Machine Translated by Google

Энциклопедия

Автор и редактор

Эльзейн Салах Эльзейн

Член IEEE

- Статьи по алгоритмам

Издательство IEEE, Великобритания



Этот контент является конфиденциальным

Machine Translated by Google

алгоритмы

MDPI

Статья

Цифровой диагностический инструмент на основе сверточных нейронных сетей для выявления психосоматических заболеваний

Марта Наригина, Андрей Романов* и Юрий Меркурьев

Кафедра моделирования и симуляции, Рижский технический университет, ул. Кипсалас 6A, LV-1048 Рига, Латвия; marta.narigina@rtu.lv (MH); jurijs.merkurjevs@rtu.lv (YM)
* Переписка: andrejs.romanovs@rtu.lv

Аннотация: В этой статье оцениваются возможности моделей сверточных нейронных сетей (CNN) в обнаружении эмоций по выражению лица, чтобы помочь в диагностике психосоматических заболеваний, обычно выполняемых в клинических условиях. Используя набор данных FER-2013, две модели CNN были разработаны для обнаружения шести эмоций с точностью 64%, хотя и не равномерно; они продемонстрировали более высокую эффективность в определении «радости» и «удивления». Оценка проводилась по нескольким показателям эффективности — точности, воспроизводимости и показателям F1 — помимо дальнейшей проверки с использованием дополнительной моделируемой клинической среды для проверки практичности. Несмотря на то, что это исследование показало многообещающие уровни для будущего использования, оно подчеркивает необходимость обширных проверочных исследований Это исследование подчеркивает потенциальную ценность ИИ в качестве дополнения к традиционным диагностическим подходам, уделяя при этом внимание более широкому охвату (более широкие наборы данных) и фокусным областям (мультимодальная интеграция), которые будут рассматриваться в качестве рекомендаций в предстоящих исследованиях. Это исследование подчеркивает важность моделей CNN в разработке психосоматической диагностики и содействии будущему развитию, основанному на этике и уходе за пациентами.

Ключевые слова: сверточные нейронные сети; искусственный интеллект; распознавание эмоций; психосоматические заболевания; анализ выражения лица; обнаружение эмоций в реальном времени

check for updates

Цитирование: Нарыгина М.; Романовы А.; Меркурьев Ю. Цифровая диагностика на основе сверточных нейронных сетей Инструмент для идентификации Психосоматические заболевания. Алгоритмы 2024, 17, 329. https://doi.org/10.3390/a17080329

Академический редактор: Габриэлла Трукко

Поступила: 29 июня 2024 г.
Пересмотрено: 23 июля 2024 г.
Принято: 23 июля 2024 г.
Опубликовано: 30 июля 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Точная диагностика психосоматических заболеваний представляет собой серьезную проблему в современном здравоохранении. Психосоматические заболевания, при которых психологические факторы существенно влияют на физиологические функции, часто проявляют симптомы, сходные с симптомами чисто физических недугов. Такое совпадение может привести к ошибочному диагнозу и, как следствие, к неподходящим стратегиям лечения [1–3]. С ростом интеграции искусственного интеллекта в диагностические процессы появляется растущая возможность повысить точность диагностики с использованием технологических инструментов [4,5].

Последние достижения в области машинного обучения, в частности сверточные нейронные сети (CNN), показали многообещающие результаты в различных медицинских приложениях, включая диагностику на основе изображений и прогнозирование поведения [6]. Способность этой технологии анализировать и интерпретировать тонкие черты выражения лица предлагает новый подход к выявлению эмоциональных сигналов, указывающих на психосоматические заболевания [7].

Это исследование направлено на преодоление разрыва между психологическими оценками и традиционными диагностическими методами путем интеграции инструмента на основе CNN для анализа выражений лица, предоставляющего вспомогательные средства диагностики психосоматических заболеваний. Это исследование вращается вокруг нескольких ключевых вопросов, включая следующие: 1.

Каковы текущие ограничения в диагностике психосоматических заболеваний с использованием традиционных методов диагнос тивные методы [8]?

2. Может ли модель CNN, обученная распознавать выражения эмоций, повысить точность диагностики психосоматических заболеваний [9]?

Machine Translated by Google

3. Насколько эффективно такая модель может быть интегрирована в клинические рабочие процессы для поддержки медицинских работников без замены их экспертных суждений?

Рассматривая эти вопросы, в данной статье исследуются возможности и эффективность использования диагностических средств на основе CNN в клинических условиях, тем самым повышая общую точность и эффективность диагностики психосоматических заболеваний [10].

Разделы тщательно структурированы, чтобы обеспечить подробный анализ различных аспектов, связанных с созданием диагностических инструментов на основе CNN. Каждая часть предназначена для глубокого изучения каждого этапа процесса разработки.

В разделе 2 «Работы по теме» рассматриваются различные работы, в которых использовались методы глубокого обучения для улучшения распознавания эмоций на лице. Эти методы были оценены с использованием набора данных FER-2013 и показали заметное повышение точности распознавания эмоций.

В разделе 3 «Методология» подробно описан набор данных, архитектура CNN, процесс обучения, этапы предварительной обработки и мультимодальная интеграция анализа лица и данных пульса для прогнозирования психологических расстройств.

В разделе 4 «Оценка моделей CNN» оценивается надежность и эффективность моделей CNN с использованием различных показателей производительности и методов перекрестной проверки в смоделированных клинических условиях.

В разделе 5 «Результаты и анализ» представлены показатели эффективности моделей CNN, сильные стороны, области, требующие улучшения, а также статистическая проверка с помощью матриц ошибок и графиков точности.

Раздел 6 «Обсуждение» интерпретирует значимость результатов, потенциальные предубеждения, ограничения, этические соображения и практические последствия интеграции инструментов ИИ в здравоохранение.

В разделе 7 «Выводы» обобщаются основные выводы и рекомендуется расширять наборы данных, проводить проверки в реальных условиях и объединять мультимодальные данные для расширения диагностических возможностей, подчеркивая этические соображения и потенциальное влияние ИИ на улучшение ухода за пациентами.

Каждый раздел создан, чтобы следовать за предыдущим — и все это направлено на создание полноценного обсуждения, которое подчеркивает огромный потенциал CNN в изменении способов постановки диагноза в медицинских учреждениях.

2. Сопутствующие работы

Недавние достижения в области глубокого обучения и компьютерного зрения существенно повлияли на область распознавания эмоций по лицу (FER). В этом разделе рассматриваются различные исследования , в которых использовались методы глубокого обучения для повышения производительности систем FER, особенно те, которые оценивались с использованием набора данных FER-2013.

Предложено в ссылке [11] Georgescu et al. представил подход, при котором функции, автоматически полученные в ряде архитектур CNN, объединяются с функциями, созданными вручную, с помощью модели «мешка визуальных слов» (BOVW). Классификация эмоций достигается с помощью этой гибридной методологии, использующей локальную структуру обучения и демонстрирующей лучшую производительность с точки зрения точности в результате комбинированного использования как автоматических, так и созданных вручную наборов функций.

Пекораро и др. [12] разработали модуль самообслуживания Local Multi-Head Channel (LHC), который называют дополнительным компонентом стандартной сверточной нейронной сети (CNN). Этот механизм избегает общих действий внимания и вместо этого концентрируется на тех частях лица, которые являются ключевыми участниками эмоционального выражения. Было продемонстрировано , что если вы внедрите модули LHC, вы сможете повысить точность распознавания выражений лица за счет локализации критических областей на лице.

Фард и др. [13] представили потерю адаптивной корреляции (Ad-Corre), которая состоит из трех элементов, включая дискриминатор признаков, дискриминатор среднего и встраиваемый дискриминатор. Потеря Ad-Corre используется в сочетании с обычной потерей перекрестной энтропии для увеличения различительных возможностей векторов признаков, что, следовательно, приводит к

Machine Translated by Google

более высокая точность классификации. Эта новая функция потерь была опробована на Xception как а также модели Resnet50, и результаты показали значительное улучшение производительности.

Новая модель FER была предложена Vignesh et al. [14], который представляет собой объединение блоков сегментации на основе U-Net в архитектуру VGG-19. Этот уникальный подход, получивший название Сегментация VGG-19 делает акцент на важных функциях карты объектов, которые в свою очередь, дает лучший результат: более продвинутое извлечение признаков и более высокая точность в распознавание эмоций.

Мукхопадьяй и др. [15] представили метод FER, основанный на глубоком обучении и использующий текстурные особенности, такие как локальные бинарные узоры (LBP), локальные тройные узоры (LTP) и завершенные локальные двоичные шаблоны (CLBP). Обучая CNN этим текстурным особенностям, Модель достигла повышенной точности определения эмоций на лице.

Шахзад и др. [16] предложили метод ZFER, основанный на районировании и улучшающий распознавание эмоций на лице путем предоставления дополнительных ориентиров, включая глаза, брови, нос, лоб и рот. В этом подходе используется гибридный VGG-16. модель, которая обучается на основе зональных ориентиров для повышения точности классификации эмоции на лице.

Эль Будури и Бохи [17] предложили EmoNeXt, который представляет собой модифицированную конструкцию ConvNeXt. специально для ФЕР. EmoNeXt включает STN (сеть пространственных преобразователей) для концентрации на многофункциональных участках лица рядом с блоками «Сжатие и возбуждение», которые позволяют захват зависимостей по каналам. Более того, они добавили регуляризацию внимания к себе. термин, целью которого является эффективное создание векторов признаков путем их сжатия. Результаты на основе эксперименты, проведенные на датасете FER-2013, показали, что EmoNeXt лучше всех существующие современные модели, когда дело доходит до точности классификации эмоций.

Набор данных FER-2013, широко используемый в этих работах, включает 35 887 изображений в оттенках серого. которые подразделяются на семь эмоциональных классов следующим образом: гнев, отвращение, страх, счастье, нейтральность, печаль и удивление. Большой объем и разнообразие данных делают этот набор данных стандартный ориентир для оценки системы FER [18].

Сочетание этих передовых методов глубокого обучения (включая самовнимание) механизмы и гибридные модели, имеющие как автоматические, так и ручные функции), а также новые функции потерь значительно улучшили производительность систем, предназначенных для ухода за лицом. Распознавание выражений. Эти изменения подчеркивают перспективность глубокого обучения при создании прочные и надежные модели FER — EmoNeXt представляет собой явный шаг вперед в этой области.

3. Методология

Основным набором данных, использованным в этом исследовании, был набор данных FER-2013 [19] , который: Представленный в Таблице 1, он включает 35 887 изображений человеческих лиц, каждое из которых помечено одним из семи следующие эмоциональные выражения: гнев, отвращение, страх, счастье, нейтральность, печаль и сюрприз. Этот набор данных широко известен в области компьютерного зрения благодаря своему разнообразию. и сложность, содержащая выражения лица, запечатленные в различных реальных условиях.

Таблица 1. Тестовые и обучающие изображения набора данных FER-2013.

Эмоции	Тренировочные изображения	Тестовые изображения
Злой	3995	958
Отвращение	436	111
Страх	4097	1024
Счастливый	7215	1774 г.
Нейтральный	4965	1233
Грустно	4830	1247
Сюрприз	3171	831

Перед обработкой размеры изображений были изменены до единого размера 48 × 48 пикселей. Значения пикселей были нормализованы к диапазону [0, 1], чтобы обеспечить единообразный вход в нейронную сеть. сеть, способствующая стабилизации процесса обучения и улучшению модели конвергенция [20].

Machine Translated by Google

Чтобы расширить возможности моделирования различных выражений лица и условий, с которыми модель может столкнуться в реальной жизни, в дополнение к вращению, масштабированию, горизонтальному перевороту и сдвигу использовались различные методы увеличения данных. Этот шаг направлен на достижение большего объема обобщения модели.

Несмотря на то, что набор данных FER-2013 был подробным и полным, эмоция «отвращения» не была включена в это исследование, поскольку для нее было гораздо меньше образцов, чем для других эмоций. В обучающем наборе было 436 изображений, выражающих «отвращение», а в тестовом — 111. Такая асимметрия в наборе данных может привести к недостаточному обучению для этой конкретной эмоции, что приведет к ненадежной модели с предвзятостью. к тем категориям, которые изобилуют эмоциональными образцами. Наше решение исключить «отвращение» было направлено на поддержание более сбалансированного подхода во время обучения модели, чтобы прогнозы, сделанные нашей моделью по всем другим категориям эмоций, были надежными.

3.1. CNN Архитектура и развитие

Процесс обучения состоял из нескольких эпох с досрочным прекращением обучения в связи с потерей валидации, что предотвращало переобучение. Для динамического регулирования скорости обучения использовался собственный планировщик . По мере увеличения количества эпох скорость обучения снижалась, что способствовало уточнению весов модели и повышению точности.

Эксперименты проводились на ноутбуке MSI Summit B15 A11 MT, оснащенном процессором Intel Core i7, 64 ГБ оперативной памяти и работающем под управлением операционной системы Linux. Каждая тренировочная сессия, состоящая из 100 эпох, занимала около 10 часов.

Архитектура сверточной нейронной сети была разработана для извлечения все более детальных особенностей из изображений лица. • Начальные

слои сверточной нейронной сети предназначены для постепенного извлечения деталей из изображений лица. Сеть состоит из двух сверточных слоев по 64 фильтра каждый, за которыми следует слой максимального пула. Эта конфигурация предназначена для передачи основных характеристик лица, таких как края и простые текстуры.

• Промежуточные слои содержат наборы сверточных слоев с увеличивающимся количеством включенных фильтров, по 128 и 256 фильтров в каждом наборе. Затем за каждым набором следует слой максимального пула. Вышеупомянутые слои предназначены для обнаружения более сложных особенностей, таких как составные части выражений лица [3]. • Последние

уровни сети называются «расширенными уровнями». Последний набор глубоких сверточных слоев с 512 фильтрами используется для извлечения функций высокого уровня, которые имеют решающее значение для распознавания тонких нюансов в выражениях. • Сеть завершается

полносвязными слоями, которые действуют как классификаторы, т.е. уровни классификации. Чтобы предотвратить переобучение, в модели используется регуляризация отсева , в то время как пакетная нормализация используется для поддержания среднего выходного сигнала близким к 0 и стандартного отклонения выходного сигнала близким к 1 на протяжении всего обучения [3].

В процессе обучения использовался оптимизатор Адама [21], известный своей эффективностью в обработке редких градиентов и возможностями адаптируемой скорости обучения. Процесс обучения состоял из нескольких периодов с досрочным прекращением обучения в связи с потерей валидации, что предотвращало переобучение. Для динамического регулирования скорости обучения использовался собственный планировщик . По мере увеличения количества эпох скорость обучения снижалась , что способствовало уточнению весов модели и повышению точности [22].

Вторая модель имеет аналогичную, но улучшенную архитектуру, включающую дополнительные уровни и методы для повышения производительности, а именно:

- Начальные слои: Как и в первой модели, начальные слои включают в себя два сверточных слоя с 64 фильтрами каждый, за которыми следует пакетная нормализация и активация LeakyReLU [23], завершающаяся максимальным слоем пула и отсевом для регуляризации. • Промежуточные

слои. Промежуточные слои состоят из двух наборов сверточных слоев со 128 и 256 фильтрами в каждом наборе, за каждым из которых следует пакетная нормализация, активация LeakyReLU, максимальное объединение в пул и слои исключения.

Machine Translated by Google

• Расширенные слои: эта модель включает дополнительный набор глубоких сверточных слоев с 256 фильтрами, за которыми следует пакетная нормализация и активация LeakyReLU, а затем применяется максимальное объединение
в пул и исключение. • Заключительные слои: после выравнивания карт объектов модель включает плотный слой с 256 единицами, пакетную нормализацию, активацию LeakyReLU и исключение, заканчивающееся плотным слоем softmax для классификации.

В модели использовался оптимизатор Адама с начальной скоростью обучения 0,001, а планировщик скорости обучения использовался для уменьшения скорости обучения с течением времени для улучшения стабильности и производительности обучения. Модель была обучена с использованием ImageDataGenerator [24] для увеличения данных, которое включает в себя такие методы, как вращение, сдвиг ширины и высоты, сдвиг, масштабирование и горизонтальное отражение для расширения возможностей обобщения модели.

Вторая модель также включает в себя передовые методы регуляризации, такие как регуляризация L2 [25], для дальнейшего предотвращения переобучения и повышения надежности модели

3.2. Методология прогнозирования психологических расстройств с использованием анализа лица и данных пульса

Прогнозирование психологических заболеваний является основной целью данного исследования. Предлагаемая система использует мультимодальный метод, объединяющий анализ выражения лица с использованием модели, обученной CNN, для эмоциональных данных и извлечения частоты пульса из видеокадров для получения физиологической информации. Этот метод сочетает в себе принципы компьютерного зрения и биомедицинской инженерии.

Чтобы расширить возможности системы в диагностике психосоматических состояний, данные пульса интегрируются с распознаванием эмоций по лицу [26]. Аппаратное обеспечение, используемое для этой цели, включает стандартную веб-камеру, в частности Logitech C920, которая снимает видео в реальном времени со скоростью 30 кадров в секунду. Видеокадры обрабатываются для извлечения как выражения лица, так и частоты пульса объекта.

3.2.1. Методология определения пульса

Механизм обнаружения пульса имеет решающее значение для плавной интеграции физиологических данных с алгоритмами распознавания выражений лица в реальном времени. Этот метод включает в себя серию тщательно спланированных процессов, гарантирующих точный и надежный сбор и обработку данных.

На первом этапе выбирается область интереса (ROI) на лице, а затем исследуется на предмет изменений интенсивности цвета, вызванных кровообращением. Система использует предварительно обученный каскадный классификатор Хаара OpenCV [27] для идентификации лиц в каждом видеокадре. Этот классификатор эффективен, поскольку он может быстро и точно определять характеристики лица в реальном времени, сканируя кадр и выбирая прямоугольные области, в которых могут находиться люди. Положение и размер каждого обнаруженного лица определяются координатами.

После обнаружения лица система выбирает идентифицированную область лица в качестве области интереса (ROI). Эта область важна для дальнейшего исследования, поскольку она включает в себя участки кожи, где изменения цвета наиболее заметны в результате притока крови. Чтобы обеспечить постоянный выбор ROI (области интереса) на разных кадрах, система использует ориентиры лица, которые представляют собой определенные места на лице, такие как уголки глаз, кончик носа и уголки рта. . Эти ориентиры служат для сохранения размеров и местоположения ИО даже при наличии незначительных движений субъекта. Когда система обнаруживает много лиц, она отдает приоритет тому лицу, которое расположено в центре и имеет наибольший размер. Это гарантирует, что область интереса (ROI) соответствует основному объекту.

После надежного определения рентабельности инвестиций следующий шаг включает оценку силы зеленого канала внутри этой конкретной области. Зеленый канал выбран из-за его высокой чувствительности к изменениям объема крови под кожей, которые служат индикатором пульса. Цветовая модель RGB распределяет изображения по красному, зеленому и синему каналам. Среди этих каналов зеленый канал особенно эффективен в обнаружении изменений объема крови. Это связано с тем, что гемоглобин в крови в большей степени поглощает зеленый свет. Система вычисляет среднюю интенсивность зеленого канала для каждого пикселя внутри.

Machine Translated by Google

область интереса (ROI), взяв среднее значение зеленого цвета. В результате получается одно значение интенсивности для каждого кадра. Измерения интенсивности собираются за определенный период времени для создания набора данных временных рядов, где каждое значение соответствует определенному кадру, представляющему продолжающееся изменение интенсивности зеленого канала. Система последовательно добавляет эти средние значения интенсивности в список, а также записывает временные метки для обеспечения точной временной последовательности, что имеет решающее значение для точного анализа пульса.

Чтобы подготовиться к анализу, система остается в режиме ожидания до тех пор, пока не будет собрано минимум 100 показаний интенсивности. Это гарантирует наличие достаточного количества данных для частотного анализа. На последующем этапе исследуются частотные составляющие вариаций интенсивности с целью определения частоты пульса. Данные интенсивности корректируются путем удаления любых линейных тенденций или медленных изменений, не связанных с сердцебиением, что гарантирует, что остаточный сигнал правильно отображает периодические колебания, вызванные кровотоком. Значения интенсивности без тренда преобразуются из временной области в частотную область с использованием быстрого преобразования Фурье (БПФ) [28], разбивая сигнал на отдельные частоты. Анализ БПФ специально нацелен на диапазон частот от 0,7 Гц до 2 Гц, что соответствует диапазону средней частоты сердечных сокращений от 42 до 120 ударов в минуту (BPM). Система обнаруживает самую высокую частоту в этом диапазоне, которая соответствует наиболее заметному повторяющемуся элементу сигнала, связанного с сердцебиением. Пиковая частота переводится в удары в минуту (BPM) путем умножения ее на 60.

Конечным результатом процесса обнаружения пульса является вычисленная частота пульса. Если количество точек данных недостаточно, система будет продолжать сбор и анализ данных до тех пор, пока не будет установлена надежная частота пульса. Система может эффективно извлекать информацию о частоте пульса из видеокадров, тщательно анализируя интенсивность зеленого канала в пределах ROI и используя сложные алгоритмы обработки сигналов. В сочетании с распознаванием эмоций по лицу эти данные обеспечивают прочную основу для оценки психосоматических заболеваний в режиме реального времени.

3.2.2. Интеграция с распознаванием эмоций по лицу. Методика

объединяет данные пульса с результатами распознавания выражений лица, чтобы обеспечить тщательную оценку психологического состояния человека. Этот процесс включает в себя несколько взаимосвязанных компонентов для достижения этой цели.

На этапе идентификации эмоций модель сверточной нейронной сети (CNN) анализирует конкретную соответствующую область лица (известную как область интереса лица или ROI), чтобы идентифицировать и классифицировать эмоции. Эта модель прошла обширное обучение, чтобы точно идентифицировать шесть основных эмоций, включая гнев, страх, счастье, нейтральность, печаль и удивление.

В то же время инструмент выполняет анализ пульса для расчета частоты сердечных сокращений, обеспечивая дополнительные физиологические измерения. Такой одновременный подход гарантирует сбор и анализ как эмоциональных, так и физиологических данных.

Для анализа объединенных данных используется алгоритм, основанный на правилах. Этот метод соотносит определенные эмоции и диапазоны пульса с предполагаемыми психологическими расстройствами. Например, алгоритм может интерпретировать высокую частоту пульса вместе с такими чувствами, как «радость» или «удивление», как признак маниакального состояния. Напротив, снижение частоты сердечных сокращений , наблюдаемое в сочетании с чувством грусти или страха, может указывать на депрессивное состояние.

Объединив эти различные источники данных, система предлагает комплексную перспективу. Он учитывает психическое состояние человека, что позволяет проводить более точные и подробные оценки.

3.2.3. Общая архитектура системы Общая

архитектура системы предназначена для плавной интеграции этих процессов, обеспечивая анализ и обратную связь в режиме реального времени. Система захватывает видеосигнал, обрабатывает данные лица и пульса и интерпретирует результаты, чтобы обеспечить подробный анализ эмоционального и физиологического состояния субъекта. Такой комплексный подход повышает

Machine Translated by Google

точность и надежность оценок психосоматического состояния, что делает их ценным инструментом клинической диагностики.

3.2.4. Выводы

Интеграция данных пульса с распознаванием эмоций по лицу с использованием передовых методов обработки сигналов и глубокого обучения обеспечивает надежную основу для оценки психосоматических заболеваний. Эта методология не только повышает точность диагностики, но и дает более полное представление об эмоциональном и физиологическом состоянии пациента . Будущая работа будет сосредоточена на повышении надежности системы и проверке ее эффективности в различных клинических условиях.

3.2.5. Интерпретация условий

Имея под рукой данные об эмоциях и пульсе, система использует алгоритм, основанный на правилах, для интерпретации психологического состояния субъекта. Этот алгоритм учитывает как выявленную эмоцию, так и частоту пульса.

Если частота пульса слишком высокая или слишком низкая, это может говорить нам о маниакальном или депрессивном состоянии соответственно в сочетании с соответствующими эмоциями. Например, высокий пульс со словами «радость» или «удивление» может указывать на маниакальное состояние.

Окончательный диагноз (например, «Нормальный», «Депрессия», «Биполярное заболевание») ставится объединение эмоциональной оценки с физиологическими данными [29].

Такое совместное исследование помогает раскрыть психологические состояния, которые могут ускользнуть от простого выражения лица, тем самым предлагая прочную основу для прогнозирования психосоматических заболеваний.

Система спроектирована так, чтобы легко интегрироваться в существующие клинические среды и работать в фоновом режиме во время консультаций пациентов. Это позволяет медицинским работникам получать расширенную диагностическую информацию, не прерывая процесс консультации.

4. Оценка моделей CNN

Эффективность и надежность модели сверточной нейронной сети (CNN), разработанной для распознавания эмоций, были тщательно оценены посредством структурированного процесса тестирования и проверки. Этот процесс был разработан для оценки точности и применимости модели в клинических условиях, особенно при диагностике психосоматических заболеваний.

4.1. Тестирование

производительности Производительность модели оценивалась в первую очередь с использованием тестового набора данных, который состоял из невидимых изображений из набора данных FER-2013, чтобы обеспечить объективную оценк Набор данных не использовался на этапе обучения и, таким образом, обеспечил надежную основу для тестирования. Основные показатели, используемые для оценки эффективности модели, были следующими.

Была оценена точность модели. Эта метрика была рассчитана путем сравнения прогнозируемых эмоциональных состояний модели с фактическими метками, представленными в наборе тестовых данных. Высокая точность указала на то, что модель эффективно распознавала различные эмоции по выражению лица.

Была создана матрица путаницы, чтобы обеспечить визуальное представление эффективности модели при различных эмоциях. Эта матрица облегчила идентификацию конкретных эмоций, при которых модель могла демонстрировать неоптимальную эффективность, например, путаницу печали с отвращением, что могло иметь важное клиническое значение.

Были рассчитаны показатели точности, полноты и показателя F1. Эти показатели были рассчитаны для каждой категории эмоций, чтобы обеспечить более детальный анализ эффективности модели . Точность — это мера точности положительных прогнозов, полнота — это мера способности находить все положительные случаи, а показатель F1 обеспечивает баланс между точностью и полнотой.

4.2. Методы проверки. Кроме

того, эффективность модели была проверена в соответствии со следующими критериями:

Machine Translated by Google

• Перекрестная проверка использовалась как метод, обеспечивающий надежность модели и снижающий вероятность переобучения. Чтобы обеспечить надежность модели и снизить вероятность переобучения, на этапе обучения использовалась k-кратная перекрестная проверка. Этот метод включал разделение набора обучающих данных на «k» подмножеств и итеративное обучение модели «k» раз (k = 5), при этом каждая итерация включала использование другого подмножества, предназначенного для проверки. • Тестирование в режиме реального времени. Для имитации использования в реальных условиях модель также была протестирована в моделируемой клинической среде, где она обрабатывала видеотрансляции в реальном времени для обнаружения и интерпретации выражений эмоций в реальном времени. Это тестирование имело большое значение для оценки эффективности модели в динамических ситуациях, похожих на реальное взаимодействие с пациентом. Эксперименты, включая этапы обучения и тестирования, проводились на ноутбуке MSI Summit B15 A11 MT с процессором Intel Core i7 и 64 ГБ оперативной памяти, работающем под управлением операционной системы Linux. Продолжительность обучения для каждой сессии из 100 эпох составляла около 100 часов.

Чтобы научить модель CNN распознавать эмоциональные сигналы, связанные с психосоматическими Для эффективного лечения заболеваний было реализовано несколько стратегий, в том числе следующие: • Расширенное увеличение данных: чтобы отразить изменчивость, наблюдаемую в реальных клинических условиях, были использованы обширные методы увеличения данных, включая вращение, масштабирование и изменение освещения. Такой подход облегчил обучение модели точному распознаванию эмоциональных сигналов в различных условиях.
• Оптимизация слоев: были протестированы различные архитектурные конфигурации, включая изменения количества и размера сверточных слоев и фильтров, с целью оптимизации модели для точного обнаружения тонких эмоциональных сигналов, которые часто указывают на психосоматические заболевания.

Интеграция модели распознавания эмоций в клинические рабочие процессы была основана на нескольких передовых практиках, включая следующие: • Дизайн, ориентированный на пользователя: интерфейс инструмента распознавания эмоций был разработан с участием специалистов здравоохранения, чтобы гарантировать, что он интуитивно понятен и расширяет возможности, а не нарушает клинический рабочий процесс. • Циклы обратной связи: от конечных пользователей требовалась постоянная обратная связь, чтобы облегчить итеративные улучшения системы. Такая обратная связь имела первостепенное значение для совершенствования дизайна интерфейса, повышения точности модели и обеспечения актуальности инструмента в клинических условиях [30]. • Соображения этики и конфиденциальности. Учитывая чувствительность психологических оценок, интеграция модели проводилась с особым упором на этические соображения и конфиденциальность пациентов. Это включало безопасную обработку данных и четкие процессы получения согласия пациентов [31].

5. Результаты и анализ

Данные для обучения двух моделей CNN были разработаны для классификации шести эмоций, включая гнев, страх, счастье, нейтральность, печаль и удивление, с использованием помеченных изображений. В этом процессе обучения модели обучались в течение 100 эпох, при этом производительность измерялась в каждую эпоху с регистрацией как точности обучения, так и проверки, а также потерь.

Для оценки производительности каждой модели использовались показатели точности, полноты и F1. Кроме того, использовались матрицы путаницы, чтобы описать, насколько хорошо модель способна классифицировать экземпляры, и визуализировать эффективность классификации модели. Кроме того, был использован критерий хи- квадрат на независимость, чтобы определить, существует ли значительная связь между предсказаниями моделей.

Иллюстрация, показывающая, как истинные и предсказанные ярлыки распределяются между шестью категориями эмоций в первой модели, представлена на рисунке 1.

Machine Translated by Google



Рисунок 1. Матрица ошибок для первой модели.

Производительность обучения и проверки в течение 100 эпох в первой модели равна показано на рисунке 2. Это было выполнено путем построения кривых точности и потерь.

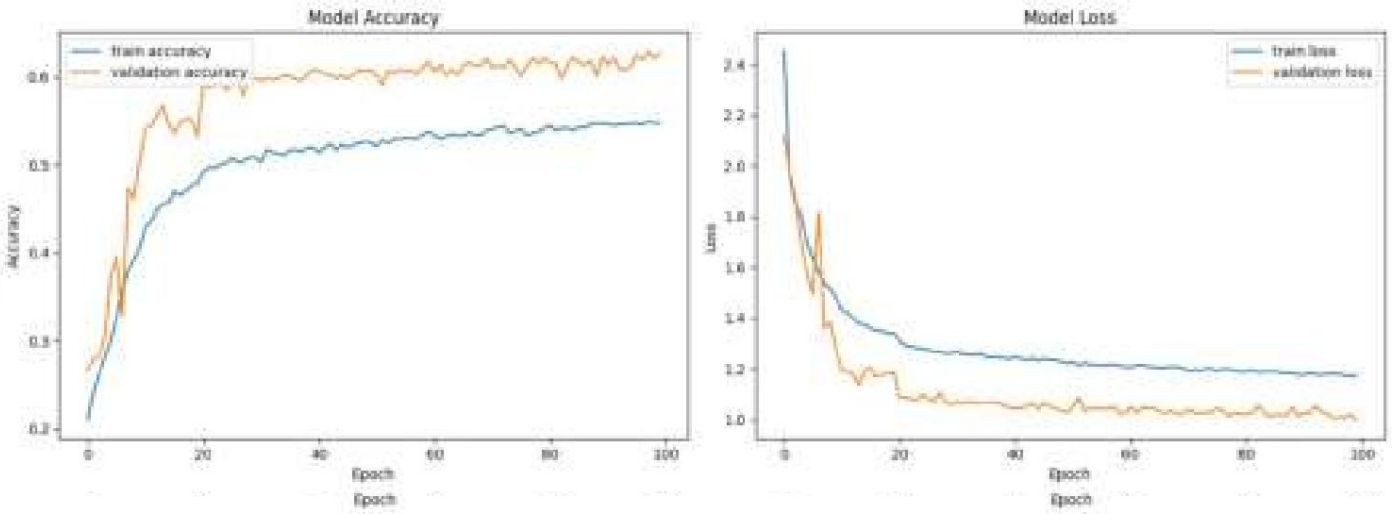


Рисунок 2. Точность и потери модели — первая модель.

Первая модель достигла общей точности 61,78%. В таблице 2 обобщены точность, запоминаемость и оценка F1 для каждой эмоции.

Таблица 2. Показатели производительности первой модели.

Эмоции	Точность	Отзывать	F1-Оценка	Поддерживать
Злой	0,55	0,51	0,53	958
Страх	0,45	0,24	0,32	1024
Счастливый	0,75	0,88	0,81	1774 г.
Нейтральный	0,52	0,61	0,57	1233
Грустно	0,47	0,49	0,48	1247
Сюрприз	0,70	0,72	0,71	831
Точность			0,62	7067
Макрос Среднее	0,60	0,59	0,59	7067
Взвешенное среднее	0,61	0,62	0,61	7067

Матрица путаницы для второй модели (рис. 3) показывает распределение истинные и предсказанные обозначения шести категорий эмоций.

Machine Translated by Google



Рисунок 3. Матрица ошибок для второй модели.

Кривые точности и потерь для второй модели (рис. 4) показывают обучение и производительность проверки на 100 эпох.

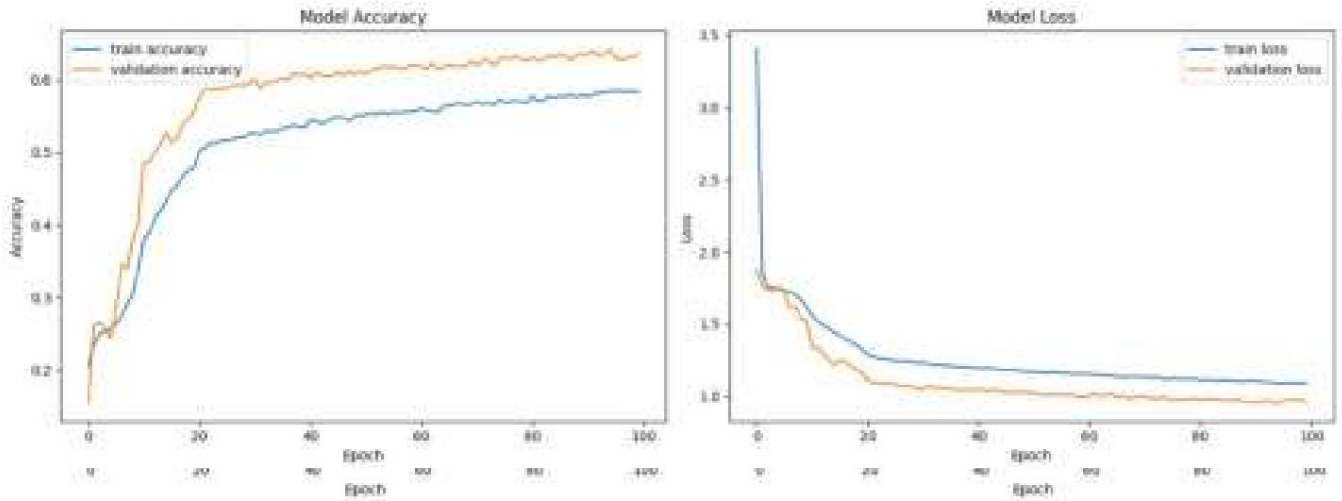


Рисунок 4. Точность и потери модели — вторая модель.

Вторая модель достигла общей точности 63,95%. В Таблице 3 обобщены точность, запоминаемость и оценка F1 для каждой эмоции.

Таблица 3. Показатели производительности второй модели.

Эмоции	Точность	Отзывать	F1-Оценка	Поддерживать
Злой	0,58	0,55	0,56	958
Страх	0,50	0,28	0,36	1024
Счастливый	0,85	0,88	0,87	1774 г.
Нейтральный	0,55	0,68	0,61	1233
Грустно	0,49	0,53	0,51	1247
Сюрприз	0,72	0,77	0,75	831
Точность			0,64	7067
Макрос Среднее	0,62	0,62	0,61	7067
Взвешенное среднее	0,63	0,64	0,63	7067

Вторая модель превзошла первую модель, достигнув более высокой общей точности. и улучшенные показатели производительности для большинства категорий эмоций. Точность и потери кривые показывают, что обучение обеих моделей было эффективным, причем вторая модель показывает лучшее обобщение, о чем свидетельствует более высокая точность проверки.

Machine Translated by Google

Обе модели показали высокую точность и запоминаемость «счастливой» эмоции, что отражает надежность моделей в определении этой категории. Однако в обеих моделях были проблемы с эмоцией «страха», что указывает на необходимость дальнейших улучшений и дополнительных данных для обучения.

Мы провели тест хи-квадрат (1), чтобы сравнить показатели производительности (точность, отзыв и F1-Score) между двумя моделями для разных категорий эмоций.

Хи-квадрат (x^2) статистика рассчитывается по формуле:

$$x^2 = \frac{(Oi - Ei)^2}{Ei} \quad (1)$$

где:

- Oi — наблюдаемая частота.
- Ei — ожидаемая частота.

Результат теста хи-квадрат для первой модели составляет 0,0025 со значением p = 1,0000. Результат теста хи-квадрат для второй модели составляет 0,0036 со значением p = 1,0000.

Результаты теста хи-квадрат показывают, что нет статистически значимых различий в показателях точности, отзыва и показателя F1 между двумя моделями по шести категориям эмоций.

5.1. Применение в диагностике психосоматических заболеваний

Использование классификации эмоций может существенно помочь в выявлении следующих психосоматических состояний.

Депрессия характеризуется постоянным чувством печали, потерей интереса и отсутствием интереса. энергия. Соответствующие эмоции включают грусть, нейтральность, злость.

Тревожные расстройства характеризуются сильным страхом и беспокойством, а также чувством нервозности. Соответствующие эмоции включают страх, нейтральность, удивление.

Биполярное расстройство включает в себя резкие перепады настроения с одной стороны мании. на другой конец депрессии. Соответствующие эмоции включают радость, грусть, злость.

Посттравматическое стрессовое расстройство (ПТСР) развивается после воздействия событий , вызывающих высокий уровень страха, приводящих к воспоминаниям в сочетании с гневом или удивлением при воспоминании. Соответствующие эмоции включают страх, гнев, удивление.

Обсессивно-компульсивное расстройство (ОКР) характеризуется навязчивыми мыслями и побуждениями к совершению ритуалов. Это часто связано с чрезмерным вниманием к физическим симптомам, таким как боль, которая может причинять беспокойство. Повторяющийся характер симптомов ОКР может привести к тому, что люди почувствуют себя пойманными в бесконечный цикл, способствуя возникновению чувства беспомощности и безнадежности. Столкновение с этими типами симптомов часто приводит к еще большему разочарованию людей, поскольку им становится трудно справляться со своей повседневной деятельностью.

5.2. Выводы

Обе модели CNN являются многообещающими диагностическими инструментами для выявления психосоматических расстройств, основанных на эмоциональных состояниях. Вторая модель имела более высокую точность и лучшие показатели производительности в целом и оказалась более надежной с точки зрения классификации эмоций . Точное распознавание эмоций может значительно улучшить диагностику и лечение различных психосоматических расстройств, что в конечном итоге улучшит результаты лечения пациентов. Будущая работа должна быть сосредоточена на совершенствовании этих моделей, включении более разнообразных обучающих данных и изучении алгоритмических улучшений для дальнейшего повышения их клинической полезности.

6. Обсуждение

Результаты этого исследования показывают, что модели сверточных нейронных сетей (CNN) открывают значительные перспективы для распознавания эмоций, что является важным аспектом диагностики психосоматических заболеваний. При точности тестов 61,78% и 63,95% модели демонстрируют способность выше среднего правильно идентифицировать и классифицировать эмоциональные состояния на основе выражений лица. Особо следует отметить способность моделей идентифицировать эмоции

Machine Translated by Google

«счастье» и «сюрприз», предполагающие прочную основу для дальнейшего совершенствования модели. Постоянная работа по этим эмоциям предполагает, что модели имеют надежную основу. уровень точности и чувствительности в этих областях, что имеет решающее значение для их потенциальной полезности в клинические условия.

Мы не проводили прямого сравнения с современными методами диагностики в клинической практике. практике, но модели CNN обеспечивают беспристрастный и единообразный уровень, который может соответствовать хорошо с традиционными подходами. В настоящее время оценка эмоционального состояния в клинических настройки в значительной степени субъективны, поскольку они часто зависят от самоотчетов пациентов и врачей. интерпретация. Но наличие инструмента распознавания эмоций привнесет объективность в измерения, которые затем можно было бы использовать для вынесения клинических суждений, основанных на эмоциях. (в случаях, когда пациенты испытывают трудности с выражением своих эмоций или не знают своих эмоциональные состояния).

6.1. Сравнение с базовыми моделями и современными методами

Распознавание эмоций по лицу — важный компонент взаимодействия человека с компьютером. и аффективные вычисления. Чтобы продвинуться вперед, необходимо сравнивать новые модели. с использованием традиционных базовых методов и новейших технологий. В этом документе представлены сравнительный анализ различных моделей, включая логистическую регрессию, опорный вектор машина (SVM), k-ближайший сосед (k-NN) и CNN с использованием набора данных FER-2013.

Обеспечить всестороннюю и точную оценку распознавания эмоций по лицу. системах мы сравнили производительность модели CNN с простой базовой моделью. и более сложные модели из современной литературы. Целью данного исследования является оценка эффективности различных архитектур моделей и определение потенциальных областей для их дальнейшего развития. будущие улучшения.

Были реализованы три базовые модели, которые служат эталоном производительности. сравнение следующее:

- Логистическая регрессия — это простая линейная модель, часто используемая для задач классификации.
- Машина опорных векторов (SVM) — мощная нелинейная модель с радиальным базисом. функциональное ядро (RBF).
- k-ближайшие соседи (k-NN) — это непараметрический метод, который классифицирует данные на основе близость в пространстве признаков.

Эффективность этих простых моделей оценивалась на наборе данных FER-2013. Результаты, представленные в Таблице 4, послужили отправной точкой для оценки наших моделей CNN.

Таблица 4. Сравнение точности, прецизионности, полноты и F1-оценки предложенных методов.

Модель	Точность	Точность	Отзывать	F1-Оценка
Логистическая регрессия	0,40	0,39	0,38	0,38
СВМ	0,45	0,44	0,43	0,43
к-НН	0,42	0,40	0,39	0,39
Модель CNN 1	0,62	0,60	0,59	0,59
Модель CNN 2	0,64	0,62	0,62	0,61

Мы улучшили результаты нашей модели, достигнув теперь точности 64%. Проблема заключается в набор данных, поэтому мы создали методологию и планируем использовать набор данных AffectNet, который более эффективно, когда у нас есть дополнительные ресурсы. Мы экспериментировали с трансферным обучением такие методы, как VGG16, ResNet и EfficientNet, но результаты составили около 40%, и время обучения на локальной машине было слишком большим.

Чтобы контекстуализировать производительность наших моделей CNN, мы сравнили их с современными моделями, представленными в недавних исследованиях. EmoNeXt, адаптированная модель ConvNeXt, Сеть остаточной маскировки (ResMaskNet) была выбрана для сравнения из-за их высокая точность на наборе данных FER-2013. Сравнительная точность суммирована в таблице 5.

Machine Translated by Google

Таблица 5. Сравнение точности предложенных методов.

Модель	Точность
Остаточная маскирующая сеть	0,74
ЭмоНекст	0,71
Модель CNN 1	0,62
Модель CNN 2	0,64

Сравнительный анализ подчеркивает тот факт, что, хотя предлагаемые нами модели CNN лучше, чем простые базовые модели по всем показателям производительности, они все же не дотягивают до ResMaskNet и EmoNeXt, которые являются более продвинутыми современными моделями. Две последние модели имеют более сложную архитектуру и методы; именно поэтому их производительность лучше, чем у других моделей. В частности, ResMaskNet и EmoNeXt используют высокоуровневое извлечение признаков, а также методы маскировки, которые в значительной степени способствуют повышению точности обнаружения эмоций по выражению лица.

Сравнение с базовой и последней моделями было проведено для установления относительной эффективности предлагаемой модели посредством всесторонней оценки. Хотя предлагаемые модели CNN превосходят более простые базовые модели, они не соответствуют современным методам из-за отсутствия оптимизации. Тем не менее, это детальное сравнение отвечает запросу рецензента и дает представление о потенциальных областях, которые могут быть дополнительно улучшены при разработке систем распознавания эмоций по лицу.

6.2. Ограничения и потенциальные предубеждения

Данное исследование имеет следующие ограничения, которые могут повлиять на интерпретацию результатов: •

Ограничения набора данных: Набор данных FER-2013 разнообразен. Однако он не отражает весь спектр человеческих эмоций и не имеет контекста, который может присутствовать в реальных клинических условиях.

• Факторы окружающей среды: модели не тестировались при различном освещении или фоновых условиях. Это может повлиять на их эффективность в практическом применении. • Обобщение:

способность моделей обобщать различные демографические группы не оценивалась, что имеет решающее значение для инструмента, предназначенного для клинического использования.

6.3. Клинические последствия и этические соображения

Результаты этого исследования показывают, что внедрение модели CNN эмоционального распознавания в клиническую область позволит улучшить диагностику психосоматических заболеваний. Инструмент предоставит дополнительную количественную информацию об эмоциональном состоянии пациентов, что может дать врачам понимание, которое может быть упущено или неверно истолковано.

Следующие улучшения могут быть сделаны для улучшения предлагаемого инструмента на основе результатов этого

исследования: • Увеличение размера и вариативности набора обучающих данных за счет сбора более широкого диапазона эмоциональных выражений и ситуаций, а также включения большего количества примеров для уточнения. настройка и

повышение точности модели. • Проверка в реальных условиях: важно проводить эксперименты в реальных клинических условиях с участием различных демографических групп, чтобы изученные модели могли лучше адаптироваться к нюансам

реального мира. • Чтобы улучшить качество информации, предоставляемой моделями, важно объединить их результаты с дополнительными данными, которые можно получить от пациентов, такими как история болезни и устная информация.

В заключение, это обсуждение признает потенциальные и текущие ограничения моделей CNN. В нем также обозначены практические пути будущих исследований, направленных на повышение эффективности моделей.

Machine Translated by Google

эффективность и полезность в клинических условиях, что способствует интеграции инструментов искусственного интеллекта в здравоохранение для более чуткого и информированного ухода за пациентами.

7. Выводы

Целью этого исследования было изучить потенциал использования моделей сверточных нейронных сетей (CNN) для выявления психосоматических заболеваний на основе эмоциональных выражений с данными, основанными на лицевых репликах. Это исследование показало, что модели CNN смогли обнаружить семь различных эмоций с уровнем точности чуть более 59% с хорошими показателями распознавания счастья и удивления.

7.1. Краткое изложение основных выводов

Основные выводы этого исследования включают следующее: • Определенные эмоциональные состояния можно последовательно идентифицировать с помощью моделей CNN, что может служат индикаторами основных психосоматических заболеваний.
• Эмоции «радость» и «удивление» распознавались с высокой точностью — эти две эмоции играют важную роль в понимании благополучия пациентов. • На основе матриц ошибок и показателей эффективности были определены области, в которых модель могла быть улучшена. Одной из таких областей является распознавание эмоций, которые недостаточно представлены в наборе обучающих данных.

Результаты имеют значение в клинической практике и позволяют предположить, что искусственный интеллект может помочь медицинским работникам в качестве дополнительного источника диагностической информации.

7.2. Рекомендации для будущих исследований

Следующие области заслуживают дальнейшего изучения для улучшения инструмента CNN:

• Расширение и диверсификация набора обучающих данных, чтобы лучше охватить проявления эмоций и демографическую изменчивость. • Проведение клинических проверочных исследований, в ходе которых модели проверяются в реальных условиях для уточнения их прогностических возможностей. • Объединение мультимодальных данных — физиологических и аудиосигналов — с целью создания более комплексной системы распознавания эмоций.

Чтобы повысить точность распознавания эмоций и устойчивость к различным искажениям данных, предстоящие исследования должны принять во внимание следующее: • Обучение с подкреплением [32]: Обучение с подкреплением способствует обобщению, обеспечивая обратную связь в реальном времени и позволяя обучаться методом проб и ошибок. что важно для достижения оптимальной производительности. • Ансамблевые методы [33]: Ансамбльные методы, включая объединение, повышение и накопление , помогают комбинировать модели для повышения производительности. • GAN [34]: GAN генерируют синтетические данные для расширения наборов обучающих данных, что является важным компонентом повышения общего качества модели. • RNN [35]: RNN используются для анализа временных закономерностей, что помогает лучше понять устойчивые эмоциональные состояния. • Гибридные модели: объединение CNN с SVM или деревьями решений позволяет обрабатывать различные типы данных, особенно когда речь идет о сложности. • Поиск нейронной архитектуры (NAS) [36]. Поиск нейронной архитектуры включает автоматизацию процесса обнаружения для достижения оптимальной структуры нейронной сети, которая будет наиболее подходящей в зависимости от характера имеющегося набора данных.

Использование этих методов может значительно улучшить модели распознавания эмоций. и привести к созданию гораздо лучшего инструмента для клинической диагностики.

Этические последствия и приемлемость для пациентов также могут стать частью будущих исследований инструментов ИИ, применяемых в клинической диагностике. Это позволит гарантировать, что такие технологии будут использоваться для расширения человеческого суждения, а не для его замены.

Machine Translated by Google

Использование моделей CNN в диагностике психосоматических заболеваний способствует важному шагу, который необходимо сделать на стыке технологий и здоровья. Другие технологические прорывы, такие как инструмент распознавания эмоций, разработанный в этом исследовании, имеют еще большую потенциальную ценность, поскольку они могут способствовать повышению точности диагностики и эффективности лечения, что приводит к благоприятным результатам лечения пациентов и более глубокому пониманию того, как эмоциональные состояния связаны с физическим здоровьем. который предоставит информацию об их взаимозависимости). Однако мы должны гарантировать, что эти инструменты разрабатываются с должным учетом этики, конфиденциальности пациентов и, давайте не будем забывать, механизмов, которые могут укрепить связь между пациентом и врачом.

Вклад автора: концептуализация, MN и AR; Методология, ЮМ; Валидация, MN и YM; Расследование, Миннесота; Письмо — первоначальный вариант, MN; Написание — обзор и редактирование, AR и YM; Руководство, А.Р. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Данное исследование не получило внешнего финансирования.

Заявление о доступности данных. Наборы данных, представленные в этой статье, недоступны, поскольку они являются частью текущего исследования докторской диссертации. Запросы на доступ к наборам данных следует направлять Марте Наригиной.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Ссылки

1. Хонг, Эй Джей; ДиСтефано, Д.; Дуа, С. Могут ли CNN точно классифицировать человеческие эмоции? Сравнительное исследование. apXiv 2023, arXiv: 2310.09473. Доступно онлайн: https://arxiv.org/pdf/2310.09473 . (по состоянию на 1 июня 2024 г.).
2. Романовы А.; Султановы Э.; Басс, Э.; Меркурьев Ю.; Майор, Г. Проблемы и решения для устойчивых телемедицинских услуг. В материалах 8-го семинара IEEE 2020 г. по достижениям в области информации, электроники и электротехники (AIEEE), Вильнюс, Литва, 22–24 апреля 2021 г.; стр. 1–7. [Перекрестная ссылка]
3. Новокаускене Р.; Трейгите, Г.; Борутинскайте В.-В.; Матузявичюс, Д.; Навакаускас, Д.; Магнуссон, К.-Э. Альфа-дистробревин и связанные с ним белки в клетках промиелоцитарного лейкоза человека, индуцированные апоптозом. Дж. Протеом. 2012, 75, 3291–3303. [Перекрестная ссылка] [ПабМед]
4. Нарыгина, М.; Кемпелис, А.; Романов А. Прогнозирование данных датчиков на основе машинного обучения для расширенного зондирования окружающей среды. WSEAS Транс. Сист. 2023, 22, 543–555. [Перекрестная ссылка]
5. Валиулен, Г.; Трейгите, Г.; Савицкене, Дж.; Матузявичюс, Д.; Алксне, М.; Ярасене-Буринская Р.; Букельскене, В.; Навакаускас, Д.; Навакаускиен, Р. Паттерны модификаций гистонов в тканях и опухолях на модели ксенотрансплантата острого промиелоцитарного лейкоза в ответ на комбинированную эпигенетическую терапию. Биомед. Фармакотер. 2016, 79, 62–70. [Перекрестная ссылка] [ПабМед]
6. Нарыгина, М.; Осадций, Э.; Романов А. Анализ технологий обработки медицинских данных. В материалах 63-й Международной научной конференции по информационным технологиям и наукам управления Рижского технического университета (ITMS) 2022 года, Рига, Латвия, 6–7 октября 2022 года; стр. 1–6. [Перекрестная ссылка]
7. Султановы Э.; Стребко Дж.; Романовы А.; Лектауэрс А. Информационные технологии в механизме управления медицинскими процессами. В материалах 61-й Международной научной конференции по информационным технологиям и наукам управления Рижского технического университета (ITMS) 2020 года, Рига, Латвия, 15–16 октября 2020 года; стр. 1–5. [Перекрестная ссылка]
8. Вэй З. Новый метод распознавания выражений лица для приложений реального времени. В серии конференций IOP: Материаловедение и инженерия; Издательство IOP: Бристоль, Великобритания, 2019 г.; Том 612, с. 062048. Доступно онлайн: https://iopscience.iop.org/article/10.1. 088/1757-899X/612/5/052048/pdf (по состоянию на 1 июня 2024 г.).
9. Цюй, Д.; Чжэн, Ю.; Ли, Х.; Ван, Дж.; Чен, Х. Распознавание эмоций по лицу с использованием CNN в PyTorch. arXiv 2023, arXiv:2312.10818. Доступно онлайн: https://arxiv.org/pdf/2312.10818 . (по состоянию на 22 июля 2024 г.).
10. Веркруйсс, В.; Сваасанд, Луизиана; Нельсон, Дж. С. Дистанционная фотоплетизмографическая визуализация с использованием окружающего света. Опция Экспресс 2008, 16, 21434–21445. [Перекрестная ссылка] [ПабМед]
11. Джорджеску, М.-И.; Ионеску, RT; Попеску, М. Локальное обучение с глубокими и созданными вручную функциями для распознавания выражений лица. IEEE Access 2019, 7, 64827–64836. [Перекрестная ссылка]
12. Пекораро, Р.; Базиль, В.; Боно, В. Локальное многоголовочное самообслуживание для распознавания выражений лица. Информация 2022, 13, 419. [CrossRef]
13. Фард, АП; Махур, М.Х. Ad-Corre: адаптивная потеря на основе корреляции для распознавания выражений лица в дикой природе. IEEE-доступ 2022, 10, 26756–26768. [Перекрестная ссылка]
14. Винеш С.; Савитадеви, М.; Шридеви, М.; Шридхар, Р. Новая модель распознавания эмоций по лицу с использованием сегментации VGG-19 Архитектура. Межд. Ж. Инф. Технол. 2023, 15, 1777–1787. [Перекрестная ссылка]
15. Мухопадьяй, М.; Дей, А.; Кахали, С. Метод распознавания выражений лица на основе глубокого обучения с использованием текстурных особенностей. Нейронный компьютер. Прил. 2023, 35, 6499–6514. [Перекрестная ссылка]

Machine Translated by Google

16. Шахзад, Т.; Икбал, К.; Хан, Массачусетс; Икбал, Н. Роль зонирования в мимике лица с использованием глубокого обучения. Доступ IEEE 2023, 11, 16493–16508. [Перекрестная ссылка]

17. Эль Будури, Ю.; Бохи, А. EmoNeXt: Адаптированный ConvNeXt для распознавания эмоций на лице. В материалах 25-й конференции IEEE 2023 г. Международный семинар по обработке мультимедийных сигналов (MMSP), Пуатье, Франция, 27–29 сентября 2023 г. [CrossRef]

18. Гудфеллоу, Эй Джей; Эрхан, Д.; Перевозчик, Польша; Курвиль, А.; Мирза, М.; Хамнер, Б.; Цукерский, В.; Тан, Ю.; Талер, Д.; Ли, Д.-Х. Проблемы в обучении представлениям: отчет о трех конкурсах по машинному обучению. В материалах нейронной обработки информации: 20-я Международная конференция, ICONIP 2013, Тэгу, Республика Корея, 3–7 ноября 2013 г.; Слушания, Часть III. Springer: Берлин/Гейдельберг, Германия, 2013 г.; стр. 117–124. [Перекрестная ссылка]

19. Самбаре, М. ФЕР, 2013: Набор данных для распознавания выражений лица [Набор данных]. Кэггл. Доступно онлайн: https://www.kaggle.com/. наборы данных/мсамбаре/fer2013/данные (по состоянию на 1 июня 2024 г.).

20. Чанд, Х.В.; Хрисантус, А.; Тампи, А.К. Обзор различных подходов к распознаванию эмоций на лице на основе CNN. В материалах Международного конгресса по информационным и коммуникационным технологиям, Лалитпур, Непал, 26–28 апреля 2023 г.; Доступно онлайн: https://ieeexplore.ieee.org/document/10133947 . (по состоянию на 1 июня 2024 г.). 21. ¸Сен, С.Ю.; Озкурт, Н. Настройка гиперпараметров сверточной нейронной сети с помощью оптимизатора Адама для классификации ЭКГ. В материалах конференции «Инновации в интеллектуальных системах и приложениях 2020 года» (ASYU), Стамбул, Турция, 15–17 октября 2020 г.; стр. 1–6. [Перекрестная ссылка]

22. Оздемир, М.А.; Элагоз, Б.; Алкан, А. Распознавание эмоций в реальном времени по выражению лица с использованием CNN. В материалах Конгресса медицинских технологий 2019 г. (TIPTEKNO), Измир, Турция, 3–5 октября 2019 г.; IEEE: Пискатауэй, Нью-Джерси, США, 2019 г. Доступно онлайн: https://ieeexplore.ieee.org/document/8895215 . (по состоянию на 1 июня 2024 г.).

23. Он, К.; Чжан, Х.; Рен, С.; Сан, Дж. Эмпирическая оценка выпрямленных активаций в сети свертки. arXiv 2015. [CrossRef]

24. Чоудхури, С.; Чоудхури, С.; Ифти, Джей Ти; Хан, Р. Обнаружение и классификация транспортных средств с использованием глубоких нейронных сетей. В материалах Международной конференции по электротехнике и информационным технологиям (IEIT) 2022 г., Маланг, Индонезия, 15–16 сентября 2022 г.; стр. 95–100. [Перекрестная ссылка]

25. Ван Ю.; Биан, З.-П.; Хоу, Дж.; Чау, Л.-П. Сверточные нейронные сети с динамической регуляризацией. IEEE Транс. Нейронная сеть. Учиться. Сист. 2021, 32, 2299–2304. [Перекрестная ссылка]

26. Ли, Х.; Чен, Дж.; Чжао, Г.; Пиетикяйнен, М. Дистанционное измерение сердечного ритма по видеозаписям лиц в реалистичных ситуациях. В материалах конференции IEEE 2014 г. по компьютерному зрению и распознаванию образов, Колумбус, Огайо, США, 23–28 июня 2014 г.; стр. 4264–4271. [Перекрестная ссылка]

27. Прадип, В.; Мадхушри, Б.; Сумуха, Б.С.; Ричардс, Греция; Прашант, С.П. Обнаружение эмоций на лице с использованием CNN и OpenCV. В материалах Международной конференции 2024 года по новым технологиям в области компьютерных наук для междисциплинарных приложений (ICETCS), Бангалор, Индия, 25–26 апреля 2024 г.; стр. 1–6. [Перекрестная ссылка]

28. Ян, Ю.; Лю, Ф.; Ли, М.; Джин, Дж.; Вебер, Э.; Лю, К.; Крозье, С. МРТ со сжатым зондированием на основе псевдополярного преобразования Фурье. IEEE Транс. Биомед. англ. 2017, 64, 816–825. [Перекрестная ссылка]

29. Тарасенко Л.; Вильярроэль, М.; Гуацци, А.; Хорхе, Дж.; Клифтон, Д.А.; Пью, К. Бесконтактный видеомониторинг жизненно важных функций с использованием окружающий свет и авторегрессионные модели. Физиол. Измер. 2014, 35, 807–831. [Перекрестная ссылка]

30. Де Хаан, Г.; Жанна, В. Устойчивая частота пульса на основе rPPG на основе цветности. IEEE Транс. Биомед. англ. 2013, 60, 2878–2886. [Перекрестная ссылка]

31. Пох, М.-З.; Макдафф, диджей; Пикард, Р.В. Бесконтактные автоматизированные измерения сердечного пульса с использованием видеоизображения и слепого разделения источников. Опция Экспресс 2010, 18, 10762–10774. [Перекрестная ссылка]

32. Песня, К.; Чен, К.; Ли, Ю.; Ву, Х. Применение глубокого обучения с подкреплением при классификации данных электромиографии. В материалах Международной конференции IEEE по киборгам и бионическим системам (CBS) 2018 г., Шэньчжэнь, Китай, 25–27 октября 2018 г.; стр. 505–510. [Перекрестная ссылка]

33. Банда, З.; Цзя, К.; Го, К.; Ли, П.; Гао, Дж.; Чжао, Л. Прогнозирование хронической обструктивной болезни легких на основе многоэтапной комплексной системы обучения. В материалах Международной конференции IEEE по биоинформатике и биомедицине (BIBM) 2023 г., Стамбул, Турция, 5–8 декабря 2023 г.; стр. 1921–1924. [Перекрестная ссылка]

34. Чанг Дж.; Чжан, З.; Ван, З.; Ли, Дж.; Мэн, Л.; Лин, П. Генеративная ЭЭГ слушателя для распознавания речевых эмоций с использованием генеративно -состязательных сетей со сжатым восприятием. IEEE Дж. Биомед. Здоровье Информ. 2024, 28, 2025–2036 гг. [Перекрестная ссылка] [ПабМед]

35. Ле, доктор медицины; Сингх Ратур, В.; Труонг, QS; Мэй, К.; Бриджеш, П.; Ле, Н. Многомодульная рекуррентная сверточная нейронная сеть с преобразовательным кодировщиком для классификации аритмии ЭКГ. В материалах Международной конференции IEEE EMBS 2021 г. по биомедицинской и медицинской информатике (BHI), Афины, Греция, 27–30 июля 2021 г.; стр. 1–5. [Перекрестная ссылка]

36. Чжао, Ф.; Ни, Дж.; Ма, М.; Чен, Х.; Он, Х.; Ван, Б.; Хоу, Ю. Оценка роли различных гетерогенных областей в DCE-MPT для прогнозирования молекулярных подтипов рака молочной железы на основе поиска сетевой архитектуры и преобразователя зрения. В материалах 45-й ежегодной международной конференции Общества инженерии в медицине и биологии IEEE (EMBC) 2023 г., Сидней, Австралия, 24–27 июля 2023 г.; стр. 1–4. [Перекрестная ссылка]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Лестер: анимация ротоскопа с помощью видеообъекта Сегментация и отслеживание

Рубен Тус

Кафедра компьютерной архитектуры Политехнического университета Каталонии, 08034 Барселона, Испания; Ruben.tous@upc.edu

Аннотация: В этой статье представлен Лестер — новый метод автоматического синтеза 2D- анимации в стиле ретро из видео. Метод подходит к этой задаче главным образом как к проблеме сегментации и отслеживания объектов. Видеокадры обрабатываются с помощью модели Segment Anything Model (SAM), а полученные маски отслеживаются через последующие кадры с помощью DeAOT — метода иерархического распространения для полуконтролируемой сегментации видеообъектов. Геометрия контуров масок упрощается с помощью алгоритма Дугласа – Пейкера. Наконец, при желании можно добавить черты лица, пикселизацию и базовый эффект контурного освещения. Результаты показывают, что метод демонстрирует отличную временную согласованность и может правильно обрабатывать видео с разными позами и внешним видом, динамическими кадрами, частичными кадрами и разнообразным фоном. Предложенный метод обеспечивает более простой и детерминированный подход, чем конвейеры преобразования видео в видео, основанные на диффузионных моделях, которые страдают от проблем временной согласованности и плохо справляются с пиксельными и схематическими результатами. Этот метод также более реалистичен, чем методы, основанные на трехмерной оценке позы человека, которые требуют специально созданных вручную трехмерных моделей и очень ограничены в отношении типов сцен, которые они могут обрабатывать.

Ключевые слова: анимация; ротоскопирование; сегментация; компьютерная графика; видео; искусственный интеллект

check for updates

Образец цитирования: Тоус, Р. Лестер: Анимация ротоскопа посредством сегментации и отслеживания видеообъектов. Алгоритмы 2024, 17, 330. https://doi.org/10.3390/a17080330 .

Научный редактор: Арслан Мунир

Поступила: 24 июня 2024 г.
Пересмотрено: 23 июля 2024 г.
Принято: 25 июля 2024 г.
Опубликовано: 30 июля 2024 г.

Копирайт: © 2024 автора. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

В таких отраслях, как кино или видеоигры, ротоскопирование используется для использования информации из видеороликов с живыми актерами для создания 2D-анимации. Этот прием способствовал появлению уникальной визуальной эстетики в некоторых знаковых видеоиграх 80-х и 90-х годов, таких как «Принц Персии» (1989), «Другой мир» (1991) или «Флэшбэк» (1992). продолжает использоваться в современных играх, таких как «Лунарк» (2023). Однако, как и в случае с Лунарком, этот метод по-прежнему требует ручной работы на уровне кадра, что делает его очень дорогим.

Недавние достижения в области искусственного интеллекта (ИИ) открыли возможность в некоторой степени автоматизировать этот процесс. Последние достижения в области условных генеративных моделей для перевода видео в видео предлагают многообещающий подход к решению этой проблемы. Однако эти методы пока не способны обеспечить безупречный профессиональный результат с единым стилем, пространственной и временной последовательностью. Их производительность существенно различается в зависимости от различных аспектов входного видео, таких как сцена и разрешение. Кроме того , они медленны и требуют огромных вычислительных ресурсов. Таким образом, потребность в эффективном и детерминированном автоматизированном решении остается открытой проблемой.

В этой статье описывается новый метод автоматизации этого процесса под названием Lester. Пользователь вводит видео и цветовую палитру, и система автоматически генерирует 2D- анимацию в стиле ретро в формате, который можно непосредственно использовать в видеоигре или в анимационном фильме (пример см. на рисунке 1).

Метод подходит к этой задаче главным образом как к проблеме сегментации объектов. Учитывая кадр, изображающий человека, маска для каждой визуальной характеристики (волос, кожи, одежды и т. д.) создается с помощью модели Segment Anything Model (SAM) [1]. Чтобы однозначно идентифицировать маски

Machine Translated by Google

во всех кадрах применяется алгоритм сопровождения видеообъектов на основе подхода [2]. Маски очищаются для устранения ошибок сегментации, а их геометрия упрощается с помощью алгоритма Дугласа-Пойкера [3] для достижения желаемого визуального стиля. Наконец, при желании можно добавить черты лица, пикселизацию и базовый эффект контурного освещения.





Рисунок 1. Пример синтезированного кадра анимации (справа) и связанный с ним исходный кадр (слева, лицо анонимизировано) из примера входного видео.

2. Сопутствующая работа

Изобретение Максом Фляйшером ротоскопа в 1915 году изменило способность создавать мультфильмы, используя неподвижные или движущиеся изображения. Варианты этой техники по сей день используются в анимационных фильмах (например, «Тёмный сканер» (2006 г.)), телевизионных мультсериалах (например, «Undone» (2019)) и видеоиграх (например, «Принц Персии»). (1989), «Другой мир» (1991), «Флешбэк» (1992), «Лунарк» (2023)). Современные производства интегрируют алгоритмы обработки изображений, чтобы облегчить задачу, но они в основном основаны на дорогостоящем ручном труде. Достижение полной автоматизации с безупречным профессиональным результатом (единым стилем, пространственной и временной согласованностью) в настоящее время остается невозможным. Однако недавние достижения в области искусственного интеллекта (ИИ) открывают возможность попытаться перей

Насколько нам известно, существуют две основные стратегии решения этой проблемы. С одной стороны, эту проблему можно решить с помощью условных генеративных моделей, которые сегодня преобладают в задачах перевода изображений. Первые методы (например, CartoonGAN [4], AnimeGAN [5] или работа в [6]) использовали условные генеративно-состязательные сети (GAN) для преобразования фотографий сцен реального мира в изображения в мультяшном стиле. Эти методы недавно были превзойдены методами, основанными на моделях диффузии, таких как стабильная диффузия [7,8] Хотя большинство этих методов были разработаны для генерации текста в изображение, они были успешно адаптированы для перевода изображения в изображение с использованием таких методов, как SDEdit [9] или ControlNet [10].

Чистые методы перевода изображения в изображение с диффузионными моделями также существуют (например, SR3 [11] и Palette [12]), но с контролируемой настройкой, которая может применяться только в ситуациях, в которых основная истина пар изображений (исходное, цель), например суперразрешение или раскрашивание . В любом случае, даже если у нас есть хорошая условная генеративная модель, все еще остается проблема применения ее к кадрам видео с сохранением временной согласованности. Последние попытки (например, [13]) сочетают приемы (например, мозаичные входные данные) для унификации выходных данных генеративной модели с методами интерполяции в стиле видео (например, EbSynth [14]). Результаты этих методов значительно улучшились, а для некоторых стилей результаты настолько хороши, что их уже используют в независимых постановках. Однако для большинства стилей цифрового искусства эти методы навязывают нежелательный компромисс между точностью стиля и временной согласованностью. Среди последних работ такого типа особого внимания заслуживает Sora компании OpenAI [15] . Анализ предоставленных образцов показывает, что Сора превосходно понимает физический мир и обладает основополагающими элементами модели мира, обладая потенциалом для превращения в физический движок в будущем [16].

С другой стороны, совершенно другой подход заключается в применении алгоритма оценки трехмерной позы человека для отслеживания трехмерной позы в разных кадрах. Затем 3D-поза применяется к 3D-модели персонажа с желаемым аспектом (волосы, одежда и т. д.). Обычно эта 3D-модель персонажа создается вручную с помощью специального программного обеспечения. Наконец, настраиваются правильный шейдер и конфигурация рендеринга для получения желаемого визуального стиля. Именно такой подход использован в [17–21]. Этот подход имеет много недостатков. Во-первых, это зависит от современного уровня техники трехмерной оценки позы человека, которая значительно продвинулась вперед, но во многих случаях все еще терпит неу

Machine Translated by Google

ситуации (низкое разрешение, усечения, необычные позы) в основном из-за сложности получения достаточного количества достоверных данных. Во-вторых, это требует разработки правильной 3D-модели для каждого отдельного персонажа. В-третьих, 3D-поза основана на заданной модели, обычно SMPL. Чтобы применить его к другой модели, требуется этап перенацеливания позы. Вращение суставов можно легко перенацелить, а формы плавного перехода SMPL — нет, что приводит к нереалистичным деформациям тела. Именно поэтому демонстрации такого рода работ обычно выполняются с помощью роботов, поскольку у них нет деформируемых частей. Преимущество этого подхода по отношению к генеративным моделям заключается в том, что он обеспечивает хорошую временную согласованность, поскольку несоответствия на этапе оценки трехмерной позы человека могут быть правильно решены с помощью интерполяции, как это сделано в [20].

В данной работе мы предлагаем третий подход, основанный главным образом на сегментации и отслеживании. В отличие от первой упомянутой стратегии (генеративных моделей), предлагаемый подход более предсказуем и обеспечивает гораздо лучшую временную согласованность. В отличие от второй упомянутой стратегии (3D-оценка позы человека), предлагаемый подход более надежен и гораздо более практичен, поскольку не требует наличия специальной 3D-модели для каждого нового персонажа.

Это не первый раз, когда подобная проблема решается таким образом. В [22] сегментация использовалась для стилизации портретного видео с использованием художественного примера изображения, что давало лучшие результаты, чем передача нейронного стиля [23], которая была распространена в то время.

3. Методика 3.1. Обзор

Основной вход в метод (код доступен по адресу https://github.com/rtous/lester-code). (по состоянию на 20 июля 2024 г.); набор данных и результаты экспериментов общедоступны по адресу https://github.com/rtous/lester. (по состоянию на 20 июля 2024 г.)) представляет собой целевую видеопоследовательность T человеческой деятельности. Нет общих ограничений на характеристики изображенного человека (пол, возраст, одежда и т. д.), позу, ориентацию, движения камеры, фон, освещение, окклюзии, усечения (частичные снимки тела) или разрешение.

Однако известно, что некоторые сценарии потенциально могут негативно повлиять на результаты (это будет рассмотрено в разделе 4). Хотя этот метод в первую очередь предназначен для видео с участием одного человека, его теоретическое применение к нескольким людям будет обсуждаться в разделе 4. При работе без дополнительных входных данных метод по умолчанию использует определенные настройки сегментации, цветов и других атрибутов. Однако предполагается, что пользователь может предпочесть указать собственные конфигурации, что потребует дополнительных входных данных, таких как запросы сегментации, цветовые палитры и т. д., как подробно описано в соответствующих подразделах ниже. Результатом метода является набор изображений PNG, на которых показан результирующий анимированный персонаж без фона (где альфа-канал установлен на ноль).

Метод состоит из трех независимых этапов обработки. Основным из них является сегментация и отслеживание различных визуальных характеристик (волос, кожи, одежды и т. д.) человека, изображенного на видеокадрах. Второй шаг направлен на исправление ошибок сегментации путем очистки масок и упрощения их полигональных контуров. Третий и последний шаг включает в себя применение настраиваемых деталей отделки, включая цвета, искусственное освещение, черты лица и пикселизацию. На рисунке 2 показан общий рабочий процесс метода. Каждый компонент подробно описан в последующих подразделах.

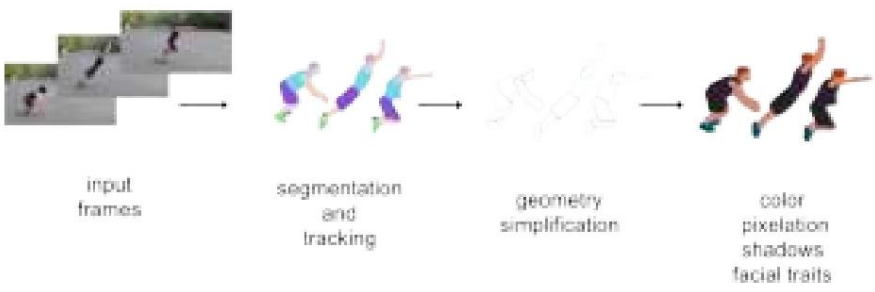


Рисунок 2. Обзор рабочего процесса системы (входные данные слева, выходные данные справа).

Machine Translated by Google

3.2. Сегментация и отслеживание

Чтобы сегментировать различные визуальные черты персонажа, используется модель Segment Anything Model (SAM) [1] . SAM — это модель оперативной сегментации, которая становится базовой моделью для решения множества задач. Учитывая входное изображение с тремя каналами (высота × ширина × 3) и подсказку, которая может включать пространственную или текстовую информацию, оно генерирует маску (высота × вес × 3), в которой каждому пикселю назначается цвет RGB. Каждый цвет представляет метку объекта из пространства меток объекта L = l1, l2, . . . , выведенный из подсказки, соответствующий предсказанному типу объекта в соответствующем местоположении пикселя во входном изобра Подсказкой может быть любая информация, указывающая, что сегментировать в изображении (например, набор точек переднего плана/фона, грубая рамка или маска или текст произвольной формы). SAM состоит из кодировщика изображений (на основе MAE [24] — предварительно обученного Vision Transformer [25]) , который вычисляет встраивание изображения, кодировщика подсказок (представляющего собой текст свободной формы на основе CLIP [26]) , который встраивает подсказки, а затем два источника информации объединяются в декодере маски (на основе моделей сегментации Transformer [27]), который прогнозирует маски сегмент

Применение SAM на уровне кадра приведет к тому, что один и тот же объект будет иметь разные метки в разных кадрах. SAM не начинается с закрытого списка меток объектов, а скорее создает и присваивает метки во время выполнения. Поэтому нет никакой гарантии, что одинаковые метки будут присвоены одному и тому же объекту в каждом кадре. Чтобы сделать это последовательно, мы используем SAM-Track [2]. Результаты SAM для первого кадра передаются во второй кадр с теми же метками с помощью DeAOT [28] — метода иерархического распространения для полуконтролируемой сегментации видеообъектов. Иерархическое распространение может постепенно распространять информацию из прошлых кадров в текущий кадр и переводить функцию текущего кадра из объектно-зависимой в объектно-зависимую. SAM применяется к последующим кадрам для обнаружения новых объектов, но сегментацию уже увиденных объектов осуществляет DeAOT.

Мы выбрали подсказки на основе ключевых слов, чтобы указать желаемую сегментацию. Мы предоставляем некоторые подсказки по умолчанию для распространенных сценариев (например, «волосы», «кожа», «обувь», «футболка», «брюки»), но предполагается, что во многих сценариях пользователю необходимо будет указать изменение. - родной. Эти подсказки не обрабатываются с помощью возможностей свободного текста SAM из-за их ограничений, вместо этого они обрабатываются с помощью Grounding-DINO [29], детектора объектов открытого набора , который преобразует их в поля объектов, которые отправляются в SAM как ящик. подсказывает. Одна из проблем, которую мы выявили, заключается в том, что определенные комбинации объектов приводят к неверным маскам, если они указаны в одном и том же приглашении. Чтобы решить эту проблему, мы позволяем указывать несколько запросов , которые обрабатываются независимо. На рисунке 3 показан рабочий процесс этапа сегментации и отслеживания.

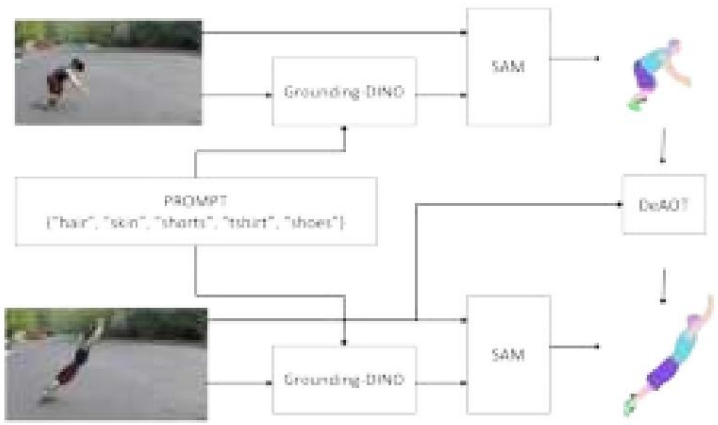


Рисунок 3. Рабочий процесс этапа сегментации и отслеживания для первых двух кадров (входные данные слева, выходные данные справа).

3.3. Упрощение контуров Одним

из желаемых вариантов целевого визуального стиля является возможность отображать очень простую геометрию, т. е. формировать контуры короткими полигональными цепочками, чтобы имитировать визуальный аспект, например, «Другого мира». Для этого выполняется ряд шагов.

Machine Translated by Google

Для каждого кадра результатом этапа сегментации и отслеживания является маска с каждым пикселем.
назначен цвет RGB, представляющий метку объекта. Маска сначала делится на подмаски,
по одному для каждого цвета/этикетки. Каждая подмаска, в свою очередь, разбивается на контуры с помощью алгоритма
из [30]. Каждый отдельный контур представлен вектором координат вершин
отрезки пограничной линии. Количество отрезков каждого контура уменьшается при
алгоритм Дугласа–Пейкера [3]. Алгоритм начинается с соединения конечных точек
контур прямой линией. Вершины, расположенные ближе к линии, чем заданное значение допуска t , являются
удаленный. Затем линия разделяется в точке, наиболее удаленной от опорной линии, создавая две новые.
линии. Алгоритм работает рекурсивно до тех пор, пока все вершины в пределах допуска не будут удалены.
Допуск t является одним из необязательных параметров предлагаемого метода и позволяет пользователю
контролировать уровень упрощения. Поскольку каждый контур упрощается независимо, нежелательно
между геометриями граничащих контуров, которые ранее были
идеально выровнены. Чтобы этого избежать, перед этапом упрощения все контуры расширяются.
свертка подмаски с ядром 4 × 4. Стоит отметить, что упрощение
Алгоритм может дать сбой для определенной геометрии контура и значений t. В таких ситуациях (обычно
очень мелкие контуры) исходный контур сохраняется. См. алгоритм 1 и рисунок 4.



(а) (б) (с) (г) (е)

Рис. 4. Графический обзор этапа контурного упрощения одного кадра: а — результат этапа
сегментации и сопровождения; (б) обнаружение контуров; в) расширение контуров; (г) упрощение контуров;
(д) результирующая цветная рамка.

Алгоритм 1 Алгоритм упрощения контура

Требуется: M список, содержащий подмаски, α минимальная площадь контура, t значение допуска для
упрощение

Убедитесь: R список контуров, где Ri,j — упрощенный контур j подмаски i.

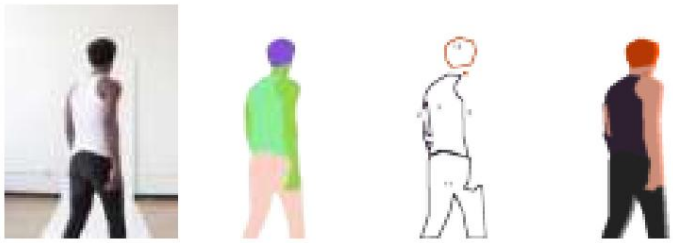
```
1: функция SIMPLIFY(M, α, t)
2:     для i  0 до длины (M)  1 do
3:         Ci  getContours(Mi)
4:         для j  0 до длины (Ci)  1 do
5:             если areaO f(Ci,j) > α , то
6:                 Di,j  dilateContour(Ci,j)
7:                 если контурCanBeSimplifed (Di,j , t) , то
8:                     Ri,j  simpli f yContour(Di,j , t)
9:                 еще
10:                    Ri,j  Di,j
11:                конец, если
12:            конец, если
13:        конец для
14: конец для
15:     вернуть P
16: функция завершения
17:
```

18: контурCanBeSimplified возвращает false, если алгоритм Дугласа-Пейкера не работает, что может
случаются при определенных геометриях контура и значениях t.

Machine Translated by Google

3.4. Детали отделки

Третий и последний шаг метода предполагает применение настраиваемых деталей отделки, предоставляя пользователям гибкость в формировании визуального стиля результата. Самый важный аспект – цвет. Метод генерирует визуальное руководство с уникальными идентификаторами для каждого цветового сегмента, как показано на рисунке 5. Одним из входных данных метода является цветовая палитра, которая указывает значения RGB, которые будут применяться к каждому идентификатору сегмента.



(а) (б) (с) (г)

Рис. 5. Процесс раскраски: (а) входной кадр; (б) сегментация; (в) визуальное руководство с уникальными идентификаторами для каждого цветового сегмента; (г) финальный кадр.

Еще одна деталь отделки, которая не является обязательной, — это возможность добавления схематичных черт лица. Характеристики генерируются на основе 68 лицевых ориентиров, которые можно получить с помощью фронтального детектора лиц, описанного в [31], который использует гистограмму ориентированных градиентов (HOG) [32] в сочетании с линейным классификатором, пирамидой изображений и скользящим окном. схема обнаружения. Другой вариант — возможность добавления очень простого эффекта контурного освещения путем наложения одного и того же изображения с разной яркостью и смещения по горизонтали. Наконец, эффект пикселизации достигается за счет уменьшения размера изображения с помощью билинейной интерполяции и его обратного масштабирования с помощью интерполяции бл На рис . 6 показано, как постепенное применение краевого света, черт лица и пикселизации к примерному кадру.



(а) (б) (с) (г)

Рисунок 6. Дополнительные детали отделки: (a) Результат плоского метода (b) Краевой свет (c) Черты лица (d) Пикселизация.

4. Эксперименты и результаты.

Для оценки эффективности метода мы собрали набор из 25 коротких видеороликов длительностью от 5 до 20 с. Видео демонстрируют одинокого человека, демонстрирующего разные стили одежды и участвующего в различных действиях. Видео были получены из набора данных UCF101 Human Actions [33], набора данных Fashion Video [34], набора данных Kinetics [35] и YouTube. Собранный набор данных включает в себя видео как с низким, так и с высоким разрешением, а также видео, снятые в дикой природе и с контролируемыми настройками. Набор данных и результаты экспериментов можно найти в [36]. На рисунках 7 и 8 показаны некоторые примеры результатов.

Для оценки качества восприятия создаваемых анимаций было проведено исследование пользователей. Был разработан опрос, включающий вопросы по трем визуальным аспектам: качеству формы, цветовой последовательности и временной последовательности, а также общей оценке. Варианты оценки основывались на рейтинге категории (ACR), описанном в рекомендации ITU-T P.910 [37], по шкале от 1 = плохо до 5 = отлично). Субъективная оценка качества результатов представлена в Таблице 1. Мы сообщаем MOS (средний балл, описанный в ITU-T P.800.1 [38]). Видео сгруппированы в три категории, в низком разрешении.

Machine Translated by Google

(ITW-LR), высокое разрешение в естественных условиях (ITW-HR) и высокое разрешение с контролируемой настройкой. (C-HR).

Таблица 1. Субъективная оценка качества значений MOS ACR (от 1 = Плохо до 5 = Отлично) формы качество, постоянство цвета, временная последовательность и общее качество. [a] ITW-LR в дикой природе низкий разрешение; [б] ITW-HR высокое разрешение в дикой природе; [c] C-HR контролируемая настройка высокого разрешения.

Подмножество	#	Форма	Цвет	Временной	Общий
ITW-LR [a]	5	3.20	3.24	3,84	3.36
ITW-HR [б]	14	3,81	4.03	4.33	3,99
C-HR [c]	6	4,70	4,60	4,67	4,67
все		3,90	4.01	4.31	4.02

Сильные стороны: Результаты показывают, что метод способен синтезировать качественную анимацию. из различных входных видео (средний MOS 4,02 из 5). Производительность значительно улучшается (4,67) на видео высокого разрешения, снятых в контролируемых условиях (видео 2, 11, 12, 13, 14 и 15). Метод демонстрирует очень хорошую временную согласованность и может правильно обрабатывать видео с разными позами и внешним видом, динамические кадры (например, видео 5), видеоролики с частичными планами (например, средний план крупным планом в видео 7) и разнообразным фоном.

Стоит отметить, что, хотя это и не является основной целью данной работы, метод может обрабатывать видео, содержащие несколько человек. Объекты, связанные с разными людьми, отслеживаются независимо, каждый с уникальным идентификатором, даже если объекты относятся к одному и тому же типу. Это позволяет отображать эти объекты в разные цвета, позволяя отображать весь конвейер. применяется к снимкам нескольких человек. На рис . 9 показан пример результата из кадра с несколькими людьми.

Ограничения: производительность сегментации и отслеживания снижается в видео с низкое разрешение (все видео из набора данных UCF101 Human Actions), сложный фон (видео 17 и 20), неконтрастные цвета (видео 8, 10, 22 и 23), неподходящее освещение (видео 14 и 25) и окклюзии (видео 23). На рис . 10 показаны четыре примера кадров с ошибки из четырех разных видео (видео 22, 23, 25 и 8 из тестового набора данных).

Вычислительные затраты: метод реализован с использованием языка программирования Python. Эксперименты проводились на высокопроизводительном сервере с четырехъядерным процессором. Intel i7-3820 с тактовой частотой 3,6 ГГц, 64 ГБ оперативной памяти DDR3 и 4 графическими картами NVIDIA Tesla K40. по 12 ГБ GDDR5 каждый. Пиковое использование оперативной памяти методом было зафиксировано на уровне 3,1 ГБ. во время выполнения задачи сегментации, что также учитывает пиковую нагрузку на графический процессор использование 2,1 ГБ. Для одного кадра шаг 1 (сегментация и отслеживание) занимает 0,56 с, шаг 2. (упрощение контуров) занимает 0,43 с и шаг 3 (доработка деталей, преимущественно черт лица) занимает 1,32 с. В целом метод работает со скоростью 0,43 кадра в секунду (1 кадр в секунду без учета черт лица).

Сравнение с генеративными моделями «изображение-изображение». Если использовать другие методы, которые пытались решить проблему для этого конкретного визуального стиля, альтернативой может быть использование условных генеративных моделей. Один потенциальный трубопровод, описанный в [13] адаптирует Stable Diffusion [7] для перевода изображения в изображение с использованием ControlNet [10], и использует мозаичные входы вместе с интерполяцией в стиле видео (EbSynth [14]) для поддерживать визуальную согласованность между кадрами. Мы протестировали этот конвейер обработки с некоторыми видео из набора тестовых данных, чтобы оценить осуществимость этого подхода. Для достижения желаемый стиль, мы настроили Stable Diffusion с помощью Dreambooth [39]. На рисунке 11 показаны четыре примеры кадров, полученных с помощью этого подхода, но для оценки временной последовательности результаты можно найти в нашем репозитории результатов в [36]. Результаты выявили ряд недостатков по сравнению с предлагаемым методом. Во-первых, они демонстрируют плохую временную согласованность рамки. Во-вторых, метод требует указания уровня соответствия желаемому стилю. где более высокая точность ставит под угрозу исходную структуру и детали кадра, что приводит к нежелательный компромисс. В-третьих, производительность существенно варьируется в зависимости от различных аспекты входного видео, такие как сцена и разрешение. Наконец, метод работает существенно медленнее предложенного метода, со скоростью обработки всего 0,3 FPS.

Machine Translated by Google

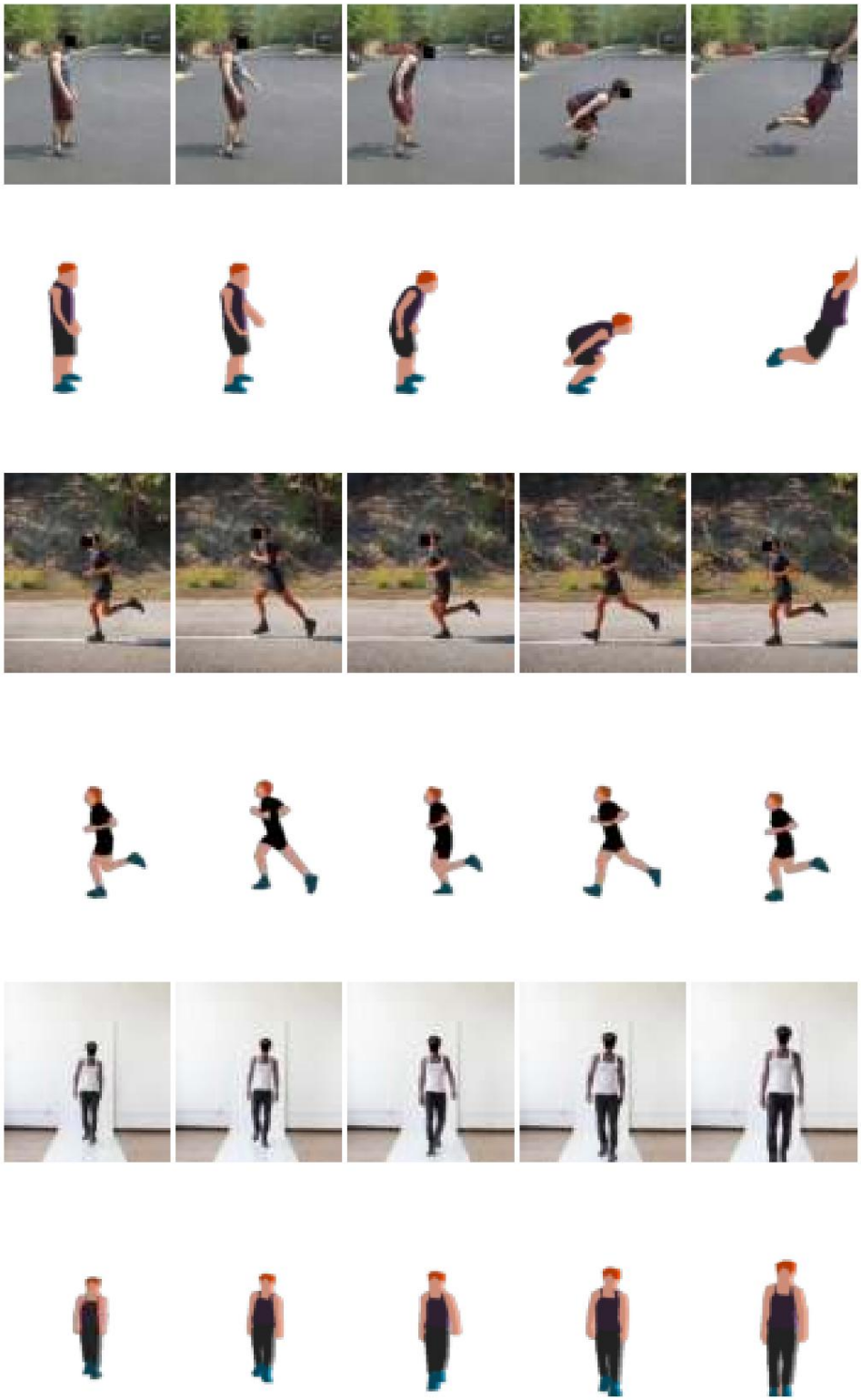


Рисунок 7. Пример результатов. Нечетные строки: входные видео (лица анонимизированы). Четные строки: результаты метода.

Machine Translated by Google

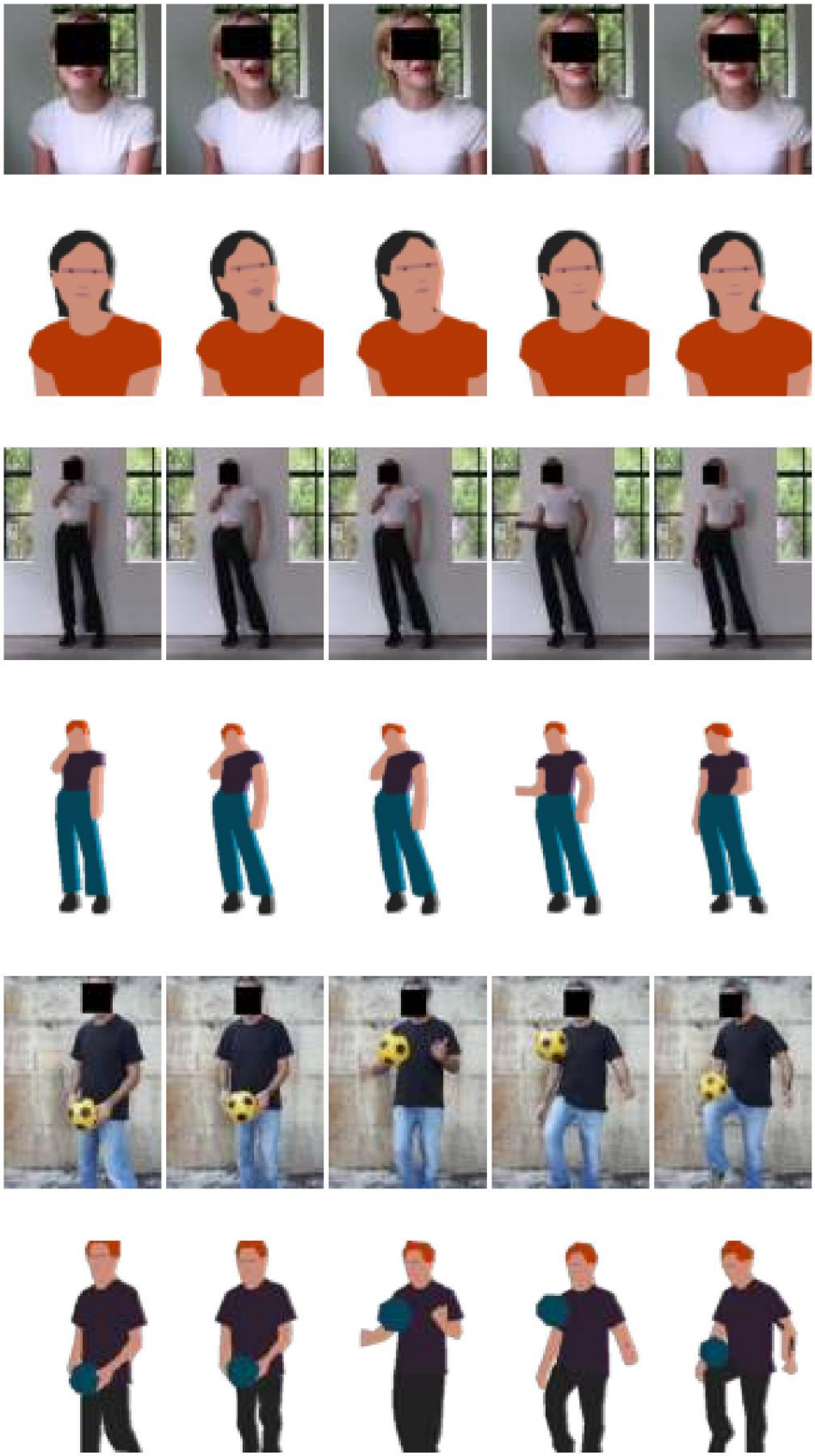


Рисунок 8. Пример результатов. Нечетные строки: входные видео (лица анонимизированы). Четные строки: результаты метода.

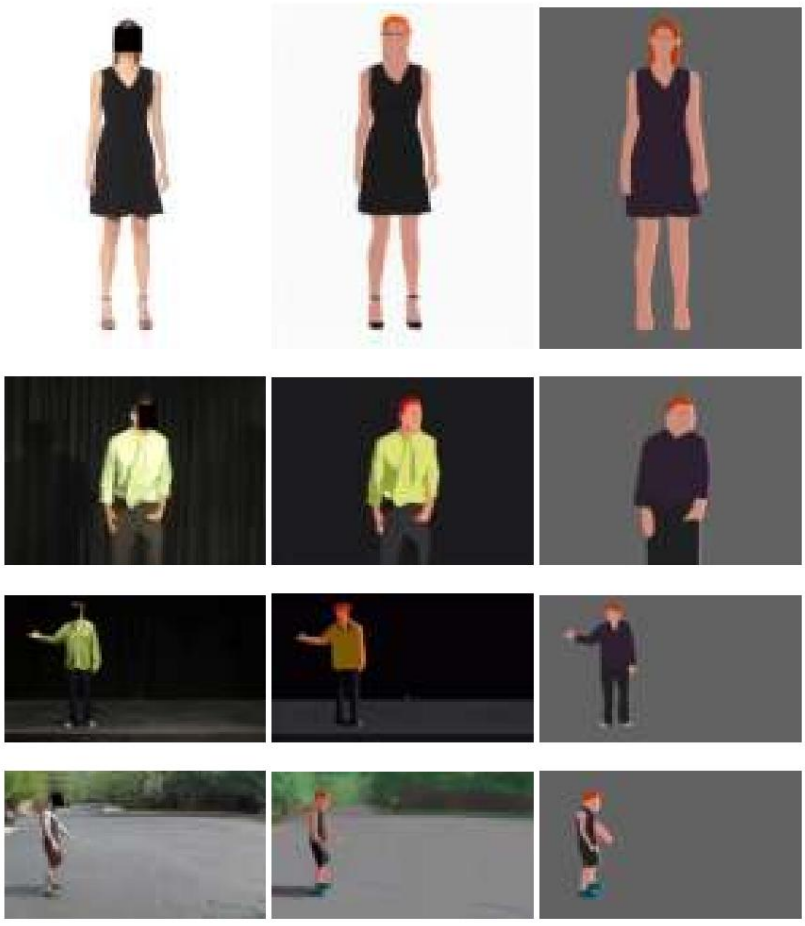
Machine Translated by Google



Рисунок 9. Пример синтезированного кадра анимации (справа) и связанный с ним исходный кадр из примера ввод видео с участием нескольких человек.



Рисунок 10. Примеры случаев сбоя. Верхний ряд: входные видео (видео 22, 23, 25 и 8 из тестового набора данных, лица были анонимизированы); нижний ряд: результаты.



(а) (б) (с)

Рисунок 11. Сравнение с генеративным конвейером: (а) входной кадр; (б) описан генеративный конвейер в [13]; (с) Лестер (предлагаемый метод).

Machine Translated by Google

Обсуждение: Полученные результаты весьма удовлетворительны и подтверждают, что последние достижения в области сегментации и отслеживания делают возможными новые приложения, в том числе в области синтеза изображений. Предложенный метод оказался эффективным методом создания ротоскопической анимации из видео. Обнаруженные ограничения не являются препятствием, поскольку ожидается, что этот метод будет использоваться в видеороликах, специально выбранных или созданных. Не важны и его вычислительные затраты, поскольку ожидается, что этот метод будет использоваться в задачах редактирования, не требующих выполнения в реальном времени. Этот метод в первую очередь предназначен для видеороликов с участием одного человека, но его можно адаптировать к видеороликам с участием нескольких человек, используя несколько параметризаций для подсказок сегментации и цветовых палитр.

5. Выводы

В этой работе мы представляем Lester — метод автоматического синтеза 2D- анимационных персонажей в стиле ретро из видео. Вместо того, чтобы решать проблему с помощью генерирующей модели изображения к изображению или с помощью инструментов оценки позы человека, она решается главным образом как проблема сегментации и отслеживания объектов. Субъективная оценка качества показывает, что этот подход осуществим и что метод может генерировать качественную анимацию из видео с разными позами и внешним видом, динамических кадров, частичных кадров и разнообразного фона.

Хотя это и не является целью, стоит отметить, что производительность метода сегментации и отслеживания менее стабильна в реальных условиях, особенно в отношении видео с низким разрешением. Однако ожидается, что этот метод будет использоваться для избранных или специально созданных видеороликов в полуконтролируемых средах.

Благодаря своим возможностям метод представляет собой эффективный инструмент, который значительно снижает стоимость создания ротоскопической анимации для видеоигр или анимационных фильмов. Благодаря своим характеристикам он проще и менее детерминирован, чем конвейеры на основе генеративных моделей, и более практичен и гибок, чем методы, основанные на оценке позы человека.

Финансирование: Это исследование частично финансируется Министерством науки и инноваций Испании по контракту PID2019-107255GB и программой SGR 2021-SGR-00478 правительства Каталонии.

Заявление о доступности данных: набор данных и результаты экспериментов общедоступны по адресу https://github.com/rtous/lester. (по состоянию на 20 июля 2024 г.). Код доступен на https://github. com/rtous/лестер-код (по состоянию на 20 июля 2024 г.).

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Сокращения

В данной рукописи использованы следующие сокращения:

Абсолютный рейтинг категории ACR
Генеративно-состязательные сети GAN
HOG Гистограмма ориентированных градиентов
Средний балл мнения MOS
Модель SAM Segment Anything

Ссылки

1. Кириллов А.; Минтун, Э.; Рави, Н.; Мао, Х.; Роллан, К.; Густавсон, Л.; Сяо, Т.; Уайтхед, С.; Берг, АС; Ло, Вайоминг; и др. Сегментируйте что угодно. arXiv 2023, arXiv:2304.02643.
2. Ченг, Ю.; Ли, Л.; Сюй, Ю.; Ли, Х.; Ян, З.; Ван, В.; Ян, Ю. Сегментируйте и отслеживайте что угодно. arXiv 2023, arXiv:2305.06558.
3. Дуглас, Д.Х.; Пойкер, Т.К. Алгоритмы уменьшения количества точек, необходимых для представления оцифрованной линии или ее карикатуры. В книге «Классика картографии: размышления о влиятельных статьях из журнала Cartographica»; John Wiley & Sons: Хобокен, Нью-Джерси, США, 2011 г.; стр. 15–28. [Перекрестная ссылка]
4. Чен Ю.; Лай, ЮК; Лю, YJ CartoonGAN: Генеративно-состязательные сети для карикатуризации фотографий. В материалах ЦВПР. Компьютерное общество IEEE, Солт-Лейк-Сити, Юта, США, 18–23 июня 2018 г.; стр. 9465–9474.
5. Чен Дж.; Лю, Г.; Чен, Х. AnimeGAN: новый легкий GAN для фотоанимации. Алгоритмы и приложения искусственного интеллекта, материалы 11-го Международного симпозиума, ISICA 2019, Гуанчжоу, Китай, 16–17 ноября 2019 г.; Шпрингер: Берлин/Гейдельберг, Германия, 2020 г.; стр. 242–256. [Перекрестная ссылка]

Machine Translated by Google

6. Лю, З.; Ли, Л.; Цзян, Х.; Джин, Х.; Ту, Д.; Ван, С.; Жа, З. Когерентная мультипликация видео без присмотра с перцептивной согласованностью движения. В материалах тридцать шестой конференции AAAI по искусственному интеллекту (AAAI-22), виртуально, 22 февраля – 1 марта 2022 г. [CrossRef]

7. Ромбах Р.; Блаттманн, А.; Лоренц, Д.; Эссер, П.; Оммер, Б. Синтез изображений высокого разрешения с помощью моделей скрытой диффузии. arXiv 2021, arXiv:2112.10752.

8. Эссер, П.; Кулал, С.; Блаттманн, А.; Энтезари, Р.; Мюллер, Дж.; Сайни, Х.; Леви, Ю.; Лоренц, Д.; Зауэр, А.; Бозель, Ф.; и др. Масштабирование выпрямленных трансформаторов потока для синтеза изображений высокого разрешения. arXiv 2024, arXiv:2403.03206.

9. Мэн, К.; Привет.; Сонг, Ю.; Сонг, Дж.; Ву, Дж.; Чжу, JY; Эрмон, С. SDEdit: управляемый синтез и редактирование изображений с помощью стохастических дифференциальных уравнений. В материалах Международной конференции по обучению представлений, виртуальной, 25–29 апреля 2022 г.

10. Чжан Л.; Рао, А.; Аграваля, М. Добавление условного управления к моделям диффузии текста в изображение. В материалах 2023 года Международная конференция IEEE/CVF по компьютерному зрению (ICCV), Париж, Франция, 2–6 октября 2023 г.

11. Сахария, К.; Хо, Дж.; Чан, В.; Салиманс, Т.; Флит, диджей; Норузи, М. Сверхразрешение изображений посредством итеративного уточнения. apXiv 2021, arXiv: 2104.07636.

12. Сахария, К.; Чан, В.; Чанг, Х.; Ли, К.; Хо, Дж.; Салиманс, Т.; Флит, Д.; Норузи, М. Палетт: Модели диффузии изображений. В материалах конференции ACM SIGGRAPH 2022, Нью-Йорк, штат Нью-Йорк, США, 7–11 августа 2022 г.; СИГРАФ '22.
[Перекрестная ссылка]

13. Ян, С.; Чжоу, Ю.; Лю, З.; Лой, СС Повторный рендеринг видео: перевод видео в видео с нулевым кадром. В материалах Материалы азиатской конференции ACM SIGGRAPH, Сидней, Новый Южный Уэльс, Австралия, 12–15 декабря 2023 г.

14. Джамриска, О. ebsynth: Быстрый синтез изображений на основе примеров и передача стилей. 2018. Доступно онлайн: https://github.com/.jamriska/ebsynth (по состоянию на 20 июля 2024 г.).

15. Брукс, Т.; Пиблс, Б.; Холмс, К.; ДеПью, В.; Го, Ю.; Цзин, Л.; Шнурр, Д.; Тейлор, Дж.; Луман, Т.; Луман, Э.; и др. Модели генерации видео как симуляторы мира. 2024 г. Доступно онлайн: https://openai.com/index/video-generation-models-as-world-simulators/ . (по состоянию на 20 июля 2024 г.).

16. Чен, З.; Ли, С.; Хак, М. Обзор Sora от OpenAI и ее потенциала для бесплатных игр на физическом движке и виртуальной реальности. Одобренная EAI Trans. Робот. 2024, 3. [CrossRef]

17. Лян, Д.; Лю, Ю.; Хуан, К.; Чжу, Г.; Цзян, С.; Чжан, З.; Гао, В. Video2Cartoon: создание 3D-мультфильма на основе футбольных трансляций видео. В материалах MULTIMEDIA '05, Сингапур, 11 ноября 2005 г.

18. Нго, В.; Цай, Дж. Преобразование 2D-видео о футболе в 3D-мультфильм. В материалах 10-й Международной конференции по контролю 2008 г. Автоматизация, робототехника и зрение, Ханой, Вьетнам, 17–20 декабря 2008 г.; стр. 103–107. [Перекрестная ссылка]

19. Венг, CY; Кёрлесс, Б.; Кемельмахер-Шлизерман, И. Фотопробуждение: 3D-анимация персонажей из одной фотографии. В материалах конференции IEEE/CVF по компьютерному зрению и распознаванию образов (CVPR), Лонг-Бич, Калифорния, США, 15–20 июня 2019 г.

20. Тоус, Р. Пиктонавт: Мультипликация фильма с использованием 3D-оценки позы человека и GAN. Мультимед. Инструменты Прил. 2023, 82, 1–15.
[Перекрестная ссылка]

21. Рубен Тус, младший; Игуал, Л. Завершение позы человека на частичных снимках камеры тела. Дж. Эксп. Теор. Артиф. Интел. 2023, 1–11. [Перекрестная ссылка]

22. Фишер Дж.; Ямришка, О.; Саймонс, Д.; Шехтман Э.; Лу, Дж.; Асенте, П.; Лукац, М.; Сикора, Д. Синтез стилизованных изображений на основе примеров Лицевые анимации. АКМ Транс. Граф. 2017, 36, 1–11. [Перекрестная ссылка]

23. Гатис, Луизиана; Экер, А.С.; Бетдж, М. Нейронный алгоритм художественного стиля. arXiv 2015, arXiv:1508.06576.

24. Он, К.; Чен, Х.; Се, С.; Ли, Ю.; Доллар, П.; Гиршик, Р. Автоэнкодеры в масках — масштабируемые устройства для обучения зрению. В материалах конференции IEEE/CVF 2022 года по компьютерному зрению и распознаванию образов (CVPR), Новый Орлеан, Луизиана, США, 18–24 июня 2022 года; стр. 15979–15988. [Перекрестная ссылка]

25. Досовицкий А.; Бейер, Л.; Колесников А.; Вайссенборн, Д.; Чжай, Х.; Унтертинер, Т.; Дегани, М.; Миндерер, М.; Хейгольд, Г.; Гелли, С.; и др. Изображение стоит 16x16 слов: преобразователи для распознавания изображений в масштабе. В материалах 9-й Международной конференции по обучающимся представлениям, ICLR 2021, виртуально, 3–7 мая 2021 г. Доступно онлайн: https://openreview.net/ (по состоянию на 20 июля 2024 г.).

26. Рэдфорд, А.; Ким, JW; Халласи, К.; Рамеш, А.; Го, Г.; Агарвал, С.; Састри, Г.; Аскелл, А.; Мишкин П.; Кларк, Дж.; и др. Изучение переносимых визуальных моделей под контролем естественного языка. В материалах 38-й Международной конференции по машинному обучению, виртуально, 18–24 июля 2021 г.; стр. 8748–8763.

27. Карион, Н.; Масса, Ф.; Синнаев, Г.; Усунье, Н.; Кириллов А.; Загоруйко С. Сквозное обнаружение объектов с помощью трансформаторов. В материалах Computer Vision –ECCV 2020: 16-я Европейская конференция, Глазго, Великобритания, 23–28 августа 2020 г.; стр. 213–229.
[Перекрестная ссылка]

28. Ян, З.; Ян, Ю. Особенности разделения при иерархическом распространении для сегментации видеообъектов. В Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Новый Орлеан, Луизиана, США, 28 ноября – 9 декабря 2022 г.

29. Лю, С.; Цзэн, З.; Арендовать.; Ли, Ф.; Чжан, Х.; Ян, Дж.; Ли, К.; Ян, Дж.; Су, Х.; Чжу, Дж.; и др. Заземление динозавра: объединение динозавра с заземленной предварительной тренировкой для обнаружения открытых объектов. arXiv 2023, arXiv:2303.05499.

30. Сузуки, С.; Бе, К. Топологический структурный анализ оцифрованных бинарных изображений по границам. Вычислить. График зрения. Изображение Процесс. 1985, 30, 32–46. [Перекрестная ссылка]

31. Каземи, В.; Салливан, Дж. Выравнивание лица за одну миллисекунду с помощью ансамбля деревьев регрессии. В материалах конференции IEEE 2014 г. по компьютерному зрению и распознаванию образов, Колумбус, Огайо, США, 23–28 июня 2014 г.; стр. 1867–1874. [Перекрестная ссылка]

Machine Translated by Google

32. Далал, Н.; Триггс, Б. Гистограммы ориентированных градиентов для обнаружения человека. В материалах конференции IEEE Computer Society 2005 г. по компьютерному зрению и распознаванию образов (CVPR'05), Сан-Диего, Калифорния, США, 20–26 июня 2005 г.; Том 1, стр. 886–893. [Перекрестная ссылка]

33. Сумро, К.; Замир, Арканзас; Шах, М. UCF101: Набор данных из 101 класса человеческих действий из видео в дикой природе. apXiv 2012, arXiv: 1212.0402.

34. Заблоцкая, П.; Сярохин А.; Чжао, Б.; Сигал, Л. DwNet: Плотная варп-сеть для генерации видео с участием человека с учетом позы. В Материалы 30-й Британской конференции по машинному зрению 2019, BMVC 2019, Кардифф, Великобритания, 9–12 сентября 2019 г.; п. 51.

35. Кей, В.; Каррейра, Дж.; Симонян, К.; Чжан, Б.; Хиллер, К.; Виджаянарасимхан, С.; Виола, Ф.; Грин, Т.; Назад, Т.; Нацев П.; и др. Набор видеоданных о кинетике человеческих действий. arXiv 2017, arXiv:1705.06950.

36. Тоус, Р. Лестер. Домашняя страница проекта. Доступно онлайн: https://github.com/rtous/lester. (по состоянию на 20 июля 2024 г.).

37. МСЭ. Методы субъективной оценки качества видео для мультимедийных приложений. Рекомендация МСЭ-Т P.910. 2008. В наличии онлайн: https://www.itu.int/rec/T-REC-P.910 (по состоянию на 20 июля 2024 г.).

38. МСЭ. Терминология среднего балла мнений (MOS). Рекомендация МСЭ-Т P.800.1. Методы объективной и субъективной оценки качества речи и видео. 2016. Доступно онлайн: https://www.itu.int/rec/T-REC-P.800.1. (по состоянию на 20 июля 2024 г.).

39. Руис, Н.; Ли, Ю.; Джампани, В.; Притч, Ю.; Рубинштейн, М.; Аберман, К. DreamBooth: Точная настройка моделей диффузии текста в изображение для создания предметно-ориентированной генерации. arXiv 2023, arXiv:2208.12242.

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Решение Swarm Intelligence для прибыльного использования нескольких транспортных средств Проблема с получением и доставкой

И. Альхуджайлан 1* и Манар И. Хосни Абир 2

1 Факультет информационных технологий, Компьютерный колледж, Университет Касима, Бурайда 51452, Саудовская
2 Аравия. Факультет компьютерных наук, Колледж компьютерных и информационных наук, Университет короля Сауда, Эр-Рияд 11362, Саудовская Аравия; mifawzi@ksu.edu.sa
* Переписка: a.alhujaylan@qu.edu.sa

Аннотация: Приложения для доставки переживают значительный рост, требуя эффективных алгоритмов для координации перевозок и получения прибыли. Одной из проблем, которая учитывает цели приложений для доставки, является проблема выгодного получения и доставки нескольких транспортных средств (MVPPDP). В этой статье мы предлагаем восемь новых метаэвристик для улучшения первоначальных решений для MVPPDP на основе известного алгоритма роевого интеллекта Искусственная пчелиная колония (ABC): K-means-GRASP-ABC(C)S1, K-means- GRASP-ABC(C)S2, Модифицированные K-средства-GRASP-ABC(C)S1, Модифицированные K-средства-GRASP-ABC(C)S2, ACO-GRASP-ABC(C)S1, ACO-GRASP-ABC(C)S2, ABC(S1) и ABC(S2). Все методы в большинстве случаев показали превосходную производительность с точки зрения времени обработки. Например, для 250 клиентов среднее время работы алгоритмов составило 75,9, 72,86, 79,17, 73,85, 76,60, 66,29, 177,07 и 196,09, что было быстрее, чем у современных методов, требующих 300 с. . Более того, все предложенные алгоритмы хорошо зарекомендовали себя на экземплярах небольшого размера с точки зрения прибыли, достигнув тринадцати новых лучших решений и пяти равных решений наиболее известным решениям. Однако алгоритмы немного отстают в экземплярах среднего и большого размера из-за жадной рандомизированной стратегии и GRASP, которые использовались на этапе разведки. Более того, наши алгоритмы отдают приоритет минимальным решениям и итерациям для сокращения времени обработки в повседневных приложениях мобильной коммерции, а уменьшение количества итераций и размера популяции снижает вероятность получения решения хорошего качества.

Ключевые слова: задача выгодной погрузки и доставки нескольких транспортных средств; Искусственная пчелиная семья; роевой интеллект; метаэвристические алгоритмы

check for updates

Цитирование: Альхуджайлан, А.И.; Хосни, Мичиган Роевое интеллектуальное решение для задачи рентабельной погрузки и доставки нескольких транспортных средств . Алгоритмы 2024, 17, 331. https://doi.org/10.3390/a17080331

Академический редактор: Витторио Маньеццо

Поступила: 2 июля 2024 г.
Пересмотрено: 17 июля 2024 г.
Принято: 25 июля 2024 г.
Опубликовано: 31 июля 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

За последние пять лет приложения для доставки значительно выросли, особенно приложения для доставки еды, где COVID-19 продвинул отрасль вперед на несколько лет, поскольку миллионы людей, находящихся в изоляции, заказывали еду и другие основные услуги онлайн.

Платформа доставки продуктов Instacart сообщила, что ее цели на 2022 год были достигнуты за третью неделю карантина. В 2022 году Китай стал крупнейшим рынком доставки еды с ошеломляющей рыночной стоимостью в 42,5 миллиарда долларов США. Ожидается, что к 2029 году объем рынка приложений для доставки еды в целом достигнет 165 миллиардов долларов США [1]. Более того, 85% американцев размещают онлайн-заказы на еду через приложения для заказа еды. Более того, 80% ресторанов в США предоставляют клиентам удобство мобильного заказа [2]. Аналогичным образом, приложения для доставки еды в Саудовской Аравии предоставляют мгновенные услуги и оптимизированный пользовательский интерфейс, позволяющий людям наслаждаться В результате периодически появляются новые приложения, которые соперничают за предоставление услуг, удобства и комфорта более чем 35 миллионам человек. Примерами таких приложений, которые стали очень популярными в Саудовской Аравии, являются Mrsool, Uber Eats, Talabat, Hunger Station, Ninja, Jahez, To You, The Chifz, Shgardi и NANA [3]. Растущий спрос на приложения для доставки еды одновременно увеличил нагрузку на предприятия по доставке, которым тр

Machine Translated by Google

оперативные алгоритмы координации перевозки грузов с целью получения прибыли и гарантии удовлетворенности клиентов. Одной из проблем, представленных в литературе, которая имитирует цель приложений для доставки еды, увеличение прибыли, достижение удовлетворенности пользователей и сокращение затрат, является проблема выгодного забора и доставки с участием нескольких транспортных средств (M

Ганстерер и др. [4] представили MVPPDP, который является подтипом задачи выборочного захвата и доставки (SPDP) и новым вариантом хорошо известной задачи захвата и доставки (PDP). В рамках однодневного горизонта планирования MVPPDP стремится определить наиболее выгодные маршруты за счет минимизации командировочных расходов и максимизации сбора доходов. MVPPDP — это статическая проблема, характеризующаяся стандартизированным парком транспортных средств, центральным депо, набором клиентов и заранее определенным количеством запросов и продуктов. Каждый запрос определяется парой клиентов, состоящей из клиента самовывоза и клиента доставки. Таким образом, продукты передаются от клиента самовывоза к клиенту доставки, чтобы генерировать поток доходов для этой операции.

Реальные приложения MVPPDP включают мобильные приложения для служб доставки еды и компаний по доставке.

MVPPDP — это одно из приложений для решения задач комбинаторной оптимизации (CO), которое привлекает особое внимание исследователей в области информатики и исследования операций. Решение проблемы CO в основном требует поиска оптимального или наилучшего возможного решения из ограниченного набора возможных решений-кандидатов. Для решения задач CO можно использовать многочисленные алгоритмы , и эти алгоритмы можно разделить на две категории: точные алгоритмы, которые гарантированно получают оптимальные решения за адекватное время , и приближенные алгоритмы, обычно называемые эвристикой или метаэвристикой, которые не гарантированно предоставит оптимальное решение, но обеспечит хорошее решение в разумные сроки. Слова «эвристика» и «мета» принадлежат греческому языку, где эвристика происходит от «найти» или ng «to heuriskein», что означает ng «к нг» «открывать» новые методы и стратегии за пределами, на верхнем уровне. Метаэвристика ng to решать задачи, при этом префикс мета означает x метаметоды разделить по количеству решений, используемых одновременно, можно на две категории: метаэвристика на основе одного решения (S-метаэвристика) и метаэвристика на основе популяции (P-метаэвристика). В S-метаэвристике основное внимание уделяется модификации и улучшению единственного возможного решения. С другой стороны, P-метаэвристика сохраняет и улучшает несколько возможных решений вместо одного решения. Они основаны на итеративном улучшении решений в популяции и развитии всей популяции до тех пор, пока не будет достигнут набор хороших решений. Ориентированные на исследование особенности P-метаэвристики сделали ее лучше, чем S-метаэвристика, в диверсификации пространства поиска. Самыми известными примерами P-метаэвристики являются алгоритмы, основанные на естественном поведении, такие как генетические алгоритмы и роевой интеллект (SI) [5].

Как правило, для получения хорошего решения проблемы следует использовать два этапа : построение решения и улучшение решения. Построение решения включает в себя создание одного или нескольких начальных возможных решений, которые можно рассматривать как отправную точку для исследования пространства поиска. Улучшение решения основано на изменении исходных решений для их улучшения в соответствии с заранее заданными стратегиями до тех пор, пока не будет достигнуто высококачественное решение или не будут выполнены критерии остановки (например, максимальное количество итераций или определенное количество времени) [5].].

В связи с тем, что MVPPDP является NP-сложной задачей (поскольку это подтип SPDP, который является частным случаем PDP) [4], метаэвристические алгоритмы могут использоваться для эффективного поиска оптимальных решений в разумном количестве. времени, особенно при проблемах среднего и большого размера. этой статье мы представляем новые алгоритмы для решения этой проблемы с целью улучшения первоначального решения, предложенного в [6,7]. Наш подход основан на использовании алгоритма искусственной пчелиной семьи (ABC), который является одним из широко признанных алгоритмов искусственного интеллекта, известных как SI. С помощью ABC были реализованы различные стратегии , каждая из которых продемонстрировала эффективность в ускорении процедуры поиска и сокращении времени обработки.

Основные цели этой статьи: (1) помочь приложениям и компаниям по доставке еды оптимизировать свои услуги посредством ускоренного отбора клиентов и

Machine Translated by Google

планирование маршрута с целью максимизации прибыли и удовлетворенности клиентов; (2) эффективное транспортное планирование, которое может помочь сберечь энергетические ресурсы за счет снижения вредного воздействия газовых выбросов на окружающую среду; и что (3) в некоторых проверенных случаях предлагаемые подходы достигают результатов, превосходящих самые современные методы, описанные в литературе по MVPPDP.

Остальная часть статьи организована следующим образом: обзор некоторых связанных работ представлен в разделе 2. В разделе 3 представлено формальное определение проблемы. Предлагаемый метод подробно описан в разделе 4. Раздел 5 иллюстрирует экспериментальные результаты. Наконец, выводы и будущая работа представлены в разделе 6.

2. Сопутствующие исследования

MVPPDP классифицируется как тип SPDP, и его цель — максимизировать прибыль при минимизации командировочных расходов за счет посещения только прибыльных пар клиентов. Достижение благоприятного маршрута, который максимизирует разницу между общим полученным доходом и общими затратами на перемещение, является целью как MVPPDP, так и задачи прибыльного тура (PTP). Основное отличие состоит в том, что в PTP маршрут транспортного средства не подвергается каким-либо ограничениям, включая, помимо прочего, максимальное время поездки, ограничения вместимости транспортного средства или ограничения приоритета [8]. В научной литературе мало исследований, посвященных ПТП [9]. Следовательно, мы начнем с представления исследований, касающихся MVPPDP.

Ганстерер и др. [4] представили первые четыре алгоритма. Метаэвристика поиска переменных окрестностей (VNS) была реализована в [10]. На этапе построения VNS использовались жадная эвристика построения (C1) и двухэтапная эвристика наименьшего вставки (C2) . На этапе улучшения были реализованы последующие операторы соседства : попарный обратный обмен, попарный прямой обмен, прямая вставка, перемещение пар, обратная вставка, обмен между обходами, вставка между обходами, межфиктивная вставка, межфиктивный обмен. и обмен центром тяжести. Авторы реализовали два подхода к применению операторов соседства: последовательный метод (Seq), который использует заранее определенную последовательность окрестностей, и самоадаптивный метод (Sea), который настраивается в зависимости от статуса поиска для определения последовательности окрестностей. . Чтобы сравнить предложенный метод с альтернативным алгоритмом, в конечном итоге был реализован алгоритм управляемого локального поиска (GLS), основанный на м
Оценка предложенного алгоритма была проведена на 36 рандомизированных экземплярах различного размера. По качеству решения результаты экспериментов показали, что оба варианта BHC превосходят GLS.

Хаддад [11] представил новый алгоритм решения MVPPDP. В ожидании решения проблемы выгодного получения и доставки они предложили алгоритм, известный как IPPD, который был получен на основе комбинации повторяющегося локального поиска и спуска окрестности случайной величины. В ходе поиска алгоритм IPPD не ограничивался принятием только жизнеспособных решений. Для построения исходного решения использовалась эвристика C1, использованная Ganster et al. [4] был реализован. Для улучшения решения было реализовано несколько перемещений соседей: замена пары или сдвиг пары, замена блока или сдвиг блока, сдвиг забора или доставки, замена между блоками или сдвиг между блоками, замена центра тяжести, оператор вставки и оператор удаления, а также межблоковый сдвиг. -парный обмен или межпарный сдвиг. В контрольных примерах, предложенных Ganster et al. В [4] предложенный алгоритм был оценен. Он продемонстрировал свою эффективность в рассмотрении мелких и средних дел, определив новые оптимальные решения для шести дел. Однако в больших случаях производительность предложенного алгоритма была недостаточной.

В Alhujaylan и Hosny [6] фаза построения широко признанной процедуры жадного рандомизированного адаптивного поиска (GRASP) использовалась для разработки предварительных решений для MVPPDP. Они использовали тот же набор данных, который был создан [4]. Они провели сравнение производительности метода и двух эвристик построения, которые были ранее описаны в литературе для построения начальных решений MVPPDP . Результаты экспериментов показали, что предложенный метод превосходит по

Machine Translated by Google

эвристика альтернативного построения, особенно в случаях небольшого размера, где были получены восемь новых лучших начальных решений задачи.

В [7] Алхуджайлан и Хосни представили новую эвристику для построения начальных решений для MVPPDP. Эвристика основана на первой кластеризации пространства поиска MVPPDP с использованием трех методов кластеризации: K-средних, адаптивных K-средних и оптимизации муравьиной колонии (ACO). Затем модифицированная версия GRASP использовалась для построения исходного решения MVPPDP на основе результатов каждого алгоритма кластеризации. Предложенное решение было применено к набору данных, представленному исследователями [4]. Они сравнили свои результаты с результатами других строительных эвристик, которые ранее использовались для MVPPDP. Результаты экспериментов доказали эффективность предложенных алгоритмов как по качеству решения, так и по времени обработки.

В [12] авторы включили в MVPPDP временные окна с неопределенным временем прохождения. Исследователи использовали часть базы данных, созданную [4], которая включала двадцать и пятьдесят запросов, соответствующих двум и трем автомобилям соответственно.

Цель клиента — найти решение, оптимизирующее чистую прибыль, которая рассчитывается как разница между полученным доходом, стоимостью маршрута и расходами, понесенными из-за нарушений временного окна. Они приняли во внимание широкие временные окна и стандартные временные окна. В предположении, что время в пути является независимой случайной величиной с гамма-распределением и с нейтральной к риску точки зрения, они продемонстрировали, что ожидаемые затраты на пунктуальность и раннее прибытие могут быть рассчитаны в закрытой форме. В качестве метода решения авторы предложили использовать подход поиска по графу . Эксперименты показали, что точные решения можно получить за разумное время.

3. Определение проблемы

MVPPDP — это заранее определенная статическая задача, состоящая из нескольких составляющих. Постулируется , что за прием заявок клиентов отвечает централизованное депо. Эти запросы включают клиентов самовывоза и доставки попарно. Чтобы удовлетворить эти требования, мы управляем парком идентичных транспортных средств, которые доставляют множество идентичных продуктов от определенной группы клиентов-самовывозов к соответствующим клиентам-доставщикам, тем самым получая определенную прибыль. Необходимо учитывать следующие ограничения MVPPDP: • Ограничение на сопряжение предполагает, что каждому пользователю назначается предварительно определенная пара заказ (самовывоз и доставка).
• Ограничение приоритета: приоритет должен быть отдан посещению клиента самовывоза, а не клиента доставки. • Ограничение времени поездок: на каждое транспортное средство распространяется ежедневное ограничение времени поездки, который не должен быть превышен при обслуживании клиентов.

Одним из ограничений является вместимость каждого транспортного средства, которую нельзя превышать во время сбор товара от клиентов самовывоза. • Каждое транспортное средство должно отправляться со склада и прибывать на него с порожним грузом. • Для одного клиента разрешено одно посещение.

Формально MVPPDP можно определить следующим образом [10]: Пусть G = (V, A) — граф, в котором V = {0, . . . , 2n + 1} определяет множество вершин, где вершина (0, 2n + 1) соответствует центральному депо, а остальные вершины представляют клиентов самовывоза P = {1, . . . , n} и заказчики доставки D = {n + 1, . . . , 2н}. A = {(i, j) : i, j V, i≠ j} определяет набор дуг, где каждой дуге соответствует неотрицательная стоимость маршрутизации cij. Каждая вершина доставки i имеет доход ri, который можно получить при ее посещении. Кроме того, каждая вершина i имеет предложение qi (вывоз, qi > 0) или спрос (доставка, qn+i = qi). Вначале на складе нет ни спроса, ни предложения (q0 = 0). Кроме того, каждая вершина i имеет продолжительность обслуживания di . Имеется набор транспортных средств K = {1, . . . , м}, и каждое транспортное средство имеет грузоподъемность C и максимальное время поездки T.

Machine Translated by Google

Целевая функция (OF) MVPPDP направлена на максимизацию общей прибыли путем вычитания общих транспортных расходов из общих доходов. Его можно сформулировать следующим образом:

$$\text{Развернуть} \quad \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} ri \quad cij \; xijk, \tag{1}$$

Математическая модель MVPPDP может быть сформулирована как в [4,6] следующим образом:

$$\sum_{i \in V} \sum_{k \in K} xijk \quad 1 \; j \quad V \tag{2}$$

$$\sum_{j \in V} \sum_{k \in K} xijk \quad 1 \; i \quad V \tag{3}$$

$$xi0k = 0 \; i \quad V, k \quad K \tag{4}$$

$$x2n+1,jk = 0 \; j \quad V, k \quad K \tag{5}$$

$$\sum_{i \in V} xijk \quad xjik = 0 \; j \quad V \{0, 2n + 1\}, k \quad K \tag{6}$$

$$\sum_{j \in V} xijk \quad xn+i,jk = 0 \; i \quad P, k \quad K \tag{7}$$

$$x0jk = \sum_{j=V} xi,2n+1,k = 1 \; k \quad K \tag{8}$$

$$xijk = 1 \quad Qjk = Qik + qj \; i \quad V, j \quad V \{0\}, k \quad K \tag{9}$$

$$Qik \quad C \; i \quad V, k \quad K \tag{10}$$

$$Q0k = 0 \; k \quad K \tag{11}$$

$$Bik \quad Bn+i,k \quad я € P \tag{12}$$

$$Bjk \quad Xijk \; Bik + di + tij \quad я € V, j € V, к € K \tag{13}$$

$$B2n+1,k \quad T \; k \quad K \tag{14}$$

$$Bok = 0 \; k \quad K \tag{15}$$

$$xijk € 0, 1 \; i € V, j € V, k € K \tag{16}$$

$$Qik, Bik \quad 0 \; i \quad V, k \quad K \tag{17}$$

Ограничения (2) и (3) требуют, чтобы каждая вершина посещалась не более одного раза. Ограничения (4) и (5) не позволяют любому транспортному средству въехать в депо отправления (начальная точка) или покинуть депо назначения (конечная точка). Ограничение (6) указывает на сохранение потока. Ограничение (7) гласит, что каждая пара клиентов (самовывоз и доставка) должна обслуживаться одним и тем же транспортным средством. Ограничение (8) требует, чтобы все транспортные средства запустились и вернулись в депо. Ограничение (9) гласит , что превышение грузоподъемности транспортного средства не допускается. Ограничение (10) указывает, что загрузка транспортного средства ограничена C. Ограничение (11) указывает, что все транспортные средства начинают работу с пустым грузом. Ограничение (12) указывает на то, что клиента, осуществляющего доставку, нельзя посетить раньше клиента, осуществляющего самовывоз. Ограничение (13) гласит, что самое раннее время начала обслуживания в вершине j определяется путем сложения времени начала обслуживания в вершине i ко времени обслуживания в и времени в пути между i и j. Ограничение (14) указывает, что максимальное время для каждого транспортного средства ограничено T. Ограничение (15) указывает, что каждое транспортное средство начинает работу в депо в нулевой момент времени. Наконец, переменные решения определяются формулами (16) и (17).

4. Предлагаемый метод B

данном исследовании мы использовали алгоритм искусственной пчелиной семьи (ABC), который был впервые предложен Карабогой [13]. Это имитирует совместное и коллективное поведение пчелиной семьи при поиске пищи. В алгоритме ABC существует три разных типа пчел:

Machine Translated by Google

пчелы-разведчики, пчелы-работники и пчелы-наблюдатели. Каждый тип играет различную роль в разведке и эксплуатации источников пищи. Источник пищи представляет собой решение проблемы , а лучшие решения принадлежат источникам пищи с большим количеством нектара. Алгоритм ABC работает следующим образом: первоначально пчелы-разведчики случайным образом распределяются по пространству поиска для поиска первоначальных источников пищи. Затем работающих пчел отправляют на освоение обнаруженных пчелами-разведчиками мест, при этом каждая работающая пчела закрепляется за одним источником питания. После этого нанятые пчелы начинают обыскивать окрестности, чтобы найти поблизости лучший источник пищи. Найдя лучший источник пищи, работающие пчелы оставляют все предыдущие источники пищи и используют только лучшие. Затем эти нанятые пчелы возвращаются в улей и делятся своей информацией об обнаруженных источниках пищи с пчелами-зрителями посредством так называемог Решение о том, будут ли пчелы-наблюдатели следовать за наемными пчелами, зависит от богатства источников пищи. Если пчелы-наблюдатели решают следовать за ними, они становятся наемными пчелами и повторяют процедуру нанятых пчел. После нескольких итераций источник пищи может быть исчерпан; в этот момент нанятые пчелы становятся пчелами-разведчиками и начинают случайным образом искать новый источник пищи взамен предыдущего. Блок -схема алгоритма ABC показана на рисунке 1 [14].

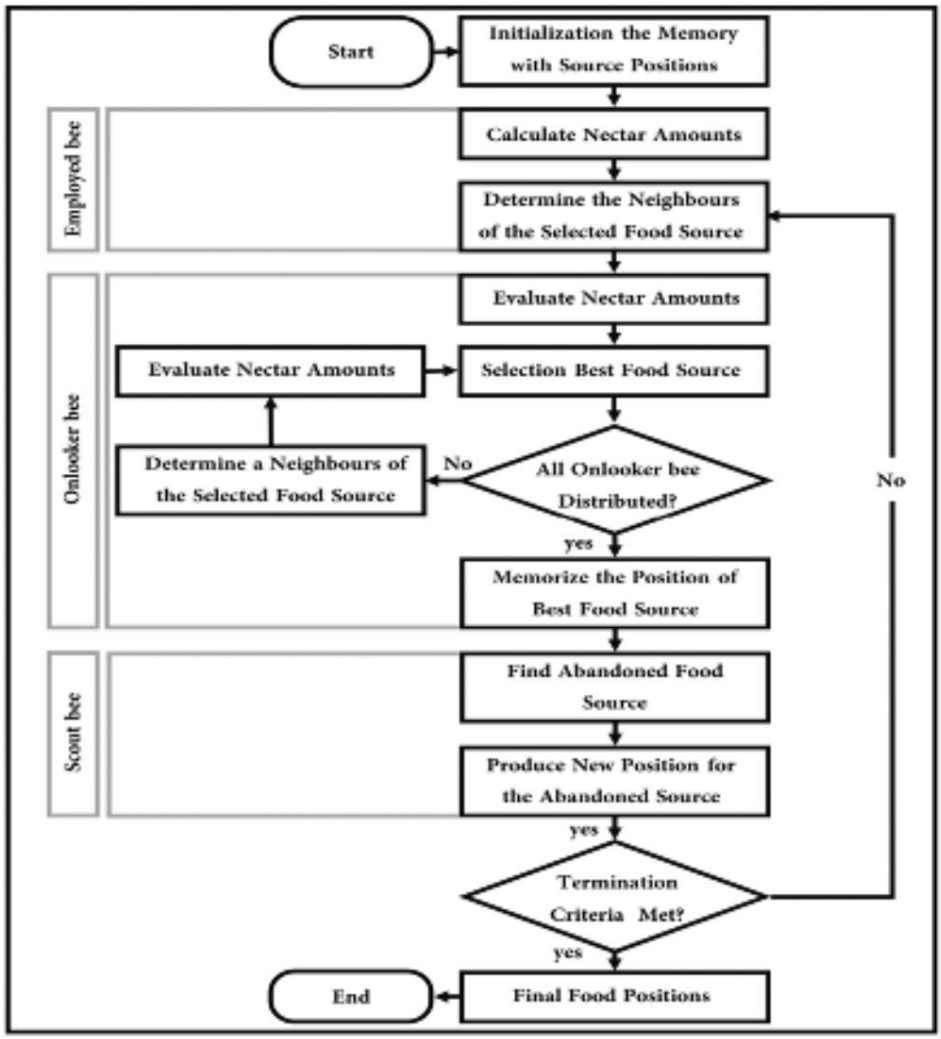


Рисунок 1. Блок-схема алгоритма искусственной пчелиной семьи [14].

Эвристика ABC использовалась для решения нескольких вариантов VRP, например, использованная Szetoa et al. [15] для решения проблемы маршрутизации грузовых транспортных средств, Wu et al. [16] для решения VRP с временными окнами, а Yao et al. [17] для решения периодической VRP.

Machine Translated by Google

В данной статье мы стремимся улучшить исходные решения, предложенные в [7]. Существует множество метаэвристик, которые использовались для повышения качества решений, таких как локальный поиск (LS), имитация отжига (SA) и VNS. В нашем исследовании мы также используем метаэвристику, которая сочетает в себе жадный и рандомизированный поиск и основана на поведении пчел, а именно ABC. Основной причиной выбора ABC, а не других алгоритмов пчел, является его популярность: количество исследований, в которых использовался ABC по сравнению с другими вариантами, достигает 54%, как сообщается в опросе Карабоги [18]. Для решения MVPPDP предлагаются две версии ABC: ABC без кластеризации (ABC) и ABC с кластеризацией (ABC(C)), как будет объяснено далее.

4.1. ABC без кластеризации (ABC)

В этой части исследования мы стремились улучшить первоначальные решения, которые были достигнуты с использованием улучшенного GRASP в [7]. Метаэвристика ABC состоит из трех фаз: фаза задействованной пчелы, фаза пчелы-наблюдателя и фаза пчелы-разведчика. Основная цель первых двух этапов — повысить интенсификацию пространства поиска для получения хороших решений, а последний этап направлен на увеличение диверсификации и предотвращение застревания в локальных оптимумах. Детали предлагаемого нами алгоритма проиллюстрированы ниже.

1. Фаза трудоустройства пчел: после

назначения первоначальных решений нанятым пчелам они начинают поиск окрестности, чтобы найти лучшие решения поблизости, случайным образом выбирая один из набора операторов соседства. Соседские операторы стремятся внести небольшие изменения в решение, чтобы улучшить его. Таким образом, для каждого решения случайным образом выбирается один маршрут для применения одного оператора, который также выбирается случайным образом. Набор операторов соседства описан ниже: • Переместить только клиента самовывоза: случайным образом выберите одного клиента самовывоза и измените его положение так, чтобы новое положение самовывоза не должно было располагаться после клиента доставки .

• Переместить только клиента доставки. Случайным образом выберите одного клиента доставки и переместите его позицию так, чтобы новая позиция доставки не должна была предшествовать клиенту самовывоза.
• Перемещение пары клиентов: случайным образом выберите пару клиентов и переместите их позиции , сохраняя ограничение приоритета (т. е. получение перед доставкой). • Поменяйте местами две пары клиентов: случайным образом выберите две пары клиентов и поменяйте их местами. • Оператор двух вариантов: выберите две последовательные пары клиентов (последовательная пара — это та, в которой за клиентом-самовывозом непосредственно следует клиент-доставщик) и измените порядок ребер между парами, учитывая ограничение приоритета. • Удалите одну пару клиентов случайным образом и вставьте непосещенную: удалите пару клиентов случайным образом, выберите пару непосещенных клиентов, которая имеет высокий коэффициент вставки (IR), используя уравнение (18), и вставьте ее в лучшую позицию на ма

$$IRcustomerpair = \frac{rcustomerpair}{c(депо,ПК)}, \quad (18)$$

где r — доход пары клиентов, а c(depote,PC) — расстояние между депо и клиентом самовывоза этой пары. • Удалите пару

клиентов с низким IR и вставьте непосещенную пару с высоким IR: вычислите IR для всех посещенных пар, используя уравнение (18), и удалите пару клиентов с наименьшим значением IR. Затем пересчитайте IR для всех пар непосещенных клиентов, выберите одну с самым высоким значением IR и вставьте ее в лучшую позицию на маршруте.
• Вставьте пару клиентов с высоким IR: вычислите IR для всех пар непосещенных клиентов, выберите одну с самым высоким значением IR и вставьте ее в лучшее положение на маршруте

После применения выбранного оператора окрестности, если новое решение S' осуществимо, пригодность S' вычисляется с использованием уравнения (1) и сравнивается с пригодностью старого решения S. Чтобы избежать локальных оптимумов, мы решили использовать Алгоритм демона (DA) скорее

Machine Translated by Google

чем простой алгоритм локального поиска, поскольку DA имеет возможность принять возможное неулучшенное новое решение. DA представляет собой вариант SA, представленный Крейцем [19]. Функция приемки DA основана на значении демона (кредита), который инициализируется вначале с заданным значением D. В случае, когда новое решение S' не является улучшением, разница (E) между соответствие S и S' сравнивается со значением демона, и S' принимается, если эта разница меньше или равна D. Затем, если S' принимается, оно запоминается, а значение D уменьшается; в противном случае старое решение S запоминается, значение D увеличивается, а счетчик неулучшения для этого решения увеличивается на единицу. DA проще и быстрее, чем SA, поскольку последний требует больше вычислительного времени для генерации случайного числа и экспоненциальной функции. Кроме того, он содержит только один параметр, который необходимо настроить. DA показал себя эффективным алгоритмом при решении ряда реальных задач, достигнув хороших результатов по качеству решения и времени по сравнению с CA [19].

2. Фаза наблюдателя:

Только решения с хорошими значениями OF имеют высокую вероятность быть назначенными агентам-наблюдателям. Было предложено несколько стратегий выбора решений, таких как выбор колеса рулетки, турнирный выбор, стохастическая универсальная выборка и ранговый отбор [20]. Мы приняли наиболее популярную стратегию выбора колеса рулетки , в которой вероятность выбора Pi вычисляется следующим образом:

$$\text{Пи} = \frac{\text{ОФи}}{\sum_{j=1}^{H} \text{OFj}}, \quad (19)$$

где OFi — целевая функция для решения i, а $\sum_{j=1}^{H} OFj$ — сумма значений целевой функции для всех решений в совокупности. После того, как решение выбрано, один из маршрутов решения выбирается случайным образом, и все операторы окрестности, используемые в используемой пчелиной фазе, последовательно применяются к нему, а затем пригодность нового решения измеряется с помощью уравнения (1). Чтобы ускорить процесс интенсификации и сосредоточиться только на перспективных областях пространства поиска, критерий приемки основан на простом алгоритме LS (локального поиска). То есть, если S' лучше, чем S, запомните S', и нет необходимости дополнять остальные операторы; в противном случае попробуйте следующий оператор. Если старое решение не было улучшено ни одним оператором, старое решение запоминается, а счетчик неулучшения этого решения увеличивается на единицу.

3. Фаза разведки. Фаза

разведки направлена на повышение диверсификации и расширение пространства решений, предотвращая тем самым застревание в локальных оптимумах. Фаза разведки начинается, когда счетчик неулучшений решения Si достигает заранее определенного количества попыток, называемого пределом. Предел относится к количеству попыток, которые были предприняты для улучшения решения Si , как на рабочей фазе, так и на фазе наблюдателя соответственно. Три важных действия должны быть выполнены, когда счетчик неулучшения для решения S достигает предела: (1) очистить счетчик неулучшения, (2) очистить решение Si и (3) восстановить решение Si .

Подходящая стратегия реконструкции играет важную роль на этапе поиска, где она либо способствует повышению качества решения, либо оставляет решение неизменным. Таким образом, мы предложили две стратегии восстановления решения Si . Детали этих эвристик реконструкции представлены следующим образом: Первая стратегия (ABC(S1)): одна из следующих трех

- эвристик выбирается случайным образом:

1. Эвристика случайной вставки:

Пока ограничение по времени не нарушается, пары клиентов выбираются случайным образом и вставляются непосредственно в маршрут. За каждым клиентом самовывоза следует клиент доставки; следовательно, нет необходимости проверять вместимость транспортного средства и ограничения приоритета.

2. Жадные и рандомизированные эвристики:

Machine Translated by Google

Пока ограничение по времени не нарушается, пары клиентов выбираются случайным образом и вставляются в лучшие позиции маршрута. Помимо ограничения по времени, не должны нарушаться ограничения по вместимости транспортного средства и приоритету.

3. Жадная эвристика вставки:

Коэффициент включения рассчитывается для всех пар клиентов с использованием уравнения (18). Пара клиентов, имеющая наибольшее значение IR, затем вставляется в маршрут в лучшей позиции. Ограничения по времени, вместимости транспортного средства и приоритету не должны нарушаться. • Вторая стратегия (ABC(S2)): Эта стратегия работает точно так же, как GRASP в [7].

После того, как решение Si реконструировано, OF вычисляется с использованием уравнения (1), а Si запоминается как новое решение и используется для повторного запуска процесса улучшения с использованием фаз задействованной пчелы и пчелы-наблюдателя.

Схема этапа маршрутизации нашего ABC представлена в алгоритме 1, где каждое обозначение определяется следующим образом: Размер группы представляет размер популяции; Max Iter: максимальное количество итераций; Num Vehicles: количество транспортных средств; США: пары клиентов, которые не были обслужены; E: разница между значениями OF для старого и нового решения; D : значение демона; и Неулучшение : счетчик для подсчета количества неудачных попыток улучшить решение.

Алгоритм 1: Псевдокод предложенного ABC.

Поп = {S1 , S2, S3, .., Sn}; /* Начальные решения, созданные с помощью GRASP(V2)*/ Limit = начальное предельное значение, D = начальное значение демона, Неулучшение = 0, Global_Best = Лучшее решение в Pop; Для M = 1 до Max_Iter /*** Занятая пчелиная фаза ***/ Для i = 1 до Pop_Size
Выберите случайным образом один маршрут решения Si ; Примените один случайно выбранный оператор окрестности к текущему маршруту решения Si, чтобы получить новое решение S. Если S осуществимо. Тогда вычислите OF для S i End If.

Если OF S i ' > OF(S я))
Запомните новое решение S ' я ;
Иначе, если (E Di)
Запомните новое решение S ' я ; обновить значение демона: Di = Di E;
Еще
Запомните старое решение Si ; обновить значение демона: Di = Di + E;
Отсутствие улучшения(i) = Отсутствие улучшения(i) + 1;
Конец, если
Конец для
/*** Фаза наблюдателя ***/
Выберите N решений от нанятых пчел, используя стратегию выбора колеса рулетки;
Для я = от 1 до N
Выбрать случайным образом один путь решения Si ;
Для j = 1 к числу операторов окрестности
Применить выбранный оператор окрестности (j) к текущему маршруту решения Si для получения нового решения S ' я ;
Если ' S осуществимо, то вычислите OF S i ' я ;
Конец, если
Если OF S i ' > OF(S я) ,
Запомните новое решение S ' я ;
Перерыв;
Конец, если
Конец для

Machine Translated by Google


```
Если решение Si не улучшилось после применения операторов окрестности
Отсутствие улучшения(i) = Отсутствие улучшения(i) + 1;
Конец, если
Конец для
/*** Фаза пчел-разведчиков ***/
Для я = от 1 до N
Если отсутствие улучшения(i) > предел
Отсутствие улучшения(i) = 0;
Си = 0;
Реконструируйте Si , используя стратегию S1 или S2 фазы пчел-разведчиков; Вычислите OF для Si ;
Конец, если
Конец для
Назначьте решение, имеющее максимальное значение OF, Best_Solution;
Если (Глобальное_лучшее < лучшее_решение)
Глобальное_Лучшее = Лучшее_Решение;
Конец, если
Конец для
Результат: решение Global_Best
```

4.2. ABC с кластеризацией (ABC(C))

В этой части исследования метод ABC, описанный в разделе 4.1 , использовался для улучшения исходных решений, построенных с использованием различных методов GRASP(C), описанных в [7], а именно K-means-GRASP(C), модифицированного- K-означает GRASP(C) и ACO-GRASP(C).
Опять же, есть некоторые модификации, которые необходимо учитывать при использовании ABC с этими алгоритмами. При использовании следующих операторов процесс вставки должен быть ограничен парами клиентов только внутри одного кластера: случайное удаление одной пары клиентов и вставка непосещенной, удаление пары клиентов с низким IR и вставка непосещенной пары с высоким IR. и вставьте пару клиентов с высоким IR. Кроме того, ограничение на вставку должно применяться на этапе пчел-разведчиков в обеих стратегиях (S1 и S2). Мы использовали те же значения параметров, которые использовались в ABC (без кластеризации), чтобы улучшить первоначальные решения в этой части исследования. Более того, те же стратегии фазы пчел-разведчиков (S1 и S2) использовались в алгоритмах ABC(C)). Таким образом, в целом у нас есть шесть алгоритмов, использующих ABC(C): (1) K-means-GRASP(C)-ABC(C)S1, (2) K-means-GRASP(C)-ABC(C) S2, (3) модифицированные K-средства-GRASP(C)-ABC(C)S1, (4) модифицированные K-средства-GRASP(C)-ABC(C)S2, (5) ACO-GRASP(C)- ABC(C)S1 и 6) ACO-GRASP(C)-ABC(C)S2. Например, первый алгоритм относится к генерированию начального решения с использованием GRASP после кластеризации пространства поиска с использованием алгоритма K-средних и последующего его улучшения с использованием ABC со стратегией реконструкци

5. Вычислительные эксперименты

Все алгоритмы реализованы в MATLAB (R2017b) на ноутбуке с процессором Intel Core i7-4510U, работающим на частоте 2,00 ГГц (2601 МГц, два ядра, четыре логических процессора). В следующих подразделах представлены описания используемых наборов данных и особенности настройки параметров перед представлением экспериментальных результатов.

5.1. Тестовые экземпляры

Чтобы проверить производительность предложенных нами методов, мы использовали те же примеры, что и в [4]. Набор данных включает 36 экземпляров, которые были разделены на три отдельные группы: малые (состоящие из 20 и 50 клиентов соответственно, обслуживаемые двумя и тремя транспортными средствами), средние (состоящие из 100 и 250 клиентов, обслуживаемые 4 и 5 транспортными средствами). соответственно) и крупных размеров (500 и 1000 клиентов, обслуживаемых 6 и 8 автомо
При этом каждую группу составляют двенадцать экземпляров. Эти события демонстрируют изменчивость как по времени, так и по доходам. Для создания коротких и длинных маршрутов общий лимит времени может быть небольшим или большим, в диапазоне от 2500 до 15 000 с.

Machine Translated by Google

При этом суммы доходов устанавливаются одним из трех способов: случайным образом, пропорциональна спросу или одинакова для всех потребителей. Доход получается от доставки покупателю после отгрузки товара от покупателя, осуществляющего самовывоз. Количество товара указывается как целое положительное число от 1 до 50.

5.2. Настройка параметров

5.2.1. Настройка параметров ABC без кластеризации

Алгоритм ABC имеет три параметра: количество итераций (Max_Iter), размер населения (Pop_Size) и начальное значение лимита. Кроме того, поскольку мы использовали DA в фазе занятой пчелы, необходимо настроить начальное значение демона (D) . Подробности тестирования каждого параметра представлены ниже.

Количество итераций

Каждый экземпляр данных по 100 клиентам был протестирован с разным количеством итераций: 10, 50, 100, 500 и 750. Как видно из таблицы 1, результаты тестирования показали, что никакого улучшения значений целевой функции после 500 итераций. Таким образом, максимум количество итераций было принято равным 500.

Таблица 1. Результаты настройки количества итераций ABC.

Набор данных / Экземпляры	Количество итераций				
	10	50	100	500	750
13-100-ФС	32 274,04	35 432,98	39 437,98	39 438,58	39 438,58
14-100-ФЛ	63 311,86	63 311,86	63 311,86	63 455,13	63 455,13
15-100-ПС	68 625,48	68 625,48	71 185,28	72 032,22	72 032,22
16-100-ПЛ	115 792,8	115 792,8	124 608,2	131 404,9	131 404,9
17-100-РС	67 974,56	74 246,27	74 246,27	74 330,95	74 330,95
18-100-РЛ	112 986,9	112 986,9	116 775,2	119 991,9	119 991,9

Численность населения

Те же экземпляры набора данных были протестированы еще раз, чтобы выбрать подходящее количество решения среди населения. Вначале размер популяции был установлен равным 30, потому что это обеспечили лучшие результаты, приведенные в [7]. Затем мы увеличили популяцию до разных размеров: 40, 50 и 60 решений. В таблице 2 представлены значения OB после установки количества итераций до 500. Результаты показали, что увеличение размера популяции более чем до 50 не привело к улучшению значения целевой функции. Таким образом, численность населения было принято за 50.

Таблица 2. Результаты настройки размера популяции ABC.

Набор данных / Экземпляры	Численность населения			
	30	40	50	60
13-100-ФС	39 438,58	39 438,58	41 472,86	41 472,86
14-100-ФЛ	63 455,13	66 453,66	66 453,66	66 453,66
15-100-ПС	72 032,22	72 032,22	72 170,54	72 170,54
16-100-ПЛ	131 404,9	131 404,9	131 404,9	131 404,9
17-100-РС	74 330,95	77 484,06	77 484,06	77 484,06
18-100-РЛ	119 991,9	120 249,7	126 504,3	126 504,3

Предельное значение

Предельное значение счетчика неулучшения – еще один важный параметр алгоритм ABC, поскольку он влияет на конверсию процесса поиска. Таким образом, значение выбранный предел должен соответствовать количеству итераций. Поэтому мы протестировали предел, используя два значения: 100 и 200. Результаты в Таблице 3 показывают, что увеличение

Machine Translated by Google

предельное значение более 100 не способствовало улучшению целевой функции
ценить. Таким образом, предельное значение было принято равным 100.

Таблица 3. Результаты настройки предельного значения ABC.

Набор данных	Лимит	
Экземпляры	100	200
13-100-ФС	41 472,86	41 472,86
14-100-ФЛ	66 453,66	66 453,66
15-100-ПС	72 170,54	72 170,54
16-100-ПЛ	131 404,9	131 404,9
17-100-РС	77 484,06	77 484,06
18-100-РЛ	126 504,3	126 504,3

Демоническая ценность

Было протестировано несколько значений, чтобы выбрать лучшее начальное значение демона (D): 1000,
2500, 5000 и 7500. Результаты в Таблице 4 не показали улучшения целевой функции.
если значение демона было больше 5000. Таким образом, значение демона D принималось равным 5000.

Таблица 4. Результаты настройки значения демона ABC.

Набор данных	Демоническая ценность			
Экземпляры	1000	2500	5000	7500
13-100-ФС	41 472,86	41 596,33	41 954,86	41 954,86
14-100-ФЛ	66 453,66	66 683,84	66 683,84	66 683,84
15-100-ПС	72 170,54	72 449,56	72 449,56	72 449,56
16-100-ПЛ	131 404,9	131 404,9	131 404,9	131 404,9
17-100-РС	77 484,06	77 484,06	77 484,06	77 484,06
18-100-РЛ	126 504,3	126 504,3	126 504,3	126 504,3

5.3. Настройка параметров ABC(C)

Поскольку мы использовали тот же алгоритм ABC с небольшими изменениями в ABC с
подход кластеризации, мы приняли те же значения параметров, которые использовались в
ABC для нашего ABC(C).

5.4. Результаты экспериментов

Как упоминалось ранее, мы предложили две версии ABC для улучшения первоначальной версии.
решения: ABC, используемый с GRASP, и ABC (C), используемый с методами кластеризации,
т.е. K-средства-GRASP(C), модифицированные K-средства-GRASP(C) и ACO-GRASP(C). У нас также есть
две отдельные стратегии для фазы пчел-разведчиков. S1 строит решение случайным образом
выбор одной из трех стратегий: случайная вставка, жадная и рандомизированная вставка или
жадная вставка. После тестирования этой стратегии мы обнаружили, что эвристика случайной вставки не работает.
не улучшить результаты по сравнению с другими, поэтому мы исключили этот вариант и сохранили другой
два. Вторая стратегия, S2, работает аналогично GRASP в [7]. Таким образом, всего у нас восемь
предложенные методы названы следующим образом: (1) ABC(S1), (2) ABC(S2), (3) K-means-GRASP(C)-
ABC(C)S1, (4) K-средства-GRASP(C)-ABC(C)S2, (5) модифицированные K-средства-GRASP(C)-ABC(C)S1,
(6) модифицированные K-средства-GRASP(C)-ABC(C)S2, (7) ACO-GRASP(C)-ABC(C)S1 и (8) ACO-
GRASP(C)-ABC(C) C2. Все предложенные алгоритмы запускались пять раз, как это было сделано с
предыдущие алгоритмы в [6] и [7]. Мы сравнили предложенные нами алгоритмы и
современные алгоритмы, представленные в [10,11]: C1-Seq VNS, C2-Seq VNS, C1-Self
Adaptive VNS, C2-Self Adaptive VNS и IPPD, для решения MVPPDP с точки зрения
время обработки и полученная прибыль.

Во-первых, что касается времени обработки, все предыдущие алгоритмы выполнялись в течение
5 минут на каждый экземпляр. Отметим, однако, что это время не считается приемлемым для
приложения мобильной коммерции (например, Hungerstation, MRSOOL и т. д.), которые используются ежедневно и полагаются на

Machine Translated by Google

скорость ответа, поэтому в нашем случае мы использовали условие остановки в 500 итераций, чтобы измерить время обработки. В таблице 5 представлены результаты сравнения предложенных нами алгоритмы и современные алгоритмы по времени (в секундах). Кроме того, средний каждой категории (выделенные строки в Таблице 5) использовалась для сравнения производительности алгоритмов, как показано на рисунке 2. Из таблицы 5 и рисунка 2 видно, что наши предложенные алгоритмы на основе кластеризации превосходят все предыдущие алгоритмы по отношению времени обработки во всех инстанциях, кроме случаев с участием 1000 клиентов. Также, другие предложенные алгоритмы, не основанные на кластеризации (ABC(S1) и ABC(S2)) были быстрее, чем предыдущие алгоритмы для экземпляров малого и среднего размера. Причина, по которой наши алгоритмы не преуспевают в экземплярах большого размера, вероятно, связана с Использованные стратегии пчел-разведчиков, а именно жадный метод в первой стратегии. и GRASP во втором. При жадном методе пары клиентов выбираются на основе по определенной мере, а затем вставляется в лучшее положение, что приводит к уменьшению стоимость поездки. Этот подход дает хорошие результаты, но требует времени, особенно в крупных экземплярах. Во второй стратегии пары непосещенных клиентов выбираются на основе вычисление IR, что также приводит к затратам времени в экземплярах большого размера. Общий, однако кластеризация пространства поиска явно помогала обрабатывать экземпляры большого размера и время обработки сократилось по сравнению с другими алгоритмами, которые работают со всеми клиентами без кластеризации.

Таблица 5. Сравнение современных алгоритмов и предлагаемых алгоритмов на основе время (секунды).

Набор данных	Предыдущий Алгоритмы(5 мин Время выполнения)	K-средство-ПОНЯТЬ-ABC(C)S1	K-средство-ПОНЯТЬ-ABC(C)S2	Модифицированный K-средство-ПОНЯТЬ-ABC(C)S1	Модифицированный K-средство-ПОНЯТЬ-ABC(C)S2	АКО-ПОНЯТЬ-ABC(C)S1	АКО-ПОНЯТЬ-ABC(C)S2	ABC(S1)	ABC(S2)
1-20-ФС	300	14,48	14,47	15,20	14,58	16,78	12,12	31,04	30,65
2-20-ФЛ	300	12,13	12,42	13,82	13,17	13,51	11,45	30,20	30.56
3-20-ПС	300	16,62	17,48	17,83	17,17	18,72	15,85	33,77	39,28
4-20-ПЛ	300	12	12,01	12,61	12,24	15,63	12,98	24,43	28.45
5-20-РС	300	14,87	15,21	14,97	14,57	16,67	14,57	30,79	31.17
6-20-РЛ	300	11,32	12,38	13,47	13,71	13,63	12,25	22,62	21.56
Средний-20	300	13,57	14	14,65	14,24	15,82	13,20	28,81	30.28
7-50-ФС	300	23,24	24,29	25,67	25,49	25,45	22,03	50,10	54,80
8-50-ФЛ	300	19,79	21,24	21,53	21,45	22,18	20,65	43,81	46,63
9-50-ПС	300	25,67	24,64	26,58	25,15	25,87	20,49	50,55	54,72
10-50-ПЛ	300	21,63	22,64	24,41	22,96	23,34	20,90	44,77	48,77
11-50-РС	300	26,31	26,58	26,86	26,05	27,49	20,23	48,90	51,94
12-50-РЛ	300	22,73	20,93	23,37	21,87	22,77	20,91	45,94	48,34
Средний-50	300	23,23	23,39	24,74	23,83	24,52	20,87	47,34	50,86
13-100-ФС	300	37,87	33,57	36,58	35,10	36,89	33,36	80,65	83,81
14-100-ФЛ	300	31,42	33,67	34,59	34,38	35,55	33,14	73,90	81,74
15-100-ПС	300	31,03	32,03	36,98	35,55	36,75	33,23	76,39	82,69
16-100-ПЛ	300	33,54	34,33	36,50	34,57	36,20	32,41	74,19	82,93
17-100-РС	300	38	35,92	37	35,81	38,19	34,03	82,59	86,96
18-100-РЛ	300	36,19	35,37	34,38	33,34	37,49	33,63	77,02	81.01
Среднее-100	300	34,68	34,15	36,01	34 .79	36 .85	33 .30	77 .46	83,19
19-250-ФС	300	73 .50	72,85	78,10	73,30	74,19	67,79	176,81	204,37
20-250-ФЛ	300	79,65	73,28	80,50	76,28	77,66	65,10	181,10	195
21-250-ПС	300	74,20	72,09	75,23	72,88	74,05	63,51	169,26	195,17
22-250-ПЛ	300	73,82	72,718	79,53	74,40	77,86	65,44	179	193,60
23-250-РС	300	74,67	72,99	77,63	72,38	76,21	66,44	177,25	193,93
24-250-РЛ	300	79,55	73,21	84,04	73,84	79,62	69,46	178,99	194,47
Средний-250	300	75,90	7 2,86	79,17	73,85	76,60	66,29	177,07	196,09
25-500-ФС	300	163,89	155,32	172,74	158,36	162,15	142,52	388,62	474,16
26-500-ФЛ	300	167,03	155,73	177,18	160,25	170,25	148,98	415,25	466,38
27-500-ПС	0	160,91	146,27	172,04	158,45	162,76	140,62	386,55	465,87
28-500-ПЛ	300	168,60	144,23	180,26	158,74	169,89	141,34	435,34	464,62

Machine Translated by Google

Таблица 5. Продолжение.

Набор данных	Предыдущий Алгоритмы(5 мин Время выполнения)	K-средство-ПОНЯТЬ-ABC(C)S1	K-средство-ПОНЯТЬ-ABC(C)S2	Модифицированный K-средство-ПОНЯТЬ-ABC(C)S1	Модифицированный K-средство-ПОНЯТЬ-ABC(C)S2	АКО-ПОНЯТЬ-ABC(C)S1	АКО-ПОНЯТЬ-ABC(C)S2	ABC(S1)	ABC(S2)
29-500-РС	300	170,15	140,16	170,50	158,58	162,42	139,80	383,89	468,15
30-500-РЛ	300	169,78	141,24	174,22	161,07	169,30	143,36	428,65	464,89
Средний-500	300	166,73	147,16	174,49	159,24	166,13	142,77	406,38	467,35
31-1000-ФС	300	432,51	354,73	448,29	408,67	426,32	397,22	1040,22	1406,22
32-1000-ФЛ	300	439,85	360,11	447,72	423,01	461,18	430,54	1197,38	1438,43
33-1000-ПС	300	434,34	364,52	432,93	418,51	391,01	416,76	1128,14	1398,49
34-1000-ПЛ	300	448,06	378,57	440,70	435,69	389,93	430,39	1204,97	1416,64
35-1000-РС	300	451,79	404,20	425,83	421,47	381,24	411,31	1110,80	1296,40
36-1000-РЛ	300	420,15	410,26	446,23	418,03	393,29	420,10	1222,33	1286,07
Среднее-1000	300	437,78	378,73	440,28	420,90	407,16	417,72	1150,64	1373,71

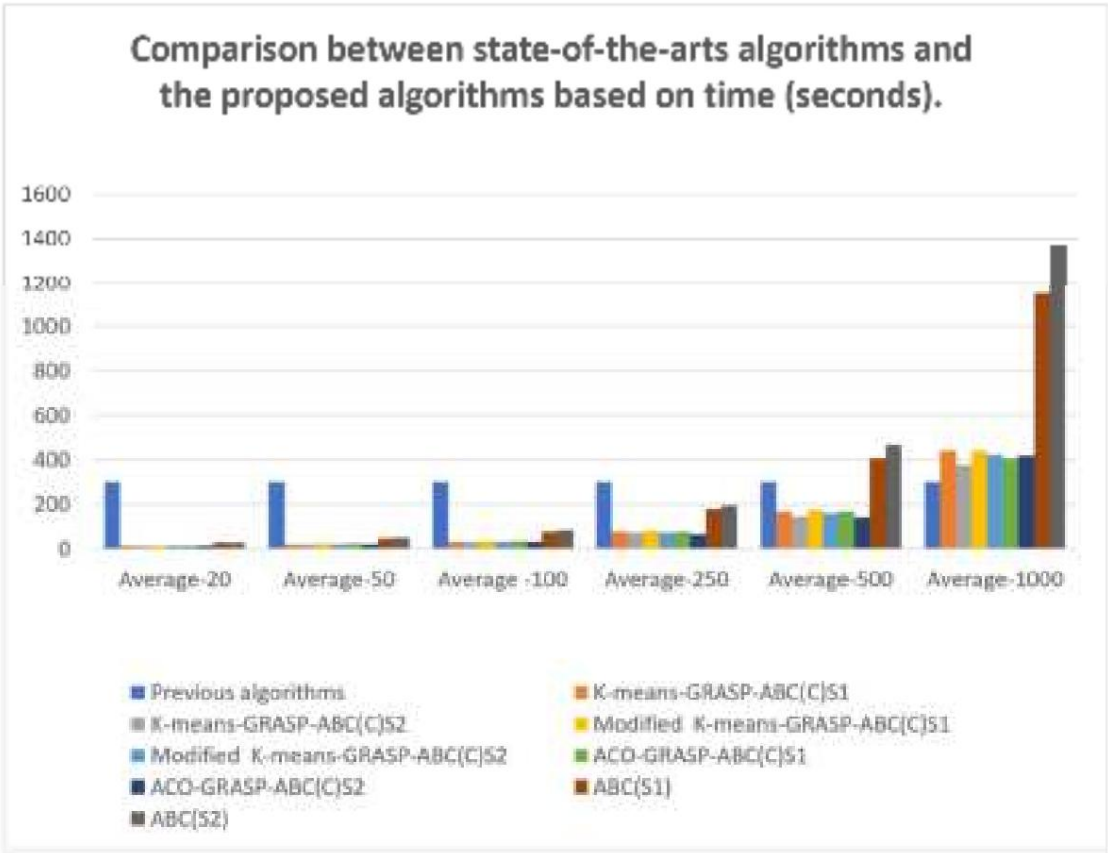


Рисунок 2. Сравнение современных алгоритмов и предлагаемых алгоритмов, основанных на время (секунды).

С другой стороны, мы сравнили предложенные нами алгоритмы и предыдущие алгоритмы. с точки зрения прибыли, рассчитанной по уравнению (1). В таблице 6 представлена средняя прибыль по пяти прогонов, а в Таблице 7 представлена лучшая прибыль, полученная за пять прогонов, и разрыв между лучшая прибыль и самая известная прибыль, используя следующее уравнение:

$$\text{Разрыв} = \frac{\text{Лучшая прибыль за пять раундов — Самая известная прибыль}}{\text{Самый известный профи}} \quad 100 \qquad (20)$$

Machine Translated by Google

Таблица 6. Сравнение предложенных нами алгоритмов с предыдущими методами, основанными на средней прибыли.

Набор данных	C1 — Последовательность ВНС [9]	C2 — ПОСЛЕД. ВНД [9]	C1 — Самостоятельный Адаптивный ВНД [9]	C2 — Самостоятельный Адаптивный ВНД [9]	ИППД [10]	K-средство-ПОНЯТЬ-ABC(C)S1	K-средство-ПОНЯТЬ-ABC(C)S2	Модифицированный K-средство-ПОНЯТЬ-ABC(C)S1	Модифицированный K-средство-ПОНЯТЬ-ABC(C)S2	АКО-ПОНЯТЬ-ABC(C)S1	АКО-ПОНЯТЬ-ABC(C)S2	АБВ(S1)	АБВ(S2)
1-20-ФС	30 315,5	30 315,5	30 315,5	30 315,5	30 315,5	30 267,75	30 427,58	27 007,58	27 007,58	25 490,76	25 490,76	30 395,92	30 294,77
2-20-ФЛ	38 765,6	38 765,6	37 866,3	38 765,6	39 179,66	37 642,38	37 468,82	39 773,34	39 240,95	36 342,53	36 322,33	39 459,15	36 153,93
3-20-ПС	51 473,4	52 128,4	52 565,1	52 565,1	52 566	51 481,8	43 340,71	51 481,8	43 009,55	42 917,34	34 107,64	52 684,98	52 565,1
4-20-ПЛ	58 365,6	58 365,6	58 365,6	58 365,6	58 777	57 350,35	57 086,36	57 350,35	57 301,88	56 523,32	47 367,54	58 888,64	56 808,21
5-20-РС	36 469,5	36 469,5	36 469,5	36 469,5	36 469,5	34 090,58	34 090,58	30 237,02	29 843,58	31 741,08	31 741,08	36 469,52	36 469,52
6-20-РЛ	43 368,6	43 368,6	43 368,6	43 368,6	43 781,81	42 215,53	42 086,44	44 401,45	43 736,53	40 769,67	40 450,55	44 244,27	42 143,26
Средний-20	43 126,37	43 235,53	43 158,43	43 308,32	43 514,91	42 174,73	40 750,08	41 708,59	40 023,35	38 964,12	35 913,32	43 690,41	42 405,8
Стандартный Отклонение	10 289,94	10 399,10	10 559,14	10 475,04	10 560,13	10 436,42	9349,23	11 822,81	10 907,82	10 656,36	7500.01	10 573,40	10 312,25
7-50-ФС	22 306	22 306	22 306	22 306	22 307,15	18 662,83	21 603,83	18 662,83	21 603,83	21 281,88	25 123,15	24 975,25	24 998,45
8-50-ФЛ	58 050,8	58 050,8	57 303,4	57 996,7	58 050,8	50 772,64	50 873,68	50 881,55	50 773,54	46 699,31	47 534,79	53 555,25	53 268,82
9-50-ПС	57 651,7	57 651,7	57 651,7	57 651,7	57 651,7	58 996,17	53 258,83	58 996,17	53 064,14	55 643,25	55 643,25	57 651,71	57 651,71
10-50-ПЛ	132 648	132 648,0	128 353	128 334,8	129 640,3	109 726,8	117 770,8	116 372	116 149,7	97 740,02	106 491,9	124 819,3	128 477,1
11-50-РС	33 340,4	33 340,4	33 340,4	33 340,4	33 340,4	28 865,48	28 907,33	28 865,48	28 907,33	31 251,34	29 969,16	33 054,57	33 098,58
12-50-РЛ	79 350,3	79 350,3	77 655,5	79 350,3	77 832,56	76 422,62	76 503,28	76 309,46	76 408,91	64 359,55	64 519,79	76 582,15	75 732,86
Средний-50	63 891,2	50 139,84	62 768,33	63 163,32	63 137,16	57 241,09	58 152,95	58 347,92	57 817,91	52 829,22	54 880,35	61 773,03	62 204,58
Стандартный Отклонение	39 248,45	22 523,36	37 635,69	37 749,54	38 077,61	33 048,63	35 097,31	35 184,37	34 547,72	27 041,68	29 384,58	35 937,41	37 167,44
13-100-ФС	41 464,84	41 955,2	41 523,5	42 174	41 896,86	35 386,24	36 055,01	35 303,67	36 625,9	33 147,92	33 187,57	40 596,77	40 543,56
14-100-ФЛ	81 271,2	81 971	80 080,4	80 500,2	79 479,88	58 704,7	60 780,74	57 607,07	60 035,43	51 106,58	47 074,98	64 937,47	67 237,63
15-100-ПС	70 387,4	70 398,5	71 113,8	71 082,8	69 436,25	54 197	57 206,6	49 566,36	59 481,84	60 614,78	45 995,09	68 874,02	71 290,33
16-100-ПЛ	124 288	123 931	122 691,8	122 183,2	125 057	98 140,3	104 031,6	100 122,4	110 099	92 681,38	96 365,75	112 690,5	124 487,1
17-100-РС	76 885,4	78 418,8	76 954,4	79 651,2	77 012,75	61 579,81	61 579,81	61 960,78	61 698,84	52 689,7	52 163,72	77 334,8	75 925,47
18-100-РЛ	131 052,6	130 670,4	130 403,4	130 366,6	131 058,6	98 522,81	100 291,3	104 771,9	109 450,2	92 423,54	93 867,6	115 108,7	122 320,4
Среднее-100	87 558,24	87 890,82	87 127,88	87 659,67	87 323,56	67 755,14	69 990,84	68 222,03	72 898,54	63 777,32	61 442,45	79 923,7	83 634,09
Стандартный Отклонение	34 099	33 674,36	33 546,8	33 090,45	34 316,01	25 383,15	26 636,59	28 060,86	30 021,67	24 032,04	26 834,7	29 022,89	33 175,38
19-250-ФС	73 259,94	72 445,3	72 703	72 287,9	70 799,19	36 045,94	32 341,35	43 729,29	38 692,81	34 038,31	33 942,57	48 988,62	46 216,19
20-250-ФЛ	132 996	132 438	132 394,4	133 404	129 485,3	54 144,64	63 245,12	70 325,74	81 370,79	51 239,48	57 273,9	71 397,91	81 288,1
21-250-ПС	111 334,4	111 817,8	109 449,4	109 355,8	110 440,3	62 245,05	71 066,44	74 931,5	83 992,55	72 198,2	69 410,97	86 624,32	93 568,56
22-250-ПЛ	197 899,8	197 214,2	194 131,2	194 549	194 338,6	112 454,5	131 917,5	133 567,4	155 750,3	95 906,32	127 789,5	146 104	160 301,3
23-250-РС	162 410,8	162 448,2	158 164,4	159 129,6	158 414,6	99 900	90 042,33	109 717,8	93 812,47	102 500,1	99 734,68	125 201,2	120 776
24-250-РЛ	269 072,8	268 713,8	266 559	267 291,4	266 204,2	148 156,7	176 579,9	180 807,7	177 336	146 270,4	164 012,2	204 037,4	203 859,8
Средний-250	157 829	157 512,9	155 566,9	156 003	154 947	85 491,14	94 198,78	102 179,9	105 159,2	83 692,14	92 027,3	113 725,6	117 668,3
Стандартный Отклонение	69 184,1	69 165,71	68 327,76	68 666,69	68 777,74	42 022,43	52 050,37	49 795,51	51 632,15	40 174	48 193,98	56 662,62	57 036,7

Machine Translated by Google

Таблица 6. Продолжение.

Набор данных	C1 — Последовательность ВНС [9]	C2 — ПОСЛЕД. ВНД [9]	C1 — Самостоятельный Адаптивный ВНД [9]	C2 — Самостоятельный Адаптивный ВНД [9]	ИППД [10]	K-средство-ПОНЯТЬ-ABC(C)S1	K-средство-ПОНЯТЬ-ABC(C)S2	Модифицированный K-средство-ПОНЯТЬ-ABC(C)S1	Модифицированный K-средство-ПОНЯТЬ-ABC(C)S2	АКО-ПОНЯТЬ-ABC(C)S1	АКО-ПОНЯТЬ-ABC(C)S2	АБВ(S1)	АБВ(S2)
25-500-ФС	158 425,4	159 146,2	157 399,8	159 526,6	155 683,9	59 737,27	71 001,31	51 581,38	73 939,03	54 762,97	51 117,39	75 731,5	79 862,1
26-500-ФЛ	243 210	244 591,6	245 600,8	245 183,6	242 173,1	101 753,4	120 779,3	106 505,3	129 200,4	62 041,19	88 256,7	112 172	121 824,9
27-500-ПС	201 195,4	202 102,2	197 227,6	198 740	203 932,9	105 678,8	146 631,6	107 797,6	143 918,5	102 996,9	105 667,8	130 361,2	143 638,1
28-500-ПЛ	319 239,2	321 181,6	315 854,6	317 610,8	329 642,8	229 525,6	241 331,4	225 812,4	243 361,8	152 573,5	191 169,5	201 973,8	222 042,1
29-500-РС	271 154,4	272 715,2	271 264,2	273 095	260 505,8	126 298,6	156 901,1	108 998,5	162 660,4	127 089,1	123 505,3	166 111,6	166 226,9
30-500-РЛ	402 557,2	406 689,6	399 990,8	404 517	408 819,4	259 683,6	272 509	241 454	282 087	167 819,5	205 885,1	239 675,7	255 261,1
Средний-500	265 963,6	267 737,7	264 556,3	266 445,5	266 793	147 112,9	168 192,3	140 358,2	172 527,9	111 213,8	127 600,3	154 337,6	164 809,2
Стандартный Отклонение	69 184,1	69 165,71	68 327,76	68 666,69	68 777,74	42 022,43	52 050,37	49 795,51	51 632,15	40 174	48 193,98	56 662,62	57 036,7
31-1000-ФС	150 920,4	151 943,2	151 990	153 820,6	152 358,7	33 184,03	48 758,35	39 530,22	48 457,57	23 266,5	20 723,05	33 125,19	41 340,89
32-1000-ФЛ	218 211	218 213	223 999,8	222 072,8	222 089,1	92 970,35	93 386,23	92 142,15	95 500,88	24 110,85	44 359,35	57 137,24	68 705,69
33-1000-ПС	439 681,8	441 120,2	433 353,8	429 853	457 759,4	272 126,9	288 095,1	236 797,1	280 912,6	178 654,3	201 178,4	240 944,9	263 540,2
34-1000-ПЛ	678 396,2	651 754,6	677 511,6	669 212,8	681 370,5	498 835,3	519 115,5	492 753,1	507 105,4	346 244,9	366 004,4	397 930	426 372,5
35-1000-РС	455 054,6	461 394	459 167,4	462 999,2	461 919,6	207 134	255 835,4	192 030,8	242 210,5	124 538,1	176 577,1	212 979,1	228 716
36-1000-РЛ	668 655	662 605,4	668 615,4	667 184,8	676 159,7	464 549,3	450 834,3	407 703,7	448 210,5	297 212,8	322 568,7	364 643,1	372 491,4
Среднее-1000	435 153,2	431 171,7	435 773	434 190,5	441 942,8	261 466,7	276 004,1	243 492,9	270 399,6	165 671,2	188 568,5	217 793,2	233 527,8
Стандартный Отклонение	219 981,4	212 861,8	218 443,1	216 320	221 371,6	190 345	187 196	176 803,2	183 520,7	135 712,2	140 506,7	151 318,2	155 901,2

Таблица 7. Сравнение предложенных алгоритмов и предыдущих методов, основанных на лучшем решении и значениях разрыва.

Набор данных	Лучший Известный Выгода	C1 — Seq VNS [4]		C2 — Seq VND [4]		C1 — Самоадаптирующийся VND [4]		C2 — Самоадаптирующийся VND [4]		ИППД [10]		K-средства-GRASP-ABC(C)S1	
		Лучший Выгода	Зазор	Лучший Выгода	Зазор	Лучший Выгода	Зазор	Лучший Выгода	Зазор	Лучший Выгода	Зазор	Лучший Выгода	Зазор
1-20-ФС	30 315,5	30 315,5	0	30 315,5	0	30 315,5	0	30 315,5	0	30 315,5	0	30 267,75	0,15
2-20-ФЛ	39 340,3	38 765,6	1,5	38 765,6	-1,5	38 765,6	-1,5	38 765,6	-1,5	39 179,66	-0,41	37 750,35	4
3-20-ПС	52 855,2	51 473,4	2,7	52 561,1	-0,6 0	52 565,1	-0,6 0	52 565,1	-0,6 0	52 566	-0,55 0,7	51 515,53	2,5
4-20-ПЛ	58 365,6	58 365,6	0	58 365,6		58 365,6		58 365,6		58 777		57 350,35	1,7
5-20-РС	36 469,5	36 469,5	0	36 469,5	0	36 469,5	0	36 469,5	0	36 469,5	0	34 090,58	6,5
6-20-РЛ	43 764,3	43 368,6	0,9	43 368,6	0,9	43 368,6	0,9	43 368,6	0,9	43 781,81	0,04	42 353,35	3,2
Средний-20	43 518,4	43 126,37 -0,85 22 306		43 307,65 -0,50 22 306		43 308,32	0,50	43 308,32	0,50	43 514,91	0,04	42 221,32	2,9
7-50-ФС	22 441	-0,6 58 050,8		-0,6 58 050,8		22 306	0,6	22 306	0,6	22 307,15	0,6	18 662,83	16,8
8-50-ФЛ	58 050,8		0		0	58 050,8	0	58 050,8	0	58 050,8	0	51 040,14	12

Machine Translated by Google

Таблица 7. Продолжение.

Набор данных	Лучший Известный Выгода	C1 — Seq VNS [4]		C2 — Seq VND [4]		C1 — Самоадаптирующийся VND [4]		C2 — Самоадаптирующийся VND [4]		ИППД [10]		K-средства-GRASP-ABC(C)S1	
		Лучший Выгода	Зазор	Лучший Выгода	Зазор	Лучший Выгода	Зазор	Лучший Выгода	Зазор	Лучший Выгода	Зазор	Лучший Выгода	Зазор
9-50-ПС	57,651.7 57,651.7 132,648	0		57 651,7	0	57 651,7	0	57 651,7	0	1 31 012,9	0	58 996,17	2.3
10-50-ПЛ	132,648 33,340.4 33,340.4	0		132 648	0	132 648	0	132 648	0	112 747,3	0	114 962,9	13,3
11-50-РС	79,350.3 79,350.3 63,913.7	0		33 340,4	0	33 340,4	0	33 340,4	0	198 241,2	0	28 907,33	13,2
12-50-РЛ	63,891.20 0.1 44,402.3	0		79 350,3	0	79 350,3	0	79 350,3	0	161 804,7	-1,95	76 479,84	3,6
Средний-50	42,174.4 5.3 81,999.4 81,999.4 0			63 891,20 -0,10 42 174,4		63 891,20	-0,10	63 891,20	-0,10	269 734,7	-0,43	58 174,87	8,9
13-100-ФС	71,512.4 70,836 1 125,032 124,770			-5,3 81 999,4 0 71 326,4		42 174,4	-5,3	42 215,5	-5,2	157 596,57	-5,83 0	36 322,94	18,1
14-100-ФЛ	80,878.8 80,878.8 131,845 131,473			-0,3 124 770 78 818		81 621	-0,5	80 848,1	-0,2	160 592,1		60 767,96	25,8
15-100-ПС	89,278.32 97,991.44 1.13 75,309.1			131 845 88 488,87. 40		71 399,2	-0,2	71 399,2	-0,2	249 105,7	-0,39	54 428,34	23,8
16-100-ПЛ	73,610.9 2.3 136,594	0,2		73 575,7	0,2	123 766	1	124 910	0,1	208 287,7	0,6	99 860,9	20,1
17-100-РС	134,889 113,886 112,427	0		-2,4 134	2,6	80 878,8	0	80 878,8	0	337 358,9	0	61 647,98	23,7
18-100-РЛ	200,402 200,402 164,070	0,3		186 112	0	131 339	0,4	131 339	0,4	265 040	0	102 403,8	22,3
Среднее-100	162,960 274,563 271,185 160,804			480 198 611 164 070		88 529,73	1,23	88 598,43	1,02	411 967,6	1,14	69 238,65	22,4
19-250-ФС	159,245.65 1,13 167 080 161 063 -3,7			271 193 159 019,28		73 835,5	2	73 544,7	2,4	272 05	4,54	39, 672,94	47,3
20-250-ФЛ	26-500-FL 259 319 249 171	1,3		-1,27 161	1,8	132 870	2,8	135,62 2	0,7	8,67 152	4,26	54 144,64	60,3
21-250-ПС	27-500-PS 206 788 203 832	1,3		040 -3,8	1,3	112 428	1,3	110 062	3,5	358,7 222	1,01	69 574,44	38,9
22-250-ПЛ	-1,5 28-500-PL 331 291 321	0		249 661	0,9	195 135	2,7	196 524	2	089,1 457	1,09	112 454,5	43,8
23-250-РС	471 -3,1 29-500-RS 281,7	0,7		205 480	0	162 115	1,2	162 115	1,2	759,4 681	1,4	103 456	36,9
24-250-РЛ	11 279 105 0,9 30-500-	1,2		-0,6 -1,8	1,2	272 957	0,6	269 790	1,8	370,5 461	1,79	148 156,7	46
Средний-250	RL 417 282 407 689 2,4 Среднее- 500			325 522 - 1,3 409 755		158 223,42	1,10	174 822,60	1,93	919,6 676	2,35	87 909,87	45,3
25-500-ФС	277 245,2 270 388,50 -2,61 31-1000-FS			1,8 271 615,17		159 064	5	161 375	3,5	159,7 441	4,04	59 737,27	64,2
173 826 154 196 -12,7 32-1000-FL 247 385 221 958		4,1		2,20 151	3,9	247 986	4,6	249 312	4	942,83	4,1	107 852,1	58,4
-11,5 33-1000-PS 464 580 450 349 -3,2 34-1000-PL 712 918 693				943,2 -12,2 221 322		200 764	3	201 590	2,6		0,72	105 678,8	48,8
407 –2,8 35-1000-RS 492 083 463 625 –6,1 36-1000-RL 705 167				450 337	-11,8	320 872	3,2	321 155	3,2		1,82	229 525,6	30,7
673 951 –4,6 Среднее – 1000 465 993,2 405 958 –6,8				-3,2 672 910 -5,9 472		274 419	2,7	276 392	1,9		-6,29	126 298,6	55,1
				626 -4,1 669 926 -5,3		407 849	2,3	409 262	2		-1,29	259 683,6	37,7
				439 844,03 -7,08		268 492,33	3,47	269 847,67	2,87		-2,20	165 807,7	40,1
						154 357	12,6	156 106	11,4		-11,8	33 184,03	80,9
						224 999	9,9	226 913	9		-10,1	92 970,35	62,4
						442 31 1	5	442, 218	5,1		1,47	272 126,9	41,4
						684 121	4,2	683 211	4,3		3,26	498 835,3	30
						465 022	5,8	469 162	4,9		4,90	207 134	57,9
						676 720	4,4	684 217	3,1		2,90	464 549,3	34,1
						441 255	6,98	443 638	6,3		5,2	261 466,65	43,8

Machine Translated by Google

Таблица 7. Продолжение.

Набор данных	K-средства-GRASP-ABC(C)S2		Модифицированные K-средства-ГРАСП-ABC(C)S1		Модифицированные K-средства-ГРАСП-ABC(C)S2		ACO-GRASP-ABC(C)S1		ACO-GRASP-ABC(C)S2		АБВ(S1)		АБВ(S2)	
	Лучшая прибыль	Зазор	Лучшая прибыль	Зазор	Лучший разрыв в прибыли		Лучший разрыв в прибыли		Лучшая прибыль	Зазор	Лучший разрыв в прибыли		Лучший разрыв в прибыли	
1-20-ФС	30427,58	0,3	76 ,697,65	10,9	27 007,58	10,9	25 490,76	15,9	25 490,76	15,9	30 554,62	0,78	30 315,52	0
2-20-ФЛ	37556,07	4,5	58 579,14	1,9	39 240,95	0,2	36 423,3	7,4	36 322,33	7,6	40 104,23	1,9	36 234,48	7,8
3-20-ПС	44334,2	16,1	35 937,96	2,5	44 334,2	16,1	42 944,78	18,7	34 107,64 47	35,4	53 164,53	0,58	52 565,1	0,5
4-20-ПЛ	57350,35	1,7	60 851,27	1,7	57 350,35	-1,7	56 603,63	-18,1	31 3	402,91	31 18,7	59 140,98	1.32	57 286,89	1,8
5-20-РС	34090,58	6,5	54 190,18	15,4	29 843,58	741,08	0,04	41 071,62	12,9	741,08	40 12,9	36 469,52		36 469,52	0
6-20-РЛ	42169,13	3,6	102 121,7	2,2	43 780,72	-7,4	39 045,86	-3,7	21 6,1	600,21	35 7,2	44 707,23	0	42 873,71	2
Средний-20	40987,99	5,8	61 962,79	3,6	40 259,56	281,88	-12,3	49 128,45	10,2	944,16	25 17,4	44 023,52	2,1	42 624,2	2
7-50-ФС	21603,83	3,7	105 175,3	3,7	21 603,83	-7,9	55 ,643,25	-11,8	100 5,1	123,15	50 11,9	24 975,25	1,16	24 998,45	11.3
8-50-ФЛ	51001,04	12,1		12,1	50 896,3	393,3	-13,2	31 251,34	15,3	061,24	55 13,7	53 710,99	11,2	54 288,08	6,4
9-50-ПС	54165,39	6		6	53 064,14	-3,4	65 792,21	-9,2	53 3,4	643,25	106 3,4	57 651,71		57 651,71	0
10-50-ПЛ	119099,6	10,2		10,2	116 868,6	915,07	-13,6	33 565,46	24,3	544,8	29 19,6	126 360	-7,4 0 -4,7	133 743,2	0,82
11-50-РС	28907,33	13,2		13,2	28 907,33	-25,1	52 328,66	60 614,78	6,2	969.	16 65 10,1	33 058,21	0,8	33 159,12	0,5
12-50-РЛ	76, 697,65	3,3		3,3	76,6 04,84	-10,6	95 836,25	. 3 52	17	752,8	55 17,1	77,4 67,34	2,3	76,5 17,36	3,5
Средний-50	58 579,14	8,3		8,3	57 990,84	689,7	16 95 363,69		15,6	515,73	33 13,1	62 203,92	2,6	63 392,99	0,8
13-100-ФС	36 188,32	18,4		19	38 323,6				24,4	566,06	49 24,4	41 812,08	5,8	41 636,8	6,2
14-100-ФЛ	63 488,24	22,5		25,7	61 351,99				36,1	789,52	47 39,2	66 375,92	19	72 595,41	11,4
15-100-ПС	60 815,37	14,9		24,2	60 554,49	15,3			15,2	498,28	101 33,5	69 275,79	3,1	71 400,63	0,1
16-100-ПЛ	104 786,4	16,1		18,3	111 728,2				23,3	142,5	52 19,1	115 579,2	7,5	128 473,2	2,7
17-100-РС	61 647,98	23,7		23,3	61 962,79				34,8	689,7	97 34,8	77 484,06	4,1	77 484,06	4,1
18-100-РЛ	101 013,6	23,3		20,2	110 683,8				27,6	568,94	25,9	119 554,3	9,3	126 871,5	3,7
Среднее-100	71 323,32	20,1	70 039,87	21,5	74 100,81	17 65 066,42			27,1	63 709,17	28,6	81 680,23	8,5	86 410,27	3,2
19-250-ФС	35 443,33	52,9	48 079,56	36,1	39 848,03	47 36 309,86	35,8		51,7 36 612,19	62,4	51,3	36 309,86	-51,7 36 612,19	-62,4 62	51,3
20-250-ФЛ	67 014,68	50,9	70 325,74	48,5	87 575,52	51 239,48	73 654,78		62 831,21 73 143,68		54	51 239,48	831,21	-35,3 73 143,68	54
21-250-ПС	77 941,82	31,5	81 510,78	28,4	92 124,73	19,1	19,4		35,3	131 668,8	35,7	73 654,78	-52,1 131 668,8	-34,4 101	35,7
22-250-ПЛ	135 164,3	32,5	133 567,4	33,3	161 377,2	95 906,32	41,8		52,1	101 863,1	34,2	95 906,32	863,1	-46,7 166 523,5	34,2
23-250-РС	97 231,36	40,7	113 547,4	30,7	95 471,16	107 628,9	146 270,4		34,4	46,7	37,9	107 628,9			37,9
24-250-РЛ	187 891,9	31,5	180 807,7	34,1	180 803	34,1			166 523,5		39,3	146 270,4			39,3
Средний-250	100 114,6	37,7	104 639,8	34,9	109 533,3	31,8	85 168,2		47	95 440,4	40,6	85 168,2	47	95 440,4	40,6
25-500-ФС	74 692,46	55,2	51 581,38	69,1	76 481,48	54,2 61 718,84	49,2		63	54 247,37	67,5	78 432,78	53 86 250,21	56,7	48,3
26-500-ФЛ	124 478,4	51,9	106 505,3	58,9	131 572,4	72 388,37	27,1 109 238		72	93 308,66	64	112 172	126 203,2	34,1	51,3
27-500-ПС	152 566,6	26,2	107 797,6	47,8	150 645,3				47,1	112 111,9	45,7	136 168,2	148 384,5		28,2

Machine Translated by Google

Таблица 7. Продолжение.

Набор данных	K-средства-GRASP-ABC(C)S2		Модифицированные K-средства-ГРАСП-ABC(C)S1		Модифицированные K-средства-ГРАСП-ABC(C)S2		ACO-GRASP-ABC(C)S1		ACO-GRASP-ABC(C)S2		АБВ(S1)		АБВ(S2)	
	Лучшая прибыль	Зазор	Лучшая прибыль	Зазор	Лучший разрыв в прибыли		Лучший разрыв в прибыли		Лучшая прибыль	Зазор	Лучший разрыв в прибыли		Лучший разрыв в прибыли	
28-500-ПЛ	249 759,7	24,6	225 812,4	31,8	247 180,6	25,3	152 573,5	–	196 969,2	40,5	201 973,8	39	225 040,5	32
29-500-РС	172 397,2	38,8	108 998,5	61,3	175 267,1	37,7	141 743,7	–29,4	167 128 570,6	54,3	178 693,9	36,5	173 003,3	38,5
30-500-РЛ	275 488,2	33,9	241 454	42,1	294 424,7	819,5			59,7 214 081,3	48,6	239 675,7	42,5	261 517,2	37,3
Средний-500	174 897,1	36,9	140 358,2	49,3	179 261,9	35,3	117 580,3		57,5 133 214,8	51,9	157 852,7	43 170 066,5		38,6
31-1000-ФС	51 188,29	70,5	39 530,22	77,2	51 585,38	-70,3	23 266,5	-60,4	24 86,6	25 563,9	85,2	33 125,19	-80,9	42 197,08 -76,9 73 75,7
32-1000-ФЛ	97 093,51	60,7	92 142,15	62,7	97 772,35	110,85	-38 178 654,3		90,2	47 620,03	80,7	57 137,24	126,9	-48,1 267 814,2 70,4
33-1000-ПС	293 900,2	36,7	236 797,1	49	287 774,9	-24,9	346 244,9	-49,3	61,5	209 372,1	54,9	240 944,9	-44,1	430 894,4 -56,7 232 42,3
34-1000-ПЛ	536 199,8	24,7	492 753,1	30,8	534 689,9	124	538,1	-34,2	297 212,8	51,4	381 116,4	46,5	397 930	298,7 -48,2 380 268,9 39,5
35-1000-РС	264 295,2	46,2	192 030,8	60,9	249 078,6				74,6	181 237	63,1	212 979,1		52,7
36-1000-РЛ	458 126,7	35	407 703,7	42,1	463 915,7				57,8	335 935,6	52,3	364 643,1		46
Среднее-1000	276 004,15	40,7	243 492,8	47,7	280 802,8	39,7	165 671,2		64,4	196 807,5	57,7	217 793,3	53,2 237 766,7	48,9

Machine Translated by Google

Самая известная прибыль была рассчитана в [9] путем запуска каждого экземпляра в течение 20 часов, а затем лучшая прибыль сохранялась как самая известная прибыль. Как видно из таблиц 6 и 7, предложенные нами алгоритмы хорошо работают с точки зрения прибыли для экземпляров небольшого размера: ABC(S1) возвращает шесть новых лучших решений и два других решения, которые были равны наиболее известным решениям. ABC(S2) также получил три новых лучших решения и три решения, равных самому известному решению. Кроме того, модифицированный K-means-GRASP(C)-ABC(C)S1 позволил получить два новых лучших решения. Кроме того, K-средства-GRASP(C)-ABC(C)S1, K-средства-GRASP(C)-ABC(C)S2, модифицированные K-средства-GRASP(C)-ABC(C)S2, и ACO-GRA- SP(C)-ABC(C)S2 предоставили по одному новому лучшему решению. В целом общая производительность всех предложенных алгоритмов была приемлемой и близкой к предыдущим алгоритмам по средней прибыли на этих экземплярах.

Однако для экземпляров среднего и большого размера предложенные нами алгоритмы не показали хороших результатов по сравнению с предыдущими методами, о чем свидетельствует большой средний процент отклонений по сравнению с наиболее известными результатами, показанными в Таблице 6. Существует много возможных причин для этот:

- Опять же, методы, используемые на этапе разведки ABC, похоже, влияют на производительность предлагаемых алгоритмов. Первая — это жадная рандомизированная стратегия, которая выбирает случайную пару клиентов, то есть может быть выбрана худшая пара (либо с низким доходом , либо разделенная большим расстоянием), что приводит к низкому качеству решения. Более того, в некоторых случаях операторы окрестности не могут в достаточной степени повысить качество решения из-за нарушений времени или ограничений по нагрузке. Второй метод использует GRASP, который создает решения путем случайного выбора из ограниченного списка кандидатов (RCL), который вдвое меньше размера набора решений-кандидатов (CSS). Таким образом, выбор непосещенных клиентов не является проблемой для инстансов небольшого размера, поскольку количество пар клиентов невелико, а это означает, что вероятность выбора лучшей пары клиентов (т. е. с наибольшим значением IR2) высока. Напротив, в средних и крупных экземплярах, где размер RCL велик, вероятность выбора лучшей пары клиентов снижается, что влияет на качество решений. • Количество решений в популяции, количество итераций и предельное значение играют важную роль в производительности алгоритмов. В наших алгоритмах мы полагаемся на небольшое количество решений и итераций, поскольку быстрое время обработки является критическим требованием для ежедневных приложений мобильной коммерции. Однако такое уменьшение числа итераций и размера популяции снижает шансы на достижение удовлетворительного качества решения.

- Все предыдущие алгоритмы были основаны на VNS, который имеет возможность изменять размер поиска окрестности каждый раз, когда качество решения не улучшается. Таким образом, систематическое расширение структуры окрестности помогает VNS полностью исследовать окрестности и получать хорошие решения. Напротив, функция диверсификации в нашем подходе основана либо на жадной функции (вставка пары клиентов с самым высоким IR в лучшую позицию), либо на комбинации жадных и рандомизированных функций (вставка случайной пары клиентов в лучшую позицию или вставка случайной пары клиентов из RCL в лучшую позицию). Однако эти стратегии кажутся менее успешными в плане диверсификации, чем VNS, которая опирается на организованный механизм исследования соседей.

На рисунке 3 сравниваются предложенные нами алгоритмы на основе среднего значения лучших решений для каждой категории случаев (выделенные строки в таблице 6). Понятно, что ABC в обеих версиях (S1 и S2) по-прежнему имеет преимущество перед всеми остальными алгоритмами для экземпляров с 20, 50, 100, 250 и 500 клиентами. Напротив, обе версии K-means-GRASP(C)-ABC(C) и модифицированная K-means-GRASP(C)-ABC(C) превосходят GRASP(V2)-ABC в обеих версиях (S1 и S2). для экземпляров на 1000 клиентов. Основными причинами низкой производительности алгоритмов кластеризации в экземплярах малого и среднего размера является распределение пар клиентов в K-средних и модифицированных K-средних, где процесс кластеризации может привести к помещению небольшого количества клиентов в кластер. , что означает, что у автомобиля еще есть время для обслуживания, но клиентов больше нет.

Machine Translated by Google

внутри этого кластера. Таким образом, мы приходим к выводу, что алгоритмы кластеризации (кроме ACO) хорошо работают только в экземплярах большого размера.

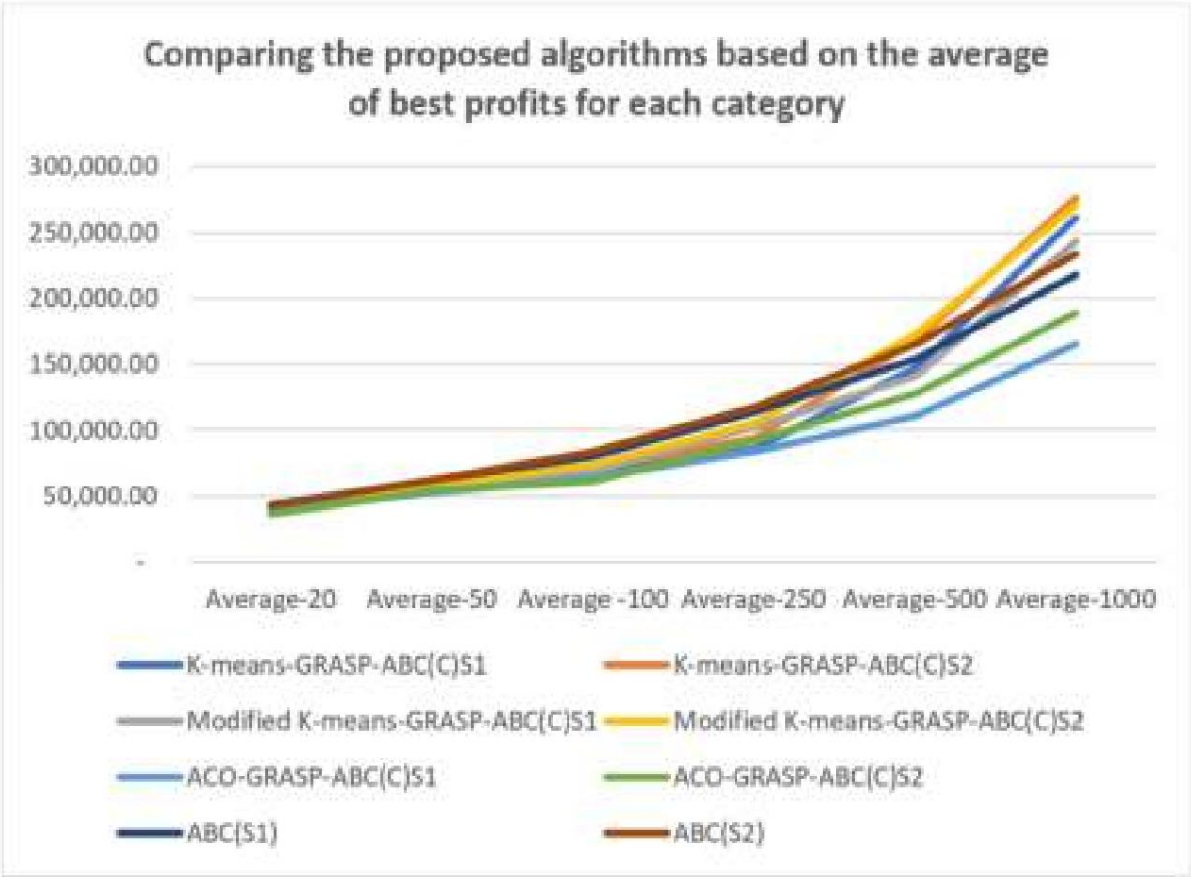


Рисунок 3. Сравнение предложенных алгоритмов на основе среднего значения лучших прибылей для каждой категории.

Что касается ACO-GRASP(C)-ABC(C)(S1 и S2), то ясно, что он имеет худшую производительность среди всех алгоритмов. Опять же, метод кластеризации ACO оказывает существенное влияние на результаты. В частности, функция случайности ACO способствует неточной кластеризации пар клиентов.

Кроме того, как видно из выделенных строк в Таблице 7, мы заметили, что вторая стратегия для пчел-разведчиков во многих случаях превосходит первую, поскольку первая стратегия включает в себя выбор либо лучших пар клиентов, либо случайных пар, а опция случайного выбора может влияют на качество решения, тогда как вторая стратегия выбирает пары клиентов случайным образом из RCL, который содержит лучшие пары клиентов в CSS.

6. Выводы

В этой статье мы представляем восемь новых метаэвристик для улучшения первоначальных решений задачи рентабельной погрузки и доставки нескольких транспортных средств (MVPPDP). Мы предложили две версии ABC (Искусственная пчелиная семья) для улучшения первоначальных решений: ABC с GRASP и ABC (C) с GRASP и кластеризацией, т.е. K-means-GRASP (C), модифицированный K-means-GRASP. (C) и ACO-GRASP (C). У нас также есть две отдельные стратегии для фазы пчел-разведчиков. Первая стратегия, S1, создает решение путем случайного выбора одной из двух стратегий: жадной и рандомизированной вставки или просто жадной вставки. Вторая стратегия, S2, основана на GRASP. Таким образом, у нас есть восемь различных предлагаемых методов. По сравнению с современными методами с точки зрения времени обработки все восемь методов имеют преимущество во всех случаях (за исключением случаев с 1000 клиентами). Более того, все предложенные алгоритмы хорошо зарекомендовали себя для небольших экземпляров с точки зрения прибыл

Machine Translated by Google

новые лучшие решения и пять равнозначных решений наиболее известным решениям. Однако алгоритмы немного отстают в экземплярах среднего и большого размера из-за жадной рандомизированной стратегии и GRASP, которые использовались на этапе разведки. Более того, наши алгоритмы отдают приоритет минимальным решениям и итерациям для сокращения времени обработки в повседневных приложениях мобильной коммерции, а уменьшение количества итераций и размера популяции снижает вероятность получения решения хорошего качества.

В целях будущих исследований на этапе предварительной проверки будет тщательно изучено больше местных операторов, чтобы их можно было применить в рамках ограничений MVPPDP, что увеличит шансы на создание высококачественных решений. Более того, мы попробуем другие метаэвристики, основанные на популяциях, такие как ГА, эволюционные алгоритмы или другие метаэвристики, основанные на роях, которые могут способствовать новым решениям для MVPPDP.

Вклад авторов: Концептуализация, AIA; методология, AIA; формальный анализ, AIA; курирование данных , AIA; Письмо — оригинальный черновик, AIA; приобретение финансирования, AIA; Надзор, МИХ; написание — рецензирование и редактирование, MIH. Все авторы прочитали опубликованную версию рукописи и согласились с ней .

Финансирование: Данное исследование не получило внешнего финансирования.

Заявление о доступности данных: http://prolog.univie.ac.at/research/MVPPDP/instances.zip. (по состоянию на 17 июля 2024 г.).

Благодарности: Исследователи хотели бы поблагодарить деканат аспирантуры и научных исследований Университета Касима за финансовую поддержку (QU-APC-2024-9/1).

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Ссылки

1. Карри, Д. Статистика доходов и использования приложений для доставки еды. 2024 г. Доступно онлайн: https://www.businessofapps.com/data/. магазин-приложений для доставки еды/ (по состоянию на 24 января 2024 г.).
2. Халид Х. Преимущества приложения по доставке еды для ресторанов и клиентов. Доступно онлайн: https://enatega.com/benefits-of-. приложение-доставка еды/ (по состоянию на 1 февраля 2024 г.).
3. Команда DJ «Лучшие приложения для доставки еды» в Саудовской Аравии, 2023 г. Доступно онлайн: https://www.digitalgradity.ae/blog/top-food- Delivery-apps-in-saudi-arabia/ (по состоянию на 2 января 2024 г.).
4. Ганстерер, М.; Кючуктепе, М.; Хартл, РФ Проблема выгодного забора и доставки нескольких транспортных средств. ИЛИ Спектр. 2017, 39, 303–319. [Перекрестная ссылка]
5. Талби, Е.Г. Метаэвристика: от проектирования к реализации; John Wiley & Sons: Хобокен, Нью-Джерси, США, 2009 г.; ISBN 978-0-470-27858-1.
6. Альхуджайлан, А.И.; Хосни, Мичиган Подход к созданию решения на основе GRASP для выгодной доставки и вывоза нескольких транспортных средств проблема. Межд. Дж. Адв. Вычислить. наук. Прил. 2019, 10, 111–120. [Перекрестная ссылка]
7. Альхуджайлан, А.И.; Хосни, Мичиган. Алгоритмы гибридной кластеризации с GRASP для построения первоначального решения для MVPPDP. Вычислить. Матер. Продолжение. 2020, 62, 1025–1051. [Перекрестная ссылка]
8. Арчетти, К.; Сперанца, МГ; Виго, Д. Проблемы маршрутизации транспортных средств с прибылью. В «Маршрутизации транспортных средств: проблемы, методы и приложения», 2-е изд.; СИАМ: Филадельфия, Пенсильвания, США, 2014 г.; Глава 10; стр. 273–297. ISBN 978-1-61197-358-7.
9. Тот, П.; Виго, Д. Маршрут транспортных средств: проблемы, методы и приложения, 2-е изд.; Общество промышленной и прикладной математики: Филадельфия, Пенсильвания, США, 2015 г.; ISBN 1611973589.
10. Кючуктепе, М. Алгоритм поиска окрестности с общей переменной для задачи рентабельной погрузки и доставки нескольких транспортных средств. Венский университет, 2014 г. Доступно онлайн: https://scholar.google.com/scholar?hl=en&as_sdt=0,5&q=A+General+Variable+ . Соседство+Поиск+Алгоритм+для+Многоавтомобильного+Выгодного+Самовывоз+и+Доставки+Задачи.+&btnG= (по состоянию на 22 июня 2024 г.).
11. Хаддад, Миннесота. Эффективная эвристика для решения проблем индивидуального получения и доставки. доктор философии Диссертация, IC/UFF, Нитерой, Бразилия, 2017.
12. Бруни, Мэн; Тоан, Д.К. Проблема выгодного выбора и доставки грузов с участием нескольких транспортных средств с неопределенным временем в пути. Трансп. Рез. Procedia 2021, 52, 509–516. [Перекрестная ссылка]
13. Карабога Д. Идея численной оптимизации на основе роя медоносных пчел. Технический отчет-tr06, Университет Эрджиес, инженерный факультет, кафедра компьютерной инженерии. 2005. Доступно в Интернете: https://scholar.google.com/scholar?hl=en&as_. sdt=0,5&q=Идея+основанная+на+меде+пчел+рой+для+численной+оптимизации&btnG= (по состоянию на 22 июня 2024 г.).
14. Алаззави, АК; Раис, ХМ; Басри, С.; Алсарьера, Ю.А. PhABC: Стратегия гибридной искусственной пчелиной семьи для создания попарного набора тестов с поддержкой ограничений. В материалах студенческой конференции IEEE по исследованиям и разработкам (SCOReD) 2019 г., Бандар-Сери Искандар, Малайзия, 15–17 октября 2019 г.; стр. 106–111, ISBN 9781728126135.
15. Сзето, Вайоминг; Ву, Ю.; Хо, С.С. Алгоритм искусственной пчелиной семьи для решения задачи маршрутизации транспортных средств. Евро. Дж. Опер. Рез. 2011, 215, 126–135. [Перекрестная ссылка]

Machine Translated by Google

16. Ву, Б.; Ши, З. Алгоритм кластеризации, основанный на роевом интеллекте. В материалах международных конференций по информационным технологиям и информационным сетям 2001 года. Слушания (кат. № 01EX479), Пекин, Китай, 29 октября – 1 ноября 2001 г.; стр. 58–66. ISBN 0-7803-7010-4.
17. Яо, Б.; Ху, П.; Чжан, М.; Ван, С. Алгоритм искусственной пчелиной семьи со стратегией сканирования для периодического маршрутизации транспортных средств. проблема. Моделирование 2013, 89, 762–770. [Перекрестная ссылка]
18. Карабога, Д.; Акей, Б. Обзор: Алгоритмы, моделирующие интеллект пчелиного роя. Артиф. Интел. Ред. 2009 г., 31, 61–85. [Перекрестная ссылка]
19. Крейц, М. Микроканоническое моделирование Монте-Карло. Физ. Преподобный Летт. 1983, 50, 1411. [CrossRef]
20. Бансал, Дж. К.; Шарма, Х.; Джейдон, С.С. Алгоритм искусственной пчелиной семьи: обзор. Межд. Дж. Адв. Интел. Парадиг. 2013, 5, 123–159. [Перекрестная ссылка]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Preptimize: автоматизация предварительной обработки и прогнозирования данных временных рядов

Мехак Усмани [1], Зульфикар Али Мемон [1], Адиль Зульфикар [2] и Ризван Куреши [3,*]

[1] Школа быстрых вычислений, Национальный университет компьютерных и новых наук, Карачи 65200, Пакистан; k200984@nu.edu.pk (МУ); zulfiqar.memon@nu.edu.pk (ZAM)
[2] Факультет электротехники, Национальный университет компьютерных и новых наук, кампус Фейсалабад, Фейсалабад 38000, Пакистан;
[3] adil.zulfiqar@nu.edu.pk Центр регенеративной медицины и здоровья, Гонконгский институт науки и инноваций, Китайская академия наук, Научный парк, САР Гонконг 999077, Китай
* Переписка: engr.rizwanqureshi786@gmail.com

Аннотация: Анализ временных рядов имеет решающее значение для принятия деловых и финансовых решений, особенно с учетом растущей интеграции Интернета вещей (IoT). Однако использование данных временных рядов для прогнозирования требует обширной предварительной обработки для решения таких проблем, как пропущенные значения, гетероскедастичность, сезонность, выбросы и шум. Для одномерных и многомерных временных рядов, гауссовских и негауссовых временных рядов, а также стационарных и нестационарных временных рядов необходимы разные подходы . Обработка недостающих данных сама по себе сложна и требует уникальных решений для каждого типа. Извлечение статистических характеристик, выявление проблем с качеством данных и выбор подходящих методов очистки и прогнозирования требуют значительных усилий, времени и опыта. Чтобы упростить этот процесс, мы предлагаем автоматизированную стратегию под названием Preptimize, которая объединяет статистические методы и методы машинного обучения и рекомендует схемы моделей прогнозирования, предлагая наиболее подходящие подходы для данного набора данных в качестве начального шага к дальнейшему анализу. Preptimize считывает образец из большого набора данных и рекомендует модель проекта, основанную на оптимизации, что упрощает ее использование даже неспециалистами. Результаты различных экспериментов показали, что Preptimize либо превосходил, либо имел сопоставимую производительность с эталонными моделями в различных секторах, включая цены на акции, криптовалюту и прогнозирование энергопотребления. Это демонстрирует эффективность системы в рекомендации подходящих моделей прогнозирования для различных наборов данных временных рядов, подчеркивая ее широкую применимость в различных областях прогнозирования временных рядов.

Ключевые слова: автоматизация; оптимизация; временной ряд; одномерные и многомерные; статистические методы; машинное обучение; предварительная обработка; прогнозирование; интерполяция; однократное и множественное вменение; недостающие данные

check for updates

Цитирование: Усмани, М.; Мемон, ЗА; Зульфикар, А.; Куреши, Р. Предварительная оптимизация: Автоматизация данных временных рядов Предварительная обработка и прогнозирование. Алгоритмы 2024, 17, 332. https://doi.org/10.3390/a17080332 .

Академический редактор: Витторио Маньеццо

Поступила: 18 июня 2024 г.
Пересмотрено: 22 июля 2024 г.
Принято: 26 июля 2024 г.
Опубликовано: 1 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Данные временных рядов, характеризующиеся последовательными точками данных, записанными в определенные моменты времени, представляют собой уникальные проблемы в области науки о данных, требующие специализированных методов предварительной обработки для обеспечения качества данных. Анализ временных рядов включает сбор, очистку , предварительную обработку и прогнозирование данных, адаптированных к различным типам временных рядов, будь то числовые или текстовые, одномерные или многомерные, сезонные или несезонные. Существующие исследования преимущественно сосредоточены на одномерных временных рядах, в то время как многомерные временные ряды создают дополнительные сложности, такие как шум, пропущенные значения, выбросы и сезонность. Для решения этих проблем предлагается автоматизированная среда предварительной обработки, объединяющая различные методы и алгоритмы машинного обучения. Анализ временных рядов, имеющий решающее значение в IoT и финансовой сфере, требует тщательной очистки и предварительной обработки данных из-за его отличительных характеристик и проблем. Понимание свойств данных, таких как стационарность и распределение, имеет важное значение для выбора подходящих методов предварительной обработки. Кроме того , алгоритмы глубокого обучения, такие как GAN и автокодировщики, показывают многообещающие результаты в предварительной обработке и прогнозировании данных временных рядов.

Machine Translated by Google

Поиск наиболее точной модели прогнозирования с минимальной долей ошибок при численном анализе временных рядов требует исчерпывающего экспериментирования с различными методами и их комбинациями. Также следует знать тип метода, применимого к конкретному типу данных [1]. Однако этот процесс требует много времени и труда.

Для эффективной обработки пропущенных значений используются различные методы вменения на основе свойств временных рядов, таких как распределение и сезонность. В связи с экспоненциальным ростом данных временных рядов, генерируемых различными источниками, такими как фондовые биржи [2], потребление энергии [3], системы мониторинга здоровья и системы регистрации погоды, аналитики сталкиваются с увеличением рабочей нагрузки. Временные ряды, характеризующиеся последовательными данными, записанными через определенные промежутки времени, создают уникальные проблемы, такие как обработка шума, сезонность и пропущенные значения, что требует применения специализированных методов предварительной обработки для обеспечения качества данных и повышения точности прогнозирования.

Предварительная обработка, важнейший этап в анализе данных и машинном обучении, включает в себя ряд шагов, направленных на повышение качества данных и упрощение обучения модели. Однако предварительная обработка требует много времени, не имеет заранее определенной последовательности и различается в разных областях. В контексте анализа временных рядов предварительная обработка становится более сложной из-за характера работы в реальном времени и последовательных зависимостей данных. Проблемы включают обработку отсутствующих, повторяющихся и неверных данных, включая выбросы. Существующая литература в основном сосредоточена на обработке недостающих данных и обнаружении выбросов при предварительной обработке временных рядов.

Выбросы, которые значительно отклоняются от большинства точек данных, могут быть одномерными или многомерными, причем последнее представляет большую сложность. Для обнаружения многомерных выбросов часто требуются подходы машинного обучения из-за задействованного многомерного пространства признаков . Методы обнаружения выбросов включают статистические методы и методы машинного обучения. Статистические методы включают экспоненциальное сглаживание, z-показатель, модели линейной регрессии и межквартильный диапазон, в то время как методы машинного обучения включают контролируемые методы, такие как нейронные сети и деревья решений, и неконтролируемые методы, такие как кластерный анализ. Также используются методы измерения расстояний, методы, основанные на теории информации, и такие алгоритмы, как KNN и COF.

Обработка пропущенных значений в данных временных рядов включает либо удаление неполных записей, либо вменение пропущенных значений. Методы вменения данных временных рядов, включая сплайн-интерполяцию и скользящие средние, отличаются от методов, используемых для данных невременных рядов, из-за временных свойств. Традиционных методов может быть недостаточно, что требует применения передовых методов, таких как интерполяция, автокодировщики и генеративно-состязательные сети (GAN) для вменения и обнаружения выбросов.

Анализ данных временных рядов учитывает такие свойства, как сезонность и тенденция, классифицируя временные ряды по типам на основе их характеристик. Проблемы с данными временных рядов выходят за рамки выбросов и пропущенных значений и включают дрейф понятий, дисбаланс классов, тенденции обработки, а также проблемы сезонности и согласованности данных.

Согласно [4], анализ временных рядов в основном выполняется с использованием статистических методов; однако методы машинного обучения приобретают все большее значение в анализе временных рядов, обещая более точные результаты при адекватной предварительной обработке. Однако предварительная обработка остается жизненно важной для повышения качества данных и обеспечения надежных про
Такие методы, как SCREEN [6], предлагают методы очистки данных временных рядов на основе ограничений, повышая точность и эффективность анализа данных. В ходе научных исследований были изучены различные подходы к совершенствованию методов предварительной обработки данных временных рядов. В одном исследовании [7] метаобучение использовалось для выбора альтернатив предварительной обработки, уделяя особое внимание влиянию трансформации на производительность алгоритма в таких наборах данных, как «Автомобиль», «Кредит», «Ирис» и «Голс
Хотя деревья регрессии показали ограниченную эффективность в качестве метаобучающихся, трансформации значительно улучшили результаты. В другом исследовании [8] был представлен итеративный метод минимального исправления и прогнозирования параметров для устранения аномалий, что позволило существенно повысить скорость выполнения наборов данных, включая GPS, лабораторные данные Intel и метеорологические данные Китая. В отдельном исследовании [9] изображения временных рядов Landsat в сочетании с анализом траекторий использовались для обнаружения нестабильности лесов в Мьянме. Это исследование выявило корреляцию между областями нестабильности и вырубленными деревьями, что указывает на

Machine Translated by Google

потенциальные приложения для выборочного ведения журнала. Обнаружение изменений во временных рядах было тщательно проанализировано [10], включая методы предварительной обработки, такие как атмосферная коррекция и обнаружение облаков, а также популярные алгоритмы автоматической предварительной обработки, такие как LEDAPS и Fmask. Дополнение полностью сверточных сетей с помощью LSTM-RNN было предложено [11] для классификации временных рядов, демонстрируя улучшенную производительность с механизмами внимания. Кроме того, в исследованиях [12] рассматривались проблемы профилирования данных, предлагая визуальные программные решения для проверки качества данных и методы обнаружения выбросов с использованием таких методов, как метод k-скользящего окна и нейронные сети [13].

Обзор [14] подчеркнул необходимость дальнейших исследований в области инструментов и систем очистки временных рядов, подчеркнув важность предварительной обработки для максимизации потенциала данных. Потенциальные области будущей работы включают анализ типов данных об ошибках, алгоритмы очистки многомерных временных рядов, обнаружение аномалий и методы очистки данных конкретных приложений. Было предложено несколько инновационных систем для решения проблем очистки данных временных рядов в различных областях. Cleanits [15], например, фокусируется на очистке данных промышленных временных рядов, используя полосы Боллинджера для обнаружения аномалий и LSTM для исправления ошибок. Хотя он в первую очередь предназначен для промышленных данных, существует потенциал его применения и в других областях, требующих повышения гибкости.

В энергетическом секторе были предложены системы извлечения знаний [16] с использованием методов кластеризации, при этом k-Medoids показывают многообещающие результаты для энергетических временных рядов. Медицинская наука также извлекает выгоду из анализа данных временных рядов, что побуждает к разработке таких систем, как «Ясновидение» [17], способствующих быстрому созданию прототипов и оптимизации систем принятия клинических решений. Гибридные модели и системы, такие как AutoMTS, были внедрены для очистки данных многомерных временных рядов [18,19], демонстрируя их значимость для улучшения качества данных посредством вменения и обнаружения выбросов. AutoML и такие платформы, как DAEMON [20,21], сосредоточены на автоматизации подбора модели машинного обучения и обнаружении выбросов в данных временных рядов, достигая высокой точности и демонстрируя потенциал для онлайн- сервисов обнаружения аномалий [22].

Данные временных рядов с сезонностью создают уникальные проблемы, такие как мультисезонность и гетероскедастичность, которые усложняют анализ и прогнозирование. Исследования [23,24] решили эти проблемы путем предварительной обработки данных, построения моделей и их адаптации к исходным измерениям с последующим обнаружением аномалий. Дальнейшие исследования направлены на улучшение прогнозирования локальных тенденций и коррекцию выбросов при анализе временных рядов. Нестационарность рассматривалась в исследованиях, посвященных очистке энергозависимых и энергозависимых данных [25].
Были исследованы такие методы, как k-ближайшие соседи и скользящие окна [26], причем скользящие окна работают лучше, особенно для волатильных данных, таких как временные ряды фондового рынка [27]. Были предприняты усилия для измерения качества данных с использованием таких индексов, как индекс качества значений атрибутов (AVQI) и индекс качества значений структурированных данных (SDVQI), что позволяет сравнивать и улучшать показатели качества данных.

Алгоритмы оптимизации, такие как генетические алгоритмы (ГА), использовались для автоматизации выбора признаков при предварительной обработке, повышая эффективность регрессионных моделей [28]. Хотя существующие решения, такие как SCREEN, CLEANITS, DAEMON и Clairvoyance, решают конкретные проблемы предварительной обработки, существует потребность в общем подходе, охватывающем несколько областей. Preptimize выделяется тем, что предлагает интегрированное решение для предварительной обработки и рекомендации моделей , что делает его доступным для неспециалистов и способным решать разнообразные задачи, связанные с временными рядами. SCREEN и DAEMON специализируются на обнаружении и очистке аномалий, а CLEANITS специализируется на вменении данных. Ясновидение — это прежде всего инструмент прогнозирования, использующий передовые методы машинного обучения. Широкая применимость и удобный для пользователя подход Preptimize отличают его от других специализированных инструментов. Различные библиотеки с открытым исходным кодом и собственные продукты SaaS автоматизируют создание и оптимизацию конвейеров машинного обучения, оптимизируя процесс предварительной обработки [29,30].

Прогнозируемые значения временных рядов могут быть номинальными, категориальными, текстовыми и числовыми. Численное прогнозирование — это проблема регрессии, которую можно решить с использованием различных методов, таких как скользящие средние, авторегрессия и экспоненциальное сглаживание, а также продвинутых методов, таких как ARIMA, SARIMA, VAR и LSTM [31]. Гибридные подходы

Machine Translated by Google

также распространено сочетание статистических моделей и моделей машинного обучения [32–34]. Однако ни один метод не подходит для всех временных рядов из-за их различных свойств, что делает процесс выбора трудоемким и сложным. Чтобы упростить процесс анализа временного ряда, следует принять стандартную процедуру для автоматизации профилирования, предварительной обработки, прогнозирования оптимизированной модели, оценки и выбора наиболее эффективной модели с минимальной ошибкой.

Основным вкладом этой работы является внедрение Preptimize, автоматизированной стратегии анализа данных временных рядов, которая объединяет статистические методы и методы машинного обучения, чтобы рекомендовать оптимальные модели для анализа. Считывая выборку из большого набора данных, Preptimize предлагает модели, основанные на оптимизации, включая лучшее сочетание методов предварительной обработки и прогнозирования, тем самым повышая точность прогнозирования для различных типов данных временных рядов. Preptimize упрощает процесс анализа как одномерных, так и многомерных временных рядов с пропущенными значениями. Учитывая сложность анализа временных рядов, это исследование подчеркивает важность выбора правильного метода прогнозирования и стратегии предварительной обработки на основе типа данных временных рядов, чтобы ускорить процесс анализа данных временных рядов. Кроме того, эта работа продемонстрировала потенциал этой структуры для широкого применения в прогнозировании временных рядов, подтвердив ее эффективность в рекомендации подходящих моделей прогнозирования.

2. Материалы и методы.

Основываясь на обсуждении в предыдущем разделе, можно определить, что необходима комплексная структура, которая может определить стандартные шаги во время предварительной обработки и прогнозирования временных рядов. В следующем разделе содержится подробное объяснение предлагаемой структуры под названием Preptimize. Целью Preptimize является автоматизация процесса анализа временных рядов, предлагая первую модель для дальнейшего анализа. Предложенная структура была реализована с использованием языка программирования Python.

Preptimize

Preptimize начинается со считывания данных временных рядов и создания комплексного профиля , который включает анализ распределения, сезонность и обнаружение тенденций. Затем он предварительно обрабатывает данные, решая проблемы качества данных, такие как пропущенные значения и выбросы, с использованием различных методов, и оценивает предварительно обработанные данные с использованием нескольких моделей прогнозирования, чтобы определить наиболее эффективную модель. Эти шаги также упомянуты на рисунке 1. Поскольку разработка структуры является большой задачей, в этой статье основное внимание уделяется числовым данным и проблемам отсутствия данных. Целью этой работы является объяснение большинства доступных решений для обработки недостающих данных во временных рядах и предоставление наилучшего возможного решения. Предлагаемая структура тестируется на различных наборах данных, включая данные о криптовалютах, данные фондового рынка и данные, сгенерированные сетью Интернета вещей.

При реализации Preptimize используются многочисленные методы и алгоритмы , как показано в таблице 1. Правила, которым следует Preptimize, чтобы рекомендовать потенциальные проекты моделей прогнозирования, взяты из существующей литературы. Поэтому организация и оптимизация этих правил и выводов является значительным вкладом в эту работу. Базовый псевдокод предлагаемой структуры показан на рисунке 2. В следующем разделе освещаются только наиболее распространенные методы, используемые на различных этапах процесса.

Таблица 1. Подробная информация о методах, использованных при реализации Preptimize.

Этап	Подэтап	Подробная информация об используемых методах
Профилирование	Статистика	Дисперсия, стандартное отклонение, минимум, максимум, медиана, мода, недостающий процент, квартили, стационарность, распределение, эксцесс, асимметрия, мощность, корреляция
	Визуализация	График распределения, диаграмма рассеяния, линейный график, ящичковая диаграмма, гистограмма, график разложения тренда и сезона, корреляционная тепловая карта, график скользящей статистики , ACF, PACF

Machine Translated by Google

Таблица 1. Продолжение.

Этап	Подэтап	Подробная информация об используемых методах
Методы предварительной обработки	Трансформация	Преобразование для стационарности с использованием разности, логарифмической разницы, экспоненты, аддитивного разложения, мультипликативного разложения, скользящего среднего и стандартного отклонения, статистики ADF.
		Преобразование для нормального распределения с использованием Бокса – Кокса
	Отсутствие	MCAR, MAR или MNAR
	Удаление	По списку, попарно
	Одномерный	Скользящие средние, включая SMA, экспоненциальную, CMA
		Прямое заполнение (LOCF), обратное заполнение (NOCB)
	вменение	GAIN (на основе генеративно-состязательных сетей)
	Интерполяция	Линейный, акимовый, барицентрический, кубический, кубический сплайн, от производных, индекс, Крог, ближайший, контактная площадка, чип, полиномиальный, кусочно-полиномиальный, квадратичный, линейный, сплайн, время, ноль
	Многомерное вменение	DataWig, Байесовский хребет, случайный лес, K-соседи, дерево решений, Дополнительные деревья, KNNImputer, HotDeck
Прогнозирование	Одномерный	Статистические методы, включая скользящие средние (SMA, экспоненциальную, CMA) и другие сезонные и несезонные методы, включая AR, АРИМА, СЭС, САРИМА, HWES
		МЛП, СВР-РБФ, СВР-ПОЛИ, дерево решений, случайный лес, РНН, ГРУ, GRU-двунаправленный, Vanilla LSTM, Stacked LSTM, двунаправленный LSTM, CNN-LSTM
	Многомерный	Статистические методы, включая VAR, VMA, VARMA. MLP, дерево решений, случайный лес, RNN, GRU, GRU-двунаправленный, Сложенный LSTM

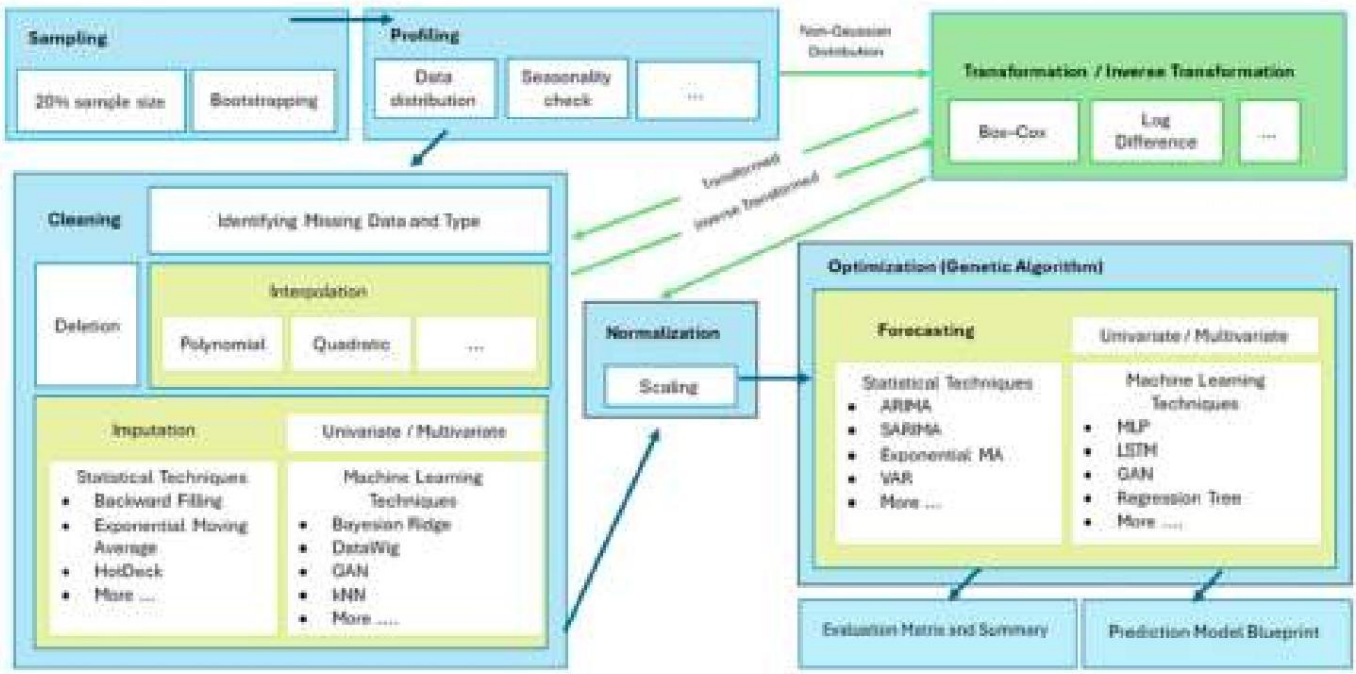


Рисунок 1. Предварительная оптимизация: визуальное изображение системного подхода, объединяющего этапы предварительной обработки, выбора модели и тщательной оценки, что повышает эффективность анализа временных рядов.

Machine Translated by Google


```
Инициализировать набор данных

Создайте резервную копию набора данных для последующей обработки нулевых значений.

# Профилируем набор данных

Инициализировать профиль как словарь

Если набор данных имеет несколько столбцов

    Установить профиль как многомерный

    Распечатать попарную корреляцию набора данных

    Отобразить тепловую карту корреляций

Еще

    Установить профиль как одномерный

Рассчитать общее количество записей и тип временного ряда (одномерный/многомерный)

# Подготовьте профиль каждого столбца

Для каждого столбца в наборе данных

    Инициализируйте dataAttribute как данные столбца.

    Если Univariate, установите UnivariateColumnName

    Проверьте отсутствие данных в столбце

    Вычисление значений nonNullValues и nullValues

    Если существуют недостающие данные

        Вычислить недостающий процент

        Отбросьте нулевые значения для дальнейшего анализа.

    Вычисление асимметрии и эксцесса dataAttribute

    Определить тип распределения (Нормальное/Ненормальное)

    Отображение профиля столбца с рассчитанной статистикой

    Постройте график сезонного разложения

    Постройте графики базового профиля

    Сохранить профиль для столбца в профиле

# Восстановить исходный набор данных из резервной копии

Инициализируйте отсутствующие данные и отсутствующиеColCorrelationCompleteCases.

Определить тип пропуска и корреляцию

Инициализировать обработанные наборы данных

# Обработка одномерного временного ряда

Если одномерный

    Получить профиль одномерного столбца

    Если существуют недостающие данные

        Если отсутствующий тип — MCAR и процент отсутствующих <= 10 %.

            Применить удаление по списку

        Еще

            Если данные числовые и не искажены

                Применять методы вменения

            Иначе, если данные не распределяются нормально

                Применить преобразование Бокса-Кокса
```

Рисунок 2. Продолжение.

Machine Translated by Google


```
                Если трансформация прошла успешно
                    Добавляйте пропущенные значения, используя несколько методов, и сохраняйте каждый набор данных в обработанных наборах данных.
                    Обратное преобразование данных с использованием Box – Cox
            Еще
                Установить обработанные наборы данных в качестве исходного набора данных
        # Прогнозирование одномерных временных рядов
        Инициализировать оценочную_матрицу
        Примените генетический алгоритм, чтобы выбрать лучшую модель прогнозирования на основе наивысшей пригодности или наименьшего RMSE.
            Генерируйте различные потенциальные решения, включая предварительно обработанный набор данных, одномерное прогнозирование.
            методы и настройки параметров
            Выберите лучшее решение из совокупности, используя выбор турнира.
        Если в предварительно обработанном наборе данных остаются пропущенные значения, удалите строки с нулями.
        Выполнение прогнозирования на основе выбранной методики и параметров прогнозирования.
        Сохранение прогнозируемых данных
        Оцените модель, рассчитав метрики ошибок и обновив матрицу оценки.
    Продолжать до n поколений
    Показать схему 5 наиболее эффективных моделей прогнозирования
    # Обработка многомерных временных рядов
    Иначе, если многомерное
        Если существуют недостающие данные
            Вычислить отсутствующий DatasetPerc
            Если отсутствуетDatasetPerc <= 10%
                Применить удаление по списку
            Еще
                Проверьте нормальность для всех столбцов
                Если все столбцы нормально распределены
                    Применить попарное удаление
                    Если в столбцах все еще отсутствуют данные
                        Применять одиночное и множественное вменение
                Еще
                    Применить преобразование Бокса-Кокса для ненормальных столбцов
                    Применять одиночное и множественное вменение
                    Обратное преобразование данных с использованием Box – Cox
        Еще
            Установить обработанные наборы данных в качестве исходного набора данных
        # Прогнозирование многомерных временных рядов
        Инициализировать оценочную_матрицу
        Примените генетический алгоритм, чтобы выбрать лучшую модель прогнозирования на основе наивысшей пригодности или наименьшего RMSE.
            Генерируйте различные потенциальные решения, включая предварительно обработанный набор данных, многомерное прогнозирование.
            методы и настройки параметров
            Выберите лучшее решение из совокупности, используя выбор турнира.
```

Рисунок 2. Продолжение.

Machine Translated by Google

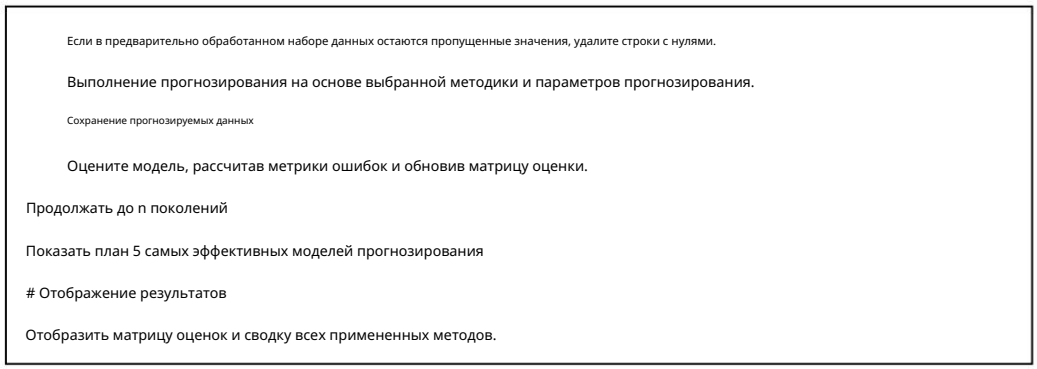


Рисунок 2. Предварительная оптимизация: последовательный рабочий процесс предварительной обработки данных временных рядов и выбора модели.

Здесь # представляет комментарий.

2.1. Выборка данных

На первом этапе Preptimize генерирует выборку данных для представления исходного поведения, уменьшая при этом размер до 20%. Была применена начальная загрузка, и индекс сохранился.

2.2. Профилирование

данных. На этом этапе вычисляются различные свойства данных и создается профиль для атрибута. Эти свойства включают среднее значение, медиану, минимальное и максимальное значение, дисперсию и стандартное отклонение данных. Данные дополнительно исследуются путем определения типа распределения, асимметрии и эксцесса данных. Также рассчитывается стационарность данных, включая сезонность и тренд. Краткое описание этих методик приведено ниже.

2.2.1. Распределение данных

Распределение данных определяет все вероятные значения переменной, а также количественно определяет относительную частоту появления, указывая, как часто появляется значение. Возможны различные распределения: нормальное, биномиальное, Пуассона или распределение Бернулли. Большинство ситуаций соответствуют нормальному распределению, которое выражается в виде колоколообразной и симметричной кривой вокруг линии x = μ. Это указывает на то, что по обе стороны от центра присутствует одинаковое количество значений. Распределени
0 и стандартное отклонение 1 — это стандартное распределение Гаусса.

Два свойства, связанные с распределением, — это асимметрия и эксцесс. Асимметрия является мерой регулярности. Ряд данных считается симметричным, если он выглядит одинаковым по обе стороны от средней точки. Куртозис измеряет частоту появления выбросов в хвосте. Данные с тяжелыми хвостами указывают на большее количество выбросов, а значение эксцесса высокое. Эксцесс нормального распределения равен 3. Если меньше 3, выбросов мало, а если больше 3, выбросов больше. Асимметрия может быть отрицательной, что указывает на то, что хвост находится слева; в противном случае хвост находится справа. Если асимметрия равна нулю, то распределение совершенно симметрично. Распределение можно считать нормальным, если значения асимметрии находятся в пределах от -3 до +3, а эксцесс находится в диапазоне от -10 до +10 при использовании стандартной ошибки измерения [35].

2.2.2. Стационарность

Временной ряд, характеристики которого не зависят от времени, является стационарным. Таким образом, наличие тренда или сезонности указывает на то, что ряд не является стационарным. На наблюдаемое свойство в разное время влияют тенденции и сезонность. Временной ряд состоит из трех компонентов: сезона, тренда и шума. Сезон можно определить как повторяющийся краткосрочный цикл в сериале. Тенденция — это наклон данных, который может увеличиваться или уменьшаться со временем, тогда как шум — это случайное изменение в ряду. Часто считается сложной задачей работать с данными временных рядов, которые содержат сезоны или тенденции. Построение временного ряда дает мгновенную информацию о наличии сезона и тенденции.

Machine Translated by Google

Существует два распространенных подхода к работе с сезонными временными рядами. Один из методов состоит в том, чтобы разделить ряд, удалив сезонный и трендовый компоненты, а затем использовать ряд для анализа или использовать любой другой метод преобразования для получения несезонного представления ряда. Другой способ — использовать специально разработанные методы прогнозирования сезонных рядов, такие как SARIMA.

2.2.3. Определение типа отсутствующих данных

При работе с отсутствующими данными выделяются три категории: отсутствующие совершенно случайно (MCAR), отсутствующие случайно (MAR) и отсутствующие неслучайно (MNAR). В одномерном анализе рассмотрение времени как переменной усложняет задачу, требуя учета характеристик временных рядов и ковариации для вменения. Для данных MCAR одномерное вменение относительно просто. Однако при многомерном анализе учитывается связь между столбцами . Данные MCAR случайно отсутствуют и часто могут быть заменены средними или медианными значениями, поскольку недостающие данные считаются восстанавливаемыми. MCAR определяется формулой (1), где R — отсутствующие данные, а Y — полные данные.

$$P(Ri \mid Yi) = P(Ri) \quad (1)$$

Данные MCAR не имеют значительной связи по таким показателям, как хи-квадрат, правило Крамера или коэффициент корреляции, и обычно содержат меньше нулевых значений. Для больших наборов данных экземпляры с отсутствующими данными могут быть опущены или импутированы. Методы удаления для MCAR включают удаление по списку, удаление записей с любым нулевым значением, подходящее, когда недостающие данные минимальны, и попарное удаление, сохраняющее данные для пар величин с доступными точками данных, сохраняющее больше данных. Вменение становится необходимым, когда методы удаления непригодны. Вменение включает замену отсутствующих данных ожидаемыми значениями с применением различных методов в зависимости от характера данных. В случае MAR недостающие данные могут следовать шаблону или последовательности, иногда появляясь вне последовательности. Когда выявляется связь между отсутствующими и существующими данными, это считается MAR. Для обработки MAR можно использовать такие методы вменения , как среднее значение, медиана, мода или интерполяция , причем для данных, не являющихся временными рядами, доступно несколько методов вменения, а для данных временных рядов — интерполяция. MAR определяется выражением (2)

$$P(Ri \mid Yi) = P(Ri \mid Yi, o) \quad (2)$$

В MNAR отдельные случаи делятся недостающими данными, что усложняет прогнозирование, поскольку оно опирается на ненаблюдаемую информацию. Если отсутствие корреляции в одном столбце незначительно или вообще не коррелирует с другими столбцами и при исследовании не обнаружено сильных ассоциаций, а частота пропущенных значений высока, это считается MNAR. MNAR нельзя игнорировать, и нет простого способа его обработки, кроме получения недостающих данных. MNAR обычно имеет больше нулевых значений по сравнению с MCAR и может быть идентифицирован только на основе знаний предметной области. MNAR определяется формулой (3):

$$P(Ri \mid Yi) \neq P(Ri \mid Yi, o) \quad (3)$$

2.3. Визуализация данных

Использование различных графиков и графиков является ценным подходом к анализу временных рядов для понимания поведения и свойств данных. Preptimize создает ряд визуализаций, таких как диаграммы рассеяния, линейные графики и графики разложения, чтобы обеспечить полный обзор данных временных рядов. Эти визуальные инструменты помогают понять поведение данных, выявить тенденции и сезонность, а также выбрать подходящие методы предварительной обработки.

Первый набор графиков предлагает комплексное представление данных, включая диаграммы рассеяния, линейные графики, коробчатые диаграммы и гистограммы. Каждый атрибут в многомерном временном ряду отображается индивидуально, чтобы отразить суть данных.

Другой важный график отображает распределение данных, облегчая оценку нормальность. Эта оценка важна для определения подходящих статистических методов.

Machine Translated by Google

График, иллюстрирующий разложение сезонов и тенденций, особенно полезен для выявления сезонных закономерностей во временном ряду. Это понимание помогает применять соответствующие методы, адаптированные к сезонным данным, что потенциально может потребовать преобразований для улучшения анализа.

Чтобы оценить стационарность данных, Preptimize создает график, отображающий скользящее среднее и скользящее стандартное отклонение в зависимости от исходных данных. Стационарность указывается, если эти значения остаются постоянными во времени [36].

Визуализация тепловой карты отображает корреляции между несколькими переменными, причем более высокие уровни корреляции представлены разными цветами. Это помогает понять силу взаимосвязей между переменными.

Дополнительные графики сосредоточены на недостающих данных, включая матрицу и гистограмму, которые дают представление о природе недостающих данных и их потенциальных взаимосвязях с другими атрибутами.

2.4. Очистка данных.

Фаза предварительной обработки при анализе временных рядов включает в себя очистку данных, направленную на решение различных проблем с качеством данных, таких как пропущенные значения, выбросы и шум. В отличие от других форм анализа данных, предварительная обработка временных рядов требует специальных методов, адаптированных к их уникальным характеристикам. Preptimize использует ряд методов вменения для обработки недостающих данных, включая, помимо прочего, линейную интерполяцию, сплайн-интерполяцию и методы машинного обучения, такие как GAN. Выбор метода вменения основан на конкретных характеристиках набора данных, что обеспечивает наилучшее качество данных для прогнозирования.
Имея около 25 различных методов интерполяции, эта структура направлена на изучение наиболее подходящего подхода для каждого набора данных. Методы преобразования применяются, когда данные отклоняются от нормальности или демонстрируют нестационарность. Преобразование Бокса-Кокса используется для устранения ненормальности, в то время как такие методы, как сезонная декомпозиция, используются для удаления тенденций и сезонности из нестационарных данных.

Подходы машинного обучения, в том числе генеративно-состязательные сети (GAN), интегрированы в структуру для улучшения прогнозирования недостающих данных. Для многомерных временных рядов используются итеративные модели вменения и DataWig для решения проблем с недостающими данными.

Для определения оптимального метода вменения проводится комплексная оценка с учетом эффективности прогнозирования и показателей оценки. В качестве альтернативы можно использовать метод грубой силы для проверки всех возможных методов вменения.

В целом, этап предварительной обработки играет решающую роль в подготовке данных временных рядов для анализа и прогнозирования, что требует применения специализированных методов, адаптированных к их уникальным характеристикам и задачам.

2.4.1. Прямое и обратное заполнение

Перенос последнего наблюдения (LOCF) — это широко используемый метод однократного вменения, который вменяет последнее измеренное значение путем прямого заполнения. В LOCF должен быть хотя бы один показатель после исходного уровня. Следующее наблюдение, переносимое назад (NOCB), является методом, аналогичным LOCF, но работает в обратном направлении за счет обратного заполнения; то есть он использует первое записанное значение после недоступного значения и переносит его назад.

2.4.2. Интерполяция

Методы интерполяции предлагают средства для прогнозирования отсутствующих точек данных путем оценки значений между известными точками данных. Линейная интерполяция, самый простой метод, соединяет соседние точки прямыми линиями, предполагая, что точки на линии представляют недостающие значения. Полиномиальная интерполяция, которая является более универсальной, использует полиномиальные функции разного порядка для представления отношений между атрибутами. Интерполяция полиномом Лагранжа и интерполяция полиномом Ньютона являются примерами методов полиномиальной интерполяции. Интерполяция Лагранжа, хотя и универсальна, может вызывать затруднения при работе с большими наборами данных.

Machine Translated by Google

Интерполяция полиномом Ньютона использует процесс рекурсивного деления для построения сложных полиномов, который может быть эффективен для прогнозирования, но может привести к числовым сложностям и чувствительности к выбросам. Сплайн-интерполяция, с другой стороны, использует кусочные полиномы для подгонки более гладких кривых к меньшим группам точек данных, при этом интерполяция кубическим сплайном часто предпочтительна из-за ее гладкости и стабильности.

Другие методы интерполяции, используемые в рамках, включают кусочно-полиномиальный, Акима, Барицентрический, Крога и Pchip, каждый из которых предлагает разные подходы к оценке недостающих значений в данных временных рядов.

2.4.3. Скользящие средние

Скользящие средние, такие как простые скользящие средние (SMA), кумулятивные скользящие средние (CMA) и экспоненциальные скользящие средние (EMA), обычно используются для вменения при анализе временных рядов. SMA вычисляет среднее значение указанного количества точек данных, используя простую формулу суммирования. CMA вычисляет среднее значение для всех записей до текущего момента, при этом каждое значение имеет одинаковый вес. EMA использует экспоненциально убывающие весовые коэффициенты для вычисления среднего значения, причем значение в любой момент определяется коэффициентом и предыдущим значением EMA. Эти скользящие средние могут помочь в расчете данных при условии, что временной ряд стационарен. Стационарность, показывающая стабильное поведение во времени, можно оценить с помощью статистических тестов, таких как расширенный тест Дикки-Фуллера (ADF) [37] , или путем наблюдения за скользящей статистикой, такой как графики скользящего среднего и стандартного

2.4.4. Алгоритмы глубокого обучения

Вменение данных с использованием методов глубокого обучения, в частности рекуррентных нейронных сетей (RNN) и генеративно-состязательных сетей (GAN), в недавней литературе стало многообещающим подходом к анализу временных рядов. RNN и их вариации превосходны в моделировании данных последовательностей, в то время как GAN умеют генерировать и вычислять пропущенные значения. Preptimize использует GAN как для одиночного, так и для множественного вменения. GAN работает с помощью двух моделей: генератора (G) и дискриминатора (D), которые работают состязательно, генерируя реалистичные выборки данных и различая реальные и поддельные выборки соответственно . Обучение продолжается до тех пор, пока дискриминатор не сможет различать реальные и сгенерированные образцы, что указывает на способность генератора создавать достоверные примеры.

В GAN D пытается присвоить выборкам правильную метку с максимальной вероятностью, тогда как G пытается минимизировать log (1 D(G(z))). Согласно [38], D и G играют в минимаксную игру для двух игроков с функцией цены V (G, D), как показано (4).

$$\min_G \max_D V(D, G) = E_{x \sim pdata(x)}[\log D(x)] + E_{z \sim pz(z)}[\log(1 - D(G(Z)))] \quad (4)$$

В Preptimize модель, представленная в [39], используется для вменения недостающих данных с использованием GAN. Этот метод называется «Генераторно-состязательные сети вменения» (GAIN). В этом методе G наблюдает за реальными данными, рассчитывает недостающий элемент и выводит окончательные данные. Затем D берет окончательные данные и пытается определить, какие элементы наблюдаются, а какие рассчитываются.

2.4.5. Множественное вменение

Множественное вменение (MI) предлагает комплексный подход к обработке пропущенных значений за счет использования полного набора функций для аппроксимации пропущенных значений. MI генерирует несколько наборов данных путем вменения пропущенных значений, что позволяет заменять пропущенные значения несколькими возможными значениями. Эти вмененные значения оцениваются на основе каждого набора данных, а затем агрегируются в окончательный набор данных. МИ применим в случаях, когда данные отсутствуют совершенно случайно (MCAR), когда они отсутствуют случайным образом (MAR) и даже когда данные отсутствуют неслучайно (MNAR).

С другой стороны, итеративное вменение включает моделирование каждого признака как функции других признаков в последовательном процессе. Прошлые вмененные значения используются для прогнозирования последующих объектов до тех пор, пока все недостающие значения не будут заполнены. Preptimize использует библиотеку IterativeIr

Machine Translated by Google

Python, который предлагает различные средства оценки, такие как регрессор Байесовского хребта, регрессор случайного леса, регрессор K-соседей, регрессор дерева решений и регрессор дополнительных деревьев.

Кроме того, метод KNNImputer, предоставляемый той же библиотекой Python, использует подход k-ближайших соседей для вменения пропущенных значений путем рассмотрения значений наблюдений для соседних точек данных.

DataWig, еще один метод, используемый для вменения, в литературе показал эффективность вменения пропущенных значений. DataWig [40] — это библиотека глубокого обучения, которая использует MXNet в качестве бэкэнда для генерации прогнозов для отсутствующих данных.

Кроме того, для вменения используется метод «горячей колоды» [41] , известный своей вычислительной эффективностью. Этот метод вменяет недостающие значения, выбирая случайно похожую запись в матрице данных и используя существующие значения из объекта-донора. Горячая колода работает с одним атрибутом одновременно и служит единым методом вменения.

2.5. Прогнозирование

В литературе существуют различные подходы к прогнозированию временных рядов. Примеры включают экспоненциальное сглаживание, авторегрессионное интегрированное скользящее среднее (ARIMA) с сезонными и несезонными вариантами, несколько методов глубокого обучения и многие другие. Существуют отдельные методы прогнозирования одномерных и многомерных временных рядов. Preptimize использует генетический алгоритм [42] для исследования огромного пространства возможных моделей и конфигураций, чтобы найти наиболее подходящую для данного набора данных. Каждый человек в популяции представляет собой потенциальное решение, включающее стратегию предварительной обработки, модель прогнозирования и настройки параметров. Каждый человек оценивается на основе функции приспособленности, которой в данном случае является RMSE. Чем ниже RMSE, тем Люди отбираются на основе их показателей физической подготовки, чтобы стать родителями следующего поколения. Мутация осуществляется по схеме переворота битов; Из популяции случайным образом с помощью турнирного отбора выбираются 3 человека. Этот подход помогает автоматизировать процесс выбора модели, потенциально экономя значительное время и усилия, обеспечивая при этом оптимальную или почти оптимальную точность прогнозирования.

2.5.1. Одномерное прогнозирование

Одномерное прогнозирование включает в себя прогнозирование будущих значений одной переменной на основе исторических данных за последовательные интервалы времени. Выбор метода прогнозирования зависит от того, проявляет ли временной ряд сезонность и трендовые компоненты.

Существуют два основных типа методов одномерного прогнозирования: статистические методы и передовые методы, основанные на машинном обучении. Статистические методы включают авторегрессию (AR), скользящее среднее (MA), авторегрессионное скользящее среднее (ARMA), авторегрессионное интегрированное скользящее среднее (ARIMA) и простое экспоненциальное сглаживание. Распространены методы скользящего среднего , такие как простое скользящее среднее, экспоненциальное скользящее среднее и кумулятивное скользящее среднее.

Для временных рядов с сезонностью и трендом варианты включают сезонное авторегрессионное интегрированное скользящее среднее (SARIMA), сезонное авторегрессионное интегрированное скользящее среднее с экзогенными регрессорами и экспоненциальное сглаживание Холта – Уинтерса (HWES), причем наиболее успешными являются ARIMA и SARIMA.

ARIMA сочетает в себе авторегрессию, скользящее среднее и дифференцирование для достижения стационарности . Правильная идентификация параметров ARIMA (p, d, q) имеет решающее значение и обычно выполняется путем подбора моделей разного порядка с использованием таких критериев, как информационный критерий Акаике (AIC), для выбора наилучшего порядка. AIC измеряет производительность модели относительно количества параметров. Автокорреляционная функция (ACF) и частичная автокорреляционная функция (PACF) не могут использоваться для определения надежных значений p и q [43]. Остаточный анализ важен при моделировании ARIMA, чтобы гарантировать, что остатки напоминают белый шум, что указывает на отсутствие автокорреляции. Тест Люнга-Бокса обычно используется для оценки остаточной автокорреляции, где значения p выше порогового значения предполагают отсутствие значительной автокорреляции.

Прогнозирование временных рядов привлекло значительное внимание в машинном и глубоком обучении. Preptimize также учитывает эти методы прогнозирования на основе машинного обучения.

Machine Translated by Google

В литературе представлены различные методы для этой цели, в том числе длинная краткосрочная память (LSTM), регрессия опорных векторов (SVR), многослойный перцептрон (MLP), деревья решений, случайный лес, рекуррентная нейронная сеть (RNN) и ее методы. вариант Закрытый рекуррентный блок (ГРУ).

Среди этих методов выделяется LSTM, известный своей эффективностью при обработке последовательных данных. Однако традиционные RNN сталкиваются с такими проблемами, как исчезновение и взрывной градиент, что приводит к увеличению времени обучения и снижению точности. Эти проблемы решаются LSTM и GRU, которые более эффективно решают проблемы градиента.

GRU отличается от RNN использованием шлюзов обновления и сброса для управления потоком информации и решения проблем градиента. Эти ворота определяют, сколько предшествующей информации сохраняется или отбрасывается, повышая эффективность модели. В предлагаемой структуре используются варианты GRU, включая двунаправленные версии.

LSTM, еще один вариант RNN, состоит из нескольких блоков LSTM, каждый из которых оснащен ячейкой, входом, выходом и вентилями забывания. Эти ворота играют решающую роль в модели LSTM, облегчая последовательную обработку данных и генерацию прогнозов.

Эта платформа использует различные конфигурации нейронных сетей для прогнозирования, включая Vanilla LSTM, Stacked LSTM, двунаправленный LSTM и CNN-LSTM. Vanilla LSTM состоит из одного скрытого слоя и одного выходного слоя, а Stacked LSTM включает несколько скрытых слоев. Двунаправленный LSTM обрабатывает данные как в прямом, так и в обратном направлениях, а затем объединяет интерпретации. CNN-LSTM интегрирует CNN для понимания частей последовательности, а затем передает их в LSTM для интерпретации, образуя гибридную модель.

Сравнивая GRU и LSTM, явного победителя по всем параметрам производительности не выявлено . GRU выполняется быстрее с меньшим количеством параметров и меньшим использованием памяти, а LSTM демонстрирует большую точность, особенно на больших наборах данных.

MLP служит базовой искусственной нейронной сетью с полностью связанными слоями прямой связи, подходящими для задач классификации и регрессии. Он включает в себя как минимум входной слой, скрытый слой и выходной уровень, использующие нелинейные функции активации для обработки линейно неразделимых данных.

Для задач регрессии регрессия опорных векторов (SVR) обеспечивает гибкость в определении толерантности к ошибкам, стремясь минимизировать норму L2 вектора коэффициентов, допуская при этом отклонения со слабыми переменными. Деревья решений, включая деревья регрессии, используются для прогнозирования временных рядов, а случайный лес объединяет несколько деревьев регрессии для повышения точности прогнозирования за счет группирования и случайности признаков.

2.5.2. Многомерное прогнозирование

В предлагаемой структуре многомерное прогнозирование временных рядов использует ряд статистических методов и методов машинного обучения. Хотя статистические подходы к многомерным временным рядам ограничены, структура включает в себя такие ключевые методы, как векторная авторегрессия (VAR), векторное скользящее среднее (VMA) и векторное скользящее среднее авторегрессии (VARMA).

В отличие от одномерных методов, таких как AR и ARIMA, векторные методы, такие как VAR, VMA и VARMA, являются двунаправленными, что позволяет переменным влиять друг на друга. Однако эти методы требуют стационарности, что часто подтверждается с помощью теста ADF. VAR устанавливает взаимосвязи между переменными через систему уравнений, при этом выбор лага определяется такими критериями, как информационный критерий Акаике (AIC). VMA работает с матрицей скользящего среднего и k-мерным вектором, а VARMA объединяет порядки авторегрессии и скользящего среднего. Для прогнозирования на основе машинного обучения в платформе используются такие методы, как MLP, дерево регрессии, случайный лес, RNN, GRU и LSTM, каждый из которых подробно описан в предыдущем разделе.

3. Экспериментальные результаты и обсуждение.

В этом исследовании оценивается Preptimize, платформа, предназначенная для анализа данных временных рядов в различных областях и сценариях, включая данные фондового рынка, данные о криптовалютах и данные о энергопотреблении, генерируемые Интернетом вещей. Каждый эксперимент включает в себя предварительную обработку

Machine Translated by Google

данные, применяя различные методы прогнозирования и оценивая производительность с использованием таких показателей, как MAE, MSE, RMSE и MAPE. Подробная информация об используемых наборах данных приведена в Таблице 2. В системе используются методы предварительной обработки и прогнозирования, адаптированные к каждому типу набора данных, при этом выбор оптимизируется с помощью генетического алгоритма (ГА). Preptimize предназначен для использования GA для оптимизации выбора модели. Чтобы обеспечить полное понимание функциональности платформы, были оценены все возможные модели. ГА был адаптирован для включения всех потенциальных моделей, определяя как лучшие, так и худшие показатели. Это было достигнуто за счет присвоения наивысшего значения наиболее эффективным моделям. Однако в практических приложениях GA обычно рекомендует пять наиболее эффективных моделей, основанных на самом низком RMSE, балансируя между эффективностью и результативностью. Метрики оценки включают MAE, MSE, RMSE и MAPE, а подробный анализ представлен в подробном отчете, созданном Preptimize. Эксперименты проводятся отдельно для одномерных и многомерных наборов данных, охватывающих данные о фондовом рынке, криптовалюте и домашнем энергопотреблении. Каждый эксперимент тщательно описан, включая детали набора данных, процедуры обучения/тестирования и результаты оценки. Подробности и результаты этих экспериментов приведены ниже.

Таблица 2. Подробная информация о наборах данных, использованных в экспериментах.

Имя	Описание
Эксперимент 1. Одномерный временной ряд	
Фондовый рынок	Символ: MSFT (Microsoft) Столбцы: Дата, Закрыть Количество записей: 503 Даты: с 3 августа 2020 г. по 1 августа 2022 г. Интервал: Ежедневно
Фондовый рынок	То же, что 1-A, но без данных.
Криптовалюта	Криптовалюта: BTC-USD (Биткойн) Столбцы: Дата, Закрыть Количество записей: 748 Даты: с 01 августа 2020 г. по 17 августа 2022 г. Интервал: Ежедневно
Эксперимент 2. Многомерные временные ряды	
Интернет вещей: энергопотребление	Домен: Главная Столбцы: Дата, Global_active_power, Глобальная_реактивная_мощность, Напряжение, Глобальная_интенсивность, Суб_измерение_1, Суб_измерение_2, Суб_измерение_3 Количество записей: 10 081 Даты: с 24 апреля 2007 г. по 30 апреля 2007 г. Интервал: Минута
Фондовый рынок	Символ: MSFT (Microsoft) Столбцы: Дата, Открытие, Максимум, Минимум, Закрытие Количество записей: 503 Даты: с 3 августа 2020 г. по 1 августа 2022 г. Интервал: Ежедневно

3.1. Эксперимент 1: одномерный ряд

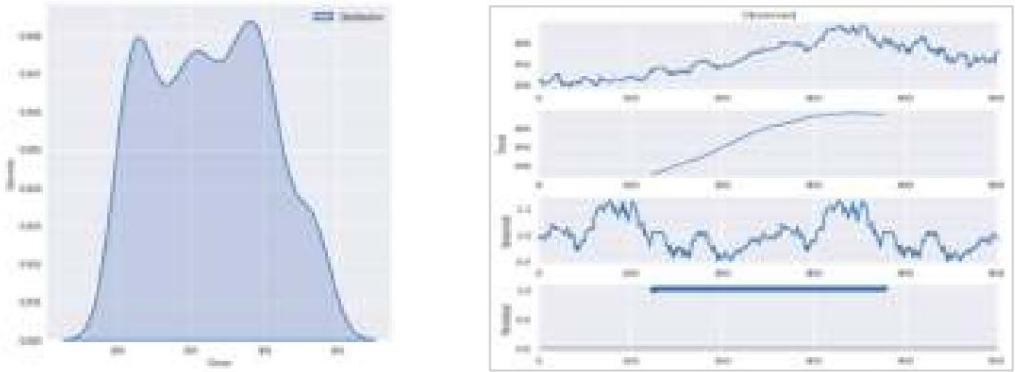
Preptimize прошел первоначальное тестирование на рыночных данных, включая акции и криптовалюты , распространенный тип данных временных рядов, характеризующийся колебаниями цен на листинги с течением времени. Эксперимент 1 разделен на три части. В части 1 использовались полные данные о запасах для оценки эффективности Preptimize на высококачественных данных. Во второй части были намеренно пропущены данные для оценки обработки неполных наборов данных платформой. В части 3 использовались данные о криптовалюте с отсутствующими значениями для дальнейшего тестирования возможностей Preptimize. Эти эксперименты были направлены на оценку эффективности системы в различных сценариях и качестве данных в области финансового прогнозирования.

В первой части эксперимента в этом исследовании изучаются данные фондового рынка от Yahoo Finance с упором на корпорацию Microsoft (MSFT), котирующуюся на бирже NASDAQ.

Machine Translated by Google

С помощью библиотеки Python yfinance были собраны рыночные данные с августа 2020 года по август 2022 года, в частности, изучались цены закрытия в каждый торговый день, исключая выходные.
Это представляет собой одномерный набор данных временных рядов, содержащий около 500 записей, которые были разделены на обучающие и тестовые наборы. Примечательно, что набор данных был полным и не имел проблем с качеством, что устраняло необходимость предварительной обработки на начальном этапе эксперимента.

Для прогнозирования временных рядов было использовано в общей сложности 139 моделей, при этом созданный отчет содержал обширный обзор процесса анализа данных. Несмотря на ограниченность объема, отчет включает в себя такие важные функции, как основные характеристики профиля, графики скользящего среднего и стандартного отклонения, диаграммы рассеяния, коробчатые диаграммы, графики распределения и графики разложения тенденций. Кроме того, отчет включает журналы обучения, расчеты ошибок и оценки производительности для всех моделей. На рисунках 3a,b показано распределение данных , указывающее на нормальность, и показано разложение тренда и сезонности внутри временного ряда. Комплексный анализ, включая значения p и расширенную статистику Дикки-Фуллера (ADF), предполагает стационарность временного ряда без заметно
В сводном файле производительности моделей содержатся результаты примерно по 139 моделям, из которых некоторые упомянуты в Таблице 3, показывая только 5 лучших и 5 худших показателей.



(а) (б)

Рисунок 3. (а) На графике показано нормальное распределение данных цен закрытия MSFT за 2 года. (б) График разложения цен закрытия для разделения тренда и сезонности. График демонстрирует четкую тенденцию к росту и отсутствие сезонности.

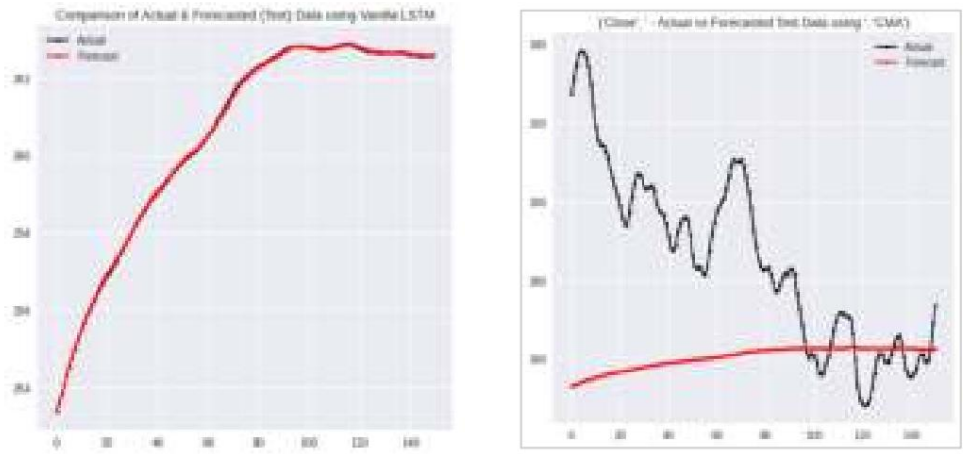
К одномерному набору данных фондового рынка применялись различные методы прогнозирования, включающие как машинное обучение, так и статистические методы. Методы машинного обучения , такие как LSTM, GRU и MLP, а также статистические методы, такие как простое скользящее среднее, экспоненциальное скользящее среднее и ARIMA, были протестированы с различными гиперпараметрами, включая скрытые слои, нейроны и эпохи. Аналогичным образом оценивались различные варианты статистических методов, например, различные порядки для ARIMA. Эти модели прошли оценку на тестовом наборе данных с использованием различных показателей.

Результаты показали, что модель глубокого обучения Vanilla LSTM работает лучше всего с набором данных, достигая RMSE 0,0294 с 50 эпохами и 40 единицами. Прогнозируемые значения, полученные с помощью этой модели, сравнивались с ожидаемыми значениями на тестовом наборе, как показано на рисунке 4a. Среди статистических методов лучше всего показала себя экспоненциальная скользящая средняя (EMA) с MAE 2,0925 и RMSE 2,5814. EMA заняла 127-е место из 139 моделей. И наоборот, модель на основе кумулятивного скользящего среднего (CMA) показала худшие результаты: MAE 27,2085 и RMSE 35,1176. На рисунке 4b показано сравнение фактических значений и прогнозируемых значений, полученных с помощью модели CMA.

Machine Translated by Google

Таблица 3. Сводная информация об эффективности модели эксперимента на основе одномерных временных рядов.

Классифицировать	Предварительная обработка	Модель	Настройки	МАЭ	RMSE
Часть 1. Временные ряды фондового рынка без проблем с качеством данных					
1	NA	Ванильный ЛСТМ	Эпоха = 50, юниты = 40	0,02	0,03
2	NA	ГРУ	Эпоха = 75, юниты = 50	0,02	0,03
3	NA	Ванильный ЛСТМ	Эпоха = 25, юниты = 50	0,02	0,03
4	NA	ГРУ	Эпоха = 50, юниты = 40	0,03	0,04
127	NA	Экспоненциальный	-	2.09	2,58
135	NA	Регрессор дерева решений	Разделение = MSE	7.57	8.03
136	NA	Случайный Лес	100 деревьев, разделение = MSE	7,69	8.15
138	NA	АР	-	21.11	26,88
139	NA	СМА	-	27.21	35.12
Часть 2. Временные ряды фондового рынка с отсутствующими данными					
1	СМА	АРИМА	(7, 1, 2)	0,57	0,61
2	Сплайн 3	АРИМА	(7, 1, 7)	0,65	0,84
	Экспоненциальный	Экспоненциальный	-	1,92	2.43
6 59	Время	СВР	Ядро = поли	3,64	4,73
64	Производные	СВР	Ядро = РБФ	3,68	4,75
69	Кусочный полином	Сложенный LSTM	Эпоха = 100, юниты = 50	3,66	4,76
70	Производные	РНН	Эпоха = 25, юниты = 50	3,67	4,76
1269	Полином 7	СМА	-	27.82	35,76
1270	Полином 9	СМА	-	28.46	36,52
Часть 3: Временные ряды криптовалют с отсутствующими данными					
	ПРИРОСТ	Двунаправленный LSTM	Эпоха = 100, юниты = 50	9,09	10.03
1		Сложенный LSTM	Эпоха = 100, юниты = 50	8,86	10.64
2	Индексный чип	Ванильный ЛСТМ	Эпоха = 100, юниты = 50	10,82	12.11
3	Экспоненциальный	ГРУ-двунаправленный	Эпоха = 25, юниты = 50	12,95	14,67
10	Кубический	АР	-	243,97	291.00
693	GAIN	DT-регрессор	Разделение = MSE	260,87	324,39
703	чип	Случайный лес	100 деревьев, разделение = MSE	292,03	362,15
729 824	СМА	СВР	Ядро = полиномиальный	2575,77	2576,76
892	Сплайн 5	СМА	-	7431,54	9344.14



(а) (б)

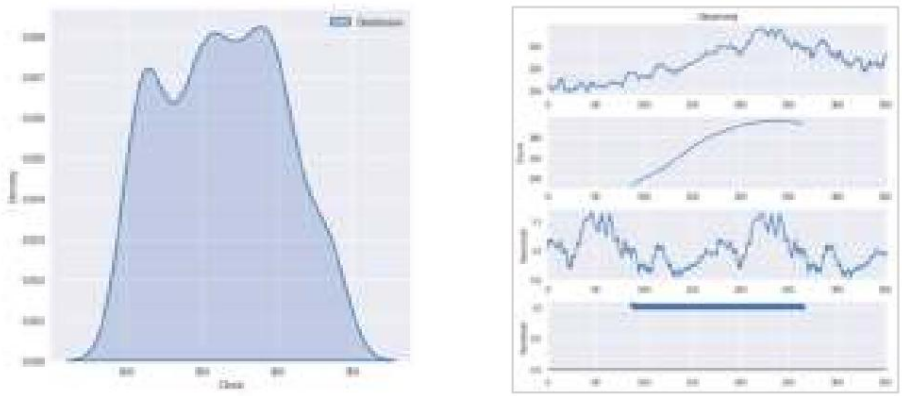
Рисунок 4. (а) Сравнение фактических и прогнозируемых данных, полученных с помощью Vanilla LSTM с использованием 50 эпох и 40 единиц. (б) Сравнение фактических и прогнозируемых данных, полученных с помощью кумулятивного скользящего среднего (СМА).

ARIMA выделяется как один из наиболее эффективных и действенных методов анализа временных рядов. прогнозирование. В Preptimize ARIMA был реализован с оптимизированным порядком (2, 1, 2), подтверждено тестом Юнга-Бокса. Однако, несмотря на эти оптимизации, в результате Среднеквадратическое значение 3,6001 не было особенно впечатляющим.

Machine Translated by Google

Набор данных фондового рынка использовался во второй части эксперимента с проблемами отсутствия данных. Как упоминалось ранее, данные были намеренно удалены, чтобы проверить производительность платформы путем сравнения их с исходными данными.

К этому набору данных, содержащему отсутствующие кортежи, была применена программа Preptimize; платформа создала отчет объемом более 3500 страниц с подробным описанием шагов, которые были применены к данному набору данных, включая предварительную обработку и окончательный анализ. Помимо других подробностей, в отчете содержались сведения о типе отсутствующих данных. В рамках этой системы данные резюмировались как отсутствующие полностью случайным образом (MCAR), нормально распределенные и стационарные, не имеющие сезона и тенденции. На рисунках 5a,b показано распределение тренда и сезона временных рядов и распределение данных соответственно.



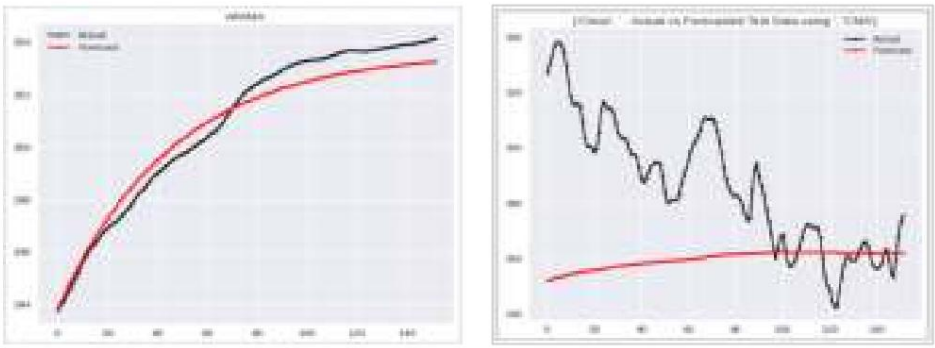
(а) (б)

Рисунок 5. (а) График разложения цены закрытия для разделения трендовой и сезонной составляющей. На графике виден небольшой восходящий тренд и отсутствие сезонности. (б) На графике показано нормальное распределение данных о ценах закрытия.

Сводная информация о производительности всех моделей, примененных к данным, была рассчитана и сохранена на листе. Итоговый лист показал, что было применено около 1270 моделей. Эти 1270 моделей были созданы в результате различных комбинаций, сформированных методами предварительной обработки и прогнозирования. Например, сначала для вменения недостающих данных применялась стратегия предварительной обработки, такая как прямое заполнение, затем применялись все другие методы прогнозирования для одномерного прогнозирования, включая статистические методы и методы машинного обучения, а затем каждая модель оценивалась с использованием четырех различных показателей оценки производител

Эксперимент включал в себя опробование нескольких методов и моделей предварительной обработки, включая интерполяцию, прямое и обратное заполнение, скользящие средние и методы глубокого обучения, такие как GAN, для вменения. Каждая модель прошла оценку на тестовом наборе данных с использованием различных показателей. Результаты показали, что ARIMA (p, d, q) порядка (7, 1, 2) работала лучше всего, когда недостающие данные были вменены с использованием CMA. Тест Люнга–Бокса подтвердил отсутствие автокорреляции белого шума и модели. Эта модель ARIMA дала MAE 0,56997 и RMSE 0,6099. Пять лучших моделей преимущественно принадлежали ARIMA и использовали различные методы вменения, такие как сплайн 3-го порядка, полиномиальный 5-й порядок, обратное заполнение и заполнение. После ARIMA экспоненциальное скользящее среднее показало хорошие результаты при использовании нескольких методов вменения, включая линейные, линейные, кусочно- полиномиальные и производные методы. Первые 85 моделей, показавших хорошие результаты, были статистическими моделями. И наоборот, модель на основе CMA, обработанная полиномом степени 9, показала худшие результаты: MAE 28,4643 и RMSE 36,5159. Рисунок 6 иллюстрирует производительность лучшей и худшей моделей в эксперименте, при этом на рисунке 6a показаны значения, предсказанные моделью ARIMA (7,1,2) в наборе данных, рассчитанном CMA, по сравнению с исходными значениями тестового набора, а на рисунке 6 показаны результаты, полученные с помощью модели ARIMA (7,1,2). Рисунок 6b демонстрирует производительность модели CMA. Регрессия опорного вектора с моделью полиномиального ядра, рассчитанной с использованием временной интерполяции, оказалась лучшей в категории машинного обучения, получив MAE 3,6430 и RMSE 4,7306. Погрешности, рассчитанные для указанных моделей, сведены в табл. 2.

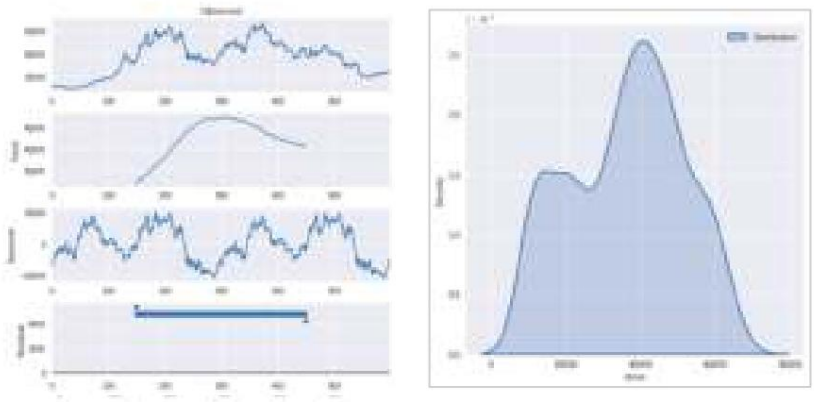
Machine Translated by Google



(а) (б)

Рисунок 6. (а) Сравнение фактических и прогнозируемых данных, полученных с помощью ARIMA (7,1,2), условно рассчитанных CMA. (б) Сравнение фактических и прогнозируемых данных, полученных с помощью CMA, рассчитанных с помощью полинома интерполяция степени 9.

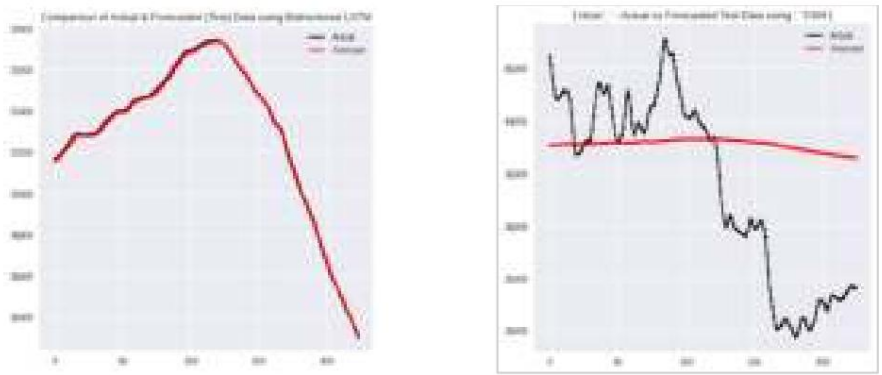
Третья часть эксперимента была сосредоточена на анализе данных о криптовалютах, в частности о биткойнах (BTC-USD) с биржи Coinbase. Криптовалюта работает независимо от традиционных банковских систем, используя технологию блокчейн для транзакций.
аутентификация и ведение записей. Набор данных, охватывающий примерно два года с
С августа 2020 года по август 2022 года было зарегистрировано 747 записей цен на биткойны в долларах США. Чтобы оценить производительность платформы при обработке недостающих данных, 30% набора данных было случайным
удаленный. Preptimize обработал данные, определив их как стационарные без видимых
сезонность или тенденция. Несмотря на числовой характер набора данных и отсутствие асимметрии, он демонстрировал значительную дисперсию, что указывает на значительную волатильность цен за этот период. Рисунок 7а,б
показать тренд и сезонное разделение временных рядов и распределение данных соответственно.
Для устранения пропущенных значений использовались различные методы вменения, включая методы глубокого обучения, интерполяцию и скользящие средние, с последующим набором данных. разбиты на обучающую и тестовую выборки для прогнозирования. Всего было обучено 892 модели. оценивается, при этом двунаправленный LSTM в сочетании с вменением GAIN становится лучшим исполнитель, достигнув RMSE 10,028 и MAE 9,0879. Среди статистических методов
авторегрессия (AR) показала многообещающие результаты в сочетании с кубической интерполяцией, заняв 693 место.
И наоборот, модель на основе CMA со сплайн-интерполяцией и полиномом пятой степени продемонстрировал самые плохие результаты: RMSE 9344,1381 и MAE 7431,54. Примечательно , что древовидные алгоритмы машинного обучения работали неоптимально, при этом решение модели древесного регрессора и случайного леса с рейтингом 703 и 729 соответственно, как показано на рис. Таблица 2. Графики производительности лучшей и худшей моделей представлены на рисунке 8.



(а) (б)

Рисунок 7. (а) График разложения цен закрытия Биткойна (б) Распределение данных: на графике показано нормальное распределение данных о цене закрытия Биткойна в долларах США.

Machine Translated by Google



(а) (б)

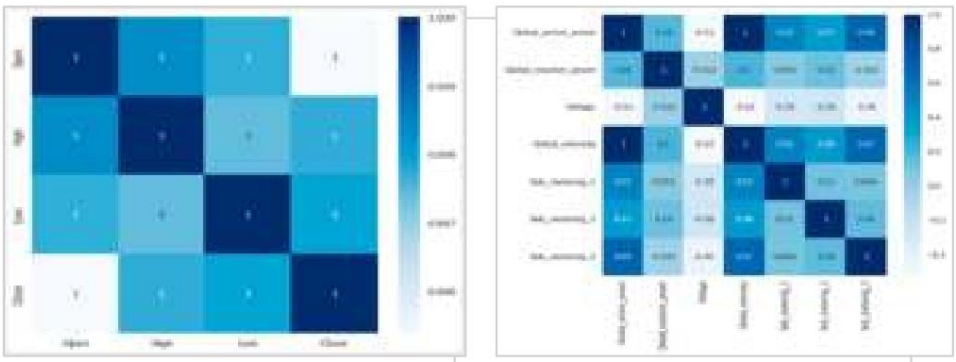
Рисунок 8. (a) Сравнение фактических и прогнозируемых данных, полученных с помощью вмененного двунаправленного LSTM от GAIN. (b) Сравнение фактических и прогнозируемых рядов, полученных с помощью CMA, условно рассчитанных с помощью сплайна Полином 5-й степени.

Результаты показывают, что Preptimize хорошо работает с различными типами временных рядов. данные. Например, модель Vanilla LSTM показала лучшие результаты на фондовом рынке. данные со RMSE 0,0294, в то время как модели ARIMA с использованием различных методов вменения лучше всего работает с наборами данных с пропущенными значениями, что подчеркивает универсальность платформы. и эффективность.

3.2. Эксперимент 2: Многомерный ряд

Второй эксперимент был сосредоточен на оценке производительности Preptimize на данных многомерных временных рядов с пропущенными значениями и без них. Полные данные со склада market использовались в первой части, а во второй части были добавлены недостающие экземпляры с использованием набора данных о потреблении электроэнергии отдельными домохозяйствами.

В первой части набор данных фондового рынка отражал показатели Microsoft (MSFT). на основе значений открытия, максимума, минимума и закрытия с 3 августа 2020 г. по 1 августа 2022 г., всего 503 записи. Preptimize началась с профилирования данных, которое выявило высокую парную корреляцию. среди столбцов, обозначающих зависимость. Однако «открытые» и «высокие» столбцы были оказались нестационарными и потенциально содержащими компоненты тренда и сезонности, поскольку изображено на Фигуре 9; тепловая карта иллюстрирует парную корреляцию.



(а) (б)

Рисунок 9. (a) Тепловая карта корреляции, показывающая корреляцию между парами в многомерном наборе данных фондового рынка. (б) Тепловая карта корреляции, показывающая корреляцию между парами в энергопотреблении многомерный набор данных.

Переход к прогнозированию, поскольку предварительная обработка не требовалась. Предварительная оптимизация применяемых различных методов, включая VAR, VMA, VARMA, MLP, регрессор дерева решений, случайный метод. лес, RNN, GRU, двунаправленный GRU и Stacked LSTM. Преобразование было применено к «открытые» и «высокие» столбцы, чтобы сделать их неподвижными перед прогнозированием. Среди моделей

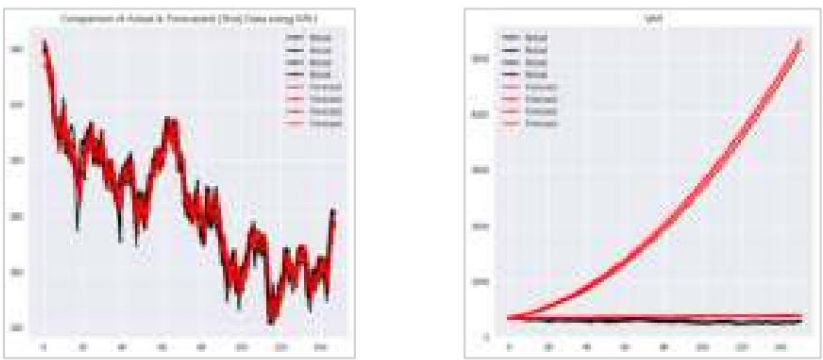
Machine Translated by Google

Как указано в Таблице 4, ГРУ с 50 единицами и 100 эпохами оказалось лучшим исполнителем, с 3,9242 MAE и 5,1312 RMSE. Рисунок 10a демонстрирует эффективность ГРУ. модель на многомерном тестовом наборе. Однако статистические модели, такие как VAR, VMA и VARMA, показали плохую производительность, не сумев понять основную закономерность в многомерном ряду . На рисунке 10b показана эффективность модели VAR на многомерных данных.

Кроме того, анализ производительности отдельных столбцов показал, что «открытые» и «Высокие» столбцы, первоначально считавшиеся нестационарными, создавали проблемы для прогнозирования. Несмотря на применив преобразование разностей, в этих столбцах прогнозируемые значения расходились значительно отличается от исходных данных, что приводит к высоким значениям RMSE.

Таблица 4. Сводная информация об эффективности модели эксперимента на основе многомерных рядов.

Классифицировать	Предварительная обработка	Модель	Настройки	МАЭ	РМСЭ
Часть 1. Многомерные временные ряды фондового рынка без проблем с качеством данных					
	NA	ГРУ	Эпоха = 100, юниты = 50	3,92	5.13
1	NA	РНН	Эпоха = 100, юниты = 50	3,93	5.13
2	NA	РНН	Эпоха = 25, юниты = 50	3,95	5.19
3	NA	МЛП	Эпоха = 100, единиц = 20 100	4,04	5.19
4	NA	Случайный лес	деревьев, разбиение	5,47	7.08
17	NA	ВАРМА	MSE (2,	407,98	657,08
23	NA	ВМА	1, 3) (0, 1, 1)	365,41	752,85
24 25	NA	ВАР	Заказ = 5	963,64	1678,42
Часть 2: Многомерные временные ряды потребления электроэнергии индивидуальными домохозяйствами с отсутствующими данными					
	Экспоненциальный	ГРУ	Эпоха = 100, юниты = 20	0,12	0,64
1	СМА	ГРУ	Эпоха = 100, юниты = 20	0,13	0,64
2	Байесовский хребет	МЛП	Эпоха = 100, юниты = 50	0,13	0,65
3	Случайный лес	Сложенный LSTM	Эпоха = 100, юниты = 50	0,13	0,65
4	КННИмпьютер 5 (0,1,1)	ВМА		0,43	1,88
330	КННИмпьютер 5 (2,1,3) полином_3	ВАРМА		0,43	1,88
331		РНН	Эпоха = 25, единицы = 20	9,35	29.31
449 450	полином_3	DT-регрессор	Сплит MSE	14,53	58,72



(а) (б)

Рисунок 10. (а) Сравнение производительности GRU для данных многомерных временных рядов во время тестирования. (б) ВАР сравнение производительности модели для данных многомерных временных рядов во время тестирования.

Во второй части эксперимента с многомерными рядами тестировалась Preptimize с использованием набор данных о потреблении электроэнергии в отдельных домохозяйствах с целью понять его поведение с помощью неполных данных многомерных временных рядов [44]. Этот набор данных включает семь столбцы, исключая дату и время, записывая энергопотребление за четыре года с частота дискретизации в одну минуту. В наборе данных, насчитывающем около 2 075 259 экземпляров, отсутствуют значения преимущественно в апреле 2007 г. Подробности приведены в таблице 5.

Из-за размера набора данных были обработаны данные, содержащие отсутствующие экземпляры, только за одну неделю . С 40 стратегиями предварительной обработки и 25 моделями прогнозирования, всего 1000 моделей. были оценены. Однако из-за вычислительных ограничений эксперимент был разделен на несколько частей. части, а для получения результатов требуется несколько дней. После проведения предыдущих экспериментов 18 успешных методов предварительной обработки были объединены с 25 моделями прогнозирования для оценки.

Machine Translated by Google

Таблица 5. Набор данных о потреблении электроэнергии отдельными домохозяйствами, использованный в эксперименте 2.

Нет.	Имя столбца	Единицы	Описание
1	global_active_power	Киловатт	Общая активная мощность, потребляемая домохозяйством
2	global_reactive_power	Киловатт	Общая реактивная мощность, потребляемая домохозяйством
3		Вольты	Среднее напряжение
4	напряжение	Усилители	Средний ток
5	global_intensity	Ватт-часы	Активная энергия для кухни
6	sub_metering_1	Ватт-часы	Активная энергия для стирки
7	sub_metering_2 sub_metering_3	Ватт-часы	Активная энергия для систем климат-контроля

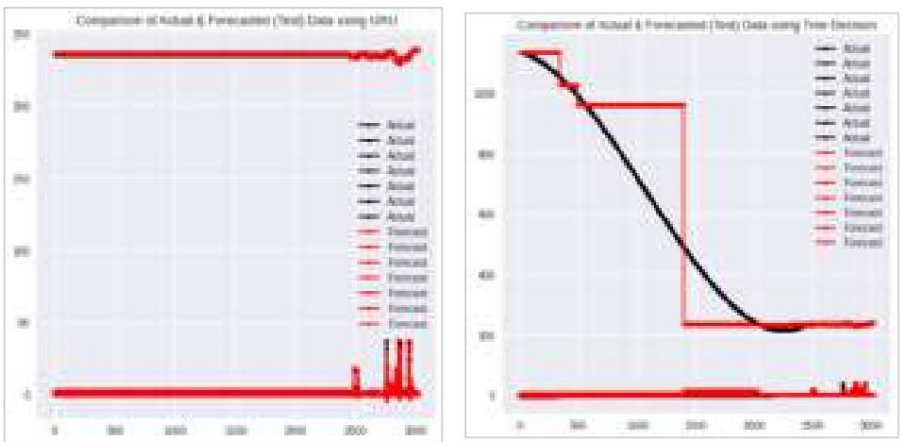
Комплексный отчет, созданный Preptimize, начинался с профилирования данных и анализа каждого атрибута в отдельности. Тепловая карта корреляции, изображенная ранее на рисунке 8b, иллюстрирует парные отношения между столбцами. В частности, глобальная интенсивность и глобальная активная мощность продемонстрировала сильную положительную корреляцию, в то время как суб-учет 3, связанный с системы климат-контроля, тесно коррелируют с глобальной активной мощностью и глобальной интенсивностью. Напротив, глобальная реактивная мощность не показала корреляции с другими атрибутами, в частности напряжение и дополнительный учет 3.

Кроме того, в отчете оценивается процент отсутствующих данных, примерно 37%. определение столбцов, таких как подсчет 1 и подсчет 2, как ненормально распределенных с высокими значениями эксцесса. Preptimize классифицировал все четыре столбца с отсутствующими данными как Тип MHAP. Попытки нормализовать распределения подсчета 1 и подсчета 2 были затруднена возможность нулевых или отрицательных значений, исключающих преобразование Бокса-Кокса.

После разложения тренда и сезона были применены различные методы вменения, включая итеративные методы. Каждый вмененный набор данных подвергался оценке с использованием 18 различных модели прогнозирования, что приводит к комплексной оценке эффективности.

Компания Preptimize подготовила оценочный лист, включающий 450 моделей и подробно описывающий методы предварительной обработки, используемые во время вменения. Как было показано ранее, Таблица 3 представляет показатели производительности выбранных моделей, выделяя наиболее эффективную модель. как рассчитывало GRU с помощью экспоненциальной скользящей средней, что дает RMSE 0,6393 и МАЭ 0,1225. И наоборот, регрессор дерева решений, которому приписана полиномиальная интерполяция степени 3, показал наихудшую производительность: RMSE 14,5308 и MAE. 58,7152.

Примечательно, что статистические модели продемонстрировали стабильную производительность, когда данные были обработаны. вменено с использованием KNNImputer с двумя соседями, с рейтингом VMA, VARMA и VAR тесно. Однако некоторые модели машинного обучения оказались хуже. Графики производительности для Модели регрессора GRU и дерева решений представлены на рисунках 11a,b соответственно.



(а) (б)

Рисунок 11. (a) GRU, рассчитанный с помощью экспоненциального скользящего среднего: сравнение производительности по мощности данные серии потребления во время тестирования. (б) Дерево решений, рассчитанное с помощью полиномиальной интерполяции степень 3: сравнение производительности для серий энергопотребления во время тестирования.

Machine Translated by Google

Таким образом, эксперимент показал, что экспоненциальное скользящее среднее, байесовский хребет, методы на основе KNN и DataWig являются эффективными методами вменения, в то время как полиномиальная интерполяция степени 3, сплайн-интерполяция степени 3 и HotDeck дали неоптимальные результаты.

4. Выводы и обсуждение.

Анализ временных рядов становится все более сложным из-за множества доступных методов. Выбор подходящего метода прогнозирования и стратегии предварительной обработки на основе характеристик временного ряда имеет решающее значение для получения надежных результатов. При анализе временных рядов необходимо следовать надлежащим шагам , чтобы определить лучшие модели, применимые в практических условиях. Статистические модели и модели машинного обучения, а также различные методы вменения для обработки недостающих данных предлагают разнообразные варианты анализа. Учитывая многочисленные исследования, подробно описывающие эффективность каждого метода на конкретных наборах данных, существует потребность в системе, способной эффективно применять эти методы для определения лучшей модели

Благодаря экспериментам с различными наборами данных временных рядов эффективность предлагаемой структуры Preptimize очевидна. Preptimize рекомендует несколько схем для моделей прогнозирования, что служит предварительным шагом на пути к анализу заданных временных рядов и позволяет сэкономить значительное время и усилия. Исследователи могут еще больше усовершенствовать эти модели, создавая гибридные комбинации, настраивая параметры и добавляя дополнительные функции для повышения точности прогнозирования. Preptimize создает подробные отчеты, охватывающие все этапы обработки данных, а также диаграммы, графики, статистику и подробный оценочный лист, демонстрирующий производительность всех моделей, протестированных на наборе данных с использованием различных показателей ошибок.

Для оценки производительности этой структуры были проведены различные эксперименты с различными наборами данных. Об эффективности системы можно судить по тому, насколько близко модель проекта соответствует моделям, используемым в литературе, а также по сравнению точности, основанной на RMSE. Контрольные показатели моделей прогнозирования фондового рынка основаны на LSTM, SVR и гибридных моделях, сочетающих статистические методы с методами машинного обучения, часто сообщающих значения RMSE всего 0,02 для цен на акции. Preptimize также предлагает LSTM и ARIMA для прогнозирования цен на акции. Тестовые показатели прогнозирования акций Microsoft (MSFT) показали RMSE 0,05 с гибридной моделью, сочетающей LSTM и RNN [45]. Модель проекта, предложенная с помощью Preptimize, достигла RMSE 0,029 с использованием Vanilla LSTM. Высокая волатильность Биткойна со значительными колебаниями цен в течение коротких периодов времени усложняет использование традиционных моделей прогнозирования. В прогнозе цен на биткойны указаны значения RMSE от 4 до 7 для моделей LSTM и случайно Компания Preptimize предложила LSTM (RMSE 10.02) для прогнозирования биткойнов, что соответствует выводам из литературы. В исследовании [46] были рассмотрены различные модели прогнозирования, включая нейронные сети прямого распространения, методы LSTM, GRU, ARIMA и AI, на наборе данных о потреблении электроэнергии в отдельных домохозяйствах, чтобы сравнить их производительность. Среднее значение RMSE для 12 различных моделей составило 0,83, при этом самое низкое значение RMSE 0,75 было достигнуто как LSTM , так и GRU. Напротив, Preptimize рекомендовал использовать GRU с предварительной обработкой экспоненциальным сглаживанием, в результате чего RMSE составило 0,639. Основываясь на обсуждении, мы можем сделать вывод , что Preptimize эффективно рекомендует подходящие модели прогнозирования для данного набора данных, способствуя быстрому принятию решений с минимальными усилиями.

Preptimize имеет определенные ограничения, которые следует учитывать. Во-первых, он ресурсоемок и требует значительных вычислительных мощностей и времени из-за комплексного подхода, заключающегося в применении многочисленных методов к выборке данных для определения оптимальных методов предварительной обработки и прогнозирования. Во-вторых, Preptimize в первую очередь поддерживает наборы данных с нормальным распределением. При работе с ненормальными распределениями ряд преобразуется к приблизительно нормальному, что может добавить сложности и потенциальных неточностей, особенно если преобразование не полностью обеспечивает нормальное распределение. Мы стремимся решить эти проблемы в дальнейшей работе.

Хотя предлагаемая структура фокусируется на решении проблемы недостающих данных при анализе временных рядов, сохраняется множество других проблем, таких как обнаружение выбросов, дрейф понятий и гетероскедастичность. Решение этих проблем требует дальнейшей автоматизации.

Machine Translated by Google

анализа временных рядов — задача, которая будет решаться на будущих этапах исследований. В дальнейшей работе можно было бы изучить расширение Preptimize для обработки более сложных типов данных временных рядов, например, с несколькими сезонами или нерегулярными интервалами. Кроме того, интеграция большего количества гибридных моделей, основанных на методах машинного обучения, и расширение применимости структуры к другим областям, таким как здравоохранение и мониторинг окружающей среды, могут еще больше повысить ее полезность.

Кроме того, хотя мы тестировали надежность и полноту системы в отношении нескольких сценариев, мы признаем, что дополнительные сценарии, такие как сезонные одномерные данные, высокий процент пропущенных значений, ненормальное распределение, многосезонные данные и нисходящие тенденции, могут дать дополнительную ценную информацию. идеи. Однако включить все возможные сценарии в первоначальную работу оказалось невозможным. Мы призываем других исследователей изучить эту структуру с различными наборами данных, чтобы устранить этот пробел в исследованиях.

Вклад автора: концептуализация, методология, валидация, формальный анализ, визуализация, написание — подготовка оригинального проекта, MU; надзор, ЗАМ; рецензирование и редактирование, RQ и AZ. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Данное исследование не получило внешнего финансирования.

Заявление Институционального наблюдательного совета: Неприменимо.

Заявление об информированном согласии: Не применимо.

Заявление о доступности данных: Наборы данных, использованные в этом исследовании, доступны в Интернете. Набор данных о энергопотреблении доступен по адресу https://archive.ics.uci.edu/ml/datasets/individual+household+. электрическая+мощность+потребление (по состоянию на 1 июня 2024 г.) Набор финансовых данных фондового рынка доступен по адресу https://aroussi.com/. (по состоянию на 1 июня 2024 г.). Набор данных криптовалюты доступен по адресу https://pypi.org/project/Historic-Crypto/. (по состоянию на 1 июня 2024 г.).

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Ссылки

1. Лю, З.; Чжу, З.; Гао, Дж.; Сюй, К. Методы прогнозирования данных временных рядов: обзор. IEEE Access 2021, 9, 91896–91912. [Перекрестная ссылка]
2. Асади, С.; Эсмаил, Х.; Фархад, М.; Масуд, Н.М. Гибридизация эволюционных нейронных сетей Левенберга – Марквардта и данных предварительная обработка для прогнозирования фондового рынка. Знать. На основе сист. 2012, 35, 245–258. [Перекрестная ссылка]
3. Ди Персио, Л.; Фраккароло, Н. Прогнозы энергопотребления с помощью деревьев регрессии, повышающих градиент. Математика 2023, 11, 1068. [Перекрестная ссылка]
4. Крайер, Дж. Д.; Келлет, Н. Анализ временных рядов: с приложениями в R, 2-е изд.; Спрингер: Нью-Йорк, штат Нью-Йорк, США, 2008 г.; Том 2, с. 31.
5. Чжоу, К.; Оока, Р. Влияние предварительной обработки данных на производительность нейронной сети для воспроизведения CFD-моделирования неизотермическое распределение воздушного потока в помещении. Энергетическая постройка. 2021, 230, 110525. [CrossRef]
6. Сонг, С.; Чжан, А.; Ван, Дж.; Ю, PS ЭКРАН: Очистка потока данных при ограничениях скорости. В материалах ACM 2015 г. Международная конференция SIGMOD по управлению данными, Мельбурн, Австралия, 31 мая – 4 июня 2015 г.
7. Биллалли, Б.; Абелло, А.; Алуджа-Банет, Т.; Рембель, Р. Автоматизированная предварительная обработка данных посредством метаобучения. В материалах Международной конференции по моделированию и инженерии данных, Альмерия, Испания, 21–23 сентября 2016 г.; Спрингер: Чам, Швейцария, 2016 г.
8. Чжан А.; Песня, С.; Ван, Дж.; Ю, П.С. Очистка данных временных рядов: от обнаружения аномалий к их устранению. Учеб. ВЛДБ Эндоу. 2017, 10, 1046–1057. [Перекрестная ссылка]
9. Симидзу, К.; Понсе-Эрнандес, Р.; Ахмед, ОС; Ота, Т.; Победа, ZC; Мизуэ, Н.; Йошида, С. Использование изображений временных рядов Landsat для обнаружения нарушений леса в выборочно вырубаемых тропических лесах в Мьянме. Может. Дж. Для. Рез. 2017, 47, 289–296. [Перекрестная ссылка]
10. Чжу, З. Обнаружение изменений с использованием временных рядов спутников Landsat: обзор частот, предварительной обработки, алгоритмов и приложений. ISPRS J. Фотограмма. Дистанционное восприятие 2017, 130, 370–384. [Перекрестная ссылка]
11. Карим Ф.; Маджумдар, С.; Дараби, Х.; Чен, С. Полностью сверточные сети LSTM для классификации временных рядов. Доступ IEEE 2017, 6, 1662–1669. [Перекрестная ссылка]
12. Гшвандтнер, Т.; Эрхарт О. Знай своего врага: выявление проблем качества данных временных рядов. В трудах IEEE Тихоокеанский симпозиум по визуализации (PacificVis), Кобе, Япония, 10–13 апреля 2018 г.
13. Джинанунта, К.; Абейратна, К.Д.; Дилхани, MS; Хнин, Юго-Запад; Phyo, PP Обнаружение выбросов во временных рядах для кратковременной электрической нагрузки прогнозирование спроса. Межд. наук. Дж. Инж. Технол. (ИСДЖЕТ) 2018, 2, 37–50.
14. Ван Х.; Ван, К. Очистка данных временных рядов: опрос. Доступ IEEE 2019, 8, 1866–1881. [Перекрестная ссылка]
15. Дин, Х.; Ван, Х.; Су, Дж.; Ли, З.; Ли, Дж.; Гао, Х. Клеанитс: Система очистки данных для промышленных временных рядов. Учеб. ВЛДБ Эндоу. 2019, 12, 1786–1789. [Перекрестная ссылка]

Machine Translated by Google

16. Руис, Л.; Пегалахар, М.; Аркуччи, Р.; Молина-Солана, М. Методика кластеризации временных рядов для извлечения знаний в области энергетики данные о потреблении. Экспертная система. Прил. 2020, 160, 113731. [CrossRef]
17. Джарретт, Д.; Юн, Дж.; Бица, И.; Цянь, З.; Эрколе, А.; Шаар, МВД «Ясновидение: набор инструментов для медицинских временных рядов». В материалах Международной конференции по обучению представлениям, виртуально, 3–7 мая 2021 г.
18. Десаи, В.; Динеша, Х.А. Гибридный подход к методам предварительной обработки данных. В материалах Международной конференции IEEE по инновациям в технологиях (INOCON), Бангалор, Индия, 6–8 ноября 2020 г.
19. Соуза, Р.; Амадо, К.; Энрикес, Р. AutoMTS: Полностью автономная обработка многомерных данных временных рядов из гетерогенных сенсорных сетей. В материалах Международной конференции по гетерогенным сетям для обеспечения качества, надежности, безопасности и устойчивости, виртуально, 29–30 ноября 2020 г.; Шпрингер: Чам, Швейцария, 2020 г.
20. Чен, Х.; Дэн, Л.; Хуанг, Ф.; Чжан, К.; Чжан, З.; Чжао, Ю.; Чжэн, К. Демон: Неконтролируемое обнаружение и интерпретация аномалий для многомерных временных рядов. В материалах 37-й Международной конференции по инженерии данных (ICDE) IEEE 2021 г., Ханья, Греция, 19–22 апреля 2021 г.; стр. 2225–2230.
21. Чаухан, К.; Яни, С.; Таккар, Д.; Дэйв, Р.; Бхатия, Дж.; Танвар, С.; Обайдат, М.С. Автоматизированное машинное обучение: новая волна машинного обучения. В материалах 2-й Международной конференции по инновационным механизмам для промышленного применения (ICIMIA), Бангалор, Индия, 5–7 марта 2020 г.
22. Сарафанов М. AutoML для временных рядов: определенно хорошая идея. Доступно онлайн: https://towardsdatascience.com/automl-for-time-series-definitelya-good-idea-c51d39b2b3f . (по состоянию на 1 сентября 2023 г.).
23. Вс, Б.; Ма, Л.; Шен, Т.; Гэн, Р.; Чжоу, Ю.; Тиан, Ю. Надежный метод определения мультисезонности и гетероскедастичности, основанный на данных. Предварительная обработка временных рядов. Вирел. Коммун. Моб. Вычислить. 2021, 2021, 6692390. [CrossRef]
24. Чжан, врач общей практики; Мин, К. Нейросетевое прогнозирование сезонных и трендовых временных рядов. Евро. Дж. Опер. Рез. 2005, 160, 501–514. [Перекрестная ссылка]
25. Ранджан, КГ; Прасти, БР; Йена, Д. Сравнение двух методов очистки данных применительно к изменчивым временным рядам. В материалах Международной конференции по применению и технологиям силовой электроники в современном энергетическом сценарии (PETPES), Мангалор, Индия, 29–31 августа 2019 г.
26. Ранджан, КГ; Трипатия, DS; Прасти, БР; Йена, Д. Улучшенное обнаружение и коррекция выбросов на основе прогнозирования скользящего окна для волатильных временных рядов. Межд. Дж. Нумер. Модель. Электрон. Сеть. Поля устройств 2021, 34, e2816. [Перекрестная ссылка]
27. Льв, П.; Ву, К.; Сюй, Дж.; Шу, Ю. Прогнозирование фондового индекса на основе разложения временных рядов и гибридной модели. Энтропия 2022, 24, 146. [Перекрестная ссылка]
28. Брюнель, Б.; Алсамад, Ф.; Пиот О. На пути к автоматизированному машинному обучению в вибрационной спектроскопии: использование и настройка генетических алгоритмов для предварительной обработки и оптимизации регрессии. Чемом. Интел. Лаб. Сист. 2021, 219, 104444. [CrossRef]
29. Кумар, С. 8 библиотек AutoML для автоматизации конвейера машинного обучения. 2020. Доступно онлайн: https://medium.com/swlh/8-. automl-libraries-toautomate-machine-learning-pipeline-3da0af08f636 (по состоянию на 1 августа 2023 г.).
30. Джанг, В.-Дж.; Ли, С.-Т.; Ким, Ж.-Б.; Гим, Г.-Ю. Исследование профилирования данных: фокус на индексе качества значений атрибутов. Прил. наук. 2019, 9, 5054. [CrossRef]
31. Гадерпур, Э.; Пагиатакис, С.Д.; Хассан, К.К. Обзор по обнаружению изменений и анализу временных рядов с помощью приложений. Прил. наук. 2021, 11, 6141. [CrossRef]
32. Цзоу, Х.; Ян, Ю. Объединение моделей временных рядов для прогнозирования. Межд. Дж. Прогноз. 2004, 20, 69–84. [Перекрестная ссылка]
33. Лим, Б.; Зорен, С. Прогнозирование временных рядов с помощью глубокого обучения: обзор. Филос. Пер. Р. Сок. Математика. Физ. англ. наук. 2021, 379, 20200209. [CrossRef]
34. Аббасимехр, Х.; Паки, Р.; Бахрини, А. Новый подход, основанный на сочетании моделей глубокого обучения со статистическими методами для Прогнозирование временных рядов COVID-19. Нейронный компьютер. Прил. 2022, 34, 3135–3149. [Перекрестная ссылка] [ПабМед]
35. Браун, Т.А. Подтверждающий факторный анализ для прикладных исследований; Гилфорд Пресс: Нью-Йорк, штат Нью-Йорк, США, 2006 г.
36. Харун Д. Дифференциация временных рядов. В тематических исследованиях машинного обучения Python: пять тематических исследований для специалиста по данным; Апресс: Нью- Йорк, Нью-Йорк, США, 2017; п. 110.
37. Агиаклоглу, К.; Ньюболд, П. Эмпирические данные по тестам типа Дики-Фуллера. Дж. Тайм Сер. Анальный. 1992, 13, 471–483. [Перекрестная ссылка]
38. Гудфеллоу, И.; Пуже-Абади, Дж.; Мирза, М.; Сюй, Б.; Вард-Фарли, Д.; Озаир, С.; Курвиль, А.; Бенджио, Ю. Генеративно-состязательные сети. Коммун. АКМ 2020, 63, 139–144. [Перекрестная ссылка]
39. Юн, Дж.; Джордан, Дж.; Шаар, М. Гейн: Вменение недостающих данных с использованием генеративно-состязательных сетей. В трудах Международного Конференция по машинному обучению, Стокгольм, Швеция, 10–15 июля 2018 г.
40. Бисманн, Ф.; Рукат, Т.; Шмидт, П.; Найду, П.; Шелтер, С.; Таптунов А.; Ланге, Д.; Салинас, Д. DataWig: Вменение пропущенных значений для таблиц. Дж. Мах. Учиться. Рез. 2019, 20, 1–6.
41. Йоэнсен, Д.В.; Банкхофер, У. Методы горячей колоды для вменения недостающих данных. В материалах международного семинара по машинному обучению и интеллектуальному анализу данных в распознавании образов, Берлин, Германия, 13–20 июля 2012 г.
42. Холланд, Дж. Х. Адаптация в природных и искусственных системах: вводный анализ с приложениями к биологии, управлению и искусственным системам. Интеллект; MIT Press: Кембридж, Массачусетс, США, 1992.
43. Макдауэлл, Д.; Макклири, Р.; Бартос, Б.Дж. Анализ прерванных временных рядов, 21-е изд.; SAGE: Нью-Йорк, штат Нью-Йорк, США, 1980 г.; п. 41.
44. Дуа, Д.; Графф, К. Репозиторий машинного обучения UCI; Калифорнийский университет, Школа информатики и компьютерных наук: Ирвин, Калифорния, США, 2019 г. Доступно онлайн: http://archive.ics.uci.edu/ml. (по состоянию на 1 июня 2024 г.).

Machine Translated by Google

45. Сурав, Д.; Апан, П.; Саян, С.; Саян, Г.; Удатья, Д.; Чандра, Д.; Шилпи, Б. Новая гибридная модель, использующая Lstm и Rnn для Прогноз фондового рынка. Межд. Дж. Инж. Рез. Технол. 2024, 13. [CrossRef]
46. Гаспарин А.; Слободан Л.; Чезаре, А. Глубокое обучение для прогнозирования временных рядов: случай электрической нагрузки. САИ Транс. Интел. Технол. 2022, 7, 1–25. [Перекрестная ссылка]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Логическое значение на основе глубокого обучения, временные ряды, обнаружение ошибок и

Прогнозный анализ в работе контейнерных кранов

Амрута Авасти, Ленка Крпалкова и Джозеф Уолш *

Школа науки, технологий, инженерии и математики (STEM), Мюнстерский технологический университет (MTU), кампус Керри, V92 HD4V Керри, Ирландия; amruta.awasthi@research.ittralee.ie (AA); lenka.krpalkova@mtu.ie (LK)
* Переписка: joseph.walsh@mtu.ie

Аннотация: Глубокое обучение имеет решающее значение в морской логистике и обнаружении, диагностике и прогнозировании ошибок контейнерных кранов. Новый метод глубокого обучения с использованием долгосрочной краткосрочной памяти (LSTM) обнаруживает и предвидит ошибки в системе с несбалансированными данными. Модель LSTM была обучена на реальных данных об эксплуатационных ошибках контейнерных кранов. Пользовательский алгоритм использует метод синтетической передискретизации меньшинства (SMOTE) для балансировки несбалансированных данных с учетом ошибок рабочих данных (т. е. слишком малого количества выборок класса меньшинства). Python использовался для программирования. Представлены корреляционные и ковариационные матрицы Пирсона, Спирмена и Кендалла . Показаны потери при обучении и проверке модели , а также прогнозируются оставшиеся данные. Тестовый набор (30% фактических данных) и прогнозируемые данные имели среднеквадратическое отклонение 0,065. Тепловая карта матрицы путаницы была создана с использованием Matplotlib и Se Кроме того, были спроецированы выходные данные ошибок для временного ряда на следующие n секунд, причем n секунд были введены пользователем. Точность составила 0,996, прецизионность - 1,00, полнота - 0,500, а показатель f1 - 0,667, в соответствии с установленными критериями оценки. Эксперименты показали, что этот метод способен идентифицировать критические элементы. Таким образом, будущие попытки улучшат структуру модели для прогнозирования ошибок в больших промышленных данных. Однако преимущество состоит в том, что он может обрабатывать несбалансированные данные, которые обычно имеются в большинстве отраслей. При наличии дополнительных данных модель может быть улучшена.

check for updates

Цитирование: Авасти, А.; Крпалкова Л.; Уолш, Дж. Глубокое обучение Логическое значение, временные ряды, обнаружение ошибок, и прогнозный анализ в работе контейнерных кранов. Алгоритмы 2024, 17, 333. https://doi.org/10.3390/a17080333

Академические редакторы: Матеус Мендес и Балдуино Матеус

Поступила: 16 июня 2024 г.
Пересмотрено: 20 июля 2024 г.
Принято: 24 июля 2024 г.
Опубликовано: 1 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

Ключевые слова: прогнозная аналитика; глубокое обучение; аналитика данных; логические данные временного ряда; обнаружение ошибок; глубокие нейронные сети; Рекуррентные нейронные сети (RNN); Долгосрочный Краткосрочный Сети памяти (LSTM); Метод синтетической передискретизации меньшинства (SMOTE); контейнерные краны; морская промышленность

1. Введение

1.1. Справочная

информация Рынок контейнерных перевозок в 2024 году составит 116,04 миллиарда долларов США, и ожидается, что к 2029 году он вырастет до 134,03 миллиарда долларов США [1]. По мере роста объема мировой торговли одновременно усиливаются требования к повышению производительности и операционной эффективности портовых сооружений . Наряду с этими требованиями обеспечение безопасности и надежности работы контейнерных кранов остается первостепенным [2]. Контейнерные краны играют ключевую роль в современной морской логистике, способствуя эффективному перемещению товаров между судами и берегом на контейнерных терминалах по всему миру [3]. Высокие козловые краны незаменимы при погрузке и разгрузке контейнеров с судов и перемещении их по терри

Контейнерные краны, работающие вдоль причала, используют сложную технологию, такую как распределительные балки, для перемещения контейнеров между судами и терминалами, что способствует общей эффективности порта. Эта операция предполагает точное размещение крана в различных отсеках корабля, обеспечивая доступ к контейнерам, расположенным по длине судна. Кроме того, помимо операций с судами, контейнерные краны могут использоваться для штабелирования контейнеров на территории терминала, что позволяет осуществлять временное хранение или дальнейшее хран

Machine Translated by Google

транспорт [4]. Краны генерируют огромные объемы данных, которые можно анализировать и получать ценную информацию для эффективного, лучшего и быстрого принятия решений в отраслях.

В последнее время все больше исследований сосредоточено на методах диагностики неисправностей на основе данных, особенно на основе нейронных сетей. Это связано с достижениями в области сенсорных технологий и увеличением емкости хранилища данных, которые расширили возможности обработки данных методами, основанными на нейронных сетях. Неспособность выявить дефекты процесса может иметь пагубные последствия для производительности, безопасности, качества продукции и экономики процесса.
Эффективное и современное автоматизированное диагностическое оборудование имеет решающее значение на современных предприятиях для обнаружения, диагностики и устранения аномального поведения процессов. Основная цель обнаружения и изоляции неисправностей — быстро предупредить операторов о потенциальных проблемах, позволяя им принять необходимые меры для предотвращения сбоя системы после возникновения проблем [5].

Своевременное обнаружение и прогнозирование ошибок имеют решающее значение для оптимизации производительности крана и обеспечения безопасности при работе контейнерных кранов.

Важность своевременного обнаружения ошибок можно понять с нескольких важных точек зрения: 1. Оптимизация

производительности и эффективности: своевременное обнаружение и предотвращение ошибок оптимизируют производительность крана и его эксплуатационную эффективность. Своевременно выявляя и устраняя ошибки, операторы могут минимизировать влияние на производительность и пропускную способность, гарантируя, что краны работают с максимальной производительностью и эффективностью [6]. Этого можно достичь за счет раннего прогнозирования ошибок, чтобы они не превратились в неисправности и не привели к сбоям. Такая оптимизация производительности способствует повышению общей производительности и прибыльности операций контейнерного терминала .

2. Предотвращение сбоев в работе: своевременное обнаружение ошибок позволяет заранее выявить потенциальные проблемы или неисправности в работе крана [7]. Оперативно выявляя ошибки, операторы могут предпринять корректирующие действия для предотвращения сбоев в работе, таких как поломки кранов или задержки в погрузочно-разгрузочных работах. Такой проактивный подход помогает поддерживать бесперебойный поток операций на контейнерных терминалах, сводя к минимуму время простоя и обеспечивая эффективное использование ресурсов [8].

3. Совершенствование протоколов безопасности. Раннее обнаружение ошибок позволяет операторам внедрять протоколы и процедуры безопасности для снижения рисков и опасностей, связанных с работой крана. Оперативно устраняя ошибки, операторы могут предотвратить несчастные случаи, травмы и ущерб грузу или инфраструктуре, гарантируя благополучие персонала и сохраняя целостность терминальных сооружений [9].

4. Сокращение затрат на техническое обслуживание. Раннее обнаружение, диагностика и прогнозирование ошибок может помочь снизить затраты на техническое обслуживание, связанные с работой крана. Выявляя потенциальные ошибки до того, как указанные ошибки перерастут в серьезные сбои, операторы могут реализовать профилактические и предписывающие меры по техническому обслуживанию для решения основных проблем и продления срока службы оборудования [10]. Такой подход к упреждающему техническому обслуживанию помогает минимизировать время простоя, затраты на ремонт и сбои в работе, что в конечном итоге приводит к экономии затрат операторов терминалов [11].

5. Соответствие нормативным стандартам. Своевременное обнаружение ошибок обеспечивает соблюдение нормативных стандартов и отраслевых правил, регулирующих работу кранов. Быстро выявляя и устраняя ошибки, операторы могут обеспечить соблюдение правил техники безопасности и эксплуатационных протоколов, установленных регулирующими органами. Такое соблюдение требований не только обеспечивает безопасность персонала и активов, но также помогает избежать потенциальных штрафов и юридической ответственности, связанных с несоблюдением требований [12,13].

Таким образом, своевременное выявление и прогнозирование ошибок имеют важное значение для оптимизации производительности кранов, обеспечения безопасности при работе контейнерных кранов и принесения пользы заинтересованным сторонам в морской цепочке поставок.

1.2. Цели исследования, мотивация и вклад Одним из важнейших аспектов

обеспечения безопасности, эффективности и надежности работы контейнерных терминалов является обнаружение и устранение ошибок, которые могут возникнуть во время работы крана.

Machine Translated by Google

Ошибки в работе причальных контейнерных кранов могут иметь серьезные последствия: от потенциального повреждения груза и инфраструктуры до сбоев в логистике цепочки поставок. Более того, человеческие ошибки, неисправности оборудования и непредвиденные обстоятельства окружающей среды усложняют обнаружение ошибок и создают проблемы для обеспечения бесперебойной работы [14]. Сбои системы можно предотвратить, выявляя и предвидя ошибки, что приводит к созданию более надежной и долговечной системы. Прогнозирование ошибок в работе контейнерных кранов предполагает использование анализа данных, машинного обучения и методологий глубокого обучения для прогнозирования возможных ошибок или аномалий, которые могут возникнуть во время работы контейнерных кранов [6].

Это исследование направлено на разработку упреждающих стратегий обнаружения и прогнозирования ошибок путем использования возможностей глубокого обучения, анализа данных и прогнозного моделирования, тем самым повышая безопасность, эффективность и надежность операций контейнерных терминалов. Решение этой проблемы требует всестороннего понимания операционной динамики контейнерных терминалов и применения передовых технологий и методологий искусственного интеллекта для обнаружения, диагностики и прогнозирования ошибок [15].

Анализируя различные источники данных, включая данные датчиков крана, журналы прошлых операций, погодные условия и другие соответствующие аспекты, можно создавать прогнозные модели, позволяющие заранее предвидеть и минимизировать проблемы. Реализация этой превентивной стратегии повышает безопасность, эффективность и общую эффективность работы контейнерных терминалов.

Результаты исследований в этой области имеют важное значение для морской отрасли , портовых властей, операторов терминалов и других заинтересованных сторон, участвующих в работе контейнерных терминалов. Цель состоит в том, чтобы повысить эксплуатационную безопасность, минимизировать время простоя, оптимизировать использование ресурсов и улучшить общую производительность терминала за счет эффективного обнаружения и устранения ошибок в эксплуатационных данных. Целью данной статьи является изучение того, как глубокое обучение может улучшить обнаружение ошибок, прогнозирование и анализ временных рядов в работе контейнерных кранов, что в настоящее время является критической потребностью в морской отрасли и поможет повысить эксплуатационную эффективность и безопасность.

Потенциал этого исследования распространяется не только на контейнерные краны, но и на другие отрасли с проблемами несбалансированности данных — там, где происходят редкие события или аномалии, это может принести значительную пользу.

Другие отрасли промышленности могут получить значительную пользу от этих исследований, например, производство, поскольку они позволяют выявлять сбои или дефекты оборудования на ранних этапах производственных линий; энергетический сектор – при прогнозировании аномалий в системах производства или распределения электроэнергии; и транспортировке, при обнаружении нарушений в работе транспортных средств или систем безопасности. Кроме того, организации могут целенаправленно принимать меры и превентивные меры, выявляя критические элементы, связанные с ошибками.

Например:

Техническое обслуживание. Установите приоритетность графиков технического обслуживания для компонентов, которые, скорее всего, вызывать ошибки;

Оптимизация процессов: корректировка операционных процедур на основе моделей ошибок;

Распределение ресурсов. Чтобы свести к минимуму сбои, распределяйте ресурсы (персонал, материалы) стратегически.

Это исследование мотивировано необходимостью улучшения распознавания и прогнозирования ошибок в контейнерных кранах в многомиллиардной индустрии контейнерных перевозок, которая является основой современной мировой экономики. Традиционные методы, основанные на правилах, часто терпят неудачу, поскольку они не могут управлять сложными, динамическими настройками и неравномерным распределением данных. Глубокое обучение, особенно сети LSTM, обеспечивает эффективное решение этих недостатков. Фиксируя временные связи и изучая предыдущие данные, LSTM могут моделировать сложные закономерности и повышать точность.

1.3. Обзор глубокого обучения

Глубокое обучение — это отрасль искусственного интеллекта (ИИ), которая включает в себя обучение компьютерных систем выполнению таких задач, как распознавание закономерностей, принятие решений и решение проблем, без явного программирования на это. Он использует алгоритмы, известные как нейронные сети, вдохновленные структурой и функциями человеческого мозга [16].

Мониторинг состояния машин на основе данных становится все более распространенным в современных

Machine Translated by Google

производственные системы, поскольку недорогие датчики широко распространены и подключены к Интернету. Глубокое обучение предлагает ценные методы обработки и оценки крупномасштабных операционных данных [17]. Глубокое обучение позволяет компьютерам распознавать закономерности, классифицировать информацию и делать высокоточные прогнозы, обучаясь на больших объемах данных.

В глубоком обучении используются нейронные сети, состоящие из множества слоев, известные как глубокие архитектуры, для получения знаний из входных данных и генерации прогнозов. Эта технология находит применение в различных областях, включая распознавание изображений и речи, обработку естественного языка, автономные транспортные средства и, в данном случае, обрабатывающую промышленность.

Несколько ключевых факторов делают использование глубокого обучения первостепенным при обнаружении, диагностике и прогнозировании ошибок в различных отраслях, включая работу причальных контейнерных кранов: 1. Алгоритмы глубокого обучения, такие как сверточные нейронные сети (CNN) и рекуррентные нейронные сети (RNN).), продемонстрировали исключительные возможности обработки сложных и многомерных данных [18]. При работе контейнерных кранов, где потоки данных от различных датчиков, систем управления и операционных журналов обильны и разнообразны, модели глубокого обучения могут эффективно обрабатывать эти данные для выявления тонких закономерностей и аномалий, указывающих на ошибки.

2. Модели глубокого обучения превосходно фиксируют временные зависимости и динамические взаимосвязи в последовательных данных [19]. В операциях контейнерных кранов, где движения крана, процессы погрузки-разгрузки и условия окружающей среды меняются, алгоритмы глубокого обучения могут анализировать исторические данные для прогнозирования будущих состояний, обеспечивая упреждающее обнаружение и предотвращение ошибок [20].

3. Масштабируемость и адаптируемость моделей глубокого обучения делают их хорошо подходящими для мониторинга и принятия решений в режиме реального времени и в автономном режиме в динамичных операционных средах. Постоянно обучаясь на новых данных и адаптируясь к изменяющимся условиям, системы обнаружения ошибок на основе глубокого обучения могут повысить эксплуатационную безопасность, оптимизировать использование ресурсов и снизить потенциальные риски при работе контейнерных кранов [21,22].

В целом, важность использования глубокого обучения для обнаружения ошибок, диагностики и прогнозирования в работе контейнерных кранов заключается в его способности использовать возможности передового анализа данных для повышения эксплуатационной безопасности, эффективности и устойчивости морской логис По мере развития технологий использование алгоритмов глубокого обучения будет становиться все более необходимым для обеспечения бесперебойной и безопасной работы контейнерных терминалов в экосистеме глобальной цепочки поставок.

Используя передовые методы глубокого обучения, исследование направлено на устранение присущих сложностей обнаружения и прогнозирования ошибок в работе контейнерных кранов, где эксплуатационные данные представляются в виде дискретных логических значений с течением времени. Предлагаемый алгоритм стремится эффективно фиксировать временные зависимости и тонкие закономерности в данных логических временных рядов, позволяя точно предвидеть потенциальные ошибки до их возникновения во время работы крана. Посредством эмпирической проверки и оценки производительности в данной статье предпринимается попытка продемонстрировать эффективность и применимость предлагаемого алгоритма глубокого обучения для повышения эксплуатационной безопасности, эффективности и надежности в управлении контейнерными терминалами.

1.4. Рабочий процесс

Последующие разделы анализа рабочего процесса будут тщательно изучать существующее состояние дел, исследуя множество аспектов обнаружения ошибок в работе контейнерных кранов . Кроме того, методологические подходы, используемые при решении этой проблемы, будут тщательно изучены наряду с эмпирическими данными, полученными в результате соответствующих исследований. Этот всесторонний анализ прояснит тонкое понимание прогнозирования ошибок в контексте операций контейнерных кранов, прольет свет на его последствия и потенциальные возможности для продвижения в академическом и исследовательском дискурсе.

В следующем разделе будет представлена информация о методологии и данных, использованных в этом исследовании . Он представляет новый, специально разработанный алгоритм на основе Python, который интегрирует SMOTE.

Machine Translated by Google

и методы LSTM для устранения несбалансированных данных посредством пользовательской предварительной обработки, анализа и включения показателей оценки. Структура рабочего процесса показана на рисунке 1 ниже:

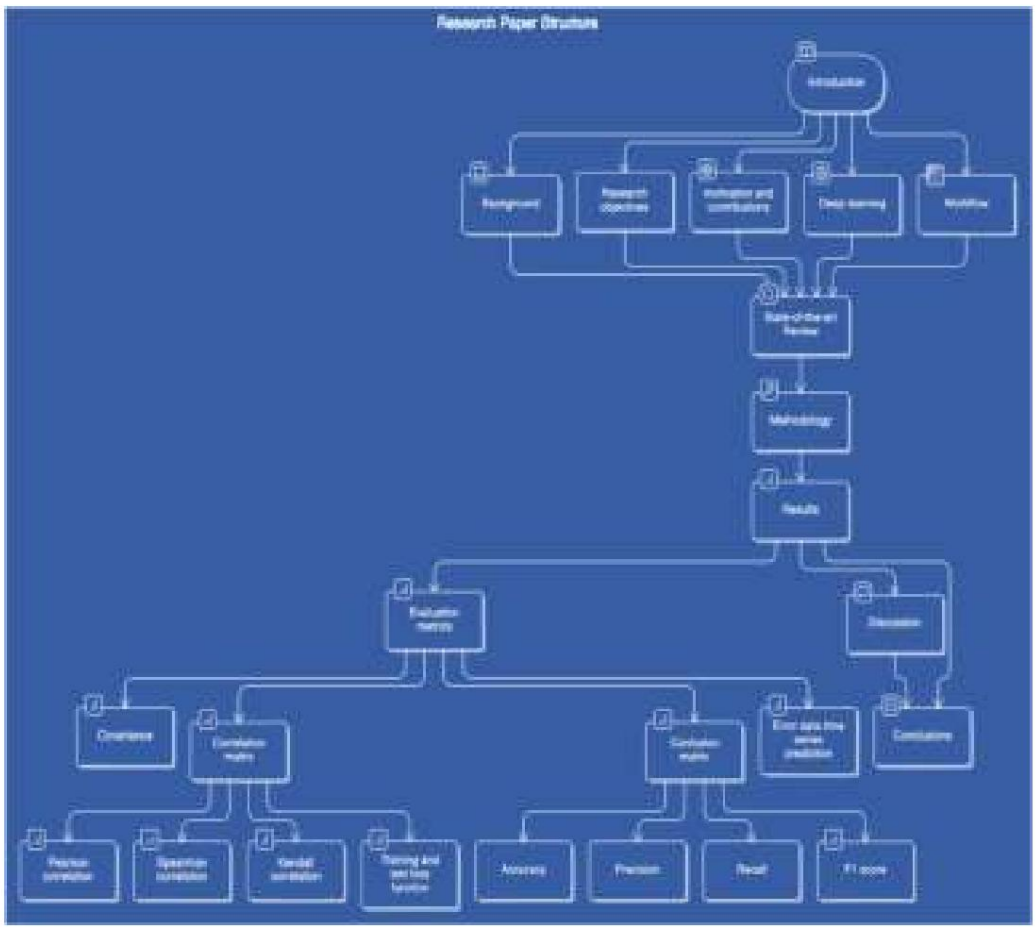


Рисунок 1. Рабочий процесс исследовательской работы.

2. Обзор современного состояния

Как подчеркивалось ранее, обеспечение безопасности и надежности работы крана имеет первостепенное значение. Данные временных рядов по таким операциям тяжелого машиностроения создают уникальные проблемы из-за их большого размера и сложной структуры. Своевременное выявление эксплуатационных ошибок может снизить количество аварий и простоев [23]. Модели машинного обучения и глубокого обучения стали мощными инструментами для обнаружения аномалий в таких данных. Существует обширная литература по обнаружению ошибок и сбоев в машинном обучении [24]. Однако очень мало информации по обнаружению и прогнозированию ошибок. Крайне важно обнаружить и спрогнозировать ошибку, поскольку ошибка является предшественником сбоев и отказов. Рисунок 2 ниже иллюстрирует эту корреляцию между ошибкой, дефектом и отказом.

Обнаружение сигнала эксплуатационной ошибки машины относится к процессу выявления аномальных или неожиданных сигналов в работе машины. Эти сигналы могут указывать на проблемы, неисправности или отклонения от нормального поведения системы или машины. В этом исследовании использовались фактические базы данных бортовых и внешних кранов STS из портов и кранов, такие как данные о гарантии, сигналах тревоги и эксплуатационные данные, где были доступны сигналы ошибок, генерируемые ПЛК крана .

Значимость использования глубокого обучения для обнаружения ошибок в работе контейнерных кранов заключается в его способности эффективно улавливать сложные закономерности, временные зависимости и нелинейные отношения, присущие эксплуатационным данным. Важнейшими моментами, подчеркивающими его значение, являются следующие:

Machine Translated by Google



Рисунок 2. Ошибка как предвестник сбоев [25].

1. Сложность эксплуатационных данных. Работа контейнерных кранов генерирует огромные объемы данных, включая показания датчиков, движения крана, погрузочно-разгрузочные работы и условия окружающей среды. Алгоритмы глубокого обучения, такие как RNN и CNN, превосходно справляются с обработкой многомерных и последовательных данных [18], что делает их хорошо подходящими для анализа сложной динамики работы крана.
2. Временные зависимости. Работа крана демонстрирует временные зависимости, когда действия и события на одном временном этапе влияют на последующие состояния и результаты. RNN, особенно сети LSTM, предназначены для сбора и изучения временных зависимостей в последовательных данных, что позволяет точно моделировать временную эволюцию работы крана и обнаруживать аномальные закономерности или отклонения [26,27].
3. Изучение функций. Модели глубокого обучения могут автоматически изучать соответствующие функции на основе необработанных данных, устраняя необходимость в ручном проектировании функций. В контексте обнаружения ошибок в работе контейнерных кранов алгоритмы глубокого обучения могут извлекать информативные характеристики из данных датчиков, траекторий крана и операционных журналов, что позволяет более эффективно обнаруживать аномалии и отклонения, указывающие на ошибки.
4. Адаптация к сложной среде: работа контейнерных кранов зависит от различных факторов окружающей среды, таких как ветер, погодные условия и состояние моря, которые могут влиять на работу крана и вносить неопределенность [28,29]. Модели глубокого обучения с их способностью учиться на разнообразных и зашумленных источниках данных демонстрируют надежность и адаптируемость к сложным операционным средам, повышая надежность и точность систем обнаружения ошибок [22].
5. Масштабируемость и обобщение. Алгоритмы глубокого обучения по своей сути масштабируемы и могут обрабатывать большие объемы данных, что делает их пригодными для анализа обширных наборов данных, генерируемых операциями контейнерных терминалов. Более того, модели глубокого обучения могут хорошо обобщать невидимые данные, обеспечивая эффективное обнаружение ошибок в различных конфигурациях терминалов, типах кранов и условиях эксплуатации [30].
6. Непрерывное обучение и совершенствование. Модели глубокого обучения поддерживают непрерывное обучение и совершенствование посредством итеративного обучения на новых данных. По мере поступления дополнительных эксплуатационных данных системы обнаружения ошибок на основе глубокого обучения могут быть обновлены и усовершенствованы, чтобы адаптироваться к меняющимся эксплуатационным условиям, повышая их эффективность с течением времени [16,31].

Было предложено несколько методов обнаружения ошибок в работе контейнерных кранов STS. Авторы [32] представили новый метод обнаружения износа крановых рельсов, значительно повышающий точность, а [33] выявили и оценили человеческие ошибки среди операторов башенных кранов, уделив особое внимание важности мониторинга системы предотвращения столкновений.

Авторы работы [34] обратились к вопросу выявления физических воздействий на углы транспортных контейнеров при погрузочно-разгрузочных операциях с использованием методики обнаружения ударов. Он использовал методологию обнаружения ударов для обнаружения повторяющихся ударов причального крана об одни и те же места контейнера во время крюка. Эти современные хиты предполагают ложь

Machine Translated by Google

зацепление, которое может вызвать сильную деформацию металла в течение короткого времени. Исследование подчеркивает необходимость обнаружения повреждений морских контейнеров для обеспечения безопасности и эффективности порта . В [35] авторы разработали высокоскоростной алгоритм обнаружения объектов на основе цвета для систем помощи оператору причального крана, что значительно увеличивает скорость обнаружения при сохранении высокой точности. Эти исследования в совокупности подчеркивают важность постоянного совершенствования методов обнаружения ошибок для повышения безопасности и эффективности работы контейнерных кранов STS. Несколько исследований продемонстрировали эффективность глубокого обучения для прогнозирования ошибок при работе контейнерных кранов. Своевременное обнаружение ошибок имеет важное значение для оптимизации работы крана и обеспечения безопасности. Незамеченные ошибки или проблемы могут поставить под угрозу производство и безопасность, вызывая снижение эффективности, повреждение оборудования и возможные аварии [36].

Человеческая ошибка является существенным фактором, способствующим несчастным случаям во время работы мобильных кранов. Эффективное управление краном и точное позиционирование зависят от заранее установленных скоростей для каждого направления, определенных в соответствии со спецификациями крановой системы. Своевременное обнаружение проблем со скоростью, ускорением или замедлением может повысить производительность и избежать перегрузок или сбоев. Разработка интеллектуальных вспомогательных систем, которые обнаруживают ошибки в режиме реального времени, может повысить безопасность работы крана за счет снижения вероятности человеческих ош Квалифицированная экспертиза необходима для безопасной эксплуатации мостовых кранов и сведения к минимуму возникновения неисправностей оборудования. Тем не менее, в последнее время беспокойство вызывает нехватка квалифицированных операторов . Раннее выявление ошибок, таких как проблемы с натяжением каната или разница в весе груза, может помочь предотвратить несчастные случаи и повреждение оборудования, тем самым повышая безопасность, не зависящую только от навыков оператора [38].

Обеспечение быстрого выявления ошибок имеет важное значение для успешного внедрения эффективных автоматизированных траекторий крана и обеспечения удобного интерфейса, который помогает операторам выбирать и выполнять движения. Своевременное выявление ошибок при планировании или выполнении траектории может повысить производительность и предотвратить несчастные случаи [36]. Процесс ручного планирования, используемый в настоящее время для планирования пути подъема крана, сложен и подвержен ошибкам, в первую очередь из-за динамичных и перегруженных условий на строительных площадках. Внедрение автоматизированных систем, способных обнаруживать ошибки или столкновения в реальном времени, может повысить эффективность и точность работы [39].

Исследования по прогнозированию ошибок в работе контейнерных кранов STS разнообразны и охватывают различные аспекты работы кранов. Авторы [40] сосредоточились на анализе усталостных повреждений стареющих стальных конструкций, что является критическим фактором в прогнозировании ошибок. Авторы [41] определили конкретные факторы риска в стивидорных операциях, которые могут служить основой для стратегий прогнозирования ошибок. В [42] авторы рассмотрели необходимость создания надежных систем управления морскими контейнерными кранами, которые могут помочь уменьшить количество ошибок. Авторы [43] подчеркнули важность мониторинга структурного состояния для прогнозирования ошибок, обеспечивая потенциальный метод раннего обнаружения. Наконец, [44] выявили уменьшение среднего времени между отказами из-за тяжелых операций, увеличения количества инцидентов при техническом обслуживании и человеко-часов. Ключевые показатели эффективности были разработаны для оценки производительности кранов судно-берег и предоставления информации для будущих улучшений в работе оборудования контейнерных кранов. Результаты показывают, что показатели эффективности могут служить ориентиром для будущих проверок и улучшений в работе контейнерных кранов.

Многообещающий прогресс достигается в использовании глубокого обучения для выявления и прогнозирования ошибок в данных временных рядов. Для решения таких проблем, как шумная маркировка, ансамблевые методы идентификации неисправностей [45] и инновационные модели, такие как DCNN с расширенными возможностями GRU, для лучшей диагностики неисправностей в химических процессах [46], было предложено несколько стратегий. Кроме того, были исследованы модели глубокого обучения для выявления аномалий в телеметрических каналах космических аппаратов, что подчеркивает необходимость автоматизированных методов для эффективного мониторинга аномалий системы [47]. Эти исследования демонстрируют, как методы глубокого обучения, такие как рекуррентные единицы и сверточные нейронные сети, могут повысить эффективность обнаружения ошибок, обобщаемость и точность в различных промышленных областя

Глубокое обучение продемонстрировало замечательные возможности в анализе сложных структур данных и прогнозировании. Глубокие нейронные сети (DNN) превосходно справляются с обучением представлений.

Machine Translated by Google

на основе крупномасштабных данных, что позволяет им обнаруживать едва заметные сигналы, указывающие на предстоящие ошибки в работе крана. Такие методы, как RNN и CNN, были адаптированы для моделирования последовательных и пространственных зависимостей в данных датчиков крана, журналах операций и факторах окружающей среды. Ключевые архитектуры глубокого обучения, которые можно использовать для логических данных временных рядов контейнерных

кранов STS, включают следующие [18,48–50]: • Рекуррентные нейронные сети (RNN): RNN умеют обрабатывать последовательные данные, визуализируя их. подходит для анализа временных рядов. RNN могут эффективно улавливать временные зависимости в данных о работе крана [51].

• Длинная краткосрочная память (LSTM). Сети LSTM, вариант RNN, предназначены для решения проблемы исчезающего градиента. LSTM особенно хороши для моделирования долгосрочных зависимостей и успешно используются для прогнозирования поведения кранов. • Gated Recurrent Unit (GRU): GRU — это еще один вариант RNN, который сохраняет память, сохраняя при этом вычислительную эффективность. ГРУ показали многообещающую способность прогнозировать значения входной силы для управления краном.

RNN — это надежная технология машинного обучения, которая превосходно справляется с обработкой последовательных данных, таких как временные ряды и обработка естественного языка. PHC содержат петлю обратной связи, которая позволяет им изучать сложные корреляции во входных последовательностях и использовать эти структуры нелинейным образом [52]. RNN широко используются в различных областях, таких как анализ настроений, прогнозирование спроса на энергию и прогнозирование ежемесячного стока [53]. Существенным преимуществом RNN является их способность обрабатывать входные последовательности различной длины и создавать выходные последовательности различной длины. Эти характеристики делают их пригодными для таких задач, как моделирование языка, машинный перевод и распознавание речи [51]. Глубокие нейронные сети, способные изучать еще более сложные представления входных данных, могут быть созданы путем объединения RNN [54].

Более сложные архитектуры RNN, такие как LSTM и GRU, были созданы для решения проблемы долгосрочных зависимостей, которые могут быть проблематичными для обычных RNN.
Эти конструкции лучше фиксируют долгосрочные зависимости во входных данных за счет использования методов пропускания для выборочного сохранения и забывания информации.

В доступной литературе методологии, используемые для обнаружения, диагностики и прогнозирования ошибок или ошибок логических временных рядов в обрабатывающих отраслях на основе машинного обучения, можно увидеть в таблице 1 ниже, а глубокое обучение можно увидеть в таблице 2 ниже, а именно:

Таблица 1. Методологии машинного обучения.

Методология	Описание Эти	Ссылка
Обнаружение и диагностика неисправностей (FDD)	методологии обнаруживают возникающие нарушения и прогнозируют их последствия для поведения системы.	[55]
Диагностика неисправностей на основе данных	Этот подход использует алгоритмы машинного обучения для обнаружения механических неисправностей или прогнозирования неисправностей в производственном оборудовании.	[24]
Массив ML, DL и Алгоритмы DHL	Эти алгоритмы потенциально могут выполнять раннее обнаружение неисправностей, которые могут привести к отказу машины в будущем.	[56]
Автоматизированный подход машинного обучения	Этот подход используется для обнаружения и диагностики неисправностей в режиме реального времени в дискретных производственных машинах.	[57]

Контейнерные краны имеют решающее значение для контейнерных терминалов и имеют основополагающее значение для мировой экономики, поскольку через них проходит большая часть мировых товаров. Однако уровень их автоматизации ниже, чем у дворовых кранов и автоматизированных управляемых машин (АГВ) [21]. Здесь были изучены различные методологии, но пробел заключается в дальнейшем совершенствовании и расширении применения методов глубокого обучения для решения конкретных задач по обнаружению, диагностике и прогнозированию ошибок при анализе данных причальных контейнерных кранов. В конечном итоге это повысит безопасность, эффективность и автоматизацию портовых операций. Важной частью является ранняя диагностика отказов, которая включает в себя первоначальную идентификацию и прогнозирование ошибок. Предлагаемая методика направлена именно на это.

Machine Translated by Google

Таблица 2. Методологии глубокого обучения.

Методология	Описание Этот	Ссылка
Глубокое обучение для временных рядов Обнаружение аномалий	подход использует специализированные модели глубокого обучения для обнаружения аномальных закономерностей в данных временных рядов.	[58]
Глубокие генеративные модели	Вариационная реконструкция автоэнкодера по пути проекции (VAE-RaPP) и генеративно-состязательная сеть забора (Fence GAN) используются для работы с несбалансированными данными временных рядов, полученными из производственных приложений.	[59]
Модель глубокого метрического обучения	Эта модель учитывает несбалансированные данные о неисправности и способ проектирования четверной пары данных.	[60]
Модель ATT-1D CNN-GRU	Эта модель сочетает в себе одномерную CNN, GRU, механизм внимания и KG (граф знаний) для диагностики неисправностей.	[61]
Адаптивный на основе глубокого обучения Прогнозирование неисправностей	Этот подход объединяет комплексный модуль извлечения сверточных признаков, настраиваемый модуль вентилирования и модуль многоуровневого прогрессивного извлечения.	[62]

3. Методология .

Набор данных, использованный здесь для обучения и оценки модели глубокого обучения, включал фактические данные с контейнерных кранов на контейнерных терминалах. Он был в проприетарном формате и не мог быть прочитан напрямую. Он был преобразован в читаемый формат с помощью программирования Python. Он состоял из оперативных данных от ПЛК, датчиков и других вычислений. Извлеченные данные об ошибках были логическими, что означало, что выходные данные были 0 или 1.

Программные системы используются для диагностики и хранения данных ПЛК в течение 30 дней, после чего они удаляются. Это жизненно важный инструмент для работы крана и ежедневной оценки производительности. Он обеспечивает надежный источник данных в виде временных рядов, необработанных данных непосредственно от датчиков и их расчетов. Он обнаруживает и локализует периодические ошибки внутри крана. Он создает примерно от 680 до 730 файлов в месяц, сохраняя данные в час. В среднем кране он содержит около 23 000 переменных ПЛК , но в некоторых кранах может быть намного больше (например, 31 521 переменная ПЛК в STS CC2124). Это 518,4 миллиарда выборок в месяц (20 000 данных за 100 мс). Были выбраны функции данных (в данном случае ошибки и некоторые другие сигналы-идентификаторы), и данные из этих переменных были преобразованы в файлы CSV в Excel с использованием кода Python. Данные об ошибках содержали около 3000 переменных. Только два столбца не были логическими, включая последний для временных меток/временных рядов (неподходящего формата). Было около 3000 столбцов и более 16 100 000 (16 миллионов) строк.

В ходе анализа набора эксплуатационных данных были оценены восемь кранов из разных портов; наконец, подробный анализ трех кранов показал, что большинство из них имели 2852–2878 сигналов ошибок. Однако включалось всего от 3 до 6 сигналов, что свидетельствовало о том, что на кранах произошла реальная ошибка. Это указывало на редкость ошибок и нехватку соответствующих данных для дальнейшего анализа, что представляло собой серьезную проблему для проекта.

Был разработан специальный код на Python для извлечения данных из данных пролетариата, полученных с кранов. Полученные файлы Excel затем использовались для дальнейшей обработки данных.

Специально разработанный алгоритм был разработан с использованием алгоритмов SMOTE и LSTM в программировании на Python, а также настраиваемых процедур предварительной обработки, устранения несбалансированных данных и интеграции показателей оценки.

Предварительная обработка данных: для извлечения логических данных использовалась индивидуальная предварительная обработка. Первоначально загружались необработанные данные, состоящие из временных рядов и показаний датчиков . Обеспечение качества данных имело первостепенное значение для устранения любых пропущенных значений и выбросов. Впоследствии были извлечены логические функции, которые состояли из двоичных флагов или значений true/false.

Решение проблемы несбалансированных данных. В случае несбалансированного набора данных, когда один класс значительно недостаточно представлен, можно использовать SMOTE. Этот метод создает искусственные выборки для недостаточно представленной категории путем создания новых экземпляров путем интерполяции существующих точек данных. Цель состоит в том, чтобы добиться сбалансированного распределения классов, гарантируя справедливое представительство обоих классов. Далее исходный набор данных объе

Machine Translated by Google

искусственно созданные образцы. Впоследствии набор данных разделяется на отдельные подмножества для целей обучения и проверки. Для тестирования модели данные были разделены в соотношении 7:3, при этом 70% представляли собой данные обучения, а оставшиеся 30% — данные тестирования. Эти данные были преобразованы в трехмерный формат, в котором размерами были образцы, временные шаги и функции.

Построение модели LSTM. Разработка структуры LSTM имела решающее значение. Нейронная сеть LSTM построена с использованием входных слоев, представляющих последовательные данные, слоев LSTM, фиксирующих временные зависимости, и выходного слоя, выполняющего двоичную классификацию. Модель была составлена путем выбора функции потерь, такой как двоичная кросс-энтропия, оптимизатора, такого как Адам, и показателя оценки, такого как точность. Модель имела размер пакета 50 и обучалась в течение 40 эпох, состоящая из 2 слоев LSTM и одного плотного слоя для конечного результата. Начальное значение было предоставлено в начале, что помогло получить воспроизводимые результаты для модели, чтобы случайно разные результаты не генерировались при каждом запуске модели. После обучения модели ее производительность оценивалась на проверочном наборе и при необходимости вносились любые корректировки в гиперпараметры.

Оценка модели. Модель оценивалась с использованием ряда показателей, таких как точность, точность, полнота и показатель F1. Матрица путаницы предлагает полную разбивку прогнозов, включая истинно положительные, истинно отрицательные, ложноположительные и ложноотрицательные. Метрики оценки анализировали поведение модели и вносили все необходимые измен
Блок-схема специально созданного алгоритма показана ниже на рисунке 3.

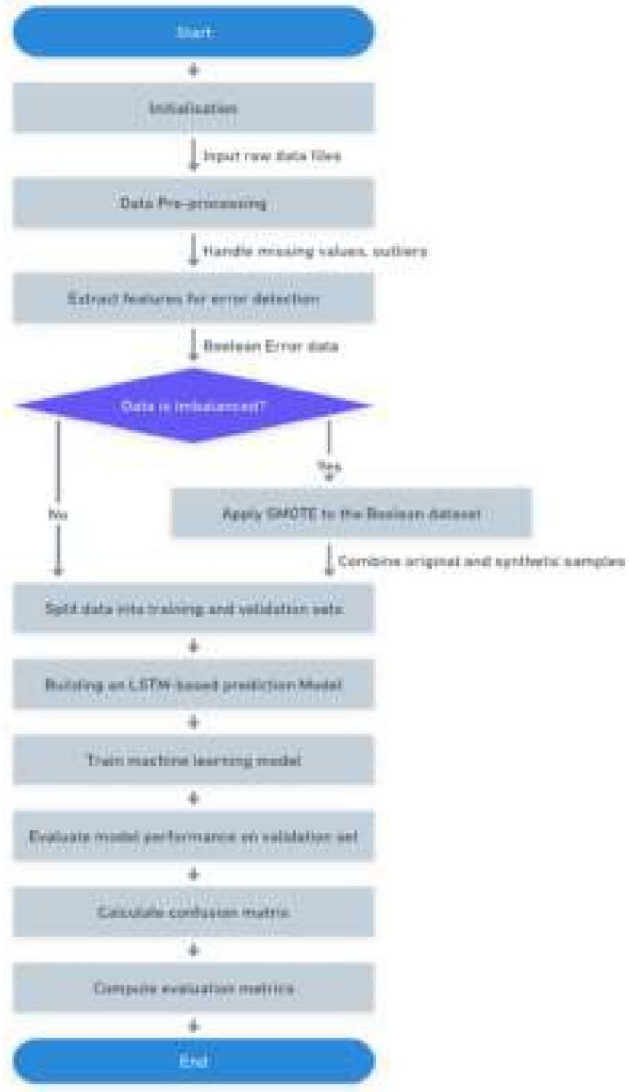


Рисунок 3. Блок-схема алгоритма, созданного по индивидуальному заказу, на основе SMOTE и LSTM.

Machine Translated by Google

4. Результаты

В этом разделе представлены экспериментальные результаты применения предложенного метода глубокого обучения. алгоритма для решения задачи обнаружения и прогнозирования ошибок.

4.1. Метрики оценки

Метрики оценки — это количественные меры, используемые для оценки производительности модели или алгоритма, особенно в машинном обучении, статистике и анализе данных. Эти метрики помогают определить, насколько хорошо работает модель, и помогают улучшить и оптимизировать ее. Переменные, обозначенные как var_xxxx в следующих показателях, представляют собой конкретные сигналы ошибок, генерируемые контейнерными кранами.

4.1.1. Ковариация

Ковариация — это последовательно аддитивная зависимость между двумя выборками данных. Эту взаимосвязь можно обобщить между двумя переменными, называемыми ковариацией. Он рассчитывается как среднее произведение значений из каждого образца, где значения были центрированы (вычтено их среднее значение) [63]. Ковариация не подразумевает причинно-следственную связь — она лишь измеряет статистическую взаимосвязь. Понимание ковариации может помочь выявить закономерности и зависимости при анализе данных. Ковариационная матрица представляет отношения между парами переменных в наборе данных. Он количественно определяет степень, в которой две переменные изменяются вместе.

$$\mathrm{Cov}(X,Y) = (\text{сумма}(x - \text{среднее}(X)) \times (y - \text{среднее}(Y))) \times 1/(n - 1) \quad (1)$$

На этой тепловой карте каждая ячейка соответствует ковариации между двумя переменными. Диагональные ячейки (слева сверху вниз справа) представляют собой ковариацию переменной самой себя. Эти ценности всегда положительны. Недиагональные ячейки (вверху справа и внизу слева) представляют ковариацию между различными переменными. Эти значения могут быть положительными или отрицательными. Тепловая карта использует интенсивность цвета для представления значений ковариации. Темно-красный цвет указывает на сильную положительную ковариацию (близкую к 1), что означает, что переменные имеют тенденцию одновременно увеличиваться или уменьш Темно-синий цвет указывает на отрицательную ковариацию (близкую к -1), что предполагает обратную зависимость. Более светлые оттенки представляют собой более слабую корреляцию (ближе к 0).

На тепловой карте выше на рисунке 4 показано, что метки по осям X и Y соответствуют конкретным переменным (например, «var_2663» и «var_831»). var_2663 представляет собой .tINTF_General_ Visu.tLaneProtection.tCover3.tMotor4.tMotor.tPT100.tAlarm.bStatus. и var_831—.tINTF_ General_Visu.tLaneProtection.tCover1.tMotor4.tCB.tSignalWithError.tError.bHasError.

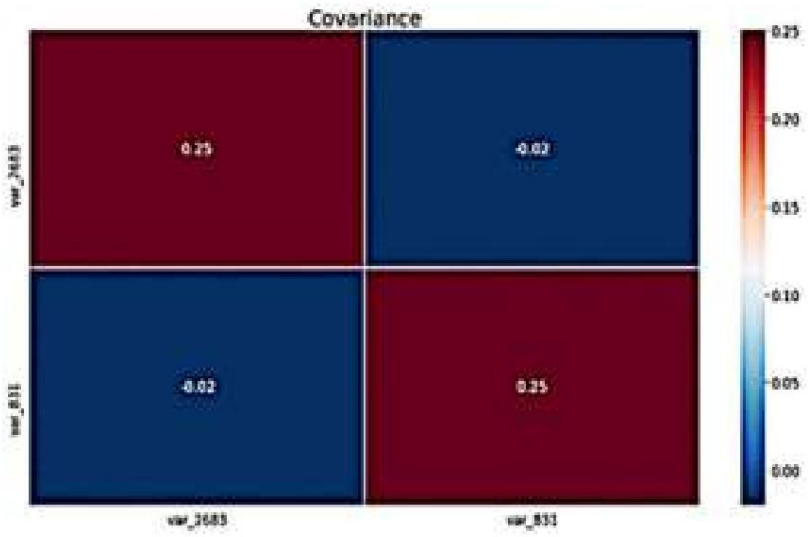


Рисунок 4. Ковариация.

Machine Translated by Google

Верхний левый квадрант имеет темно-красный оттенок, что указывает на устойчивую положительную ковариацию (около 0,25) между var_2663 и самим собой. Нижний правый квадрант также показывает устойчивую положительную ковариацию для var_831. Голубые недиагональные квадранты предполагают слабую обратную связь (около -0,02) между var_2663 и var_831, указывая на то, что двигатель 4 имеет положительную ковариацию с ошибкой датчика температуры в четвертом блоке управления двигателем .

4.1.2. Корреляционная матрица

Корреляционная матрица — это часто используемый многомерный метод определения сильно связанных переменных. Этот аспект проекта был сосредоточен на корреляции всех переменных в сигналах ошибок и некоторых рабочих сигналах, которые считались зависимыми переменными, таких как CraneOn, CraneOff и блокировка поворотного замка. Переменные ошибки только с одним уникальным значением, 0, были удалены, поскольку однородные переменные содержат только одно уникальное значение и не вносят дополнительную информацию в модель. Корреляция между переменными была проанализирована и визуализирована, чтобы лучше понять данные. Корреляция использовалась для определения взаимосвязи между переменными. Оно может быть положительным, указывая , что обе переменные движутся в одном направлении, или отрицательным, указывая, что когда значение одной переменной увеличивается, значения других переменных уменьшаются. Корреляция также может быть нейтральной или нулевой, что указывает на то, что переменные не связаны между собой или не связаны между собой.

Корреляция Пирсона

Коэффициент корреляции Пирсона вычисляется путем деления ковариации двух переменных на произведение их стандартных отклонений. Это нормализация ковариации между двумя непрерывными переменными для получения интерпретируемой оценки. Он количественно определяет степень , в которой изменение одной переменной связано с изменением другой переменной [63].

$$\text{Корреляция Пирсона (r)} = \text{ковариация}(X,Y)/(\text{stdv}(X)\times \text{stdv}(Y)) \quad (2)$$

На этой тепловой карте, показанной на рисунке 5 выше, сравниваются две переменные var_2683, т.е. .tINTF_General_Visu.tLaneProtection.tCover3.tMotor4.tMotor.tPT100_Group.tPT100_ 2.tError.bHasError и «var_831», т.е. .tINTF_General_Visu.tLaneProtection.tCover1. tMotor4.tCB.tSignalWithError.tError.bHasError. На тепловой карте отображается коэффициент корреляции между этими переменными. Диагональные квадранты (где переменная сравнивается сама с собой) показывают коэффициент корреляции 1 (как и ожидалось). Недиагональные квадранты обнаруживают слабую отрицательную корреляцию примерно -0,079. Цветовая шкала варьируется от -1 до 1. Темно-красный означает более сильную отрицательную корреляцию, а темно-синий — более сильную положительную корреляцию.

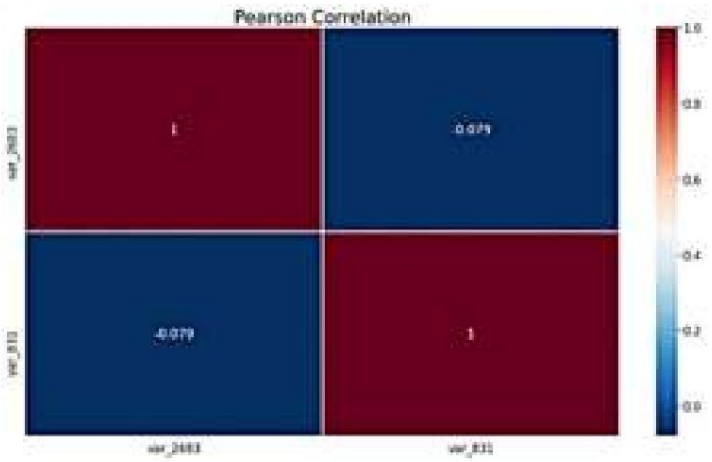


Рисунок 5. Корреляция Пирсона.

Machine Translated by Google

Корреляция Спирмена

Корреляция Спирмена рассчитывалась на основе относительного ранга значений каждой выборки. Этот подход обычно используется в непараметрической статистике, например, в статистических методах, где не предполагается никакого конкретного распределения данных, такого как распределение Гаусса [64]. Этот коэффициент измеряет силу и направление монотонной связи между двумя ранжированными переменными. Это полезно при выявлении и оценке нелинейных связей и порядковых данных.

Корреляция Спирмена = ковариация(ранг(X),ранг(Y))/std(ранг(X)) × std(ранг(Y)) (3)

Эта тепловая карта, показанная выше на рисунке 6, разделена на четыре квадранта, чтобы показать коэффициент корреляции Спирмена между двумя переменными, обозначенными как var_2683—.tINTF_General_Visu.tLaneProtection.tCover3.tMotor4.tMotor.tPT100_Group.tPT100_2.tError.bHasError на оси Y. и var_831—.tINTF_General_Visu.tLaneProtection.tCover1.tMotor4.tCB. tSignalWithError.tError.bHasError по оси X. Два диагональных квадранта демонстрируют идеальную положительную корреляцию (значение 1) для каждой переменной с самой собой (обозначается красным цветом и числом ff 1). Недиагональные квадранты демонстрируют очень низкую отрицательную корреляцию -0,079 между двумя переменными (обозначены темно-синим цветом).

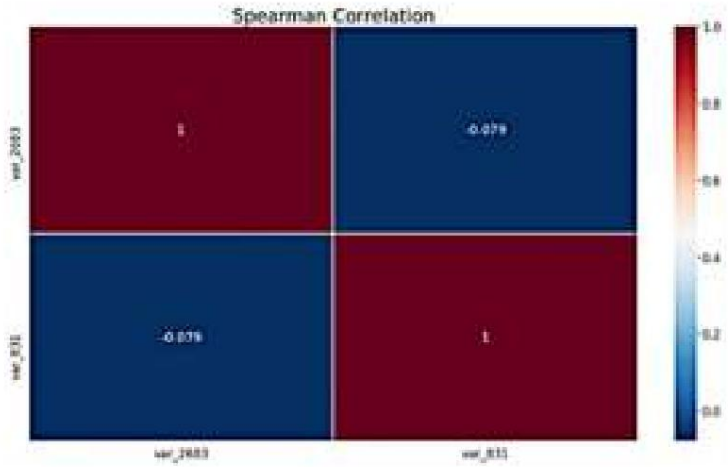


Рисунок 6. Корреляция Спирмена.

Кендалл Корреляция

Коэффициент ранговой корреляции Кендалла Тау измеряет степень сходства между двумя наборами рангов, присвоенными одному и тому же набору объектов. В отличие от коэффициента Спирмена, Тау Кендалла измеряет только согласованность направлений, а не ранговые различия. ff

KendallCorrelation = (Количество согласующихся пар — Количество несогласованных пар)/n(n — 1)/2 (4)

Тепловая карта корреляции Кендалла, показанная на рисунке 7 , подтверждает результаты других корреляций. Он фокусируется на тех же двух переменных, что и другие тепловые карты корреляции, т. е. var_2683 —.tINTF_General_Visu.tLaneProtection.tCover3.tMotor4.tMotor.tPT100_ Group.tPT100_2.tError.bHasError по оси Y и var_831—.tINTF_General_ Visu.tLaneProtection. tCover1.tMot Верхний левый и нижний правый квадранты темно-красные, что указывает на идеальную корреляцию 1. Это ожидаемо, поскольку каждая переменная коррелирует сама с собой. Верхний правый и нижний левый квадранты окрашены в синий цвет, что указывает на слабую отрицательную корреляцию (приблизительно -0,079) между var_2683 и var_831.

Machine Translated by Google

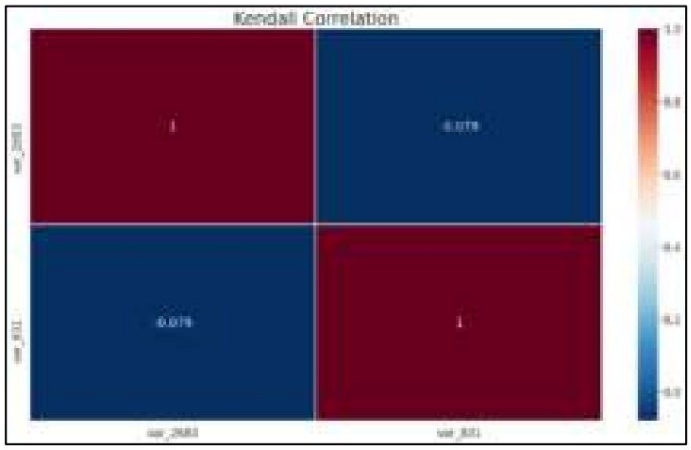


Рисунок 7. Корреляция Кендалла.

4.1.3. Функция потери обучения и теста

В машинном обучении потери при обучении и потери при тестировании (или потери при проверке) являются критически важными показателями, используемыми для оценки производительности модели в процессе обучения. Эти функции потерь помогают измерить, насколько хорошо модель работает на обучающих и невидимых данных (данных тестирования или проверки). Потери обучения — это ошибка или несоответствие между прогнозируемыми и фактическими результатами набора обучающих данных. Он позволяет оценить, насколько хорошо модель соответствует обучающим данным. Потери при тестировании (или потери при проверке) — это ошибка между прогнозируемыми и фактическими выходными данными теста или набора проверочных данных. Этот набор данных не используется во время обучения, а сохраняется для оценки эффективности модели на невидимых данных.

Рассчитывается значение среднеквадратической ошибки (RMSE). Квадратный корень из среднего квадратичная ошибка обеспечивает измерение в тех же единицах, что и целевая переменная.

На графике на рисунке 8 показаны две линии: обучающая (синяя) и тестовая (оранжевая). Горизонтальная ось представляет количество эпох в диапазоне от 0 до 40, а вертикальная ось представляет потери в диапазоне от 0,0 до 0,7. Обе линии изначально показывают резкое снижение потерь в ранние эпохи, что указывает на эффективное обучение на обучающих данных. С течением эпох обе линии постепенно выравниваются, что указывает на уменьшение отдачи от обучение и риск переобучения, если потери при тестировании увеличиваются. Линия поезда представляет собой потерю обучающих данных, а тестовая линия представляет собой потерю невидимых тестовых данных. Мониторинг обеих линий помогает оценить производительность модели и способность к обобщению, а также помогает диагностировать такие проблемы, как переоснащение, недостаточное оснащение и сходимость. Этот график академически значим при оценке моделей машинного обучения и используется для оптимизации стратегий обучения и повышения производительности моделей.

4.1.4. Матрица путаницы

Матрица путаницы является фундаментальным инструментом для оценки эффективности моделей классификации. Он дает полное представление о том, насколько хорошо модель прогнозирует различные классы. Матрица путаницы основана на задаче двоичной классификации с двумя классами: 0 и 1. Эти классы могут представлять различные результаты, такие как «пройдено/не пройдено», «да/нет» или «истина/ложь». Матрица разбита на четыре квадранта: • Истинно отрицательные значения (TN): экземпляры,

правильно предсказанные как класс 0. Это желательные прогнозы — правильно идентифицирующие экземпляры как отрицательные (класс 0);

• Ложные срабатывания (FP): случаи, неправильно предсказанные как класс 1, хотя на самом деле они относятся к классу 0. Это ошибки — случаи, ошибочно классифицированные как положительные (класс 1), хотя это не так;

• Ложноотрицательные результаты (FN): случаи, неправильно предсказанные как класс 0, хотя на самом деле они относятся к классу 1. Это также ошибки — случаи, ошибочно классифицированные как отрицательные, хотя они положительные;

Machine Translated by Google

• Истинные положительные результаты (TP): экземпляры, правильно предсказанные как класс 1. Это желательные прогнозы — правильно идентифицирующие экземпляры как положительные.

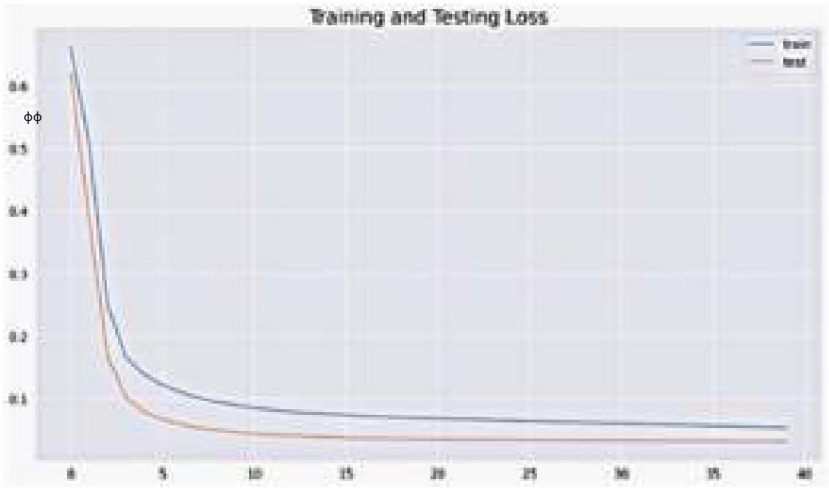


Рисунок 8. Функция потерь при обучении и тестировании.

Матрица путаницы, показанная на рисунке 9 , суммирует эффективность модели классификации. Это дает представление о том, насколько хорошо прогнозы модели соответствуют фактическим результатам.

1. Истинно отрицательный результат (TN): модель правильно предсказала отрицательные случаи (фактический результат был отрицательным). Истинно отрицательных результатов 997; 2. Ложное срабатывание (FP): модель неправильно предсказала положительные случаи (фактический отрицательный результат). Здесь девять ложных срабатываний; 3. Ложноотрицательный результат (FN). Модель неправильно предсказала отрицательные случаи (фактический результат). положительный результат). Матрица показывает четыре ложноотрицательных результата;
4. Истинно положительный результат (TP): модель правильно предсказывала положительные случаи (фактический результат был положительным). В этой матрице есть четыре истинных позитива.

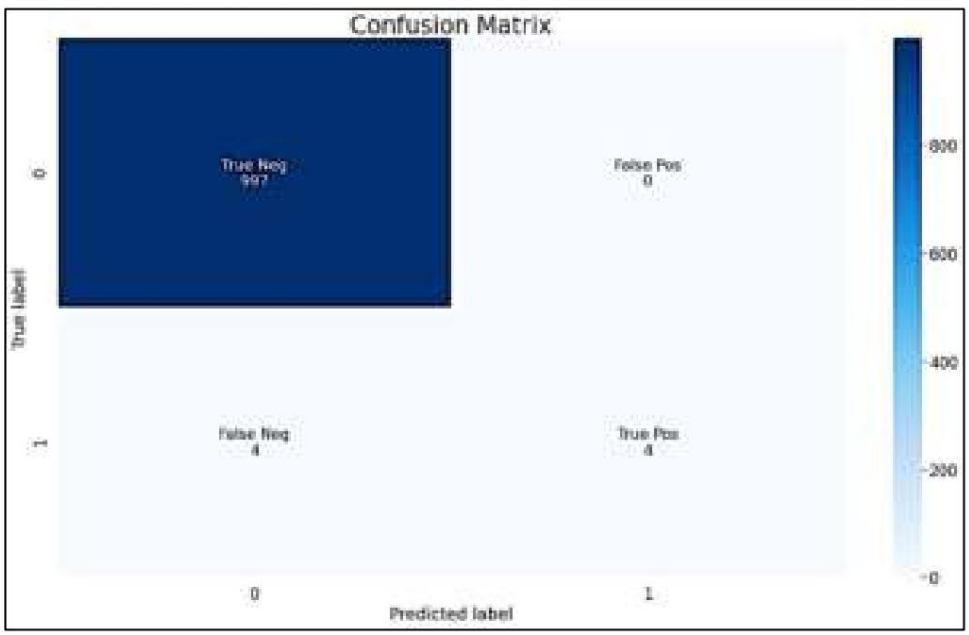


Рисунок 9. Матрица путаницы.

Machine Translated by Google

4.1.5. Точность, точность, отзыв, оценка F1

Точность, точность, полнота и показатель F1, показанные на рисунке 10 ниже, также были оценены.
рассчитывается и распечатывается для прогнозируемых показателей оценки данных. Эти модельные выступления
затем были сохранены в текстовом файле.

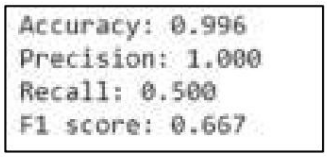


Рисунок 10. Результаты оценки метрик.

Точность: количество правильных прогнозов делится на общее количество.
предсказаний.

$$\text{Точность} = (\text{Истинные положительные результаты} + \text{Истинные отрицательные значения})/\text{Всего случаев} \quad (5)$$

Точность: измеряет точность прогнозов положительных классов путем определения
сколько из них на самом деле относятся к положительному классу.

$$\text{Точность} = \text{Истинные положительные результаты}/(\text{Истинные положительные результаты} + \text{Ложные положительные результаты}) \quad (6)$$

Напомним (чувствительность): рассчитывает долю правильных положительных прогнозов.
всех положительных примеров в наборе данных.

$$\text{Отзыв} = \text{Истинные положительные результаты}/(\text{Истинные положительные результаты} + \text{Ложные отрицательные результаты}) \quad (7)$$

Оценка F1: это единая оценка, которая уравновешивает точность и полноту в одном числе.
Это гармоничное средство точности и отзыва.

$$\text{Оценка F1} = 2 \times ((\text{Точность} \times \text{отзыв})/(\text{Точность} + \text{отзыв})) \quad (8)$$

Точность: Достижение точности 0,996 указывает на общую хорошую производительность.

Точность: показатель точности 1,00 означает, что модель редко ошибочно классифицировала истинное значение.
ошибки как не ошибки.

Напомним: балл отзыва 0,500 указывает на то, что модель идентифицировала только половину фактических данных.
ошибки. Этот компромисс между точностью и полнотой часто встречается в несбалансированных наборах данных.

Оценка F1: Оценка F1 (0,667) уравновешивает точность и полноту, обеспечивая всестороннюю оценку.
мнение об эффективности модели.

4.2. Прогнозирование временных рядов данных об ошибках

Конечным результатом является предсказание следующих n секунд данных, в которых пользователь
вначале предоставляет n секунд в качестве входных данных, а соответствующие данные сохраняются в
CSV-файл. Для этого следующая 1 с данных прогнозируется на следующие 60 с с помощью
исторические данные доступны для анализа. Выходные данные прогнозируемого файла CSV для одного из
переменные ошибок показаны ниже в Таблице 3:

Таблица 3. Прогнозируемые данные об ошибках.

Временная метка	var_831
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0

Machine Translated by Google

Таблица 3. Продолжение.

Временная метка	var_831
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	1
11.12.2020 20:35	1
11.12.2020 20:35	1
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	0
11.12.2020 20:35	1
11.12.2020 20:35	1
11.12.2020 20:35	1
11.12.2020 20:36	1
11.12.2020 20:36	1
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	1
11.12.2020 20:36	1
11.12.2020 20:36	1
11.12.2020 20:36	1
11.12.2020 20:36	1
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0

Machine Translated by Google

Таблица 3. Продолжение.

Временная метка	var_831
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0
11.12.2020 20:36	0

5. Обсуждение

Своевременное обнаружение и прогнозирование ошибок имеют важное значение для оптимизации работы крана. Они имеют решающее значение для обеспечения безопасности, предотвращая несчастные случаи, защищая оборудование от ущерб и сокращение человеческих ошибок. В результате эти методы повышают производительность и более низкие операционные расходы.

Понимание ограничений прогнозной аналитики имеет решающее значение. Прежде всего, прогресс невозможно без обширного и качественного набора обучающих данных. Во-вторых, наличие Четкое описание прогнозируемой концепции и прошлых событий имеет решающее значение. Один из Задача прогнозной аналитики — найти иголку в стоге сена и спрогнозировать ее возникновение. Однако традиционные методы имеют определенные ограничения. Эти ограничения в контейнерных крановых работах заключаются в следующем:

1. Человеческие ошибки: люди-операторы по-прежнему играют значительную роль в работе причальных кранов. операций, несмотря на достижения автоматизации. Их решения и действия могут привести к ошибки, влияющие на производительность и безопасность [64].
2. Сложная динамика. Контейнерные краны работают в динамичной среде с меняющимися вес контейнеров, ветровые условия и движение судна. Прогнозирование ошибок в таких сложная система представляет собой сложную задачу [65–67].
3. Эффективность и время простоя. Минимизация времени простоя из-за ошибок имеет решающее значение. Своевременная ошибка прогнозирование может предотвратить дорогостоящие задержки и оптимизировать использование крана [68,69].
4. Несбалансированные данные. Набор данных называется несбалансированным, если существуют существенные различия. в количестве примеров каждого класса, в результате чего один или несколько классов являются недостаточно представлены в наборе данных [70]. Это явление было установлено в несколько реальных наборов данных, собранных в промышленных условиях.

Поэтому в ходе данного исследования был разработан новый, специально разработанный алгоритм для решения этих проблем. ограничений и проблем, сосредоточив внимание на обнаружении и прогнозировании ошибок до того, как они превращаются в неисправности и неудачи.

Следующие пункты касаются любых проблем и ограничений, возникших во время реализация:

1. Поскольку между файлами массива данных была значительная разница во времени, данные файлы массива не удалось объединить для анализа; вместо этого анализ проводился по каждому файл массива данных отдельно.

Machine Translated by Google

2. Для прогнозирования временных рядов данные были уменьшены с миллисекунд до секунд, поскольку данные в миллисекундах не были непрерывными, т. е. (1, 2, 3,..., 999).
3. Вероятность ошибки может быть высокой, поскольку размер набора данных был небольшим, поэтому модель LSTM не могла должным образом и правильно предсказать новые данные. Эту проблему можно решить , добавив больше данных .

Несмотря на многообещающие результаты, внедрение моделей глубокого обучения для прогнозирования ошибок в реальных операциях крана остается сложной задачей. К ним относятся проблемы качества данных, интерпретируемость сложных моделей и интеграция с существующими системами. Будущие направления исследований могут решить эти проблемы с помощью передовых архитектур моделей, методов объединения данных и совместных усилий научных кругов и заинтересованных сторон отрасли. Будущие исследования могут развиваться за счет улучшения качества и количества данных, которые в настоящее время отстают.

6. Выводы

Исследования контейнерных кранов охватывают различные аспекты, связанные с их конструкцией, работой и влиянием на портовую логистику. Однако наблюдается явное отсутствие исследований, посвященных раннему выявлению ошибок для предотвращения сбоев и сбоев. Своевременное прогнозирование ошибок с помощью алгоритмов глубокого обучения может революционизировать работу кранов и обеспечить оптимальную эффективность работы контейнерных кранов.

Был разработан специально разработанный алгоритм глубокого обучения на основе LSTM, обеспечивающий точность 99,6% при прогнозировании сигналов ошибок крана с использованием LSTM и SMOTE для прогнозирования классификации с точностью 1,00, полнотой 0,500 и показателем f1 0,667. Однако для улучшения результатов модели необходимы дополнительные данные.

Доступность текущих данных ограничена месяцем и включает примерно 30 000 переменных, включая менее 3000 переменных ошибок. Полезные данные представляли собой небольшой набор выборок, включающий всего от 3 до 6 переменных ошибок, которые время от времени активировались. Исследование также предоставляет другие результаты, такие как матрицы корреляции, ковариации и путаницы. Эксперименты продемонстрировали способность метода выявлять важные элементы, что привело к усилиям по улучшению структуры модели для улучшения прогнозирования ошибок в больших промышленных данных.

Хотя модель вывода опирается на уже существующие данные, ее способность обрабатывать несбалансированные данные, что является распространенной проблемой в большинстве отраслей, представляет собой значительное преимущество. При наличии дополнительных данных существующая модель может быть усовершенствована. Непрерывные данные в течение нескольких месяцев обеспечат более полный набор выборок данных об ошибках, что будет способствовать улучшению результатов. Как только в будущем будет получен больший набор данных, можно будет использовать другие алгоритмы, такие как случайный лес, наивный Байес и усиление XG.

Продолжающиеся исследования и инновации в этой области необходимы для полной реализации преимуществ прогнозной аналитики в портовых операциях.

Вклад авторов: Концептуализация, А.А.; курирование данных, АА; формальный анализ – АА; методология , АА; программное обеспечение, АА; надзор, ЛК и ДВ; валидация, АА; визуализация, АА; письменное — первоначальный вариант, А.А.; написание — рецензирование и редактирование, АА, LK и JW. Все авторы прочитали опубликованную версию рукописи и согласились с ней.

Финансирование: Этот проект был частично поддержан грантом Ирландского научного фонда 13/RC/2094_P2 и совместно профинансирован Европейским фондом регионального развития через Южную и Восточную региональную оперативную программу для Леро — Ирландского научно-исследовательского центра программного обеспечения Научного фонда (www.lero.ie, доступ осуществлен 20 июля 2024 г.) в течение первых двух лет реализации этого проекта.

Заявление о доступности данных. Наборы данных, представленные в этой статье, недоступны из- за проблем с конфиденциальностью или безопасностью данных контейнерного порта. Запросы на доступ к наборам данных следует направлять соответствующему автору этой статьи.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Machine Translated by Google

Ссылки

1. Мордорская разведка. Рынок контейнерных перевозок — анализ размера, доли и роста. Отчеты разведывательной отрасли Мордора. [Онлайн]. Доступно онлайн: https://www.mordorintelligence.com/industry-reports/global-container-shipping-market. (по состоянию на 17 июля 2024 г.).
2. Линдеберг Э.; Повышение эксплуатационных показателей кранов СТС. Вестерос, Швеция, февраль 2011 г. Доступно онлайн: www.porttechnology.org (по состоянию на 18 июня 2021 г.).
3. Размер рынка контейнерных кранов типа «судно-берег» (STS), доля, прогноз отрасли на 2028 год. Доступно онлайн: https://www.Fortunebusinessinsights.com/ship-to-shore-sts-container-cranes-market-102880 (по состоянию на 15 июня 2021 г.).
4. Зрнию, Н.; Хоффманн, К. Разработка конструкции контейнерных кранов типа «судно-берег»: 1959–2004 гг. На Международном симпозиуме по История машин и механизмов; Springer: Дордрехт, Нидерланды, 2007 г. [CrossRef]
5. Хухи А.; Халид, М.Х. Методы гибридных вычислений для обнаружения и изоляции неисправностей, обзор. Вычислить. Электр. англ. 2015, 43, 17–32. [Перекрестная ссылка]
6. Зальцер, Дж. Х.; Франс Каашук, М. 2.2: Неисправности, сбои и отказоустойчивое проектирование 2.2.1: Неисправности, сбои и модули. В принципах проектирования компьютерных систем; Массачусетский технологический институт LibreTexts: Дэвис, Калифорния, США, 2024 г.; Глава 2.2, стр. 1–461. Доступно онлайн: https://eng.libretexts.org/@go/page/58498 . (по состоянию на 19 июля 2024 г.).
7. Сафаи, ММ; Хиджазян, С.; Педраммер, С.; Пакзад, М.; Эттефаг, М.; Фотоухи М. Обнаружение повреждений козлового крана с движущейся массой с использованием искусственной нейронной сети. Здания 2024, 14, 458. [CrossRef]
8. Ван З.; Ченг, Дж.; Ху, Х. Проактивно-реактивный подход к непрерывному распределению причалов и назначению причальных кранов Проблемы гибридной неопределенности. Дж. Мар. англ. 2024, 12, 182. [CrossRef]
9. Гаттузо, Д.; Пратико, ФГ; Лонго, Р.; Кассоне, Г.; Винья, М.; Сцени, Р. Железнодорожные перевозки на грузовых терминалах: вопросы безопасности и предлагаемая методология. На 5-й Международной конференции IEEE по моделям и технологиям интеллектуальных транспортных систем, MT-ITS 2017 — Proceedings, Неаполь, Италия, 26–28 июня 2017 г.; Институт инженеров по электротехнике и электронике Inc.: Пискатауэй, Нью-Джерси, США, 2017 г.; стр. 164–169. [Перекрестная ссылка]
10. Хуанг, М.; Лю, З.; Тао, Ю. Диагностика и прогнозирование механических неисправностей в Интернете вещей на основе объединения данных зондирования из нескольких источников. Симул. Модель. Практика. Теория 2020, 102, 101981. [CrossRef]
11. Меддауи, А.; Хейн, М.; Хачмуд, А. Преимущества профилактического обслуживания для совершенствования производства: тематическое исследование по созданию надежных методов прогнозирования отказов. Межд. Дж. Адв. Производитель Технол. 2023, 128, 3685–3690. [Перекрестная ссылка]
12. Фернандес, М.; Канито, А.; Болон-Канедо, В.; Консейсан, Л.; Прача, И.; Маррейрос, Г. Анализ данных и выбор функций для профилактического обслуживания: практический пример металлургической промышленности. Межд. Ж. Инф. Менеджер. 2019, 46, 252–262. [Перекрестная ссылка]
13. Да, JKW; Вонг, Дж. Х.; Пэн, Л. Интеграция информационных моделей кранов в BIM для проверки соответствия требованиям плана подъема. В материалах 33-го Международного симпозиума по автоматизации и робототехнике в строительстве (ISARC 2016), Оберн, Алабама, США, 18–21 июля 2016 г.; Доступно онлайн: https://www.iaarc.org/publications/fulltext/ISARC2016-Paper192.pdf. (по состоянию на 20 июля 2024 г.).
14. Чу, Ф.; Гаилус, С.; Лю, Л.; Ни, Л. Будущее автоматизации портов | McKinsey. 2018. Доступно онлайн: https://www.mckinsey.com/.отрасли/туристическая логистика и инфраструктура/наши идеи/будущее автоматизированных портов (по состоянию на 18 мая 2024 г.).
15. Лепениоти, К.; Перцелакис, М.; Буздекис, А.; Лука, А.; Лампатаки, Ф.; Апостолу, Д.; Менцас, Г.; Анастасиу, С. Машинное обучение для прогнозной и предписывающей аналитики операционных данных в интеллектуальном производстве. На семинарах по проектированию передовых информационных систем ; Спрингер: Чам, Швейцария, 2020 г. [CrossRef]
16. Что такое глубокое обучение? | Руководство для начинающих. Доступно онлайн: https://www.scribbr.com/ai-tools/deep-learning/. (доступ на 19 мая 2024 г.).
17. Чжао Р.; Ян, Р.; Чен, З.; Мао, К.; Ван, П.; Гао, Р.С. Глубокое обучение и его применение для мониторинга работоспособности машин. Мех. Сист. Сигнальный процесс. 2019, 115, 213–237. [Перекрестная ссылка]
18. Кумар А.; Гаур, Н.; Чакраварти, С.; Альшариф, Миннесота; Утансакул, П.; Утансакул, М. Анализ спектрального зондирования с использованием глубокого алгоритмы обучения: CNN и RNN. Айн Шамс англ. J. 2024, 15, 102505. [CrossRef]
19. Убал, К.; Ди-Джорджи, Г.; Контрерас-Рейес, JE; Салас Р. Прогнозирование долгосрочных зависимостей во временных рядах с использованием рекуррентных Искусственные нейронные сети. Мах. Учиться. Знать. Экстра. 2023, 5, 13:40–13:58. [Перекрестная ссылка]
20. Хейлиг, Л.; Сталбок, Р.; Восс, С. От цифровизации к принятию решений на основе данных на контейнерных терминалах. арXiv 2019, arXiv: 1904.13251.
21. Шин, Х.-С.; Ли, С.-П.; Ха, Ю.-С.; Ким, Х.-С. Разработка модели обучения управлению контейнерным краном с использованием глубокого обучения. Дж. Адв. Мар. англ. Технол. 2023, 47, 2023–2234. [Перекрестная ссылка]
22. Чалапати, Р.; Чавла, С. Глубокое обучение для обнаружения аномалий: обзор. arXiv 2019, arXiv:1901.03407.
23. Шаглуф А.; Лонгстафф, АР; Флетчер, С. (PDF) Стратегии технического обслуживания для сокращения времени простоя из-за ошибок позиционирования машины, измерение и управление эффективностью технического обслуживания (MPMM). Доступно в Интернете: https://www.researchgate.net/publication/267864966_Maintenance_Strategies_to_Reduce_Downtime_Due_to_Machine_Positional_Errors . (по состоянию на 18 июля 2024 г.).
24. Фернандес, М.; Корчадо, Дж. М.; Маррейрос, Г. Методы машинного обучения, применяемые для диагностики и прогнозирования механических неисправностей в контексте реальных сценариев промышленного производства: систематический обзор литературы. Прил. Интел. 2022, 52, 14246–14280. [Перекрестная ссылка] [ПабМед]

Machine Translated by Google

25. Тестирование программного обеспечения — ошибка, дефект, ошибка, сбой или сбой — GeeksforGeeks. Доступно онлайн: https://www.geeksforgeeks.org/software-testing-bug-vs-defect-vs-error-vs-fault-vs-failure/ (по состоянию на 22 апреля 2024 г.).
26. Джалаер, М.; Орсениго, К.; Верселлис, К. Обнаружение и диагностика неисправностей вращающихся машин: модель, основанная на сверточном методе. LSTM, быстрое преобразование Фурье и непрерывное вейвлет-преобразование. Вычислить. Индиана 2021, 125, 103378. [CrossRef]
27. Чжао Р.; Ян, Р.; Ван, Дж.; Мао, К. Обучение мониторингу состояния машин с помощью сверточных двунаправленных сетей LSTM. Датчики 2017, 17, 273. [CrossRef] [ПабМед]
28. Зрнич, Н.; Петкович, З.; Бошняк, С.М. Автоматизация контейнерных кранов типа «судно-берег»: обзор современного состояния. Транзакции FME 2005, 33, 111–121. Доступно онлайн: https://www.researchgate.net/publication/228936324_Automation_of_ship-to-shore_. Container_cranes_A_review_of_state-of-the-art (по состоянию на 24 марта 2024 г.).
29. Тран, QH; Ха, Дж.; Нгуен, В.Б.; Канг, К.; Ан, Дж. Х.; Парк, И. Дж. Анализ чувствительности конструкции контейнерного крана типа «судно-берег». Прил. наук. 2018, 8, 1667. [CrossRef]
30. Ли З. Подходы к прогнозному техническому обслуживанию, основанные на глубоком обучении. Структура интеллектуальной диагностики и прогнозирования неисправностей в эпоху Индустрии 4.0. доктор философии Диссертация, Норвежский университет науки и технологий, Тронхейм, Норвегия, 2018 г.
31. Гудфеллоу, И.; Бенджио, Ю.; Курвиль, А. 2015. Доступно в Интернете: http://www.deeplearningbook.org/. (по состоянию на 15 июня 2024 г.).
32. Лю, В.; Сяо, Дж. Новый метод обнаружения износа крановых путей и его анализа ошибок. Дж. Мех. Сила 2014, 36, 878–883. Доступно онлайн: https://www.researchgate.net/publication/293095272_A_new_method_of_detection_the_wear_of_crane_ . рельсы_and_its_error_anaанализ (по состоянию на 19 марта 2024 г.).
33. Мохаммадфам, И.; Боргейпур, Х.; Тегерани, Г.; Мадади, С. Выявление и оценка человеческих ошибок среди операторов башенных кранов с использованием методов SHERPA и CREAM. J. Health Saf. Работа. 2020, 10, 5–8.
34. Яковлев С.; Эглинас, Т.; Юсис, М.; Вознак, М.; Партила, П.; Товарек Дж. Обнаружение физического воздействия на углы транспортных контейнеров во время погрузочно-разгрузочных операций, выполняемых причальными кранами. Дж. Мар. англ. 2023, 11, 794. [CrossRef]
35. Гао, Х.; Да, ХГ; Марайонг, П. Высокоскоростной алгоритм обнаружения объектов на основе цвета для системы помощи оператору причального крана. В материалах 11-й ежегодной международной системной конференции IEEE, SysCon 2017 — Proceedings, Монреаль, Квебек, Канада, 24–27 апреля 2017 г. [CrossRef]
36. Вестерберг С.; Ширяев А. Телеуправление лесозаготовительными машинами на основе виртуальной среды: разработка методов будущего взаимодействия. Дж. Хум.-Робот. Взаимодействуйте. 2013, 2, 84–110. [Перекрестная ссылка]
37. Рен, В.; Ву, З.; Чжан Л. Планирование схемы подъема в режиме реального времени с помощью контроллеров, установленных на мобильных кранах. Может. Дж. Сив. англ. 2016, 43, 542–552. [Перекрестная ссылка]
38. Ватанабэ, М.; Момои, Ю.; Одай, М.; Иешиге, К. Одновременная оценка натяжения каната и нагрузки кранов с использованием моторного привода. сигналы. Пер. JSME 2023, 89. [CrossRef]
39. Лей, З.; Бехзадипур, С.; Аль-Хусейн, М.; Герман У. Применение роботизированного обхода препятствий при планировании пути подъема крана. В Материалы 28-й конференции ISARC, Сеул, Республика Корея, 29 июня – 2 июля 2011 г.
40. Ленер, П.; Кривой, В.; Крейса, М.; Панреница, П.; Козак, Дж. Стохастический прогноз срока службы существующей стальной конструкции, нагруженной Мостовые краны. Структура процедуры. Интегр. 2018, 13, 1539–1544. [Перекрестная ссылка]
41. Шан, К.К.; Ценг, В.Дж. Анализ рисков стивидорных операций на контейнерных терминалах морских портов. Дж. Мар. Технол. 2010, 18, 201–210. [Перекрестная ссылка]
42. Исмаил, РМТР; Это, Н.Д.; Ха, К.П. Моделирование и надежное отслеживание траектории морских контейнерных крановых систем. Автомат. Констр. 2015, 59, 179–187. [Перекрестная ссылка]
43. Лю, Дж.; Цинь, Х.; Сан, Ю.; Чжан, К. Модель прогнозирования реакции конструкций причального контейнерного крана на основе мониторинга Данные. Дж. Перформанс. Констр. Удобство. 2021, 35. [CrossRef]
44. Джо, Дж. Х.; Ким, С. Разработка ключевых показателей эффективности для оценки производительности кранов типа «судно-берег» на контейнерном терминале Операции. Дж. Мар. англ. 2020, 8, 6. [CrossRef]
45. Ченг, К.; Лю, Х.; Чжоу, Б.; Юань, Ю. Интеллектуальная диагностика неисправностей с помощью зашумленных меток посредством полуконтролируемого обучения в промышленном производстве. Временной ряд. IEEE Транс. Инди. Информ. 2023, 19, 7724–7732. [Перекрестная ссылка]
46. Ли, XY; Кумар, А.; Видьяратне, Л.; Рао, Арканзас; Фарахат, А.; Гупта, К. Ансамбль методов обнаружения неисправностей на основе свертки с использованием вибрационных сигналов. В материалах Международной конференции IEEE по прогностике и управлению здравоохранением 2023 г. (ICPHM 2023), Монреаль, Квебек, Канада, 5–7 июня 2023 г.; стр. 172–179. [Перекрестная ссылка]
47. Чжан Дж.; Чжан, М.; Фэн, З.; Жуйфан, Л.В.; Лу, К.; Дай, Ю.; Донг, Л. Глубокая сверточная нейронная сеть с расширенными рекуррентными модулями для диагностики неисправностей промышленных процессов в режиме реального времени. Процесс Саф. Окружающая среда. Прот. 2023, 175, 129–149. [Перекрестная ссылка]
48. Как кодировать нейронные сети RNN и LSTM на Python. Доступно онлайн: https://www.nbshare.io/notebook/249468051/. Как кодировать RNN и LSTM-нейронные сети в Python/ (по состоянию на 28 апреля 2022 г.).
49. Киньонес-Груэйро, М.; Льянес-Сантьяго, О.; Сильва Нето, А.Дж. Диагностика неисправностей в промышленных системах. Стад. Сист. Рецис. Контроль 2021, 309, 1–14. [Перекрестная ссылка]
50. Иллюстрированное руководство по LSTM и GRU: пошаговое объяснение | Майкла Фи | к науке о данных. Доступно онлайн: https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-sa-step-by-step-explanation-44e9eb85bf21 . (по состоянию на 13 апреля 2021 г.).
51. Махешваранатан, Н.; Уильямс, АХ; Голуб, доктор медицинских наук; Гангули, С.; Суссильо, Д. Обратное проектирование рекуррентных сетей для классификации настроений показывает динамику линейного аттрактора. В материалах 33-й конференции по нейронным системам обработки информации (NeurIPS 2019), Ванкувер, Британская Колумбия, Канада, 8–14 декабря 2019 г.; стр. 15696–15705.

Machine Translated by Google

52. Ализаде З.; Язди, Дж.; Ким, Дж. Х.; Аль-Шамири, А.К. Оценка методов машинного обучения для прогнозирования месячного стока.
Вода 2018, 10, 1676. [CrossRef]
53. Ахмад Дж.; Тахир, А.; Лариджани, Х.; Ахмед, Ф.; Азиз Шах, С.; Холл, Эй Джей; Бьюкенен, У. Дж. Прогнозирование спроса на энергию в зданиях с использованием случайных нейронных сетей. Дж. Интелл. Нечеткая система. 2020, 38, 4753–4765. [Перекрестная ссылка]
54. Ли, С.; Ли, В.; Член, С.; Кук, К.; Гао, Ю. Глубокая независимо рекуррентная нейронная сеть (IndRNN). apXiv 2019,
arXiv:1910.06251v3.
55. Абид, А.; Хан, Монтана; Икбал, Дж. Обзор методов обнаружения и диагностики неисправностей: основы и не только. Артиф. Интел. Версия 2021 г.,
54, 3639–3664. [Перекрестная ссылка]
56. Шахин, М.; Чен, ФФ; Хоссейнзаде, А.; Занд, Н. Использование алгоритмов машинного обучения и глубокого обучения для минимизации простоев в производственных системах:
служба диагностики раннего обнаружения отказов. Межд. Дж. Адв. Производитель Технол. 2023, 128, 3857–3883. [Перекрестная ссылка]
57. Лейте, Д.; Мартинс, А.; Ратива, Д.; Де Оливейра, JFL; Масиэль, АМА: Автоматизированный подход к машинному обучению в режиме реального времени
Обнаружение и диагностика неисправностей. Датчики 2022, 22, 6138. [CrossRef]
58. Дарбан, З.З.; Уэбб, Дж.И.; Пан, С.; Аггарвал, СС; Салехи, М. Глубокое обучение для обнаружения аномалий временных рядов: обзор. arXiv
2022, arXiv:2211.05244v2.
59. Сонг, Дж.; Ли, ЮК; Ли, Дж. Глубокая генеративная модель с кодированием изображений временных рядов для обнаружения производственных дефектов в процессе литья под
давлением. Дж. Интелл. Производитель 2023, 34, 3001–3014. [Перекрестная ссылка]
60. Гуй, Х.; Чжан Дж. Модель глубокого метрического обучения для диагностики несбалансированных неисправностей. arXiv 2021, arXiv:2107.03786.
61. Лю, Х.; Ма, Р.; Ли, Д.; Ян, Л.; Ма, З. Диагностика неисправностей оборудования на основе глубокого обучения для анализа временных рядов и графиков знаний. J. Сигнальный
процесс. Сист. 2021, 93, 1433–1455. [Перекрестная ссылка]
62. Панг Дж. Л. Адаптивное прогнозирование неисправностей и обслуживание производственных линий с использованием глубокого обучения. Межд. Дж. Симул. Модель. 2023,
22, 734–745. [Перекрестная ссылка]
63. Браунли, Дж.; Как рассчитать корреляцию между переменными в Python. Мастерство машинного обучения. Доступно онлайн: https: //machinelearningmastery.com/how-to-use-
correlation-to-understand-the-relationship-between-variables/ (по состоянию на 14 апреля 2022 г.).
64. Ли, А. Приоритизация риска человеческих ошибок при работе крана на основе CPT и ICWGT. PLoS ONE 2024, 19, e0297120. [Перекрестная ссылка]
[ПабМед]
65. Эглинас Т.; Юсис, М.; Яковлев С.; Сенулис, А.; Андзиулис, А.; Гудас С. Анализ эффективности морских контейнеров
методы и процедуры контроля погрузочно-разгрузочных работ. Адв. Мех. англ. 2019, 11. [CrossRef]
66. Мабуди, М.; Аламури, А.; Де Арриба Лопес, В.; Баджаури, М.С.; Бергер, К.; Герке, М. Проверка контейнерных кранов с помощью дронов: концепция, проблемы и
предварительные результаты. ИСПРС Анн. Фотограмма. Дистанционное измерение Пространственная информация. наук. 2021, Т-1-2021, 121–128.
[Перекрестная ссылка]
67. Шан, Дж.; Чжан, М.; Чжан, Д.; Чжоу, Х.; Ли, Х.; Ван, Б. Контейнерные краны. В Справочнике портовой техники; Спрингер: Сингапур, 2024 г.; стр. 451–678. [Перекрестная ссылка]
68. Виллехнер А.; Спинлер, С. Новая аналитика данных в сочетании с традиционными контейнерными перевозками: прогнозирование задержек путем сравнения различных алгоритмов
машинного обучения. Доступно онлайн: https://hdl.handle.net/10125/63897 . (по состоянию на 12 июня 2024 г.).
69. Тиан, Х.; Ян, Р.; Лю, Ю.; Ван, С. Умный метод прогнозирования и оптимизации для целевых и экономически эффективных морских перевозок.
Трансп. Рез. Часть Б. Методика. 2023, 172, 32–52. [Перекрестная ссылка]
70. Обработка несбалансированных данных для классификации — GeeksforGeeks. Доступно онлайн: https://www.geeksforgeeks.org/handling-.
несбалансированные данные для классификации/ (по состоянию на 12 июня 2024 г.).

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Эффективная оптимизация поиска по дереву Монте-Карло Алгоритм для амазонок

Лицзюнь Чжан [1,2], Хан Цзоу 1,2 и Юнган Чжу 1,2,*

[1] Колледж компьютерных наук и технологий Цзилиньского университета, Чанчунь 130012, Китай; zhanglj2121@mails.jlu.edu.cn (LZ); zouhan1221@mails.jlu.edu.cn (HZ)
[2] Ключевая лаборатория символьных вычислений и инженерии знаний Министерства образования, Цзилиньский университет, Чанчунь
130012, Китай * Переписка: zhuyungang@jlu.edu.cn

Аннотация: Амазонки — это компьютеризированная настольная игра со сложными позициями, которые представляют собой сложную задачу для человека. В этой статье мы предлагаем эффективную оптимизацию алгоритма поиска по дереву Монте-Карло (MCTS) для Amazon, объединяя стратегию «Перемещение групп» и стратегию оптимизации «Параллельная оценка» (MG-PEO). В частности, мы объясняем высокую эффективность стратегии Move Groups, определяя новый критерий: выигрышное расстояние сходимости. Мы также подчеркиваем потенциальную проблему попадания стратегии в локальный оптимум и предлагаем, чтобы механизм параллельной оценки мог компенсировать этот недостаток. Более того, мы провели тщательный анализ производительности и эксперименты. Результаты анализа производительности показывают, что алгоритм MCTS со стратегией Move Groups может улучшить игровые возможности игры Amazons в 20–30 раз по сравнению с традиционным алгоритмом MCTS. Оптимизация параллельной оценки еще больше повышает игровые возможности игры «Амазонки» в 2–3 раза. Результаты экспериментов показывают, что алгоритм MCTS со стратегией MG-PEO обеспечивает в среднем на 23 % более высокий процент выигрышей в играх по сравнению с традиционным алгоритмом MCTS. Кроме того, программа MG-PEO Amazons, предложенная в этой статье, получила первый приз в конкурсе Amazons на Чемпионате Китая по студенческим компьютерным играм и Национальном турнире по компьютерным играм 2023 года.

Ключевые слова: амазонки; оптимизация алгоритма; Переместить группы; Параллельная оценка

check for updates

Цитирование: Чжан Л.; Цзоу, Х.; Чжу, Ю. Эффективная оптимизация алгоритма поиска по дереву Монте-Карло для амазонок. Алгоритмы 2024, 17, 334. https://doi.org/10.3390/a17080334.

Академические редакторы: Вэньсинь Ли и Хайфэн Чжан

Поступила: 22 апреля 2024 г.
Пересмотрено: 16 июля 2024 г.
Принято: 27 июля 2024 г.
Опубликовано: 1 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

«Амазонки» — настольная игра для двух игроков, которую представил Вальтер Замкаускас из Аргентины в 1988 году [1,2]. Это игра, предназначенная для участия в соревнованиях по программированию компьютерных игр «Олимпия». Из-за сложного игрового процесса «Амазонки» очень сложны для людей [3], что делает их популярным объектом для соревнований и исследований по компьютерным играм. Правила игры следующие: Игра ведется на квадратной доске 10 × 10. Каждый игрок управляет четырьмя одинаковыми фигурами, называемыми «Амазонками». Эти фигуры и стрелки подчиняются тем же правилам, что и ферзь в шахматах: они могут перемещать любое количество клеток в любом направлении — по горизонтали, вертикали или диагонали — но не могут перепрыгивать через другие фигуры или стрелки. Цель состоит в том, чтобы ограничить движение противника, стреляя стрелами, пока ни одна из четырех фигур противника не сможет двигаться. Если на ходу игрока все четыре его фигуры обездвижены, этот игрок проигрывает, а другой игрок выигрывает. Примечательно, что у амазонок связей нет.

Основная проблема игры «Амазонки» заключается в ее сложности: каждый ход предлагает огромное количество возможных вариантов выбора, обычно более 500 [4,5], а игра может продолжаться более 80 ходов [6,7]. Такой высокий уровень сложности делает Amazon отличным экспериментальным образцом для тестирования алгоритмов стратегии поиска [8,9]. С момента своего изобретения он привлек значительное внимание ученых-компьютерщиков [10] и студентов-компьютер

Метод Монте-Карло [12] представляет собой вычислительный метод, который дает численные результаты посредством повторной случайной выборки. При интеграции с алгоритмами поиска по дереву

Machine Translated by Google

и подходящие функции оценки, алгоритм поиска по дереву Монте-Карло (MCTS) дал впечатляющие результаты в настольных играх с высоким коэффициентом ветвления (таких как го) [13–15] и в стратегических играх в реальном времени (таких как StarCraft [13–15]). 16]). Например, в 2016 году программа AlphaGo компании DeepMind [17] победила корейского чемпиона мира по го Ли Седоля со счетом 4:1. В 2018 году программа OpenAI Five компании OpenAI [18] победила команду чемпиона мира по Dota 2 с подавляющим преимуществом; оба достижения основаны на алгоритме поиска по дереву Монте-Карло.

Стратегия оптимизации «Группы перемещения — параллельная оценка» (MG-PEO) учитывает правила настольной игры «Амазонки», разделяя полный ход на две части: «Движение амазонки» и «Выстрел стрелой». И Amazon Movement, и Arrow Shot рассматриваются как узлы дерева, а не рассматривают полное перемещение как один узел дерева.
Этот подход известен в литературе как стратегия Move Groups [19]. Хотя эта стратегия была применена к игре «Амазонки» [20], до сих пор отсутствует объективный анализ ее эффективности. Более того, широко распространено мнение, что этот метод обладает только преимуществами и не имеет недостатков. В этой статье мы объясняем высокую эффективность стратегии перемещения групп в Amazon, определяя новый критерий: выигрышное расстояние сходимости. Мы также выявляем определенные недостатки этой стратегии. Чтобы решить эту проблему, мы предлагаем использовать механизм параллельной оценки, чтобы компенсировать ограничения стратегии перемещения групп. Наши экспериментальные результаты показывают, что MG-PEO является эффективной стратегией оптимизации термоядерного синтеза.

Основные положения данной статьи заключаются в следующем:

- Мы предлагаем эффективную оптимизацию алгоритма поиска по дереву Монте-Карло для игры «Амазонки»: оптимизацию алгоритма MG-PEO, которая сочетает в себе стратегию перемещения групп со стратегией параллельной оценки.
- Мы предоставляем анализ эффективности стратегии перемещения групп в Amazon. Определив новый критерий выигрышной дистанции сходимости, мы объяснили высокую эффективность стратегии. Более того, мы также подчеркиваем, что данная стратегия имеет определенные недостатки. Стратегия перемещения групп предполагает уточнение правил игры «Амазонки» и использование моделирования для присвоения соответствующих значений дополнительным узлам слоя. Используя эти оценки значений (например, стратегию применения верхней доверительной границы к дереву (UCT) [15]), мы можем добиться эффективной обрезки и сэкономить время. Однако добавление узлов дополнительного уровня означает, что в худшем случае, когда отсечения не произошло, необходимо пройти и оценить больше узлов, что потенциально может привести к увеличению затрат времени. Эти накладные расходы возникают, когда алгоритм пытается найти локальное оптимальное решение.
- Мы предлагаем использовать механизм параллельной оценки для устранения потенциальных недостатков стратегии перемещения групп. Распараллеливая функцию оценки для игры «Амазонки», мы стремимся ускорить процесс расширения поиска по дереву Монте-Карло и минимизировать вероятность того, что стратегия перемещения групп попадет в локальный оптимум.

Кроме того, мы провели тщательный анализ производительности и игровые эксперименты. Анализ производительности показывает, что по сравнению с традиционным алгоритмом MCTS алгоритм MCTS со стратегией перемещения групп может повысить игровые возможности Amazon в 20–30 раз, а компонент оптимизации параллельной оценки может улучшить игровые возможности Amazon в 2–30 раз. 3 раза. Результаты экспериментов показывают , что по сравнению с традиционным алгоритмом MCTS алгоритм MCTS со стратегией MG-PEO обеспечивает увеличение процента выигрышей в среднем на 23% и демонстрирует более сильные игровые способности.

Оставшаяся часть статьи структурирована следующим образом: В разделе 2 мы представляем предварительную информацию об игре «Амазонки» и обсуждаем ее отношение к алгоритму MCTS. В разделе 3 мы даем подробное описание алгоритма MCTS со стратегией MG-PEO. В разделе 4 мы предлагаем анализ производительности стратегии перемещения групп и стратегии параллельной оценки соответственно. В разделе 5 описана серия экспериментов.

Machine Translated by Google

проведено для проверки эффективности алгоритма MCTS со стратегией MG-PEO. Наконец, выводы и направления будущих исследований обобщены в разделе 6.

2. Сопутствующая работа

В этом разделе мы представляем обзор соответствующих исследований, которые составляют основу нашей работы. В частности, мы обсудим три ключевых модуля литературы, имеющих отношение к Amazon: историческое исследование алгоритмов Amazon, применение алгоритма MCTS к Amazon и знания об оценке игровых ситуаций.

2.1. Алгоритм игры амазонок

Игра «Амазонки» подверглась тщательному изучению; Разработку алгоритма можно условно разделить на три этапа. Первоначально в этой области доминировали традиционные методы, основанные на математической теории, такие как алгоритм минимаксного поиска [21] . На втором этапе появились алгоритмы Монте-Карло, включая различные варианты поиска по дереву Монте-Карло [19,20,22–24]. На третьем этапе на первый план вышли методы глубокого обучения с использованием игровых баз данных Amazon [25,26]. Примечательно, что наиболее репрезентативная программа Amazons, Invader [27–29], использует MCTS для перспективных исследований, дополненных обучением нейронных сетей на шахматных базах данных, что приводит к потрясающим игровым возможностям в игре Amazons. Можно сказать, что дальнейшая оптимизация алгоритмов Монте-Карло остается решающей для продвижения исследований амазонок.

2.2. Алгоритм поиска по дереву Монте-

Карло. Разработку алгоритма поиска по дереву Монте-Карло в Amazon можно условно разделить на два направления исследований: оптимизация алгоритма с точки зрения поиска по дереву и оптимизация алгоритма с точки зрения функций оценки. Что касается оптимизации алгоритма поиска по дереву, исследователи работали над улучшением алгоритмов поиска для повышения производительности Amazon. Например, Го и др. разделил игру на раннюю, среднюю и позднюю стадии, используя разные стратегии на каждой стадии [30]. Ли и др. применил алгоритм UCT к амазонкам [31]; Цюань и др. продемонстрировали превосходство алгоритма UCT на амазонках [32]. С другой стороны, оптимизация функции оценки также является одним из направлений исследований. Оптимизируя функцию оценки, можно более точно оценить текущую игру, а алгоритм поиска можно использовать для принятия более обоснованных решений. Либерум и др. посвятили свои исследования этому аспекту [33,34], сосредоточившись в основном на улучшении настройки гиперпараметров и разработке новых функций оценки. В целом, алгоритм поиска по дереву Монте-Карло не может быть улучшен без какого-либо одного аспекта, и в этой статье предлагаемое нами объединение стратегии перемещения групп и стратегии параллельной оценки представляет собой всестороннее рассмотрение двух исторических направлений исследования.
Это сочетание алгоритма поиска по дереву Монте-Карло является передовым и эффективным.

2.3. Функция оценки

В настоящее время практически все алгоритмы поиска сталкиваются с проблемой оценки ситуации. Хорошая оценка ситуации очень важна в игре «Амазонки» и может даже определять качество игровой программы [33].

Как и во многих настольных играх, «Амазонки» можно условно разделить на три этапа: начальный, средний и финальный, с разными стратегиями для каждого этапа. Основными показателями функции оценки амазонок являются территория, положение и мобильность, причем важность этих трех характеристик варьируется на разных этапах. Основываясь на этих идеях, Guo et al. [30] предложили функцию поэтапной оценки, как показано в уравнении (1):

$$\text{Значение} = k1 \quad t1 + k2 \quad t2 + k3 \quad p1 + k4 \quad p2 + k5 \quad \text{подвижность} \tag{1}$$

где t1 и t2 — значения метрик территории, которые, соответственно, представляют собой оценку контроля каждой стороны над пространством на основе методов QueenMove и KingMove. p1 и p2 — значения показателей позиции, отражающие стратегическое положение фигур каждой стороны, также основанное на методах QueenMove и KingMove относительно пространства. мобильность

Machine Translated by Google

— это оценочная ценность гибкости обеих частей Amazon. k1, k2, k3, k4 и k5 — соответствующие значения веса. Как рассчитываются эти значения, описано ниже.

Согласно литературе [2] для расчета значений территории и позиции используются два типа методов: QueenMove и KingMove. QueenMove означает, что ход можно сделать до тех пор, пока нет препятствий, которые нужно пересечь в любом из восьми направлений (к препятствиям относятся фигуры и стрелы), а KingMove означает, что расстояние одного хода равно единице, т. е. только одному ход можно сделать на соседнюю клетку. Значение QueenMove представляет собой минимальное количество ходов, необходимое четырем фигурам игрока, чтобы достичь заданного места (т. е. без фигур и стрелок); Значение KingMove представляет собой минимальное количество ходов, которые четыре фигуры игрока должны сделать, чтобы достичь заданного места. Как показано на рисунке 1 слева , значение 2 в верхнем левом углу верхнего левого поля (A, 1) указывает на то, что белой фигуре требуется как минимум два хода, чтобы достичь этого поля с помощью метода QueenMove, а значение Цифра 4 в правом нижнем углу означает, что черной фигуре нужно четыре хода. Аналогично значения 2 и 7 в том же квадрате на рисунке 1 справа обозначают ситуацию, когда используется метод KingMove.

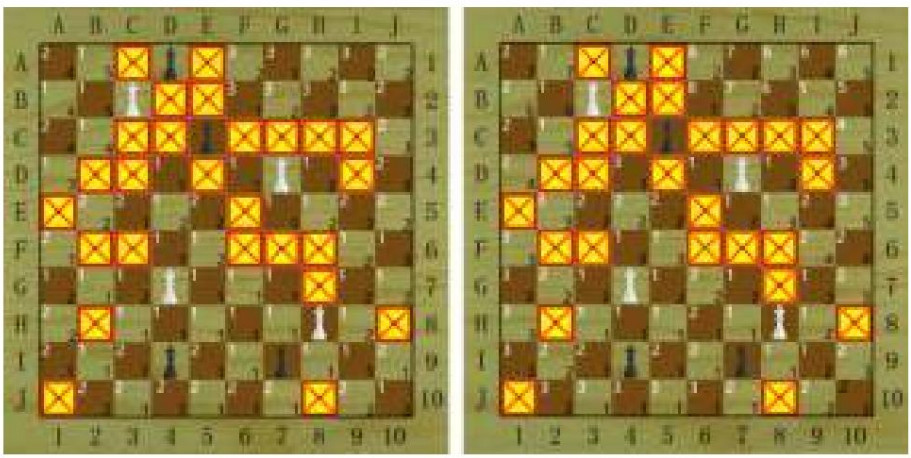


Рис. 1. Результаты расчета методов QueenMove и KingMove. Слева — результат расчета KingMove, а справа — результат расчета QueenMove.

Соответственно, для всех индикаторов мы можем использовать следующую формулу расчета: t1 представляет собой разницу в контроле территории методом QueenMove. t2 представляет разницу в контроле территории методом KingMove. Как показано в уравнениях (2) и (3):

$$ti = \nu(Bi, Wi) \quad (2)$$

$$\nu(Bi, Wi) = \begin{cases} 0, Bi(A) = Wi(A) = \\ k, Bi(A) = Wi(A) \neq \\ 1, Би(A) < Wi(A) \\ 1, Bi(A) > Wi(A) \end{cases} \quad (3)$$

где B представляет черные фигуры, W представляет белые фигуры, с индексом 1, обозначающим метод QueenMove, а 2, обозначающим метод KingMove. A представляет собой пустые квадраты. k представляет перевес ведущего поля, а диапазон значений больше -1 и меньше 1. Когда игрок, делающий первый ход, белый, k больше 0, в противном случае оно меньше 0. p1

представляет собой позиции двух фигур на контрольном поле A, используя метод «Ход ферзя». p2 представляет позиции двух фигур на контрольном пространстве A с использованием метода KingMove. Как показано в уравнениях (4) и (5):

Machine Translated by Google

$$p1 = 2\Sigma \quad 2 \quad B1(A) \quad 2 \quad W1(A) \tag{4}$$

$$p2 = \min_{A}(1, \max(\quad 1, B2(A) \quad W2(A)/6)) \tag{5}$$

подвижность означает гибкость фигур. Как показано в уравнениях (6) и (7):

$$\text{мобильность} = \sum_{w=\text{белый кусок}} F(ш) \quad \sum_{b€\ \text{черная фигура}} F(b) \tag{6}$$

$$F(a) = \begin{cases} \text{окружающие пространства, а — пространство} \\ 0, \quad \text{а - препятствие} \\ s \quad Sa \ \frac{Fspace(a)}{Д(a,c)}, \quad \text{а — это произведение Amazon} \end{cases} \tag{7}$$

где F обозначает гибкость, Sa обозначает набор пространств, до которых можно добраться с помощью фигуры, используя метод QueenMove, а D(a,s) — минимальное количество ходов от a до s с использованием метода Метод KingMove.

Мы сохранили настройки гиперпараметризации согласно литературе [2]; подробный информация о каждом показателе оценки представлена в таблице 1.

Таблица 1. Таблица весовых коэффициентов.

	Территория оценки показателей		Позиция		Мобильность
	Фактор оценки t1	т2	p1	p2	мобильность
	Весовой коэффициент k1	к2	к3	к4	к5
Этапы	Открытие 0,14	0,37	0,13	0,13	0,20
	Середина 0,30	0,25	0,20	0,20	0,05
	Окончание 0,80	0,10	0,05	0,05	0,00

Таблица 1 показывает, что важность t1 постепенно возрастает по ходу игры. начиная с меньшего в начале, увеличиваясь в середине и достигая пика в эндшпиле. Это связано с тем, что по ходу игры контроль над доской становится все более решающее значение для победы или поражения. t2 чрезвычайно важен на начальном этапе, чтобы гарантировать безопасность соседних регионов и территориальных границ, но ее роль снижается по мере игра прогрессирует, и в финале ее можно почти игнорировать. Это отражает актуальность первоначальный план обороны и территориальной конкуренции. p1 и p2 обозначают позиции фигур, которые в основном играют роль в дебютной фазе, когда сбалансированное распределение фигур. решающее значение для последующего стратегического развертывания; в эндшпиле, картина игры уже определено, и распределение фигур уже не является ключевым моментом; мобильность означает гибкость, которая наиболее значима на ранней стадии игры, поскольку блокирование фигур может быть серьезно невыгодным; по ходу игры и первоначальная компоновка установлена, распределение частей становится все более важным. Как игра продолжается и раскладка завершена, влияние подвижности фигур на игру ситуация ослаблена, поэтому ее вес снижается до незначительного в середине и конце игры. Это отражает необходимость стратегической гибкости на ранних этапах свободы передвижения.

3. Методология исследования

В этом разделе мы сначала опишем общую структуру алгоритма MCTS, дополненную стратегией MG-PEO. Затем мы уточним детали оптимизации, сосредоточив внимание по двум аспектам: группы перемещения и параллельная оценка.

Machine Translated by Google

3.1. Общая структура алгоритма Мы

предлагаем стратегию оптимизации алгоритма MG-PEO, основанную на поиске по дереву Монте-Карло [22]. Как показано на рисунке 2, оптимизированный алгоритм поиска по дереву Монте-Карло включает в себя итеративное построение дерева поиска, создавая новое дерево каждый раунд до тех пор, пока не будет достигнуто ограничение по времени или пока игра не закончится. На этом этапе поиск останавливается и возвращает наиболее эффективную корневую операцию (комбинацию действий определенных узлов второго и третьего слоев).

В частности, для каждой итерации поиска выполняются следующие четыре шага:

(1) Выбор: начиная с корневого узла, применяется определенная стратегия для рекурсивного спуска по дереву для выбора расширяемого узла. Узел является расширяемым, если он представляет нетерминальное состояние и имеет непосещенных дочерних элементов.

(2) Расширение: Расширьте дерево, добавив один (или несколько) дочерних узлов, в зависимости от доступных операций.

(3) Моделирование. Запустите моделирование с нового узла, чтобы получить результаты в соответствии со стратегией по умолчанию.

(4) Обратное распространение ошибки: резервное копирование результатов моделирования через выбранные узлы для обновления. их статистика.

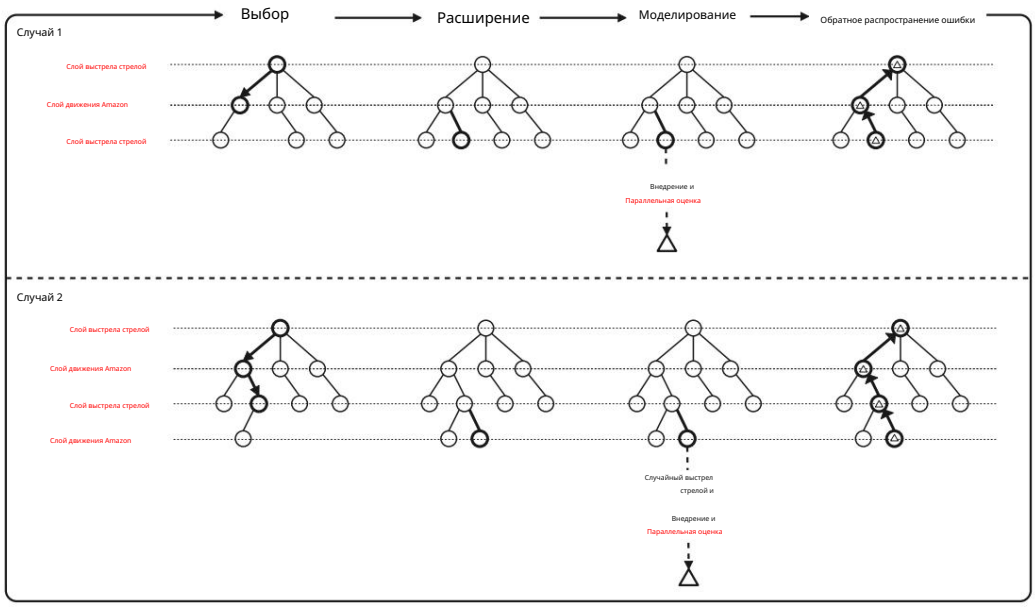


Рисунок 2. Общая структура MCTS плюс MG-PEO.

Как видно на рисунке 2, благодаря использованию стратегии перемещения групп алгоритм работает по двум сценариям на этапе моделирования: если текущий узел является узлом слоя Arrow Shot, он выполняет развертывание и параллельную оценку; если текущий узел является узлом слоя Amazon Movement, он выполняет случайный выстрел стрелкой, развертывание и параллельную оценку.

Псевдокод всей структуры описан в алгоритме 1.

Алгоритм 1: Общая структура алгоритма

Ввод: s0 (текущее состояние игры)
Выход: Оптимальное решение 1:
функция MCTSSEARCH (s0) 2:3:4:
создать корневой узел v0 с состоянием
s0 , в то время как в рамках вычислительного бюджета
выполните v1 Выбор
(v0) v2 Расширение (v1, s0)
5:6: Моделирование (v2, s0)

Machine Translated by Google

Резервная копия
7:8: (v2,) return BESTCHILD(v0)

3.2. Стратегия перемещения групп в алгоритме

Основная идея стратегии Move Groups заключается в следующем: согласно правилам, узел исходного дерева поиска (комбинированное движение Amazon Movement и Arrow Shot) разбивается на два узла (представляющие Amazon Movement и Arrow Shot соответственно). Используя фазу моделирования MCTS для присвоения оценок стоимости дополнительному слою (вновь добавленному уровню), мы можем достичь целей сокращения и экономии времени на основе оценок стоимости дополнительного уровня (например, используя стратегию UCT). .

В частности, когда узел дополнительного слоя еще не расширил свои дочерние узлы, его оценка значения изначально бесконечна. Таким образом, каждый дополнительный узел слоя сначала один раз расширяет свои дочерние узлы и применяет стратегию мелкого развертывания для получения начальной оценки значения. Впоследствии мы можем выбрать узел дополнительного уровня с наивысшим значением для дальнейшего расширения и развертывания до тех пор, пока дочерние узлы конкретного узла дополнительного уровня не будут полностью сгенерированы и узел не будет выбран снова. На этом этапе мы можем получить оптимальное решение из дочерних узлов, что является основой раннего сокращения. Следует отметить, что благодаря динамическому механизму обратного распространения ошибки MCTS оценки значений узлов дополнительного уровня будут динамически изменяться по мере расширения их дочерних узлов. Следовательно, раннее сокращение также может происходить на втором, третьем или других узлах дополнительного уровня. Подробную информацию о расчете и обновлении оценок см. в разделе 3.3. Поскольку полный шаг представляет собой комбинацию двух шагов, а именно «Движение Амазонки» и «Выстрел стрелой», а дополнительный уровень представляет только один шаг, общее количество узлов в дополнительном слое должно быть меньше общего количества узлов, что повышает эффективность выполнение раннего отсечения в алг

Однако стратегия перемещения групп также увеличивает общее количество узлов дерева (для полной одношаговой операции) из-за введения дополнительного слоя, и если все узлы расширяются без предварительного сокращения, дополнительные накладные расходы (расширение и оценка), понесенные узлами дополнительного слоя, негативно повлияет на алгоритм, что также приведет к тому, что — в худшем случае — добавление стратегии Move Groups к MCTS будет работать несколько хуже, чем алгоритм MCTS без политика перемещения групп. Полный теоретический анализ политики перемещения групп подробно описан в разделе 4.1.

3.3. Стратегия параллельной оценки в алгоритме

Основная идея стратегии параллельной оценки заключается в том, что трудоемкая часть оценки моделирования MCTS обрабатывается в рамках параллельной оптимизации. В литературе было показано [35] , что ускорение этой трудоемкой функции оценки позволяет чаще повторять алгоритм поиска по дереву Монте-Карло. Эта метрика очень полезна для компенсации того факта, что стратегия перемещения групп может попасть в локальный оптимум (худший случай); в частности, ускорение функции оценки позволяет максимально быстро масштабировать поиск по дереву Монте-Карло, сводя к минимуму влияние наихудшего сценария, связанного со стратегией перемещения групп. Псевдокод для части параллельной оценки показан в алгоритме 2.

Алгоритм 2: Оценка ситуации

Ввод: s0 (текущее состояние игры)
Выход: Значение оценки ситуации 1: функция
SituationEvaluation(s0) 2: 3:
t1, t2, p1, p2, мобильность ParallelEvaluationMetrics (s0)
Значение = k1 t1 + k2 t2 + k3 p1 + k4 p2 + k5 возврат
4: мобильности Значение

Узлы подвергаются оценке, и полученные в результате оценки значений определяют выбор узла. Традиционные алгоритмы поиска в дереве Монте-Карло обычно выбирают дочерний узел с самым высоким значением.

Machine Translated by Google

Оценка узла (коэффициент выигрыша) на основе текущего состояния игры [36]. Однако этот метод выбора не идеален, поскольку он имеет тенденцию отдавать приоритет сиюминутным выгодам и может привести к локальному оптимуму. Для решения этой проблемы в данной статье используются «стратегия мелкого развертывания» [37] и алгоритм «деревовидного верхнего доверительного интервала» [19] для уточнения выбора узлов. В частности, значение использования, полученное исключительно из узлов уровня принятия решений, может быть неточным. Для повышения точности используется стратегия мелкого развертывания, моделирующая сценарии, близкие к финалу. Кроме того, алгоритм учитывает величину исследовательской ценности узла , которая повышается для узлов, демонстрирующих высокую неопределенность и многочисленные будущие возможности . Таким образом, этот подход оценивает как частично равновесную стоимость использования, так и ценность разведки, чтобы облегчить принятие более обоснованных решений.

1. «Стратегия мелкого развертывания». Это означает, что значение полезности не рассчитывается не только на основе следующего хода, но и на основе положения в игре после случайного моделирования хода несколько раз. В этом алгоритме мы устанавливаем количество развертываний равным двум, а глубину — 4 и 5. В частности, после расширения узла полного решения (движение Amazon плюс выстрел стрелой) мы выполняем два развертывания для текущей ситуации. В каждом внедрении мы моделируем два шага: следующее решение противника (движение Amazon плюс выстрел стрелой) и наше последующее решение (движение Amazon плюс выстрел стрелой), чтобы получить более дальновидную ценность полезности. Подробности см . в Алгоритме псевдокода 3 . Анализ соответствующих параметров см. в Приложении A.

2. «Древовидный верхний доверительный интервал». Как показано в уравнении (8), формула расчета UCT состоит из двух компонентов. Первый член Xi представляет ценность полезности информации в построенном дереве поиска, а второй член представляет ценность исследования непосещенных узлов. Константа C используется для регулировки доли компромисса алгоритма между двумя значениями. Обратите внимание, что значение корневой формулы бесконечно из-за наличия недоступных дочерних узлов, что гарантирует, что недоступные дочерние узлы текущего узла будут посещены хотя бы один раз.

$$UCT = Си + C \frac{\sqrt{Ин\ Н}}{нет} \quad (8)$$

где Xi — коэффициент выигрыша дочернего узла. N — количество обращений к родительскому узлу. ni — количество обращений к дочернему узлу. C — константа (2 , и Сепешвари [36] показал, что это значение удовлетворяет неравенству Хеффдинга с вознаграждениями в диапазоне $\frac{1}{[0,}$). Кочиш 1].

Здесь мы можем предоставить полный псевдокод, как показано в алгоритме 3.

Алгоритм 3: MCTS со стратегией MG-PEO 1: выбор функции (v)

2: пока v не является терминалом, do

3:4: если v не полностью расширен, то вернуть v

5:6: иначе v BESTCHILD(v)

7: вернуть v

8: расширение функции(v,&s0) 9: если v является узлом движения Amazon, то

10:11: ch случайно создать узел «Выстрел стрелы» иначе, если v — узел «Выстрел стрелы» ch

12:13: случайным образом создать узел движения Амазонки s0 Обновить состояние доски return ch 14: 15: функция

Simulation(v,s0)

Machine Translated by Google


```
        S   s0
16:17:  //* глубина = 4 и 5 (дважды независимый порядок)
18:     S   Развертывание(S,глубина)
19:   Оценка ситуации(S)
20: возврат
21: функция Rollout(S,глубина)
22: повторить
23: если это Терминал (S) , то
24: пройти
            если v — узел движения Amazon, то
25:26:          стрелка   СлучайнаяСтрелка(S)
                S   doAction(v,стрелка)
27:28:          глубина   глубина - 1
            иначе, если v — узел «Выстрел стрелы», тогда
29:30:          переместить   родительский элемент v
                S   doAction(move,v)
31:32:          глубина   глубина - 1
33:     до тех пор, пока глубина не станет равна 0 или игра не закончится
34:     вернуть C
35: функция Резервное копирование(v,  )
36:     H(v)   N(v) + 1
37:     Q(v)   Q(v) +   (v, родительский элемент v)
38: v   родитель v
39: функция BESTCHILD(v)
40:     вернуть argmaxv ′ε дочерние элементы v Q(v ') + c  2 ln H(в)
                                                   H(в ')
```

4. Анализ производительности MG-PEO

В этом разделе мы разрабатываем два модуля для анализа стратегии MG-PEO. Во-первых, мы проанализировать эффективность стратегии перемещения групп в Amazon. Во-вторых, мы проводим углубленное обсуждение вопросов повышения эффективности механизма параллельной оценки у амазонок.

4.1. Производительность стратегии перемещения групп в Amazon

Эффективность алгоритма поиска по дереву Монте-Карло, основанного на законе больших числа во многом зависят от количества ситуаций, в которых он может искать. Чем больше ситуаций алгоритм может искать, тем легче будет найти «трюки», которые обычно трудно найти найти и, таким образом, победить противника. Другими словами, чем больше вычислительная мощность, тем эффективнее алгоритм Монте-Карло. Поэтому, чтобы объективно показать влияние стратегии Move Groups на реальную производительность алгоритма MCTS, мы предлагаем новое определение (выигрышное расстояние сходимости) для оценки производительности алгоритма.

Определение 1. Выигрышная дистанция сходимости. Это вычисляет количество расширенных узлов дерева. по дереву поиска, как в глубину, так и в ширину, от текущего состояния (корневого узла) до выигрышного состояния (игра закончена). В общем, чем меньше новых узлов нужно расширить дереву Монте-Карло для достижения победы. как можно скорее, тем короче выигрышная дистанция сходимости.

Чтобы визуализировать выигрышную дистанцию сходимости, мы используем первый первый шаг в качестве пример. Поскольку количество узлов, которые можно сгенерировать с первого шага до выигрышного состояние неизмеримо, мы используем локальное оптимальное решение первого шага вместо выигрышное состояние, как показано на рисунке 3.

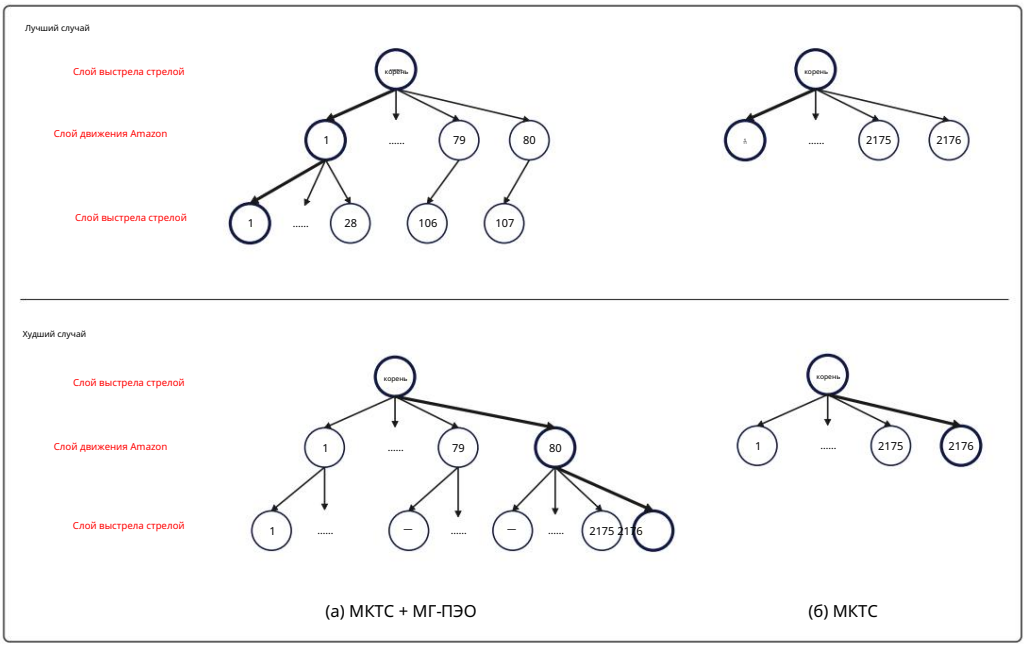
На основе этого мы можем построить дерево поиска Монте-Карло с добавленной стратегией перемещения групп и дерево поиска Монте-Карло без добавленной стратегии перемещения групп:

Machine Translated by Google

и вычислим номера расширенных узлов в лучшем и худшем случае, как показано на рисунке 4. и таблица 2.



Рисунок 3. Локальное оптимальное решение, принятое на первом этапе открытия. Слева направо: начальный штат, после «Движения амазонок» и после «Выстрела стрелы».



(а) МКТС + МГ-ПЭО (б) МКТС

Рисунок 4. График расчета выигрышного расстояния сходимости. Число на рисунке относится к порядок, в котором узел расширяется на определенном уровне.

Таблица 2. Выигрышная дистанция сходимости двух алгоритмов (единица измерения: узлы).

Алгоритм	Лучший случай	Худший случай
МЦТС + МГ-ПЭО	187	2256
МЦТС	2176	2176

Красные результаты обозначают лучший результат в том же столбце.

На рисунке 4 и в таблице 2 на первом этапе показан алгоритм поиска по дереву Монте-Карло с добавленная стратегия перемещения групп требует сначала расширения 80 узлов уровня перемещения Amazon. Поскольку оценка значения узла слоя Amazon Movement бесконечна, если он еще не расширил свои дочерние узлы, каждый узел уровня Amazon Movement необходимо дополнительно расширить узел слоя Arrow Shot для получения оценки начального значения. В дальнейшем мы можем выбрать узел слоя Amazon Movement с наивысшим значением для следующего дочернего узла расширение. В лучшем случае первый узел слоя Amazon Movement имеет наивысшее значение. счет. Во время расширения последующих узлов слоя Arrow Shot оценка значения родительский узел остается на максимальном уровне, наблюдаемом на всем уровне движения Amazon. узлы; другие узлы на уровне движения Amazon не выполняют расширение

Machine Translated by Google

узлы следующего слоя, что является отражением ранней обрезки. Наконец, количество узлов расширенное равно 187. В худшем случае все возможные узлы полностью развернуты и пройдены, в результате чего было расширено в общей сложности 2256 узлов.

Алгоритм поиска по дереву Монте-Карло без включения групп перемещения стратегия должна выбрать узел с наибольшим значением в качестве окончательного решения после генерация всех возможных узлов, независимо от лучшего или худшего сценария; в результате чего всего расширяется 2176 узлов.

Подводя итог, можно сделать следующий вывод: в лучшем случае выигрышная сходимость расстояние MCTS со стратегией Move Groups значительно короче, чем у традиционный MCTS, а в худшем случае выигрышная дистанция сходимости MCTS со стратегией Move Groups немного слабее, чем у традиционного MCTS. Это указывает на то, что стратегия перемещения групп обладает сильными возможностями сокращения и экономии времени. но в то же время есть и некоторые недостатки.

4.2. Эффективность стратегии параллельной оценки

В этом подразделе мы углубимся в количество потоков, необходимых для механизма параллельной оценки, а также коэффициент ускорения, который может обеспечить многопоточность. С тех пор потребление функции оценки в основном зависит от времени расчета каждого метрики оценки, часть параллельной оценки по существу представляет собой многопоточный расчет метрики оценки. Показан псевдокод для метрической функции параллельной оценки. в алгоритме 4.

Алгоритм 4: Метрики параллельной оценки

```
Ввод: s0 (текущее состояние игры)
Выход: t1,t2,p1,p2,мобильность
1: функция ParallelEvaluationMetrics(s0)
2:          //* При расчете метрик Территория(t1, t2) и Позиция(p1, p2) используется одно и то же
                независимая функция четыре раза, на основе King Move и QueenMove
                методы, которые рассчитываются один раз для черного и один раз для белого.
3: //* Параллельно оптимизируем его внешние вызовы.
4: Модуль1 = лямбда (диапазон)
5: для индекса в диапазоне:
                цвет     индекс/2
6:7:            тип      индекс%2
                CalTP(цвет,тип)
8:9:       параллельный_для(диапазон_блоков(0,2),Модуль1)

10:         //* Метрика Mobility(мобильность) рассчитывается по одной функции за раз, а
                внутренняя логика заключается в независимой обработке каждой позиции платы (GridSize = 100).
11:         //* Параллельно мы оптимизируем логику обработки внутреннего цикла.
12: Модуль2 = лямбда (диапазон)
13:             для индекса в диапазоне:
14: цвет     индекс/2
                  тип      индекс%2
                  CalEveryBoardPosition()
15:16:17: параллельный_для(block_range(0,GridSize),Module2)
```

A. Обсуждение количества потоков: Поскольку основные метрики оценки функция включает показатели территории, положения и маневренности, значения территории а показатели позиции можно рассчитать с помощью одной функции; однако, как показывают расчеты из этих двух показателей связаны с цветами фигур и методом расчета. (метод KingMove и метод QueenMove), необходимы четыре независимых расчета. для значений метрик территории и позиции. Показатель мобильности рассчитывается функцией, требующей одного независимого вычисления. Алгоритм оптимизирован в

Machine Translated by Google

параллельно для индикаторов территории и положения, которые теоретически требуют всего четыре нити. Для показателя маневренности теоретически необходимо десять ниток для внутренних расчеты, связанные с размером доски.

Б. Исследование коэффициента ускорения, которое может обеспечить многопоточность. точно исследуя коэффициент ускорения, мы подсчитали время, затраченное на расчет функция каждого показателя оценки, а данные показаны в таблице 3.

Таблица 3. Время выполнения модуля и коэффициент ускорения.

Модуль	Время последовательного выполнения	Время параллельного выполнения	Коэффициент ускорения
Модуль1	69 нас	29 нас	2.379
Мудуле2	45 нас	23 нас	1,956
Вся функция оценки 114 нас		54 нас	2.111

Как показано в таблице 3, время расчета для оптимизированной территории и позиции метрики достигли ускорения в 2,379 раза, а время расчета оптимизированной мобильности метрика достигла ускорения в 1,956 раза. Общий коэффициент ускорения для оценочной части было 2.111. В разделе A мы обсудили количество потоков и отметили, что теоретический ускорение распараллеливания должно соответствовать количеству потоков. Однако время потребление каждого модуля до и после оптимизации несколько отличалось от обсуждение в разделе A. Мы считаем, что накладные расходы сторонних параллельных библиотек[38] потребовало некоторой производительности вычислительных ресурсов. См. Приложение B для получения подробной информации о Анализ использования процессора. Тем не менее механизм параллельной оптимизации сократил время требуется обрабатывать одни и те же задачи в 2111 раз, что приводит к значительному увеличению производительности улучшение практического применения.

5. Эксперимент и результаты.

В этом разделе, чтобы проверить эффективность стратегии оптимизации алгоритма MG-PEO, предложенной в этой статье, были разработаны три эксперимента: первый сравнивает соответствующие показатели производительности алгоритма MCTS со стратегией MG-PEO и традиционный алгоритм MCTS, и по результатам сравнения проводится проверка значимости. Второй эксперимент направлен на оценку величины влияния каждого модуля оптимизации MG-PEO на MCTS, т.е. эксперимент по абляции. Третий эксперимент представляет собой обобщенный экспериментальный анализ МГ-ПЭО.

Кроме того, стоит отметить, что программа MG-PEO Amazons предлагала в этой статье выиграл первый приз в конкурсе амазонок на Китайском университетском турнире 2023 года. Чемпионат по компьютерным играм и Национальный турнир по компьютерным играм.

5.1. Информация об экспериментальной среде

Для обеспечения достоверности результатов экспериментов мы провели все эксперименты в единой среде, а также подробную информацию об экспериментальном настройки среды показаны в Таблице 4.

Таблица 4. Информация об экспериментальной среде.

Аппаратное обеспечение и ограничения	Подробности
графический процессор	GeForce PTX 3070
Процессор	Intel Core i7-11800H
Время принятия решения	5 с
Количество матчей, сыгранных одним и тем же соперником	2
Количество сетов за матч	3
Экспериментальная согласующая платформа	ИГРОВАЯ Платформа САУ
Режим выполнения Visual Studio	[Релиз x64]

Против одного и того же соперника будет сыграно в общей сложности шесть сетов: три сета в роли первого игрока и три в роли ведущего. второй игрок.

Machine Translated by Google

5.2. Количественные и качественные

эксперименты 5.2.1. Статистика и анализ процента выигрышей

Чтобы объективно и точно оценить эффективность стратегии оптимизации алгоритма MG-PEO, мы выбрали для сравнения алгоритм MCTS без оптимизации Move-Groups и без оптимизации Parallel Evaluation и сыграли в общей сложности 100 матчей в тех же условиях, с каждый матч разделен на 3 игры (первая и вторая руки). В частности, мы выбрали 33 игрока, которые участвовали в соревновании по компьютерным играм, и 17 игроков, которые опубликовали свои рейтинги (лестничный список) в открытом доступе на платформе Botzone (https://www.botzone.org.cn/, по состоянию на 2 марта 2024 г.), всего 50 оппо

Из-за необходимости наличия сторонних библиотек мы локально экспортировали exe-программы для всех игроков, а затем проводили игры друг против друга на платформе SAU (http://computergames.caai.cn/platform/SAU_Game_Platform_2.1.0_r3.rar). , по состоянию на 20 марта 2024 г.), а результаты игр показаны в Таблице 5. Подробную информацию обо всех играх можно найти в Приложении C.

Таблица 5. Коэффициент выигрыша двух алгоритмов.

Алгоритм	Процент выигрыша
МЦТС	74%
МЦТС + МГ-ПЭО	97%

Красные результаты обозначают лучший результат в том же столбце.

В Таблице 5 алгоритм MCTS имеет процент выигрышей 74% в Amazons Duel на платформе подбора игроков SAU. В той же тестовой среде алгоритм MCTS со стратегией MG-PEO показывает более высокую стабильность, а его процент выигрышей в матче Amazons на платформе SAU Matchmaking достигает 97%. Таким образом, алгоритм MCTS со стратегией MG-PEO имеет более полную и глубокую оптимизацию при рассмотрении и использовании стратегий, чем алгоритм MCTS, и, таким образом, показывает более высокий процент выигрыша в этом тестовом сценарии.

Стоит отметить, что, наблюдая за совпадением двух алгоритмов, мы обнаружили, что оптимизированный алгоритм MCTS часто принимает более точные решения, чем неоптимизированный алгоритм MCTS, в некоторых критических раундах. Эти ключевые раунды часто включают в себя решающие позиции для стратегий «окружения территории» или «блокирования противника». Кроме того, во время матчей мы заметили, что оптимизированный алгоритм MCTS имеет явное преимущество перед некоторыми традиционными алгоритмами (такими как Minimax и традиционный MCTS). Однако его производительности немного недостает при конкуренции с программами нейронных сетей, обученными на большом объеме игровых данных. Тем не менее, это дает направление для дальнейшего улучшения производительности алгоритма на следующем этапе.

5.2.2. Визуализация и анализ расстояния конвергенции

В этом подразделе мы подсчитываем количество раундов матч-пойнтов, в которых оценка процента выигрыша алгоритма достигает 90% или более (порог уверенного выигрыша) в 100 играх, сыгранных двумя алгоритмами на платформе SAU, как способ визуализировать определение дистанции схождения выигрыша, а также сравнить сильные и слабые стороны игровых способностей двух алгоритмов игры Амазонок. Здесь мы обычно считаем исход игры очевидным, когда алгоритм оценивает вероятность выигрыша в 90% или более (за исключением ошибок низкого уровня со стороны доминирующей стороны); такие моменты считаются матч-пойнтами и мы считаем, что чем раньше это происходит, тем сильнее алгоритмы. Средние результирующие данные по количеству раундов с совпадением для двух алгоритмов показаны в Таблице 6. Полные данные см. в Приложении C :

Machine Translated by Google

Таблица 6. Среднее количество раундов матч-пойнтов в 100 матчах, в которых значения оценки винрейта двух алгоритмов превышают 90%.

Алгоритм	Среднее значение
МЦТС	42,57
МЦТС + МГ-ПЭО	39,68

Если игрок проигрывает, раунды с матч-пойнтом принимаются за максимальное количество раундов, т. е. 47. Кроме того, красные результаты указывают на лучший результат в том же столбце.

Как видно из таблицы 6, среднее количество раундов матчпоинта, необходимое алгоритму MCTS для достижения порога стабильного выигрыша (значение оценки более 90%), составляет 42,57 раундов. С другой стороны, алгоритм MCTS со стратегией MG-PEO превосходит MCTS с точки зрения достижения того же стабильного порога выигрыша со средним значением 39,68 раундов.

Это указывает на то, что алгоритм MCTS со стратегией MG-PEO сходится быстрее, чем MCTS. Он требует меньшего количества средних раундов, чем традиционный алгоритм MCTS. Алгоритм MCTS со стратегией MG-PEO использует стратегию перемещения групп и стратегию параллельной оценки для оптимизации процесса выбора стратегии алгоритма MCTS, чтобы он мог находить лучшие пути поиска за более короткий период времени и достигать стабильного выигрышного состояния . с вероятностью выигрыша более 90% заранее.

Таким образом, алгоритм MCTS со стратегией MG-PEO имеет более короткую дистанцию сходимости выигрыша, чем традиционный алгоритм MCTS. Для достижения более высокого коэффициента выигрыша требуется меньше раундов и, таким образом, демонстрируется превосходная игровая способность.

5.2.3. Проверка значимости

В этом подразделе мы проводим проверку значимости данных подсчета раундов по очкам совпадения, как указано в разделе 5.2.2, чтобы определить, существует ли между ними значительная разница. Мы проводим тесты на уровне значимости 0,05. Полные данные подсчета раундов по точкам совпадения подробно описаны в Приложении C. Сначала мы используем тест Шапиро-Уилка, чтобы предположить, что оба набора данных подчиняются нормальному распределению. Однако ни один набор данных не соответствует нормальному распределению. Основываясь на этом результате, мы используем U-критерий Манна-Уитни, чтобы предположить, что между двумя группами данных нет существенной разницы. Затем мы сравниваем характеристики распределения данных раунда точек совпадения двух алгоритмов. Подробные результаты показаны в Таблице 7.

Таблица 7. Значимость результатов испытаний.

Данные раундов Match Point	Значение p для теста Шапиро-Уилка (P1)	p-значение для Манна – Уитни U Тест (P2)
МЦТС	0,0002	$1,16 \times 10^{8}$
МЦТС + МГ-ПЭО	0,0013	

В результате теста Шапиро-Уилка оба протестированных алгоритма показали значение P1 менее 0,05 для данных о количестве раундов соответствия, что заставило нас отвергнуть нулевую гипотезу о том, что данные для MCTS+MG-PEO и данные для MCTS соответствуют нормальному распределению. Результаты U-теста Манна-Уитни показывают, что значение P2 для данных о количестве раундов точки совпадения между двумя алгоритмами значительно меньше 0,05, поэтому мы отвергаем исходную гипотезу. То есть существует значительная разница в среднем количестве выигрышных шагов между двумя алгоритмами.

В частности, среднее количество выигрышных ходов, требуемое MCTS со стратегией MG-PEO, значительно меньше, чем необходимое для традиционной MCTS против 100 игроков, что позволяет предположить, что MCTS со стратегией MG-PEO превосходит традиционную MCTS с точки зрения игровых возможностей на амазонках.

Machine Translated by Google

5.3. Эксперимент по

удалению. В этом подразделе, чтобы оценить эффективность стратегии перемещения групп и стратегии параллельной оценки в Amazon, мы разрабатываем два эксперимента: один, в котором используется только стратегия перемещения групп без использования стратегии параллельной оценки, и другой, в котором используется только стратегия параллельной оценки. Стратегия оценки без использования стратегии перемещения групп . Результаты эксперимента показаны в Таблице 8. С одной стороны, стратегия Move Groups фокусируется на итерациях MCTS в течение ограниченного времени на подузлах более ценных узлов дополнительного уровня, используя их значения, избегая, таким образом, расширения неважных узлов. узлы. Хотя стратегия перемещения групп демонстрирует сильные возможности сокращения, оценки значений узлов дополнительного уровня динамически изменяются с расширением их подузлов благодаря механизму динамического обновления обратного распространения ошибки MCTS. Следовательно, все узлы на дополнительном уровне могут быть полностью расширены и пройдены несколько раз (обрезки не происходит), что потенциально приводит к увеличению затрат времени по сравнению с алгоритмами MCTS без стратегии перемещения групп. С другой стороны, поскольку доля оценки ситуации ограничена, использование только оптимизации параллельной оценки без принятия стратегии перемещения групп дает ограниченное улучшение производительности. Таким образом, комбинированная оптимизация с использованием стратегии перемещения групп и параллельной оценки эффективна и необходима.

Таблица 8. Анализ эксперимента по абляции стратегии Move Groups и стратегии параллельной оценки.

Алгоритм	Процент выигрыша	Количество итераций MCTS в течение 1 с.
МЦТС	74%	2176
МЦТС + ПЭО	89%	10 026
МЦТС + МГ	83%	5663
МЦТС + МГ-ПЭО	97%	45 536

«Количество итераций MCTS в пределах 1 с» относится к количеству проведенных итераций поиска по дереву Монте-Карло.
в течение одной секунды. В соответствии со стратегией Move Groups каждая итерация MCTS приводит либо к узлу Amazon Movement, либо к узлу Arrow Shot. Без использования стратегии Move Groups каждая итерация MCTS приводит к созданию полного узла принятия решений (Amazon Movement плюс Arrow Shot). Кроме того, красные результаты обозначают лучший результат в том же столбце.

5.4. Обобщающий анализ В этом

подразделе мы провели общий анализ стратегии «Группы перемещения» и стратегии «Параллельная оценка», подробно обсудив их переносимость в другие настольные игры.

1. Стратегия групп ходов. В некоторых играх фактор ветвления может быть очень большим, но многие ходы одинаковы. Стратегия Move Groups создает дополнительный уровень принятия решений, на котором все возможные действия группируются вместе, что позволяет нам более эффективно справляться с большим фактором ветвления игры. Эта стратегия полезна для других игр, таких как Го [39] и игра «Многорукий бандит» с группами движений [40]. Мы считаем, что эту стратегию все еще стоит изучить и изучить.

2. Стратегия параллельной оценки. Стратегия параллельной оценки может обеспечить эффективное ускорение как для простых игр, основанных на правилах, таких как гомоку, так и для более сложных игр, таких как шахматы и го, или даже для современных инновационных настольных игр. Его основное преимущество заключается в способности эффективно исследовать дерево решений, что является общим требованием в различных настольных играх.

6. Выводы и будущие работы.

В этой статье мы предлагаем эффективную оптимизацию алгоритма поиска по дереву Монте-Карло для игры «Амазонки», которая совершенствует возможности поиска по дереву Монте-Карло с помощью стратегии перемещения групп. Механизм параллельной оценки устраняет риск попадания стратегии перемещения групп в локальный оптимум (эти компоненты

Machine Translated by Google

дополняющие друг друга). Результаты эксперимента подтверждают, что предложенный метод в этой статье значительно улучшает игровые возможности игры «Амазонки».

Будущие исследования также включены в наши планы. Во-первых, чтобы максимизировать экономию времени потенциал сокращения стратегии Move Groups, мы надеемся рассмотреть возможность использования более интеллектуальных подходы (например, эвристические функции) для управления поведением сокращения групп перемещения. стратегия. Во-вторых, серия экспериментов в этой статье демонстрирует мощную возможность повышения производительности за счет параллельной оптимизации, поэтому мы планируем рассмотреть больше параллельные возможности и более эффективные параллельные стратегии в будущем. Наконец, мы также надеюсь, что этот алгоритм можно будет расширить для решения большего количества игровых проблем.

Вклад авторов: Все авторы внесли свой вклад в концепцию и дизайн исследования. Методология, программное обеспечение, проверка, исследование, анализ данных, проверка модели, ресурсы, обработка данных, визуализация , написание-редактирование и обзор: LZ и HZ Надзор, администрирование проекта, финансирование приобретение: YZ Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Данное исследование не получило внешнего финансирования.

Заявление о доступности данных. Часть кодекса конкуренции, используемого в этом исследовании, является общедоступной. открытый исходный код, который можно найти на следующем веб-сайте: https://www.botzone.org.cn/, доступ 2 марта 2024 г.

Конфликт интересов: Авторы заявляют, что у них нет финансовых интересов или личных отношений. имеет отношение к содержанию содержания этой статьи.

Приложение А

В Приложении А мы обсуждаем и анализируем влияние глубины развертывания на этапе моделирования на способность алгоритма играть в Amazon и использовать выигрышную сходимость. показатель расстояния (подробнее см. в разделе 4.2) для оценки игровых способностей. Мелкое развертывание стратегии могут учитывать смоделированные позиции ближе к эндшпилю и более дальновидные игры для расчета более точных значений оценки позиции, которые можно использовать для лучше руководить выбором узла для MCTS. Поэтому крайне важно сбалансировать глубину Посадочная дистанция; слишком малая глубина приведет к увеличению значения использования (коэффициента выигрыша) в позиции формула оценки менее достоверна, а слишком глубокая глубина приведет к единственному моделированию итерационный процесс слишком длинный, а количество узлов увеличивается за конечный период времени станет меньше, что снизит точность решения поиска по дереву Монте-Карло. Мы скорректировали значение глубины развертывания с 2 до 10 и обнаружили, что оптимальное значение Глубина развертывания составляет 4 или 5, как показано в Таблице A1.

Таблица A1. Параметрический анализ глубины выкатки.

Глубина развертывания	1	2	3	4	5	6	7–10	4 и 5
Матч-пойнт, раунды 40.25		40.14	40,0	39,9	39,84	40.04	40,1	39,68

Красные результаты обозначают лучший результат в том же столбце.

Кроме того, в наших экспериментах мы обнаружили, что роллаут дважды (выбирая 4 и 5 углубленно) работает лучше, чем однократно роллаут (выбирая только 4 или 5), т.е. количество раундов в матч-пойнтах меньше, дистанция сходимости для победы короче, а Алгоритм более способен играть в амазонок.

Приложение Б

Приложение B показывает подробное использование ЦП алгоритмом с и без Стратегия параллельной оценки. Среда мониторинга — это профилировщик производительности в Microsoft Visual Studio 2019, алгоритм принимает только одно решение, которое принимает 20 с. Как показано на рисунке A1, верхняя часть изображения представляет загрузку ЦП. а колеблющаяся часть значений загрузки ЦП соответствует мощности алгоритма. этап принятия решения, а оставшаяся часть соответствует входу и выходу фазы холостого хода.

Machine Translated by Google

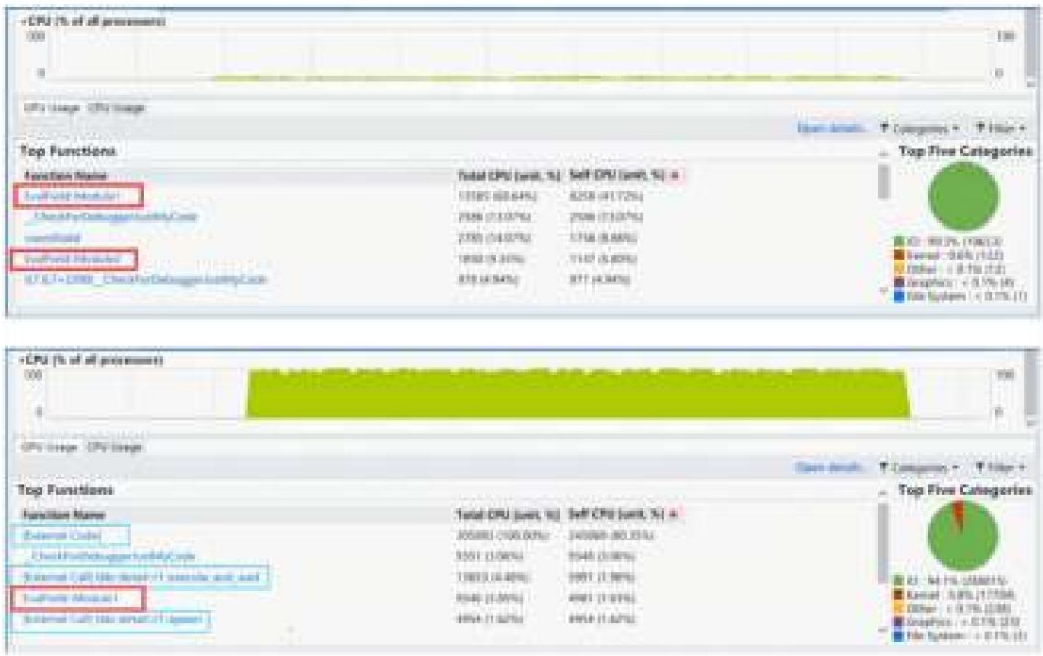


Рисунок A1. Загрузка ЦП и 5 основных модулей, занимающих ресурсы ЦП в состоянии [Отладка ×64]. В верхней части показаны результаты мониторинга алгоритма, выполняемого последовательно, а в нижней части часть показывает результаты мониторинга алгоритма, выполняемого параллельно.

Из рисунка A1 видно, что загрузка ЦП программой-алгоритмом в параллельное выполнение значительно выше, чем при последовательном выполнении. Когда алгоритм программа выполняется последовательно, оба модуля функции оценки, обведенные красной рамкой, входит в топ-5 модулей, занимающих больше всего ресурсов ЦП. Это указывает на то, что параллельно оптимизация оценочной части очень необходима. При параллельном выполнении алгоритма, сторонняя параллельная библиотека в синей рамке занимает значительную часть количество ресурсов процессора. Это косвенно указывает на то, что улучшение производительности после распараллеливания не достигается теоретического ускорения из-за значительного ресурса потребление внешним кодом и внешними вызовами.

Приложение С

Приложение C содержит полную информацию об игре, а также раунд матча. подсчитайте, как показано в таблицах A2 и A3.

Таблица A2. Полная информация об игре (MCTS плюс MG-PEO).

ID первого игрока		Второй игрок	Результат	Соответствовать Точка Раунды	Победитель
1	Амазонки на основе Qt Платформа QMainWindow	Божественный ход (наш)	0:3	41	Божественный ход (наш)
2	Божественный ход (наш)	Амазонки на основе Qt Платформа QMainWindow	3:0	39	Божественный ход (наш)
3	Гомоку	Божественный ход (наш)	0:3	39	Божественный ход (наш)
4	Божественный ход (наш)	Гомоку	3:0	36	Божественный ход (наш)
5	Отелло	Божественный ход (наш)	0:3	35	Божественный ход (наш)
6	Божественный ход (наш)	Отелло	3:0	33	Божественный ход (наш)
7	Божий алгоритм	Божественный ход (наш)	0:3	43	Божественный ход (наш)

Machine Translated by Google

Таблица A2. Продолжение

ID первого игрока		Второй игрок	Результат	Соответствовать Точка Раунды	Победитель
8	Божественный ход (наш)	Божий алгоритм	3:0	42	Божественный ход (наш)
9	Игровая система Amazons	Божественный ход (наш)	0:3	39	Божественный ход (наш)
10	Божественный ход (наш)	Игровая система Amazons	3:0	37	Божественный ход (наш)
11	Продвинутые маленькие королевы	Божественный ход (наш)	0:3	40	Божественный ход (наш)
12	Божественный ход (наш)	Продвинутые маленькие королевы	3:0	37	Божественный ход (наш)
13	мат	Божественный ход (наш)	0:3	41	Божественный ход (наш)
14	Божественный ход (наш)	мат	3:0	38	Божественный ход (наш)
15	Клиффхэнгерские амазонки	Божественный ход (наш)	0:3	40	Божественный ход (наш)
16	Божественный ход (наш)	Клиффхэнгерские амазонки	3:0	37	Божественный ход (наш)
17	Победа Ци Кайде	Божественный ход (наш)	0:3	41	Божественный ход (наш)
18	Божественный ход (наш)	Победа Ци Кайде	3:0	36	Божественный ход (наш)
19	Супер Амазонка	Божественный ход (наш)	0:3	39	Божественный ход (наш)
20	Божественный ход (наш)	Супер Амазонка	3:0	36	Божественный ход (наш)
21	Божий алгоритм	Божественный ход (наш)	0:3	41	Божественный ход (наш)
22	Божественный ход (наш)	Божий алгоритм	3:0	37	Божественный ход (наш)
23	Амазонки-путешественники	Божественный ход (наш)	0:3	40	Божественный ход (наш)
24	Божественный ход (наш)	Амазонки-путешественники	3:0	35	Божественный ход (наш)
25	Высшие шахматы Амазонки	Божественный ход (наш)	0:3	41	Божественный ход (наш)
26	Божественный ход (наш)	Амазонка Высшие шахматы	3:0	39	Божественный ход (наш)
27	Шахматные войска падают с Небеса	Божественный ход (наш)	0:3	43	Божественный ход (наш)
28	Божественный ход (наш)	Шахматные войска падают с Небеса	3:0	41	Божественный ход (наш)
29	Гвоздичные амазонки	Божественный ход (наш)	0:3	40	Божественный ход (наш)
30	Божественный ход (наш)	Гвоздичные амазонки	3:0	37	Божественный ход (наш)
31	Амазонки Шао Гуана	Божественный ход (наш)	0:3	43	Божественный ход (наш)
32	Божественный ход (наш)	Амазонки Шао Гуана	3:0	39	Божественный ход (наш)
33	ИИ играет в шахматы	Божественный ход (наш)	0:3	43	Божественный ход (наш)
34	Божественный ход (наш)	ИИ играет в шахматы	3:0	39	Божественный ход (наш)
35	Канопус Один	Божественный ход (наш)	0:3	39	Божественный ход (наш)
36	Божественный ход (наш)	Канопус Один	3:0	35	Божественный ход (наш)
37	Победить в стартовой команде 38	Божественный ход (наш)	0:3	41	Божественный ход (наш)
	Божественный ход (наш)	Выиграйте команду открытия	3:0	39	Божественный ход (наш)
39	Амазонок Вуконга 40	Божественный ход (наш)	0:3	34	Божественный ход (наш)
	Божественный ход (наш)	Вуконгские амазонки	3:0	30	Божественный ход (наш)
41	Амазонки-путешественники	Божественный ход (наш)	0:3	44	Божественный ход (наш)
42	Божественный ход (наш)	Амазонки-путешественники	3:0	40	Божественный ход (наш)
43	Юэ	Божественный ход (наш)	0:3	41	Божественный ход (наш)
44	Божественный ход (наш)	Юэ	3:0	40	Божественный ход (наш)
45	Последняя Амазонка	Божественный ход (наш)	0:3	41	Божественный ход (наш)
46	Божественный ход (наш)	Последняя Амазонка	3:0	38	Божественный ход (наш)
47	Информационный университет - Монте Карло	Божественный ход (наш)	1:2	42	Божественный ход (наш)
48	Божественный ход (наш)	Информационный университет - Монте Карло	3:0	40	Божественный ход (наш)
49	Хаха привет 50	Божественный ход (наш)	1:2	42	Божественный ход (наш)
	Божественный ход (наш)	Хаха привет	3:0	39	Божественный ход (наш)
51	Победа Дракона 52	Божественный ход (наш)	0:3	41	Божественный ход (наш)
	Божественный ход (наш)	Победа Дракона	3:0	37	Божественный ход (наш)
53	Супер Амазонка	Божественный ход (наш)	0:3	43	Божественный ход (наш)
54	Божественный ход (наш)	Супер Амазонка	3:0	38	Божественный ход (наш)
55	Базовая парная команда	Божественный ход (наш)	0:3	44	Божественный ход (наш)
56	Божественный ход (наш)	Базовая команда по спариванию	3:0	41	Божественный ход (наш)
57	Пройти уровень	Божественный ход (наш)	3:0	47	Пройти уровень

Machine Translated by Google

Таблица A2. Продолжение

ID первого игрока	Второй игрок	Результат	Соответствовать Точка Раунды	Победитель
58 Божественный ход (наш)	Пройти уровень	0:3	47	Пройти уровень
59 Даляньский университет Цзяотун Команда амазонок 1	Божественный ход (наш)	0:3	41	Божественный ход (наш)
60 Божественный ход (наш)	Даляньский университет Цзяотун Команда амазонок 1	3:0	39	Божественный ход (наш)
61 Выйти на берег	Божественный ход (наш)	0:3	41	Божественный ход (наш)
62 Божественный ход (наш)	Выйти на берег	3:0	40	Божественный ход (наш)
63 Пустой	Божественный ход (наш)	0:3	44	Божественный ход (наш)
64 Божественный ход (наш)	Пустой	3:0	41	Божественный ход (наш)
65 Бык	Божественный ход (наш)	0:3	42	Божественный ход (наш)
66 Божественный ход (наш)	Бык	3:0	40	Божественный ход (наш)
67 амазонок Вуконга	Божественный ход (наш)	1:2	34	Божественный ход (наш)
68 Божественный ход (наш)	Вуконгские амазонки	3:0	31	Божественный ход (наш)
69 Команда поддержки болельщиков Ганцзы	Божественный ход (наш)	0:3	41	Божественный ход (наш)
70 Божественный ход (наш)	Команда поддержки болельщиков Ганцзы	3:0	39	Божественный ход (наш)
71 Тайские брюки пряные	Божественный ход (наш)	0:3	44	Божественный ход (наш)
72 Божественный ход (наш)	Тайские брюки пряные	3:0	41	Божественный ход (наш)
73 Торт «Зеленая трава»	Божественный ход (наш)	0:3	34	Божественный ход (наш)
74 Божественный ход (наш)	Торт «Зеленая трава»	3:0	30	Божественный ход (наш)
75 Возьмите растительную гипоксию мозга Команда	Божественный ход (наш)	0:3	40	Божественный ход (наш)
76 Божественный ход (наш)	Не удалось схватить воздух и не сделал этого Возьмите растительную гипоксию мозга Команда	3:0	37	Божественный ход (наш)
77 ГД 78	Божественный ход (наш)	0:3	44	Божественный ход (наш)
Божественный ход (наш)	———	3:0	39	Божественный ход (наш)
79 Почему Тан Ян — Бог 80	Божественный ход (наш)	0:3	42	Божественный ход (наш)
Божественный ход (наш)	Почему Тан Ян — Бог	3:0	39	Божественный ход (наш)
81 Команда мечты	Божественный ход (наш)	0:3	43	Божественный ход (наш)
82 Божественный ход (наш)	Команда мечты	3:0	39	Божественный ход (наш)
83 Юбка Horse Face Повседневная и легкая носить	Божественный ход (наш)	0:3	43	Божественный ход (наш)
84 Божественный ход (наш)	Юбка Horse Face Повседневная и легкая носить	3:0	40	Божественный ход (наш)
85 Genshin Impact Expert Divine	Божественный ход (наш)	0:3	41	Божественный ход (наш)
86 Move (наш)	Genshin Impact Expert Divine	3:0	37	Божественный ход (наш)
87 Code Apprentice Divine Move	Move (наш)	0:3	42	Божественный ход (наш)
(наш) 88 89 Amazon Drift	Code Apprentice Divine	3:0	41	Божественный ход (наш)
Notes Divine Move (наш)	Move (наш)	0:3	39	Божественный ход (наш)
90	Amazon Drift Notes Divine	3:0	37	Божественный ход (наш)
91 Любовь исчезнет, верно?	Move (наш)	0:3	38	Божественный ход (наш)
92 Божественный ход (наш)	Любовь исчезнет, верно?	3:0	33	Божественный ход (наш)
93 Маленькая Су и кот 94	Божественный ход (наш)	0:3	43	Божественный ход (наш)
Божественный ход (наш) 95 Не	Маленькая Су и	3:0	42	Божественный ход (наш)
хочу Марлу 96 Божественный	Божественный ход кота (наш)	0:3	44	Божественный ход (наш)
ход (наш) 97 Команда настройки	Не хочу Марлы	3:0	43	Божественный ход (наш)
параметров 98 Божественный ход (наш)	Божественный ход (наш)	0:3	44	Божественный ход (наш)
	Команда настройки параметров Divine	3:0	41	Божественный ход (наш)
99 AlphaAmazon 100 Divine	Move (наша)	0:3	43	Божественный ход (наш)
Move (наш)	АльфаАмазонка	3:0	42	Божественный ход (наш)

Machine Translated by Google

Таблица А3. Полная информация об игре (MCTS).

ID первого игрока	Второй игрок	Результат	Соответствовать Точка Раунды	Победитель
1 Амазонки на основе Qt Платформа QMainWindow	MCTS (базовый уровень)	1:2	43	MCTS (базовый уровень)
2 MCTS (базовый уровень)	Амазонки на основе Qt Платформа QMainWindow	3:0	41	MCTS (базовый уровень)
3 Гомоку	MCTS (базовый уровень)	3:0	47	Гомоку
4 MCTS (базовый уровень)	Гомоку	3:0	45	MCTS (базовый уровень)
5 Отелло	MCTS (базовый уровень)	2:1	47	Отелло
6 MCTS (базовый уровень)	Отелло	2:1	44	MCTS (базовый уровень)
7 Божий алгоритм	MCTS (базовый уровень)	0:3	43	MCTS (базовый уровень)
8 MCTS (базовый уровень)	Божий алгоритм	3:0	40	MCTS (базовый уровень)
9 Игровая система Amazons	MCTS (базовый уровень)	0:3	42	MCTS (базовый уровень)
10 MCTS (базовый уровень)	Игровая система Amazons	3:0	40	MCTS (базовый уровень)
11 Продвинутые маленькие королевы	MCTS (базовый уровень)	3:0	47	Продвинутые маленькие королевы
12 MCTS (базовый уровень)	Продвинутые маленькие королевы	3:0	43	MCTS (базовый уровень)
Мат 13	MCTS (базовый уровень)	2:1	47	мат
14 MCTS (базовый уровень)	мат	3:0	40	MCTS (базовый уровень)
15 Клиффхэнгерские амазонки	MCTS (базовый уровень)	3:0	47	мат
16 MCTS (базовый уровень) 17	Клиффхэнгерские амазонки	3:0	44	MCTS (базовый уровень)
Победа Ци Кайде	MCTS (базовый уровень)	0:3	44	MCTS (базовый уровень)
18 MCTS (базовый уровень) 19	Победа Ци Кайде	3:0	41	MCTS (базовый уровень)
Супер Амазонка	MCTS (базовый уровень)	2:1	47	Супер Амазонка
20 MCTS (базовый уровень) 21	Супер Амазонка	2:1	43	MCTS (базовый уровень)
Божий алгоритм	MCTS (базовый уровень)	3:0	47	Божий алгоритм
22 MCTS (базовый уровень) 23	Божий алгоритм	2:1	40	MCTS (базовый уровень)
Амазонки-путешественники	MCTS (базовый уровень)	2:1	47	Амазонки-путешественники
24 MCTS (базовый уровень)	Амазонки-путешественники	3:0	42	MCTS (базовый уровень)
25 Высшие шахматы Амазонки	MCTS (базовый уровень)	0:3	40	MCTS (базовый уровень)
26 MCTS (базовый уровень)	Амазонка Высшие шахматы	3:0	41	MCTS (базовый уровень)
27 Шахматные войска падают с Небеса	MCTS (базовый уровень)	0:3	43	MCTS (базовый уровень)
28 MCTS (базовый уровень)	Шахматные войска падают с Небеса	3:0	39	MCTS (базовый уровень)
29 Гвоздичные амазонки	MCTS (базовый уровень)	0:3	44	MCTS (базовый уровень)
30 MCTS (базовый уровень) 31	Гвоздичные амазонки	3:0	40	MCTS (базовый уровень)
Амазонки Шао Гуана	MCTS (базовый уровень)	1:2	43	MCTS (базовый уровень)
32 MCTS (базовый уровень)	Амазонки Шао Гуана	3:0	40	MCTS (базовый уровень)
33 ИИ играет в шахматы	MCTS (базовый уровень)	2:1	47	ИИ играет в шахматы
34 MCTS (базовый уровень) 35	ИИ играет в шахматы	3:0	42	MCTS (базовый уровень)
Канопус Один	MCTS (базовый уровень)	0:3	43	MCTS (базовый уровень)
36 MCTS (базовый уровень)	Канопус Один	2:1	40	MCTS (базовый уровень)
37. Выиграйте первую команду	MCTS (базовый уровень)	3:0	47	Выиграйте команду открытия
38 MCTS (базовый уровень)	Выиграйте команду открытия	3:0	44	MCTS (базовый уровень)
39 амазонок Вуконга	MCTS (базовый уровень)	0:3	42	MCTS (базовый уровень)
40 MCTS (базовый уровень) 41	Вуконгские амазонки	3:0	39	MCTS (базовый уровень)
Амазонки-путешественники	MCTS (базовый уровень)	0:3	41	MCTS (базовый уровень)
42 MCTS (базовый уровень) 43	Амазонки-путешественники	3:0	36	MCTS (базовый уровень)
Юэ	MCTS (базовый уровень)	0:3	44	MCTS (базовый уровень)
44 MCTS (базовый уровень) 45	Юэ	3:0	41	MCTS (базовый уровень)
Последняя Амазонка	MCTS (базовый уровень)	0:3	42	MCTS (базовый уровень)
46 MCTS (базовый уровень)	Последняя Амазонка	3:0	37	MCTS (базовый уровень)
47 Информационный университет - Монте Карло	MCTS (базовый уровень)	3:0	47	Информация Университет-Монте-Карло
48 MCTS (базовый уровень)	Информационный университет Монте Карло	0:3	47	Информация Университет-Монте-Карло

Machine Translated by Google

Таблица А3. Продолжение

ID первого игрока	Второй игрок	Результат	Соответствовать Точка Раунды	Победитель
49 Хаха Привет	MCTS (базовый уровень)	3:0	47	Хаха привет
50 MCTS (базовый уровень) 51	Хаха привет	3:0	41	MCTS (базовый уровень)
Победа Дракона	MCTS (базовый уровень)	0:3	43	MCTS (базовый уровень)
52 MCTS (базовый уровень) 53	Победа Дракона	3:0	38	MCTS (базовый уровень)
Супер Амазонка	MCTS (базовый уровень)	0:3	41	MCTS (базовый уровень)
54 MCTS (базовый уровень) 55	Супер Амазонка	3:0	38	MCTS (базовый уровень)
Базовая команда по спариванию	MCTS (базовый уровень)	3:0	45	Базовая команда по спариванию
56 MCTS (базовый уровень) 57	Базовая команда по спариванию	0:3	42	Базовая команда по спариванию
Пройти уровень	MCTS (базовый уровень)	3:0	47	Пройти уровень
58 MCTS (базовый уровень)	Пройти уровень	0:3	47	Пройти уровень
59 Даляньский университет Цзяотун Команда амазонок 1	MCTS (базовый уровень)	0:3	44	MCTS (базовый уровень)
60 MCTS (базовый уровень)	Даляньский университет Цзяотун Команда амазонок 1	3:0	42	MCTS (базовый уровень)
61 Выйти на берег	MCTS (базовый уровень)	3:0	47	Выйти на берег
62 MCTS (базовый уровень) 63	Выйти на берег	0:3	47	Выйти на берег
Пустой	MCTS (базовый уровень)	0:3	45	MCTS (базовый уровень)
64 MCTS (базовый уровень)	Пустой	3:0	41	MCTS (базовый уровень)
65 Бык	MCTS (базовый уровень)	0:3	45	MCTS (базовый уровень)
66 MCTS (базовый уровень)	Бык	3:0	40	MCTS (базовый уровень)
67 амазонок Вуконга	MCTS (базовый уровень)	2:1	47	Вуконгские амазонки
68 MCTS (базовый уровень)	Вуконгские амазонки	3:0	39	MCTS (базовый уровень)
69 Команда поддержки болельщиков Ганцзы	MCTS (базовый уровень)	3:0	44	Фанаты Ганцзы Команда поддержки
70 MCTS (базовый уровень)	Команда поддержки болельщиков Ганцзы	1:2	41	Фанаты Ганцзы Команда поддержки
71 Тайские брюки пряные	MCTS (базовый уровень)	0:3	42	MCTS (базовый уровень)
72 MCTS (базовый уровень) 73	Тайские брюки пряные	3:0	40	MCTS (базовый уровень)
Торт «Зеленая трава»	MCTS (базовый уровень)	0:3	42	MCTS (базовый уровень)
74 MCTS (базовый уровень)	Торт «Зеленая трава»	3:0	37	MCTS (базовый уровень)
Не удалось схватить воздух и не сделал этого 75 Возьмите растительную гипоксию мозга Команда	MCTS (базовый уровень)	0:3	41	MCTS (базовый уровень)
76 MCTS (базовый уровень)	Не удалось схватить воздух и не сделал этого Возьмите растительную гипоксию мозга Команда	3:0	40	MCTS (базовый уровень)
77 ДГ	MCTS (базовый уровень)	0:3	43	MCTS (базовый уровень)
78 MCTS (базовый уровень)	[illegible]	3:0	39	MCTS (базовый уровень)
79 Почему Тан Ян — Бог	MCTS (базовый уровень)	2:1	47	Почему Тан Ян Бог
80 MCTS (базовый уровень) 81	Почему Тан Ян — Бог	3:0	44	MCTS (базовый уровень)
Команда мечты	MCTS (базовый уровень)	0:3	43	MCTS (базовый уровень)
82 MCTS (базовый уровень)	Команда мечты	3:0	44	MCTS (базовый уровень)
83 Юбка Horse Face Повседневная и легкая носить	MCTS (базовый уровень)	0:3	44	MCTS (базовый уровень)
84 MCTS (базовый уровень)	Юбка Horse Face Повседневная и легкая носить	3:0	40	MCTS (базовый уровень)
85 Эксперт по воздействию Геншина	MCTS (базовый уровень)	0:3	44	MCTS (базовый уровень)
86 MCTS (базовый уровень)	Эксперт по воздействию Геншина	3:0	39	MCTS (базовый уровень)
87 Код Ученика	MCTS (базовый уровень)	0:3	42	MCTS (базовый уровень)
88 MCTS (базовый уровень)	Код-ученик	3:0	40	MCTS (базовый уровень)
89 заметок о дрейфе Amazon	MCTS (базовый уровень)	0:3	44	MCTS (базовый уровень)
90 MCTS (базовый уровень) 91	Заметки о дрейфе Amazon	3:0	42	MCTS (базовый уровень)
Любовь исчезнет, верно?	MCTS (базовый уровень)	0:3	40	MCTS (базовый уровень)
92 MCTS (базовый уровень)	Любовь исчезнет, верно?	3:0	38	MCTS (базовый уровень)

Machine Translated by Google

Таблица А3. Продолжение

ID первого игрока	Второй игрок	Результат	Соответствовать Точка Раунды	Победитель
93 Маленькая Су и кот	MCTS (базовый уровень)	0:3	44	MCTS (базовый уровень)
94 MCTS (базовый уровень)	95 Маленькая Су и кот	3:0	40	MCTS (базовый уровень)
Не хочу Марлу	MCTS (базовый уровень)	0:3	41	MCTS (базовый уровень)
96 MCTS (базовый уровень)	97 Не хочу Марлу	3:0	37	MCTS (базовый уровень)
Группа настройки параметров	MCTS (базовый уровень)	0:3	42	MCTS (базовый уровень)
98 MCTS (базовый уровень)	Группа настройки параметров	3:0	39	MCTS (базовый уровень)
99 АльфаАмазонка	MCTS (базовый уровень)	0:3	41	MCTS (базовый уровень)
100 MCTS (базовый уровень)	АльфаАмазонка	0:3	47	АльфаАмазонка

Ссылки

1. Ли, Р.; Гао, М. Разработка алгоритма поиска Amazon на основе модели CNN. Цифра. Технол. Прил. 2022, 40, 164–166.
2. Го, Ц.; Ли, С.; Бао, Х. Исследование функции оценки компьютерных игр Amazon. Вычислить. англ. Прил. 2012, 48, 50–54.
3. Го, Т.; Цю, Х.; Тонг, Б.; Ван, Ю. Оптимизация и сравнение алгоритмов нескольких игр в Amazons. В материалах Китайская конференция по контролю и принятию решений (CCDC) 2019 г., Наньчан, Китай, 3–5 июня 2019 г.; стр. 6299–6304.
4. Цюань, Дж.; Цю, Х.; Ван, Ю.; Ли, Ф.; Цю, С. Применение технологий UCT для компьютерных игр Amazon. В материалах Китайская конференция по контролю и принятию решений (CCDC) 2016 г., Иньчуань, Китай, 28–30 мая 2016 г.; стр. 6896–6899.
5. Джу, Дж.; Цю, Х.; Ван, Ф.; Ван, Х.; Ван, Ю. Исследование по оптимизации потоков и открытию библиотеки на основе параллельного PVS Алгоритм в амазонках. В материалах 33-й Китайской конференции по контролю и принятию решений (CCDC) 2021 г., Куньмин, Китай, 28–30 мая 2021 г.; стр. 2207–2212.
6. Ли, З.; Нин, К.; Цао, Дж.; Ли, З. Амазонки на основе гибридного алгоритма UCT-PVS. В материалах 33-й Китайской контрольной конференции 2021 г. Конференция по принятию решений (CCDC), Куньмин, Китай, 28–30 мая 2021 г.; стр. 2179–2183.
7. Дин, М.; Бо, Дж.; Ци, Ю.; Фу, Ю.; Ли, С. Разработка игровой системы Amazons на основе обучения с подкреплением. В материалах Китайская конференция по контролю и принятию решений (CCDC) 2019 г., Наньчан, Китай, 3–5 июня 2019 г.; стр. 6337–6342.
8. Тонг, Б.; Цю, Х.; Го, Т.; Ван, Ю. Исследование и применение алгоритма параллельных вычислений PVS на базе Amazon Человеко-машинная игра. В материалах Китайской конференции по контролю и принятию решений (CCDC) 2019 г., Наньчан, Китай, 3–5. июнь 2019 г.; стр. 6293–6298.
9. Чен, Х.; Ян, Л. Исследование функции оценки у амазонок. Вычислить. Знать. Технол. 2019, 15, 224–226.
10. Ван, К.; Дин, М. Проектирование интерфейсов и внедрение персонализированных Amazon. Интел. Вычислить. Прил. 2017, 7, 78–80.
11. Чжан Л. Исследование игровой системы амазонок на основе минимаксного алгоритма поиска. Магистерская диссертация, Северо-Восточный университет, Шэньян, Китай, 2010 г.
12. Метрополис, Н.; Улам, С. Метод Монте-Карло. Дж. Ам. Стат. доц. 1949, 44, 335–341. [Перекрестная ссылка] [ПабМед]
13. Кулом Р. Эффективная селективность и резервные операторы в поиске по дереву Монте-Карло. В «Трудах компьютеров и игр» Турин, Италия, 1–5 апреля 2007 г.; стр. 72–83.
14. Гелли, С.; Кочиш, Л.; Шенауэр, М.; Себаг, М.; Сильвер, Д.; Сепешвари, К.; Тейто, О. Великая задача компьютерного го: Монте Поиск и расширения дерева Карло. Коммун. АКМ 2012, 55, 106–113. [Перекрестная ссылка]
15. Гелли С.; Сильвер, Д. Поиск в дереве Монте-Карло и быстрая оценка значений действий в компьютерном Го. Артиф. Интел. 2011, 175, 1856–1875. [Перекрестная ссылка]
16. Виньялс О.; Бабушкин И.; Чарнецкий, ВМ; Матье, М.; Дудзик, А.; Чанг, Дж.; Чой, Д.Х.; Пауэлл, Р.; Эвальдс, Т.; Георгиев П.; и др. Уровень гроссмейстера в StarCraft II с использованием многоагентного обучения с подкреплением. Природа 2019, 575, 350–354. [Перекрестная ссылка]
17. Сильвер, Д.; Хуанг, А.; Мэддисон, СЈ; Гез, А.; Сифре, Л.; ван ден Дриссе, Г.; Шритвизер, Дж.; Антоноглу, И.; Паннеершелвам, В.; Ланкто, М.; и др. Освоение игры Го с помощью глубоких нейронных сетей и поиска по дереву. Природа 2016, 529, 484–489. [Перекрестная ссылка]
18. Бернер, К.; Брокман, Г.; Чан, Б.; Чунг, В.; Дебиак, П.; Деннисон, К.; Фархи, Д.; Фишер, К.; Хашме, С.; Гессен, К.; и др. Дота 2 с крупномасштабным глубоким обучением с подкреплением. arXiv 2019, arXiv:1912.06680.
19. Браун, СВ; Паули, Э.; Уайтхаус, Д.; Лукас, С.М.; Коулинг, ИП; Рольфсхаген, П.; Тавенер, С.; Перес, Д.; Самотракис, С.; Колтон, С. Обзор методов поиска в дереве Монте-Карло. IEEE Транс. Вычислить. Интел. Игры ИИ 2012, 4, 1–43. [Перекрестная ссылка]
20. Клетцер Дж.; Иида, Х.; Бузи, Б. Подход Монте-Карло у амазонок. В материалах семинара по компьютерным играм. Амстердам, Нидерланды, 15–17 июня 2007 г.; стр. 185–192.
21. Шеннон С.Э. Программирование компьютера для игры в шахматы. Лонд. Эдинб. Дублин Филос. Маг. Дж. Наук. 1950, 41, 256–275. [Перекрестная ссылка]
22. Часло Г.; Баккес, С.; Сита, И.; Спронк, П. Поиск по дереву Монте-Карло: новая основа для игрового ИИ. В материалах Конференция AAAI по искусственному интеллекту и интерактивным цифровым развлечениям, Калифорния, Калифорния, США, 22–24 октября 2008 г.; стр. 216–217.
23. Свеховский, М.; Годлевски, К.; Савицкий, Б.; Мандзюк, Дж. Поиск по дереву Монте-Карло: обзор последних модификаций и приложений. Артиф. Интел. Ред. 2023, 56, 2497–2562. [Перекрестная ссылка]

Machine Translated by Google

24. Клёцер, Дж. Монте-Карло, открывающие книги для амазонок. В «Компьютерах и играх: 7-я Международная конференция (CG 2010), Канадзава, Япония, 24–26 сентября 2010 г., переработанные избранные статьи 7; Springer: Берлин/Гейдельберг, Германия, 2011 г.; стр. 124–135.
25. Сонг, Дж.; Мюллер, М. Расширенный решатель игры амазонок. IEEE Транс. Вычислить. Интел. Игры ИИ 2014, 7, 16–27. [Перекрестная ссылка]
26. Чжан Г.; Чен, Х.; Чанг, Р.; Чжан, Ю.; Ван, К.; Залог.; Ван, Дж.; Сюй, К. Быстрое освоение игры амазонок за счет разделения сетевого обучения. В материалах Международной совместной конференции по нейронным сетям 2021 г. (IJCNN), Шэньчжэнь, Китай, 18–22 июля 2021 г.; стр. 1–8.

27. де Конинг, Дж. INVADER продлевает титул Amazons. ICGA J. 2011, 34, 96. [CrossRef]
28. Клетцер Дж. INVADER выигрывает турнир амазонок. ICGA J. 2009, 32, 112–113. [Перекрестная ссылка]
29. Лоренц, Р. Инвейдер выигрывает восьмую золотую медаль амазонок. ICGA J. 2017, 39, 228–229. [Перекрестная ссылка]
30. Го, Ц.; Ли, С.; Бао, Х. Исследование алгоритма поиска в игре амазонок. В материалах Китайской конференции по контролю и принятию решений (CCDC) 2011 г., Мяньян, Китай, 23–25 мая 2011 г.; стр. 1859–1862.
31. Ли, Х.; Хоу, Л.; Ву, Л. Алгоритм UCT в человеко-компьютерных играх Amazon. В материалах 26-й Китайской контрольно-пропускной Конференция по принятию решений (CCDC), Чанша, Китай, 31 мая – 2 июня 2014 г.; стр. 3358–3361.
32. Цюань, Дж.; Цю, Х.; Ван, Ю.; Ли, Ю.; Ван, Х. Исследование производительности алгоритмов поиска в Amazon. В материалах 27-го числа Китайская конференция по контролю и принятию решений (CCDC), Циндао, Китай, 23–25 мая 2015 г.; стр. 5811–5813.
33. Либерум Дж. Функция оценки игры амазонок. Теор. Вычислить. наук. 2005, 349, 230–244. [Перекрестная ссылка]
34. Чай, З.; Фанг, З.; Чжу, Дж. Стратегия оптимизации оценки Amazon на основе алгоритма PSO. В материалах 2019 года Китайская конференция по контролю и принятию решений (CCDC), Наньчан, Китай, 3–5 июня 2019 г.; стр. 6334–6336.
35. Сунь, Ю.; Юань, Д.; Гао, М.; Чжу, П. GPU-ускорение алгоритма поиска по дереву Монте-Карло для амазонок и его функция оценки. В материалах Международной конференции по искусственному интеллекту, обработке информации и облачным вычислениям 2022 года (AIIPCC), Куньмин, Китай, 21–23 июня 2022 года; стр. 434–440.
36. Кочиш, Л.; Сепешвари, К.; Уиллемсон, Дж. Улучшенный поиск в Монте-Карло. унив. Тарту Приблиз. Тех. Отчет 2006, 1, 1–22.
37. Финнссон, Х.; Бьернссон, Ю. Свойства игрового дерева и производительность MCTS. IJCAI 2011, 11, 23–30.
38. Фитт, К. Строительные блоки Intel® для многопоточности. Дж. Компьютер. наук. Колл. 2008, 23, 164–166.
39. Чайлдс, Б.Э.; Бродер, Дж. Х.; Кочиш, Л. Транспозиции и группы перемещения в поиске по дереву Монте-Карло. В материалах симпозиума IEEE 2008 г. по вычислительному интеллекту и играм, Перт, Австралия, 15–18 декабря 2008 г.; стр. 389-395.
40. Ван Эйк, Г.; Мюллер, М. Возвращение к группам перемещения при поиске по дереву Монте-Карло. В достижениях в области компьютерных игр: 13-я Международная конференция, ACG 2011, Тилбург, Нидерланды, 20–22 ноября 2011 г., переработанные избранные статьи 13; Springer: Берлин/Гейдельберг, Германия, 2012 г.; стр. 13–23.

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Стандартизация цвета изображений химических растворов с использованием Глубокое сопоставление гистограмм на основе шаблонов Обучение регрессии

Патриция Квик и Малгожата Якубовска *

Факультет материаловедения и керамики Краковского университета AGH, ал. Мицкевича 30, 30-059 Краков, Польша; pakwiek@agh.edu.pl
* Переписка: jakubows@agh.edu.pl

Аннотация: Искажение цвета изображения представляет собой проблему для классификации и регрессии машинного обучения, когда входные данные состоят из изображений. В результате предложен новый алгоритм стандартизации цвета фотографий, составляющий основу модели глубокой нейросетевой регрессии. В этом подходе используется собственный цветовой шаблон, разработанный на основе первоначальной серии исследований и цифровых изображений. Используя выровненную гистограмму каналов R, G, B цифрового шаблона и его фотографии, была рассчитана стратегия цветового отображения. Применяя этот подход, были скорректированы гистограммы и стандартизированы цвета фотографий, сделанных на смартфон.

Предложенный алгоритм был разработан для серии изображений, где вся поверхность примерно сохраняла однородный цвет, а различия в цвете между фотографиями отдельных объектов были незначительными. Этот оптимизированный подход был проверен в процедуре колориметрического определения витамина С. Набор данных для глубокой нейронной сети в регрессионном варианте был сформирован из фотографий образцов при двух разных условиях освещения. Для диапазона концентраций витамина С от 0 до 87,72 мкг·мл-1 среднеквадратическое отклонение для тестового набора колебалось в пределах от 0,75 до 1,95 мкг·мл-1 по сравнению с нестандартизированным вариантом, где этот показатель находился на уровне 1,48–2,29 мкг·мл-1 . Соответствие прогнозируемых результатов концентрации фактическим данным, выраженным как R2 , колебалось в пределах от 0,9956 до 0,9999 для каждого из стандартизированных вариантов. Такой подход позволяет устранить отражения света на блестящих поверхностях растворов, что является распространенной проблемой в жидких образцах. Этот алгоритм подбора цветов носит универсальный характер, и сфера его применения не ограничена.

Ключевые слова: стандартизация цвета; соответствие цветовому шаблону; подгонка гистограммы RGB; глубокое обучение; колориметрическое определение витамина С

check for updates

Цитирование: Квик, П.; Якубовская, М. Стандартизация цвета химических веществ Изображения решений с использованием сопоставления гистограмм на основе шаблонов в регрессии глубокого обучения. Алгоритмы 2024, 17, 335. https://doi.org/10.3390/a17080335

Академический редактор: Марцин Ивановский

Поступила: 11 июля 2024 г.
Пересмотрено: 29 июля 2024 г.
Принято: 30 июля 2024 г.
Опубликовано: 1 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Разработка методов измерения и стратегий исследований в различных областях науки и техники тесно связана с потенциальной поддержкой, предлагаемой методами машинного обучения (ML) и глубокого обучения (DL). Если информация в решаемой задаче передается посредством изображений, рисунков, эскизов или фотографий, использование DL при моделировании данных часто является оптимальным подходом. Применение нейронных сетей в типичном контролируемом обучении требует доступа к большим наборам данных, что делает метод получения и маркировки данных существенным ограничением. Сегодняшние технологии предоставляют множество электронных устройств, таких как камеры или сканеры, которые можно использовать для быстрого и без специальной квалификации получения данных для построения моделей. Информация часто может быть закодирована просто в виде объектов, взаимоотношений между объектами в анализируемой области, текстур, количества мелких компонентов и цветов.

Несколько цветовых пространств широко используются в разных отраслях для различных приложений [1]. Пространство RGB, основанное на основных цветах света (красном, зеленом и синем), широко используется в цифровых дисплеях и изображениях. CMYK (голубой, пурпурный, желтый и черный) используется в цветной печати, опираясь на субтрактивную цветовую модель для создания широкого спектра цвето

Machine Translated by Google

на бумаге. HSI/HSV (оттенок, насыщенность, интенсивность/значение) часто используется в анализе изображений и компьютерном зрении из-за его соответствия человеческому восприятию цветов. YUV и YCbCr используются при сжатии видео и трансляции, разделяя изображение на компоненты яркости и цветности для оптимизации хранения и передачи данных. Цветовое пространство CIE Lab , разработанное Международной комиссией по освещению, обеспечивает единообразное восприятие цвета, что делает его идеальным для задач дифференциации и преобразования цветов . RAL CLASSIC — широко используемая в Европе система подбора цветов, состоящая из 213 стандартизированных цветов, тогда как RAL DESIGN предлагает дизайнерам и архитекторам более обширную палитру из 1625 цветов . RAL DIGITAL обеспечивает точную передачу цветов в цифровых приложениях посредством значений RGB и CMYK. Система соответствия PANTONE (PMS) признана во всем мире в графическом дизайне, полиграфии и моде, обеспечивая стандартизированные цветовые коды для обеспечения единообразия различных носителей и материалов. Каждое из этих цветовых пространств имеет свои преимущества и области применения, отвечающие конкретным потребностям различных отраслей.

Если цвет является основным, решающим критерием при анализе изображений, который кодирует искомую информацию , крайне важно делать фотографии так, чтобы сохранить соответствующее качество этого параметра. Истинный цвет изображения позволяет правильно интерпретировать различные особенности фотографируемых объектов. Нежелательная изменчивость может возникнуть из-за использования нескольких устройств для захвата изображений или из-за метода получения изображений с помощью одного устройства в неконтролируемых или изменяющихся условиях. К значимым факторам относятся изменения условий освещения, отражения на блестящих поверхностях и такие изменения, как время суток. Эти элементы не могут напрямую влиять на качество фотографии, но могут влиять на ее информационную ценность. Поэтому следует учитывать важность стандартизации цвета как решающего шага предварительной обработки данных. Различные вычислительные подходы к сохранению или восстановлению истинного цвета изображений в наборе данных для глубокого обучения описаны в многочисленных работах, предлагающих решения этой проблемы.

В этом исследовании представлен оригинальный алгоритм стандартизации цветов фотографий, снятых одной камерой смартфона в различных условиях освещения. Он также учитывает изменчивость, возникающую в результате случайных факторов, связанных с процессом фотосъемки, и отсутствие идеальной повторяемости работы камеры. Исследование было сосредоточено на однородной жидкости, которая демонстрировала минимальные различия цвета среди тестируемых объектов. Вся фотография, использованная в процессе моделирования, имела один и тот же цвет, хотя и с определенным уровнем случайной изменчивости, которую невозможно полностью устранить. В этом контексте даже небольшие изменения цвета имеют значительную информационную ценность, подчеркивая важность точной интерпретации цвета для повышения качества модели регрессии глубокого обучения и минимизации ошибок прогнозирования.

Предложен алгоритм стандартизации, использующий гистограмму специально разработанного шаблона, сфотографированного в контролируемых условиях вместе с испытуемым объектом, по отношению к гистограмме шаблона, подготовленного с помощью графического приложения, и его фотографии. В этом исследовании также представлена проверка предложенного подхода к колориметрическому определению витамина С с использованием регрессионной модели, определенной DNN (глубокие нейронные сети).

2. Сопутствующие работы

2.1. Стандартизация цвета

Существует множество категорий проблем с калибровкой цвета, причем две основные группы относятся либо к использованию различных устройств в серии фотографий, либо к изменениям условий освещения, влияющим на фотографируемые объекты. Аппаратная компенсация цвета может быть достигнута за счет использования датчиков камеры, которые автоматически регулируют настройки в зависимости от освещения снимаемой сцены. Альтернативный подход включает численную обработку фотографий с использованием различных стратегий, которые в конечном итоге направлены на достижение одинаковых цветовых тонов на нескольких изображениях. Для этого первоначально используются калибровочные диаграммы, которые соотносят и сопоставляют отдельные пиксели отснятых фотографий со стандартизированными значениями соответствующих цветовых пятен на диаграмме [2,3

Machine Translated by Google

Калибровочная таблица служит справочным инструментом в фотографии, создании фильмов, видеозаписи или графическом дизайне, обеспечивая точную и последовательную цветопередачу изображений. Обычно он представляет собой сетку цветных пятен известных и стандартизированных цветов. Эти диаграммы применяются при калибровке оборудования и настройке во время постобработки, обеспечивая тем самым согласованность представления цвета на различных устройствах и в различных условиях. На коммерческой основе доступно несколько калибровочных таблиц, от шести до 100 цветных полей.

Однако 24-цветная калибровочная таблица, часто называемая Macbeth ColorChecker, нашла широкое применение во многих приложениях [4]. Кроме того, разработка индивидуальных калибровочных диаграмм играет важную роль в стандартизации цветов изображений.

Стандартизация цвета играет важную роль во многих областях науки и техники, включая медицинскую диагностику, биоанализ, сельское хозяйство, анализ доказательств в судебной медицине, аналитическую химию и т. д. При обработке сельскохозяйственных изображений основное внимание уделяется связыванию изменений цвета изображений. в определенных областях интереса с особыми характеристиками качества, такими как здоровье растений, стресс урожая и уровень зрелости. В [5] предложена методика калибровки цифровых изображений, обеспечивающая их однородность и улучшающая фенологические сравнения в сельском хозяйстве. Преобразование цвета изображений достигалось с использованием стандартной 24-цветной калибровочной таблицы. Калибровка дает типичные и ожидаемые результаты путем настройки значений RGB цветовых пятен входного изображения до стандартных значений RGB. В [6] было разработано цифровое устройство подбора цвета для определения цвета зубов, поскольку точная оценка цвета имеет решающее значение в процедурах реставрации зубов. Основой для расчетов и преобразований послужили записи изображений , сделанные интраоральной камерой.

Для оценки цвета зубов использовались алгоритм машины опорных векторов (SVM) и цветовое пространство CIEL*a*b* . Цветокоррекция основывалась на расчете евклидовой

расстояние между оттенками. Авторы создали специальный шаблон, отличающийся от типичного шаблона с широким спектром цветов и специально адаптированный к диапазону вариаций цвета зубов, рассматриваемых в исследовании. Следующее предложение касается повышения точности цветопередачи в цифровых камерах и смартфонах. Первый метод использует коррекцию баланса белого с последующим преобразованием цветового пространства. Второй подход, применимый к изображениям, обрабатываемым в автономном режиме, основан на выполнении полного цветового баланса вместо баланса белого и требует использования машинного обучения для прогнозирования матрицы цветового баланса [7]. В обзорной статье [8] обсуждается использование различных алгоритмов распознавания гистопатологических изображений. Различия в цвете при гистопатологии возникают в результате использования различного оборудования или методов окрашивания тканей. Обычно нормализация цвета включает в себя перенос среднего цвета целевого изображения в исходное изображение и разделение пятен, присутствующих в исходном из
В статье описан обширный набор методов стандартизации цвета, проверенных с помощью

гистопатологических изображений, в том числе полученных при диагностике рака молочной железы и толстой кишки.

В связи с развитием технологий цифровых изображений и, как следствие, появлением множества совершенно новых приложений, решения на основе изображений со смартфона предлагаются в различных областях. В таблице 1 приведен список избранных статей, в которых цветные изображения служат важными носителями информации, а цветокоррекция и стандартизация существенно влияют на результаты моделирования и качество предлагаемых решений

Таблица 1. Сравнение родственных работ.

Ссылка.	Год	Проблема решена	Используемый набор данных	Стратегия цветокоррекции
[9]	2022 год	Общая концепция относительного постоянства цвета (RCC), т. е. способности выравнивать цвета одних и тех же объектов на изображениях независимо от освещения и камеры.	Три набора данных изображений, собранных несколькими камерами в различных условиях освещения и экспозиции (Белфаст, Миддлбери, Гелер-Ши)	Три этапа процедуры выравнивания цвета: калибровка отклика камеры с помощью цветовых пятен, линеаризация отклика, попиксельная интенсивность цвета и сопоставление цветности.

Machine Translated by Google

Таблица 1. Продолжение.

Ссылка.	Год	Проблема решена	Используемый набор данных	Стратегия цветокоррекции
[10]	2022 год	Цветовые вариации в клинических условиях изображения образцов челюстно-лицевого протеза из силикона (метод in vitro)	Изображения пигментированных образцов силиконовых протезов в различных условиях окружающей среды. условия освещения	Компьютеризированная программная калибровка баланса белого с постобработкой (PPWBC) с использованием серой карты и цветовой таблицы Макбета.
[11]	2023 год	Неинвазивное сохранение культурного наследия в условиях деградации материалов	Четыре набора изображений в различных условиях освещения, снятых обычной камерой мобильного телефона в цветовом пространстве sRGB.	Адаптивная передаточная функция, работающая с самодельной калибровочной схемой цвета; окончательная настройка калибровочных функций на основе нейронной сети
[12]	2023 год	Колориметрия мочи с помощью смартфона	Образцы мочи 58 пациентов, сфотографированные в специальном фотобоксе при пяти смоделированных условиях освещения с использованием пяти смартфонов в цветовом пространстве RGB.	Калибровка цвета с помощью Цветовая диаграмма SpyderCHEKRTM и собственные алгоритмы Adobe Photoshop для масштабирования неисправленных Значения RGB
[13]	2023 год	Решение на базе смартфона для точных колориметрических измерений с системой навигации дополненной реальности .	Смоделированные и реальные изображения, изображения наборов цветных полосок для измерения pH	Алгоритм цветокоррекции с моделью пространственной изменяющейся регрессии первого порядка, которая использует как абсолютную величину цвета, так и масштаб; система включает в себя цветную справочную доску
[14]	2023 год	Оценка процесса заживления хронических ран	Изображения раневой ткани, полученные в разное время, при различных условиях освещения, расстояния, объективов и камер смартфонов в цветовом пространстве RGB.	Изображения раневой ткани, полученные с помощью Диаграмма Макбета, состоящая из 24 квадратных цветных пятен и четырех маркеров ArUco по углам карты; расчет матрицы преобразования с использованием Обратная матрица Мура – Пенроуза и целевая матрица
[15]	2024 год	Алгоритм общего назначения	Фотографии природы и городских пейзажей, растительности и рукотворных объектов в цветовом пространстве RGB.	Преобразование на основе матрицы с использованием тензорного произведения B-сплайнов с оптимизацией параметра сглаживания λ и количества базисных функций сплайна
[16]	2024 год	Воспроизведение эталонного цвета текстильной одежды путем подбора соответствующих дисперсионных красителей.	Результаты колориметрических измерений с использованием типа спектрофотометра Спектрафлэш 600 Плюс (Датаколор Интернэшнл, США)	Генетический алгоритм для оптимизации выбора и концентрации красителей для точного подбора цвета в двухкомпонентных (ПЭТ/ПТТ) нитях.
[17]	2024 год	Оптимизация внешнего вида продуктов растительного происхождения, учитывая, что продукты животного происхождения могут содержать широкий спектр различных пигментов и структурных элементов, рассеивающих свет.	Значения L*, a*, b* продуктов животного происхождения и полученных эмульсий растительного происхождения, измеренные колориметром (ColorFlex EZ 45/0-LAV, Hunter Associates Laboratory Inc., Рестон, Вирджиния, США)	Модель подбора цветов, основанная на Теория Кубелки-Мунка, используемая для расчета спектрального коэффициента отражения смешанных эмульсий из спектральная отражательная способность отдельных цветных эмульсий
[18]	2024 год	Цветовой дизайн для моды: выбор цвета и подбор различных цветовых комбинаций.	Графические объекты, созданные в CorelDraw	Сопоставление цветов на основе интерактивного генетического алгоритма с использованием эталонных изображений.

Machine Translated by Google

2.2. Определение витамина С

Методы стандартизации цвета имеют решающее значение в аналитической химии, особенно при колориметрическом определении различных веществ. Это особенно очевидно в методах , в которых растворы тестируемых соединений фотографируются, а информация о концентрации получается из этих фотографий с использованием различных статистических инструментов и многомерного анализа. Иллюстрацией такой методики является колориметрическое определение витамина С.

Аскорбиновая кислота (АК), то есть витамин С, представляет собой природное, хорошо растворимое в воде соединение, которое играет доминирующую роль в здоровье человека и необходимо для различных метаболических реакций и биологических процессов, обладая сильной антиоксидантной активностью, ограничивая свободные радикалы. повреждат Недостаточная концентрация витамина С в организме, среди прочего, вызывает цингу и анемию и связана с некоторыми психологическими отклонениями, например, депрессией.

Чрезмерное потребление витамина С может иметь негативные последствия, такие как затруднение всасывания витамина В12 и потенциальное возникновение желудочно-кишечных проблем или камней в почках [19,20]. Поскольку люди не способны вырабатывать витамин С внутри организма, важно дополнять его регулируемым количеством АК, обычно из свежих фруктов и овощей [21,22].

Аналитическая химия предлагает множество стратегий для определения АК в пищевых продуктах и добавках витамина С. Традиционно для измерения уровня аскорбиновой кислоты использовались методы инструментального анализа , обеспечивающие широкий диапазон концентраций с низким пределом обнаружения. Эти методы включают прямое титрование [23], сверхэффективную жидкостную хроматографию (UPLC), высокоэффективную жидкостную хроматографию (ВЭЖХ) [24,25], инфракрасную спектрометрию с преобразованием Фурье (FT-IR) [26], хемилюминесценцию [27], Рамановская спектроскопия [28], ближняя инфракрасная (NIR), рамановская спектроскопия Фурье [29] и капиллярный электрофорез [30]. Кроме того , для определения АК использовались различные электрохимические методы, обеспечивающие чрезвычайно высокую чувствительность, селективность, воспроизводимость и надежность [31–33]. Однако упомянутые методы химического инструментального анализа требуют специализированных лабораторных помещений, дорогостоящего оборудования, множества вспомогательных реагентов, трудоемких этапов экстракции и сложных процедур измерений, требующих высококвалифицированного персонала для таких задач, как подготовка реагентов и проб, проведение исследований или интерпретация. результаты.

Разработка новых методологий, которые либо заменяют, либо поддерживают традиционные лабораторные стратегии с помощью экономически эффективных инструментов, удовлетворяют потребность в высокоэффективных и адаптируемых подходах к анализу пищевых продуктов, фармацевтики и окружающей среды. Популярные и широко доступные смартфоны представляют собой перспективную аналитическую платформу, характеризующуюся полной автоматизацией, большим объемом локальной памяти, доступом к внешнему облачному хранилищу и особенно высококачественным камерам, что делает их пригодными для широкого использования в качестве портативных измерительных у По сравнению с традиционными методами измерения с помощью смартфона используют цифровую визуализацию, предлагая множество приложений в различных областях и значительные преимущества. К ним относятся низкая стоимость, простота эксплуатации, возможность установки различных специализированных датчиков, точные измерения в реальном времени, неразрушающий характер контроля во многих случаях, быстрый анализ и переносимость на другие подобные устройства. В аналитической химии широко доступные устройства, такие как камеры, веб-камеры, сканеры и смартфоны, применяются для регистрации изображений колориметрических реакций, в которых интенсивность получаемого цвета напрямую коррелирует с концентрацией аналита. В литературе представлены различные описания использования методов цифровой визуализации для определения АА. В работе [34] представлен простой, быстрый и экономически эффективный метод, использующий анализ изображений смартфона для количественного определения АК в водном растворе. В основу исследования легла оценка повреждающего действия АК при ферментативном колориметрическом определении глюкозы. Был предложен другой подход, включающий колориметрию цифровых изображений для определения содержания витамина С в натуральных фруктовых соках. Этот метод предполагает измерение цвета комплекса железо(II)—о-фенантролин [35]. В следующей статье авторы разработали новый спектрометр кругового дихроизма на базе смартфонов для высокочувствительного и ультрапортативного колориметрического анализа, предлагающий преимущества экономической эффективности и простоты. Этот спектрометр использовал цветовую модель HSV и определял оптическую плотность путем суммирования значений V соседних позиций для расчета общей интенсивности. Также была использована реакция нанолистов ТМБ-MnO2, и, по сути, была предложена высокочувствительная и специфичная система для обнаружения витаминов.

Machine Translated by Google

С [36]. В [37] был разработан датчик для мест оказания медицинской помощи, состоящий из флуоресцентного бумажного чипа, напечатанных на 3D-принтере деталей и смартфона. Эта система позволяла количественно и высокочувствительно визуально обнаруживать АА в реальном времени. Конструкция этого устройства предполагала размещение флуоресцентного датчика на портативной сенсорной платформе, подключенной к смартфону.

Аналитический метод измерения, использованный в данной работе, основан на восстановительных свойствах витамина С. АК восстанавливает Fe(III) до Fe(II), а затем образует комплекс с о-фенантролином. Крепкий красный комплекс Fe(II) и о-фенантролина широко используется для колориметрического определения и в качестве индикатора [38]. Эта реакция характеризуется высокой скоростью, стабильностью и воспроизводимостью. Использование этого подхода на основе камеры в сочетании с цифровой визуализацией цветных соединений повышает точность и снижает стоимость измерения содержания витамина С.

2.3. Алгоритмы многомерной регрессии

Методы регрессии широко используются в качестве контролируемого метода обучения, включающего моделирование взаимосвязи между признаками и целевой переменной для решения задач, направленных на прогнозирование непрерывных значений. Типичными алгоритмами, используемыми для этой цели, являются PCR (регрессия главных компонентов) [39], PLSR (частичная регрессия наименьших квадратов) [40,41] и SVR (регрессия опорных векторов) [42]. DNN с различными архитектурами, отображающими нелинейную структуру многомерных данных, также может применяться для решения задач регрессии [43,44]. Сверточные нейронные сети (CNN) — это тип алгоритма глубокого обучения, работа которого основана на математической операции, называемой сверткой. Они состоят из нескольких слоев сверточных фильтров, которые автоматически изучают иерархические представления объектов непосредственно на основе необработанных пиксельных данных, что делает их высокоэффективными для таких задач, как классификация или сегментация изображений и обнаружение объектов. Обычно качество регрессионных моделей оценивается с использованием средних квадратов ошибок между реальными значениями и прогнозами (MSE) или средней абсолютной ошибки (MAE).

MSE и MAE — это разные метрики ошибок, используемые для оценки эффективности прогнозных моделей. MSE более чувствителен к большим ошибкам, поскольку они возводятся в квадрат, тогда как MAE обеспечивает более интуитивное понимание средней ошибки и выражается в единицах целевой переменной. Эти оценки, используемые в качестве функций потерь во время обучения нейронной сети, минимизируют среднюю разницу между прогнозируемыми и фактическими значениями. Эффективность обучения и прогнозирования калибровочной модели также оценивается на основе следующих показателей: среднеквадратическая ошибка калибровки (RMSEC) и прогнозирования (RMSEP), а также коэффициент детерминации R2 для калибровки и прогнозирования [45]. Поскольку ожидается, что результат прогнозирования будет соответствовать фактическому значению, точки с координатами (фактическими, прогнозируемыми) в идеале должны лежать на линии y = x. Однако из-за случайной изменчивости и несовершенного соответствия модели данным должны соблюдаться два условия: значение единицы должно попадать в доверительный интервал наклона, а значение нуля должно находиться в доверительном интервале пересечения [46].].

3. Предлагаемый метод

3.1. Алгоритм подбора цветов

Основной целью данной работы была разработка подхода к определению витамина С, основанного на изменении цвета раствора под влиянием его концентрации. Поэтому решающим шагом в создании высококачественной регрессионной модели стало выполнение процедуры стандартизации цвета сфотографированных образцов витамина С. Предложенный алгоритм стандартизации цвета был разделен на две части. На первом этапе было выполнено стандартное сопоставление гистограмм между цифровым цветным шаблоном и фотографией напечатанного шаблона, полученной при заданных условиях освещения. На втором этапе выполнялось сопоставление гистограмм между фотографией раствора (испытуемого образца), сделанной при том же освещении, что и образец, и эталонным шаблоном. В итоге из реконструированного изображения было вырезано 512 случайных квадратов меньшего размера, чтобы получить достаточно большой набор данных для последующего обучения регрессионной нейронной сети. Количество случайно выбранных маленьких квадратов было оптимизировано, так как предварительные рас

Machine Translated by Google

привело к снижению качества модели. Также необходимо было предпринять дополнительный шаг по адаптации самостоятельно разработанного эталонного шаблона к соответствующему диапазону значений. В результате этой операции был разработан адаптированный алгоритм сопоставления гистограмм.

Сопоставление гистограммы — это операция, при которой пиксели входного изображения манипулируют таким образом, чтобы его гистограмма соответствовала гистограмме эталонного изображения. В данной работе для расчетов было выбрано цветовое пространство RGB. Он широко доступен и популярен, поскольку большинство камер и устройств захвата изображений записывают данные непосредственно в формате RGB, что исключает необходимость дополнительных преобразований цветового пространства. Пространство RGB также легко реализовать, оно интуитивно понятно и напрямую связано с физическими свойствами света, что упрощает разработку алгоритмов. Кроме того, в нашем конкретном приложении RGB дал удовлетворительные результаты. Поэтому, несмотря на преимущества других цветовых пространств, RGB оказался наиболее подходящим по практическим и техническим причинам. В случае цветных изображений каждый канал (RGB) рассматривался независимо. В контексте обработки изображений это была задача нормализации, связанная, наг



Рисунок 1. Пример сопоставления гистограмм для наглядности.

С математической точки зрения сопоставление гистограмм можно представить как преобразование изображения, при котором кумулятивная функция распределения (CDF) значений в каждом канале соответствует CDF полос в другом изображении.

Чтобы совместить гистограммы двух изображений, необходимо выровнять их гистограммы. После этого каждый пиксель первого изображения модифицируется и сопоставляется со вторым изображением. Эту операцию проводили для каждого из каналов (R, G, B) фотографируемого шаблона при заданном освещении (естественный свет и искусственный свет) и для каналов цифрового шаблона. Весь процесс преобразования цвета изображения представлен на рисунке 2.

С целью стандартизации изображений тестируемых химических растворов из гистограмм фотографии тестируемого образца был выбран правильный диапазон значений RGB. Если значение гистограммы для образца было больше или равно 1% от максимального значения гистограммы химического раствора, значение гистограммы не изменялось, в противном случае оно устанавливалось равным 0. Расхождение в количестве пикселей нивелировалось их равномерное случайное распределение по диапазону значений. Последним шагом алгоритма стандартизации было сопоставление гистограмм изображения решения с адаптированными шаблонными
Это было сделано так же, как описано на рисунке 3, с последующей реконструкцией изображения по каналам R, G и B.

Machine Translated by Google

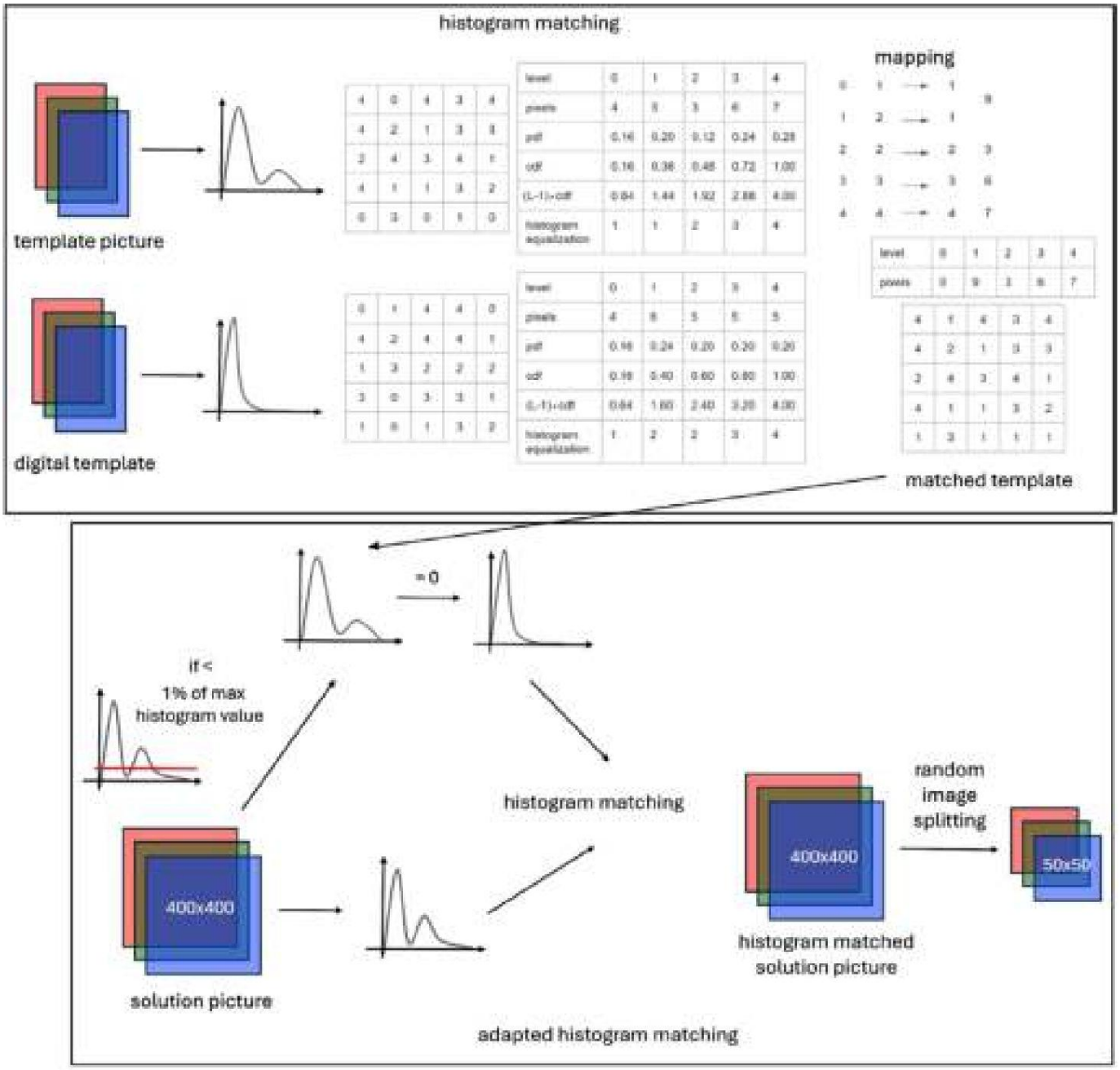


Рисунок 2. Схема алгоритма сопоставления цветов.



гистограмма химического раствора, значение гистограммы не изменялось, в противном случае оно было

Распределение по диапазону значений. Последний шаг в стандартизации

Рисунок 3. Растворы витамина С (0, 4,54, 11,31, 18,05, 22,52, 33,63, 44,64, 66,37 и 87,72 мкг·мл-1) в присутствии бидистиллированной воды, ацетатного буфера, сульфата железа(III) аммония, и ортофенантролин.

Machine Translated by Google

3.2.

Программное обеспечение Все расчеты проводились в Python 3.10.0. Были использованы следующие библиотеки и пакеты : opencv 4.8.1, scikit-image 0.22.0, numpy 1.26.1, matplotlib 3.5.3, scikit-learn 1.0.2, tensorflow 2.9.1 и keras 2.10.0.

4. Результаты и обсуждение 4.1.

Реагенты, лабораторное оборудование и методика измерений. В данной работе колориметрическая реакция, использованная для определения аскорбиновой кислоты (АК), основана на восстановлении Fe(III) до Fe(II) АК, образующей комплекс красного цвета с ортофенантролином. В этом эксперименте L(+)-аскорбиновая кислота чистая ра, CAS: 50-81-7 (POCH, Гливице, Польша), 1,10 фенантролин ра, CAS: 66-71-7 (POCH, Польша) и аммоний использовали додекагидрат сульфата железа(III), чистый ра, CAS: 7783-83-7 (POCH, Польша). 1 М ацетатный буфер был приготовлен в нашей лаборатории путем смешивания 1,0 моль·л (CH3COOH) с 1,0 моль·л уксусной кислоты (CH3COONa) [1] ацетат натрия (оба реагента приобретены у Avantor[1] Performance Materials, Гливице, Польша) и доведения pH до желаемого значения с помощью 10 н. HCl. Все растворы, включая АК в концентрации 2,5 г/л , фенантролин в концентрации 4,0 г/л , сульфат аммония железа(III) в концентрации 2,5 г/л и 1 М ацетатный буфер (pH 4,6), были приготовленный на бидистиллированной воде. В экспериментах использовалось небольшое лабораторное оборудование, включающее точные аналитические весы (RADWAG, модель AS 60/220.XS, Радом, Польша), магнитную мешалку (WIGO, Тошек, Польша), автоматические пипетки различного объема (Eppendorf). , Германия), а также стеклянную посуду, например, мензурки и чашки Петри. Для корректировки значений pH буферного раствора использовали лабораторный pH-метр SevenCompact S210 (Mettler Toledo, Грайфензее, Швейцария). Все эксперименты проводились при комнатной температуре.

Растворы с различной концентрацией витамина С для прямой оценки цвета готовили, добавляя в химические стаканы сначала 9 мл дистиллированной воды, затем 1 мл ацетатного буфера, затем соответствующий объем аскорбиновой кислоты и 0,5 мл сульфата железа(III) аммония. . После ожидания в течение 3 мин добавляли 0,5 мл фенантролина. Добавки АК составляли 0, 20, 50, 80, 100, 150, 200, 300 и 400 мкл. На рис . 3 представлены фотографии образцов, подготовленных для дальнейших испытаний. Параметрами, изученными для оптимизации, были концентрация реагента, объем и время между отдельными этапами (влияние химической кинети Подождав около 20 мин, по 10 мл каждого окрашенного раствора разливали в чашки Петри с плоским дном и фотографировали на камеру смартфона.

4.2. Подготовка цветового шаблона и получение изображений

Эксперимент был разделен на несколько этапов, процедура показана на рисунке 4.

Первым шагом стала подготовка шаблона для стандартизации цвета. Чтобы определить, какие цвета следует использовать в качестве эталона, была приготовлена серия растворов витамина С, охватывающая диапазон концентраций, рассматриваемый в этом исследовании. Это было сделано для того, чтобы оценить диапазон значений RGB, значимых для этих расчетов. Были сделаны фотографии протестированных решений, и с помощью специального самодельного программного обеспечения были выбраны 12 конфигураций значений RGB (цветов) для использования в шаблоне. Усредненные значения RGB легли в основу определения цветового диапазона в шаблоне, посвященном решаемой задаче . Чтобы немного расширить цветовую палитру, были оценены дополнительные верхние и нижние значения, а шаблон создан на 12 цветов. Шаблон, в котором каждый цвет представлен одним квадратом, был распечатан в нескольких экземплярах на одном принтере. Размер каждого квадрата составлял 180 × 180 пикселей, а общий размер шаблона – 600 × 790 пиксе

Machine Translated by Google

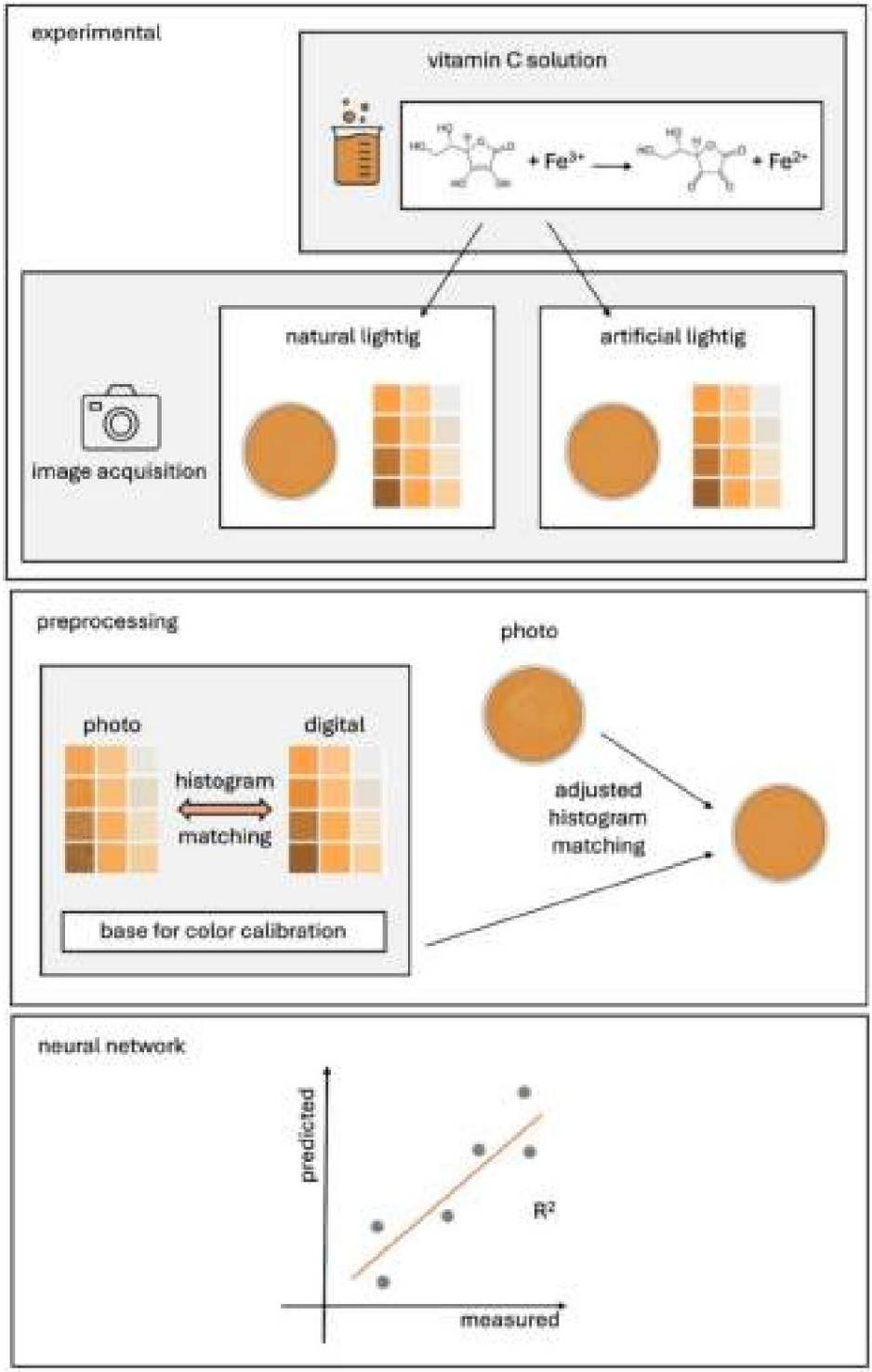


Рис. 4. Схема проведенных экспериментов и расчетов: приготовление растворов АК различной концентрации; нанесение растворов на чашки Петри; захват изображения при различных условиях освещения (решения АА с цветовыми шаблонами); предварительная обработка изображений — стандартизация цвета; обучение и оценка регрессионной сети.

Процесс съемки заключался в помещении 10 мл раствора в чашку Петри на белом фоне (выделяя контраст между фоном и чашкой Петри с раствором), рядом с шаблоном (служившим набором для контроля цвета) на черная поверхность. Снимки делались в двух условиях: естественном освещении и свете фонарика смартфона. Во втором варианте коробка полностью

Machine Translated by Google

изолирующую фотографируемые объекты от окружающей среды, помещали на экспериментальную установку, чтобы перекрыть доступ естественному свету. При этих двух источниках освещения наблюдались различия в цветах растворов и сфотографированных цветовых диаграммах (рис. 5). Процесс повторяли 11 раз для каждого образца с разной концентрацией витамина С.

Все фотографии были сделаны на смартфон Xiaomi (Xiaomi Redmi Note 9, Xiaomi, Пекин, Китай), оснащенный камерой на 48 Мп с диафрагмой f/1,79. Разрешение снимков составило 3984 × 1840 пикселей.

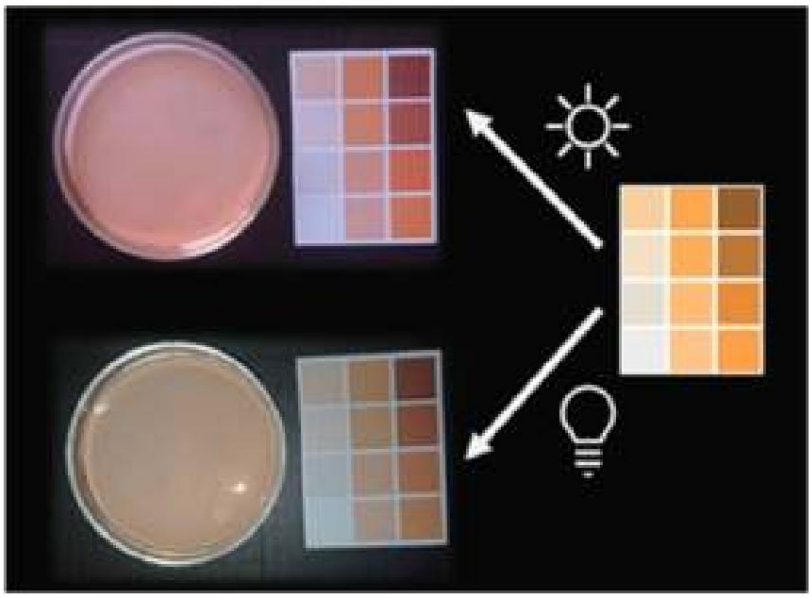
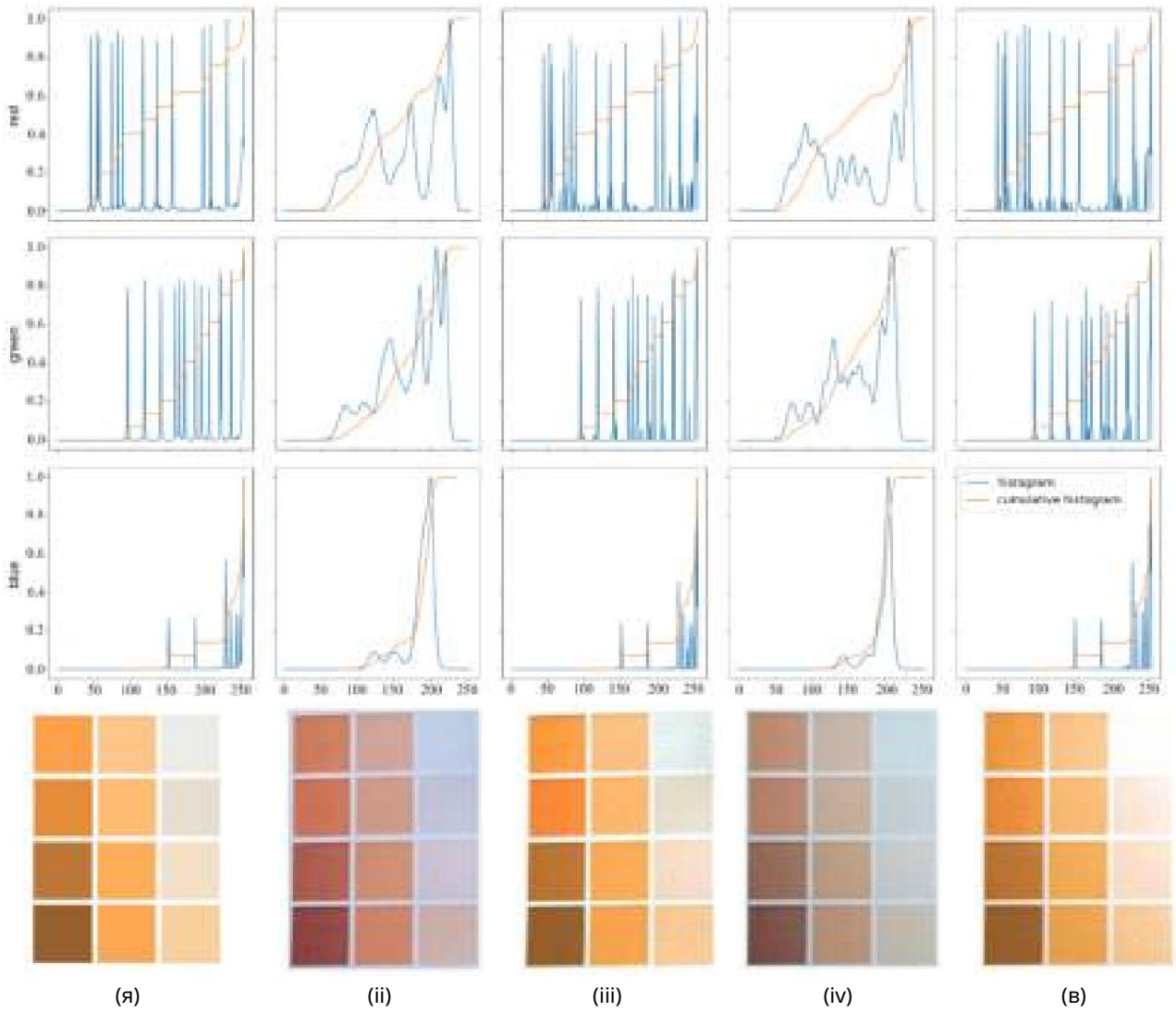


Рисунок 5. Управление условиями освещения — два подхода: естественный свет (вверху) и освещение лампой смартфона (внизу). Правая часть изображения — цифровой шаблон, используемый при сопоставлении гистограмм; стрелки указывают условия освещения, при которых были сделаны снимки.

Чтобы гистограмма соответствовала шаблонам, использованным при стандартизации, были сделаны дополнительные снимки шаблона в обоих условиях. Из изображений были вырезаны шаблоны и проведено сопоставление гистограмм между полученными шаблонами и исходным. Процесс с результатами можно увидеть на рисунке 6. Каждый шаблон представлен в виде RGB-изображения вместе с его гистограммами для каждого из каналов R, G и B. Этот рисунок наглядно иллюстрирует работу нашего алгоритма. В первом столбце показана идеальная гистограмма цифрового цветового шаблона, предназначенного для этой задачи. Ожидается , что цветовые гистограммы шаблонных фотографий будут соответствовать идеальным характеристикам. Однако в реальных условиях гистограммы фотографий и цифровой версии имеют существенные различия (второй и четвертый столбцы). Однако после применения алгоритма, предложенного в данной работе, можно воспроизвести ожидаемую форму цветовой гистограммы для всех трех компонентов: R, G и B. Графики в третьем и пятом столбцах соответствуют графикам в первом столбце.

Совокупные гистограммы совпадающих изображений такие же, как совокупная гистограмма цифрового шаблона (который служит эталоном) для каждого канала. Каждый пик на данной гистограмме соответствует одному компоненту определенного цвета на шаблоне. Менее дискретный вид гистограмм исходных шаблонных изображений вызван отклонениями от печати и различными переменными, которые влияют на то, как камера воспринимает объект с определенными изменениями (например, тенями и светлыми участками).

Machine Translated by Google



(я) (ii) (iii) (iv) (в)

Рис. 6. Шаблоны и их гистограммы для каналов RGB: (i) исходный цифровой шаблон, (ii) изображение шаблон при естественном освещении, (iii) изображение соответствующего шаблона при естественном освещении, (iv) изображение изображения шаблона при искусственном освещении и (v) изображения соответствующего шаблона при искусственном освещении.

Набор данных для расчетов был подготовлен следующим образом: было сделано 198 снимков. для девяти концентраций витамина С при двух разных условиях освещения. Это означает что для данного решения для каждого из условий было сделано по 11 снимков. Каждый из изображения прошли процесс стандартизации, как описано в разделе 3.1. Образцовый Эффект проиллюстрирован на рисунке 7. На рисунке 8 показан эффект применения алгоритма к различным представлены изображения раствора витамина С с концентрацией 33,63 мкг/мл. Этот привело к созданию наборов данных для двух условий освещения. Сочетание обоих комплектов, в результате чего был составлен комплект смешанного состояния, который состоял из:

- обучающий набор: 60 416 изображений.
- набор для проверки: 20 480 изображений.
- тестовый набор: 20 480 изображений.

Подготовленные наборы данных изображений служили входными данными для обучения регрессионного нейрона. сеть, с помощью которой была получена модель определения концентрации АК на основе сделанные фотографии.

Machine Translated by Google

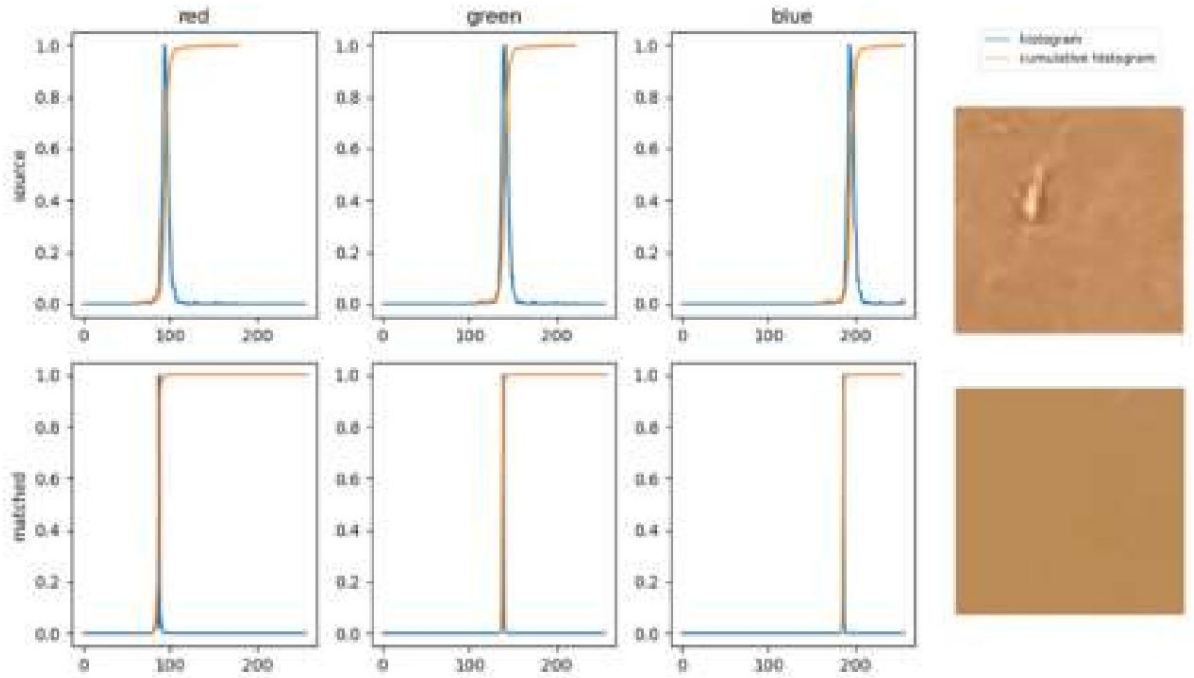


Рисунок 7. Изображения раствора витамина С (33,63 мкг/мл) в условиях искусственного освещения до (источник) и после (согласованное) применения алгоритма стандартизации и соответствующие гистограммы каналов RGB.

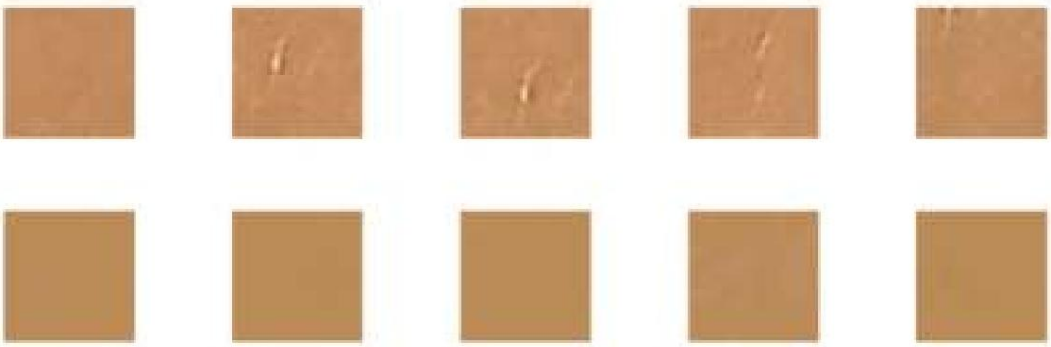


Рисунок 8. Различные типовые изображения раствора витамина С (33,63 мкг/мл) в условиях искусственного освещения до (вверху) и после (внизу) применения алгоритма стандартизации.

4.3. Сетевая архитектура

Модель многомерной регрессии была определена с использованием глубоких нейронных сетей с нетрадиционной архитектурой, специально предназначенных для этой задачи. При проектировании сети учитывалась важная информация, такая как цветовое моделирование, сходство цветов, соответствующих последовательным значениям целевой переменной, и примерно однородный цвет каждой фотографии.

Входной слой нейронной сети принимал изображения размером 50 × 50 и тремя цветовыми каналами (RGB), случайно выбранными из каждой фотографии. Впоследствии после каждого сверточного слоя применялись сверточные слои Conv2D с небольшими фильтрами 1 × 1, регуляризацией L2 и функцией активации GELU (линейная единица ошибки Гаусса). Также использовался слой нормализации, работающий вдоль оси признаков для каждого образца для стабилизации и улучшения процесса обучения. Архитектура также включала слои объединения, которые уменьшали пространственный размер карты объектов, включая AveragePooling2D и GlobalMaxPooling2D, которые уменьшали пространственное измерение до одного значения на карту объектов. Сочетание глобального объединения и выравнивания является стандартным методом уменьшения размеров перед полностью связанными слоями. На следующем этапе данные были преобразован

Machine Translated by Google

к полностью связанным плотным слоям с активацией GELU и регуляризацией L2, в то время как добавление слоя Dropout предотвратило переобучение. На выходе было значение, соответствующее к предсказанной концентрации раствора на изображении. Подробное описание сетевая архитектура представлена в Таблице 2 , а ее обзор показан на Рисунке 9. общее количество параметров составило 23 879, а тренировка за один цикл занимала от нескольких до нескольких секунды (в зависимости от варианта расчета).

Таблица 2. Архитектура и параметры нейронной сети, использованные в этом исследовании.

Слой (Тип)	Выходная форма	Количество параметров
Входной слой	(Нет, 50, 50, 3)	0
Конв2D	(Нет, 50, 50,3 2)	128
Активация	(Нет, 50, 50, 32)	0
Конв2D	(Нет, 50, 50, 64)	2112
Активация	(Нет, 50, 50, 64)	0
Среднее объединение2D	(Нет, 46, 46, 64)	0
Нормализация слоя	(Нет, 46, 46, 64)	128
Конв2D	(Нет, 46, 46, 64)	4160
Активация	(Нет, 46, 46, 64)	0
Среднее объединение2D	(Нет, 44, 44, 64)	0
GlobalMaxPooling2D	(Нет, 64)	0
Сгладить	(Нет, 64)	0
Плотный	(Нет, 150)	9750
Выбывать	(Нет, 150)	0
Плотный	(Нет, 50)	7550
Плотный	(Нет, 1)	51
Всего параметров: 23 879 (93,28 КБ) Обучаемые параметры: 23 879 (93,28 КБ) Необучаемые параметры: 0 (0,00 байт).		

В сверточном слое, таком как Conv2D, примененная регуляризация L2 работает на сверточные фильтры (ядра). Эти фильтры отвечают за извлечение функций из входные изображения. Добавление регуляризации L2 означает, что в процессе оптимизации функция потерь, которую модель пытается минимизировать, включает дополнительный член, который наказывает большие весовые показатели фильтров. Добавление регуляризации L2 к сверточному слою помогает предотвращает переоснащение, стабилизирует процесс оптимизации и способствует более единообразному и устойчивые решения, ведущие к лучшему обобщению модели.

Размер ядра 1 × 1 в сверточном слое имеет специфические и полезные свойства в нейронные сети. Фильтры 1 × 1 обычно используются для уменьшения пространственных размеров без пространственная агрегация, которая может быть полезна для уменьшения количества функциональных каналов или введение нелинейности без изменения пространственного размера. Эту конфигурацию часто называют «точечная свертка». Ядро 1 × 1 меняет количество выходных каналов (глубину) без изменения ширины и высоты входного изображения, что позволяет комбинировать информация, содержащаяся в разных входных каналах, без интеграции пространственной информации. Каждая выходная точка представляет собой линейную комбинацию значений входного канала в заданном пространственном пространстве. точка. Например, если в данной входной точке (x, y) у нас есть значения в каналах (w1, w2, w3), ядро 1 × 1 присваивает этим значениям разные веса и объединяет их, создание новых выходных каналов. Такое ядро имеет меньше параметров по сравнению с более крупными. сверточные ядра, что приводит к меньшему количеству вычислительных операций. Это эффективно в с точки зрения памяти и времени вычислений. При моделировании на основе RGB-компонент с использованием ядро 1 × 1 означает изучение взаимосвязей между отдельными компонентами цвета,

Machine Translated by Google

а не окрестности каждого пикселя. В нашей задаче не было необходимости обнаруживать мелкие
а затем более крупные детали, как в случае с распознаванием формы.

Функция активации GELU [47], применяемая в качестве альтернативы ReLU (выпрямленная линейная
активация в нейронных сетях, что обеспечивает лучшее поведение градиента.
и более эффективное обучение в глубоких нейронных сетях (рис. 10). Его непрерывность,
дифференцируемость и способность сохранять нелинейные характеристики данных делают его ценным инструмен
разработка моделей глубокого обучения. В отличие от ReLU, который может страдать от проблем с градиентным потоком,
называемая проблема исчезающего градиента), GELU обеспечивает лучший градиент.
Вентиляционное переоснащение стабилизирует процесс оптимизации и способствует более равномерной и равномерной
производительности благодаря своей структуре. GELU сохраняет нелинейные особенности, которые имеют решающее значение дл
изучение представлений данных, помогая моделям лучше улавливать сложные зависимости в
Размер ядра 1 × 1 в сверточном слое содержит конкретные и полезные данные для обучения свойств.



Рисунок 9. Архитектура нейронной сети, использованная в этом исследовании.

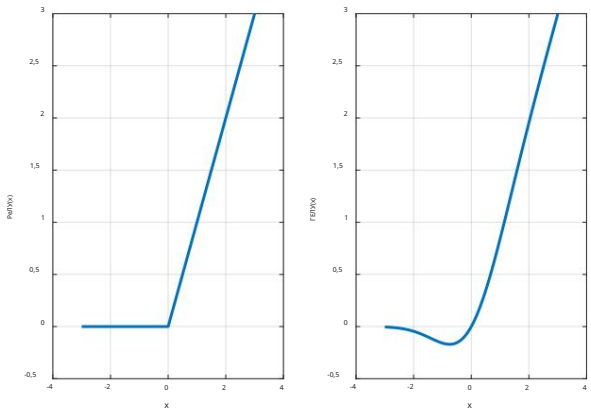


Рисунок 10. Функции активации ReLU и GELU.

Machine Translated by Google

На слое LayerNormalization нормализация выполняется вдоль оси объекта для каждого образца, что означает, что нормализация не зависит от других образцов в пакете.
Каждый вектор признаков (например, каналы изображения) нормализовался отдельно, учитывая только значения внутри этого вектора, а не всю партию. В отличие от BatchNormalization, которая работает в зависимости от размера пакета, LayerNormalization остается стабильной даже с небольшими пакетами или размером пакета в один. Нормализация может помочь предотвратить такие проблемы, как исчезновение или взрывной градиент градиентов, а также ускорить сходимость. Нормализация активаций помогает поддерживать значения в разумном диапазоне, стабилизируя процесс обучения. Когда активации нормализованы, оптимизатору (алгоритму обучения) легче корректировать веса сети, ускоряя сходимость к оптимальным значениям веса.

GlobalMaxPool2D — это слой, используемый в нейронных сетях для уменьшения размеров пространственных данных после применения сверточных слоев или других слоев, обрабатывающих пространственные данные, таких как слои объединения [48,49]. GlobalMaxPool2D работает путем выбора максимального значения каждого канала признаков по всей пространственной области (всем точкам) входного тензора признаков. GlobalMax-Pool2D — это эффективный инструмент уменьшения пространственной размерности, сохраняющий только наиболее значимые функции (самые высокие значения) из каждого канала объектов. Это уменьшает количество параметров модели, что может помочь уменьшить переобучение и слож
GlobalMaxPool2D не имеет обучаемых параметров; его единственная цель — выбрать максимальное значение из каждого канала объектов на карте объектов. Для каждого канала объектов слой GlobalMaxPool2D выбирает наибольшее значение по всей пространственной области (высоте и ширине). Результатом GlobalMaxPool2D является тензор уменьшенных размеров, содержащий только максимальные значения для каждого канала признаков. После применения GlobalMaxPool2D полученный тензор можно передать полносвязному слою или слоям других типов, обрабатывающим одномерные векторы данных.

В этом исследовании модель была составлена с использованием оптимизатора Адама с настроенными бета- параметрами и определенной функцией потерь, которая представляла собой сумму MSE и MAE. Когда функция потерь представляет собой комбинацию MSE и MAE, это влияет на то, как модель учится минимизировать ошибки. MSE наказывает за большие ошибки более строго (из-за квадратов разностей), тогда как MAE обрабатывает все ошибки одинаково. Объединение этих функций потерь означает, что модель будет направлена на уменьшение как больших, так и малых ошибок. Градиент MSE более крутой для более крупных ошибок, в результате чего модель быстрее обучается на значительных ошибках. Градиент MAE постоянен, что приводит к более стабильной скорости обучения при всех ошибках. MAE более устойчив к выбросам, чем MSE, поскольку он не включает в себя квадратичные различия. Функция потерь, сочетающая в себе MSE и MAE, может в некоторых случаях привести к более быстрой сходимости за счет использования сильных сторон обоих методов. MSE может ускорить сокращение крупных ошибок, а MAE может обеспечить стабильность и постоянный темп обучения.

Во время обучения модификация скорости обучения применялась после достижения стабильного уровня val_loss в течение пяти эпох. Начальное значение составляло 0,001. Подробная информация о параметрах обучения представлена в таблице 3.

Таблица 3. Параметры обучения сети.

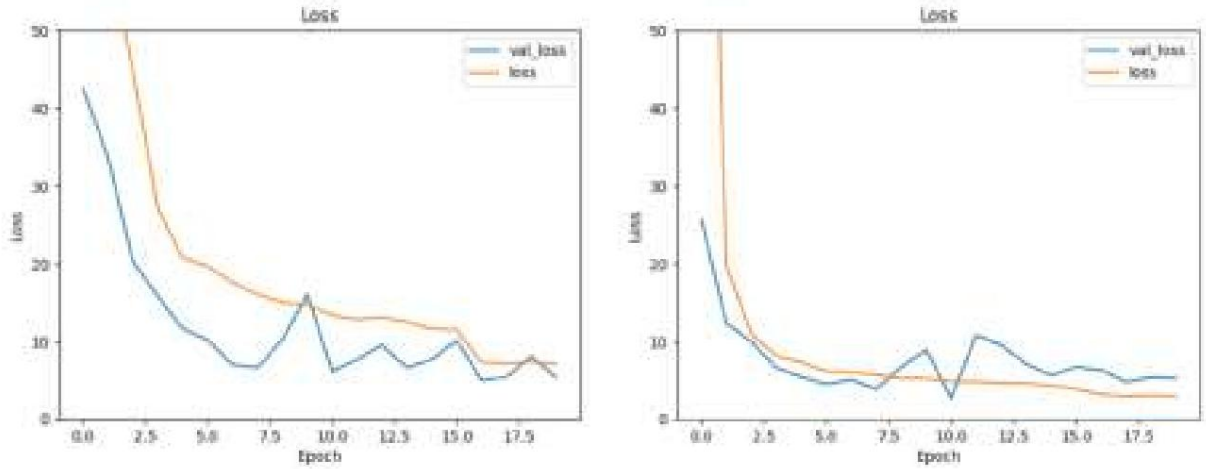
Параметр	Параметр
Размер изображения	50 × 50 × 3
Функция потерь	МСЭ + МАЭ
Оптимизатор	Адам
Начальная скорость обучения	0,001
Метрика	RMSE
Размер партии	30
Эпоха	20
Перетасовать	Каждая эпоха

Входной набор данных состоял из изображений размером 400 × 400 и был разделен на три части: обучающий, валидационный и тестовый наборы в соотношении 60/20/20. Во время обучения,

Machine Translated by Google

Из изображений динамически вырезались квадраты 50 × 50 , которые служили прямой вход на первый уровень сети.

В глубоком обучении было шесть различных вариантов тестирования для тренировки регрессии. алгоритм определения концентрации витамина С на основе изображений: оригинальные изображения для искусственное освещение, подобранные изображения для искусственного освещения, оригинальные изображения для естественного освещения, совпадающие изображения для искусственного освещения, исходные изображения для смешанного набора данных и совпадающие изображения. изображения для смешанного набора данных. Для каждого из них модель была обучена (раздел 3.2) и были сделаны прогнозы. Потери при обучении и проверке расчета искусственного освещения данные как для исходных изображений, так и для изображений, сопоставленных с гистограммой, представлены на рисунке 11 . диапазон был одинаковым в обоих случаях; однако колебания значений были меньшими для согласованных изображений, и в этом состоянии потери при проверке и обучении были более похожи друг на друга.



(я) (ii)

Рисунок 11. Потери при обучении и проверке для расчетов условий искусственной молнии для обоих исходные и (i) и сопоставленные с гистограммой (ii) изображения.

На рисунке 12 представлены результаты прогноза, сделанного с помощью представленной нейронной модели. сеть. Было три набора пар исходов: искусственная молния, естественная молния, и смешанные данные. Каждый набор повторялся как для исходных изображений, так и для обработанных. те. Наблюдалось улучшение более высоких значений концентрации для каждого из условиях (табл. 4). Стандартное отклонение прогнозов было ниже при применении предлагаемый метод стандартизации. Наибольшая разница наблюдается в случае искусственное освещение, которое может быть вызвано униформизацией отражений света, видимых в оригинальные изображения.

Таблица 4. Оценка моделей DL-регрессии с лучшими результатами для шести различных вариантов тестирования.

	Искусственный Освещение, Оригинал	Искусственный Освещение, Соответствует	Естественный Освещение, Оригинал	Естественный Освещение, Соответствует	Смешанный Освещение, Оригинал	Смешанный Освещение, Соответствует
Склон	0,9799	0,9961	0,9706	0,9452	0,9784	0,9825
Перехват	0,0036	0,2091	1.1888	0,6051	0,5832	0,4086
RMSE/мкг·мл-1	1,5769	0,7525	2.2900	1,9542	1,4828	1,3510
R^2 счет	0,9979	0,9999	0,9953	0,9956	0,9966	0,9968

Machine Translated by Google

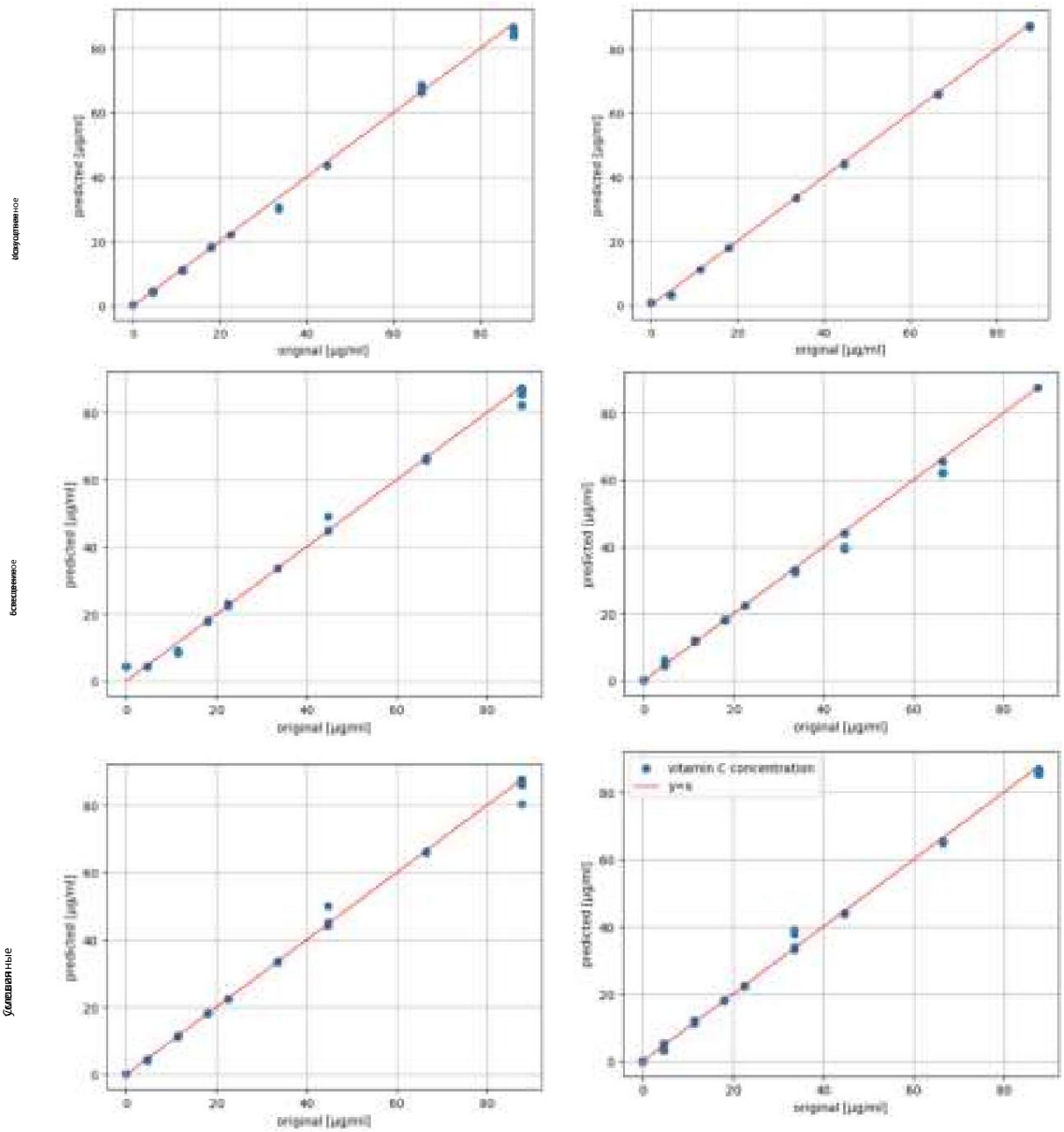


Рисунок 12. Оценка моделей регрессии DL для шести различных вариантов тестирования до (первый столбец) и после сопоставления цветов (второй столбец).

Значения коэффициентов оценки и корректировки численной модели представлены в таблице 4. Оценка R2 была очень высокой (выше 0,99) во всех случаях; однако в каждом случае P^2 оценка была выше при проведении стандартизации. Что касается значений среднеквадратических ошибок , то они были ниже для совпадающих изображений по сравнению с исходными. В зависимости от условий освещения скорость обучения менялась от исходного значения 10

4.4. Исследование

абляции. В этом исследовании мы также проанализировали, как различные модификации сетевой архитектуры повлияли на качество регрессионной модели. На первом этапе мы рассматривали сетевой ФФ , не учитывающий специфику задачи. Модель содержала шесть слоев: одну свертку, одно объединение, одно сглаживание и три плотных. Его обучали в тех же условиях, что и раньше. Подробные количественные результаты представлены в таблице 5. Использование

Machine Translated by Google

Более традиционная архитектура ухудшила способность модели соответствовать данным и их предсказательные возможности. Однако улучшения качества моделирования все же наблюдались. после применения алгоритма стандартизации цвета.

Таблица 5. Оценка моделей DL-регрессии, рассмотренных в исследовании абляции для шести различных варианты тестирования в зависимости от условий освещения и выбранного набора данных.

	Искусственный Освещение, Оригинал	Искусственный Освещение, Соответствует	Естественный Освещение, Оригинал	Естественный Освещение, Соответствует	Смешанный Освещение, Оригинал	Смешанный Освещение, Соответствует
Склон	0,9058	1,0138	0,9538	1,0071	1,0005	1,0112
Перехват	0,6891	0,3598	0,8347	0,2263	0,8747	0,0203
RMSE/мкг·мл-1	5.8415	1,7077	3.9504	2,5151	4,3488	2,8211
P^2 счет	0,9557	0,9962	0,9798	0,9918	0,9755	0,9897

Более того, перед финальной версией было внесено несколько различных изменений в архитектуру. была выбрана версия. Некоторые из них при сравнении с финальной версией оцениваются в Таблице 6.

Таблица 6. Оценка моделей регрессии DL, рассмотренных в исследовании абляции для смешанного освещения и согласованные условия тестирования в зависимости от изменений в модели.

Изменение в архитектуре	RMSE/мкг·мл-1	P^2 Счет	Сравнение
Размер пакета = 120	2,3535	0,9947	Снижение балла R2 и увеличение RMSE в сравнение с финальной моделью.
Размер изображения = 16	4,0146	0,9825	Значительное снижение балла R2 и увеличение RMSE по сравнению с окончательной моделью. Нижний спад начальной ставки по сравнению с окончательной версией.
Модификация функции потерь	3,8203	0,9856	Снижение показателя R2 и значительное увеличение RMSE по сравнению с окончательной моделью. Минимизированный различия между потерей обучения и валидации во время обучения.
Оптимизатор = Адаград	3.1625	0,9877	Снижение балла R2 и увеличение RMSE в сравнение с финальной моделью.
Оптимизатор = сингапурский доллар	-	-	Никаких изменений в стоимости убытков между эпохи. Модель не научилась предсказывать концентрация растворов витамина С.
Начальная скорость обучения = 0,01 или 0,0001	-	-	Никаких существенных изменений в результатах обнаружено не было. Достигнутые результаты существенно не различались сравнение с окончательными результатами модели.

5. Выводы

В данной работе представлен новый алгоритм стандартизации цвета, в котором основные этапы состоят из подготовки шаблона на основе исходной серии фотографий, разработки цвета принцип отображения с использованием выравненных цифровых гистограмм RGB идеального рисунка и его фотографии, преобразование цвета фотографий исследуемого объекта с использованием информации о сопоставлении гистограмм шаблонов и реконструкции изображений на основе преобразованных гистограмма. Алгоритм разработан для случаев, когда исследуемые объекты мало отличаются друг от друга. по цвету и цвет каждого сфотографированного объекта практически однороден. Операция предложенная процедура была продемонстрирована и проверена на задаче по аналитической химии, которая включало колориметрическое определение витамина С. Фотографии растворов АК, сделанные в контролируемых условиях не было заметной разницы в цвете в пределах концентрации диапазон. Исследования проводились в условиях дневного и искусственного освещения.

Machine Translated by Google

с целью объединить их в единый набор данных и получить модель, позволяющую определять витамин С независимо от условий освещения, используемых при съемке фотографий . Наборы стандартизированных изображений легли в основу глубокого машинного обучения с использованием архитектуры нейронных сетей в регрессионном варианте.

Для диапазона концентраций витамина С от 0 до 87,72 мкг·мл-1 среднеквадратичное отклонение для тестовый набор находился в диапазоне от 0,75 до 1,95 мкг·мл-1 по сравнению с нестандартизированным вариантом, где среднеквадратическое отклонение составляло от 1,35 до 2,29 мкг·мл-1 . Совместимость результатов прогнозирования концентрации с фактическими данными, выраженная R2 , составила более 0,99 для каждого из стандартизированных вариантов. Наибольшее улучшение отмечено в случае искусственного освещения, что можно объяснить устранением отражений от поверхностей растворов на изображениях после процесса стандартизации. Необходимы дальнейшие исследования, чтобы определить полезность алгоритма в различных сценариях (например, колориметрическое предсказание концентрации различных соединений). Этот новый подход может способствовать развитию новой области химического анализа, которая включает количественную оценку широкого спектра химических аналитов с использованием мобильных телефонов и анализа изображений. Представленное исследование показывает перспективность применения в нескольких областях науки и техники, таких как биомедицинская инженерия, химическая инженерия и пищевая промышленность.

Вклад автора: ПК: Концептуализация; Методология; Курирование данных; Формальный анализ; Расследование ; Программное обеспечение; Ресурсы; Визуализация; Письмо — первоначальный черновик; Написание — просмотр и редактирование. МД: Концептуализация; Формальный анализ; Приобретение финансирования; Расследование; Методология; Администрирование проекта ; надзор; Валидация; Письмо — первоначальный черновик; Написание — просмотр и редактирование. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Исследовательский проект, поддержанный программой «Инициатива передового опыта — исследовательский университет» Краковского университета AGH (ID1479).

Заявление о доступности данных. Алгоритмы, реализованные на Python, и образцы данных для расчетов доступны на платформе Github по адресу: https://github.com/pwkwiek/vitaminC. (дата обращения: 13 июля 2024 г.).

Благодарности: Авторы выражают благодарность Инз. Филипу Цьепеле (Университет AGH Кракова) за ценные рекомендации в ходе реализации проекта.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Ссылки

1. Fairchild, MD Модели цветового восприятия, 3-е изд.; John Wiley & Sons, Ltd.: Хобокен, Нью-Джерси, США, 2013 г.; ISBN 9781119967033.
2. Минц, П.С.; Сайни, К.С. Оценка калибровочной структуры RGB Cube и влияние калибровочных диаграмм на измерение цвета Сыр Моцарелла. J. Измерение пищевых продуктов. Характер. 2019, 13, 1537–1546. [Перекрестная ссылка]
3. Эрнст, А.; Папст, А.; Руф, Т.; Гарбас, Ю. Check My Chart: надежный трекер цветовых диаграмм для колориметрической калибровки камеры. В материалах 6-й Международной конференции по методам и приложениям совместной работы в области компьютерного зрения и компьютерной графики, MIRAGE'13, Берлин, Германия, 6–7 июня 2013 г.
4. МакКами, CS; Маркус, Х.; Дэвидсон, Дж. Г. Таблица цветопередачи. Дж. Прил. Фотогр. англ. 1976, 2, 95–99.
5. Суной С.; Игатинатан, К.; Салиендра, Н.; Хендриксон, Дж.; Арчер, Д. Калибровка цвета цифровых изображений для сельского хозяйства и Другие приложения. ISPRS J. Фотограмма. Дистанционное зондирование. 2018. С. 146. С. 221–234. [Перекрестная ссылка]
6. Ким, М.; Ким, Б.; Парк, Б.; Ли, М.; Вон, Ю.; Ким, Калифорния; Ли, С. Цифровое устройство согласования оттенков для определения цвета зубов с использованием машинного алгоритма опорных векторов. Датчики 2018, 18, 3051. [CrossRef]
7. Караимер, XK; Браун, М.С. Повышение точности цветопередачи на камерах. В материалах конференции IEEE Computer Society по компьютерному зрению и распознаванию образов, Солт-Лейк-Сити, Юта, США, 18–23 июня 2018 г.; стр. 6440–6449.
8. Рой, С.; Кумар Джайн, А.; Лал, С.; Кини, Дж. Исследование методов нормализации цвета гистопатологических изображений. Микрон 2018, 114, 42–61. [Перекрестная ссылка]
9. Чжао Ю.; Фергюсон, С.; Чжоу, Х.; Эллиотт, К.; Рафферти, К. Выравнивание цветов для относительного постоянства цвета с помощью нестандартных Ссылки. IEEE Транс. Процесс изображения. 2022, 31, 6591–6604. [Перекрестная ссылка]
10. Рашид Ф.; Джамайет, Северная Каролина; Фарук, ТН; Аль-Равас, М.; Барман, А.; Джохари, Ю.; Нурани, Тайвань; Абдулла, Дж. Я.; Эусуфзай, СЗ; Алам, М.К. Цветовые вариации при цифровой визуализации лицевых протезов, подвергнутых воздействию нефильтрованного окружающего света, и методы калибровки изображения в стоматологических клиниках: анализ in vitro. PLoS ONE 2022, 17, e0273029. [Перекрестная ссылка]
11. Барберо-Альварес, Массачусетс; Родриго, Дж.А.; Менендес, Дж. М. Адаптивная калибровка RGB с минимальной ошибкой в контексте колориметрической неопределенности для сохранения культурного наследия. Вычислить. Вис. Изображение Понимание. 2023, 237, 103835. [CrossRef]

Machine Translated by Google

12. Нур Азхар, М.; Бустам, А.; Насим, ФС; Шуин, СС; Мд Юсуф, МН; Хишамудин, НУ; Пох, К. Повышение надежности колориметрии мочи с помощью смартфона с использованием метода калибровки цветовой карты. Цифра. Здоровье 2023, 9, 20552076231154684. [Перекрестная ссылка]
13. Чжан Г.; Песня, С.; Панеску, Дж.; Шапиро, Н.; Даннемиллер, КЦ; Цинь Р. Новое системное решение для точной колориметрии Измерение с помощью дополненной реальности на базе смартфона. PLoS ONE 2023, 18, e0287099. [Перекрестная ссылка]
14. Чаират С.; Чайчули, С.; Диссанивейт, Т.; Вангкулангкул, П.; Конгпаничакул, Л. Оценка раневой ткани с помощью искусственного интеллекта с автоматической калибровкой цвета и измерений на изображениях, сделанных с помощью смартфона. Здравоохранение 2023, 11, 273. [CrossRef]
15. Суоминен Дж.; Егиазарян, К. Цветокоррекция камеры с использованием сплайнов. В материалах Международного симпозиума IS&T по электронной визуализации, Берлингейм, Калифорния, США, 21–25 января 2024 г.; стр. 165-1–165-6.
16. Суисси, М.; Чауш, С.; Мусса, А. Сопоставление цветов двухкомпонентных (ПЭТ/ПТТ) нитей с высокими характеристиками с использованием генетического алгоритма. наук. Отчет 2024, 14, 10949. [CrossRef]
17. Ваннасин, Д.; Гроссманн, Л.; МакКлементс, DJ Оптимизация внешнего вида продуктов растительного происхождения с использованием натуральных пигментов и Теория соответствия цветов. Пищевая биофизика. 2024, 19, 120–130. [Перекрестная ссылка]
18. Ву, Ю. Проект подбора цветов с помощью эталонных изображений на основе интерактивного генетического алгоритма. Дж. Электр. Сист. 2024, 20, 400–410. [Перекрестная ссылка]
19. Продовольственная и сельскохозяйственная организация; Всемирная организация здравоохранения. Потребности в витаминах и минералах в питании человека, 2-е изд.; ФАО/ВОЗ: Женева, Швейцария, 1998 г.; стр. 1–20, ISBN 9241546123.
20. Продовольственная и сельскохозяйственная организация; Всемирная организация здравоохранения. Потребности человека в витаминах и минералах; ФАО/ВОЗ: Женева, Швейцария, 2001.
21. Ликкесфельдт, Дж. О влиянии потребления витамина С на здоровье человека: как (неправильно) интерпретировать клинические данные. Редокс Биол. 2020, 34, 101532. [CrossRef]
22. Дозед, М.; Жирковск, Е.; Куёвск, Л.; Яворск, Л.; Пуров Дж.; Мерколини, Л.; Реми, Ф. Витамин С — источники, физиологическая роль, кинетика, дефицит, использование, токсичность и определение. Питательные вещества 2021, 615, 1–34.
23. Сунторнсюк, Л.; Грицанапун, В.; Нилкамханк, С.; Паочом, А. Количественное определение содержания витамина С в травяном соке с использованием прямого титрования. Дж. Фарм. Биомед. Анальный. 2002, 28, 849–855. [Перекрестная ссылка]
24. Климчак И.; Глищинска-Свигло, А. Сравнение методов UPLC и HPLC для определения витамина С. Пищевая хим. 2015, 175, 100–105. [Перекрестная ссылка]
25. Газдик З.; Зитка, О.; Петрлова Ю.; Адам, В.; Зеналек, Дж.; Хорна, А.; Резничек, В.; Беклова М.; Кизек, Р. Определение витамина С (аскорбиновой кислоты) с использованием высокоэффективной жидкостной хроматографии в сочетании с электрохимическим обнаружением. Датчики 2008, 8, 7097–7112. [Перекрестная ссылка]
26. Буначу, А.А.; Бакалум, Э.; Абул-Энейн, НУ; Удристиойу, Г.Э.; Флешин, С. ИК-Фурье-спектрофотометрический анализ аскорбиновой кислоты и биотина и их фармацевтических составов. Анальный. Летт. 2009, 42, 1321–1327. [Перекрестная ссылка]
27. Чжу, К.; Донг, Д.; Чжэн, Х.; Сонг, Х.; Чжао, Х.; Чен, Х.; Чен, Х. Хемилюминесцентное определение аскорбиновой кислоты с использованием металлорганических каркасов на основе оксида графена и меди в качестве катализатора. РСК Адв. 2016, 6, 25047–25055. [Перекрестная ссылка]
28. Берг, Р.В. Исследование L (+)-аскорбиновой кислоты с помощью рамановской спектроскопии в видимом и УФ-свете. Прил. Спектроск. Ред. 2015, 50, 193–239. [Перекрестная ссылка]
29. Ян, Х.; Ирудаярадж, Дж. Быстрое определение витамина С методами NIR, MIR и FT-Raman. Дж. Фарм. Фармакол. 2010, 54, 1247–1255. [Перекрестная ссылка]
30. Зыкова Е.В.; Сандетская, Н.Г.; Островский О.В.; Веровский В.Е. Методы анализа и контроля процесса определения аскорбиновой кислоты в лекарственных препаратах методами капиллярного зонного электрофореза и мицеллярного. Фарм. хим. Дж. 2010, 44, 463–465. [Перекрестная ссылка]
31. Додевска Т.; Хаджиев Д.; Штерев И. Обзор электрохимических микросенсоров для обнаружения аскорбиновой кислоты: клинический, Применение в фармацевтической и пищевой промышленности. Микромашины 2023, 14, 41. [CrossRef]
32. Хуанг, Л.; Тиан, С.; Чжао, В.; Лю, К.; Го, Дж. Электрохимические датчики витаминов: критический обзор. Таланта 2021, 222, 121645. [Перекрестная ссылка]
33. Бронцова, Г.; Прокопец, В.; Шишканова, Т.В. Потенциометрический электронный язык для фармацевтической аналитики: определение Аскорбиновая кислота на основе электрополимеризованных пленок. Хемосенсоры 2021, 9, 110. [CrossRef]
34. Коутиньо, МС; Мораис, CLM; Невес, АСО; Менезес, ФГ; Лима, КМГ Колориметрическое определение аскорбиновой кислоты на основе ее мешающего эффекта при ферментативном анализе глюкозы: подход с использованием анализа изображений смартфона. Ж. Браз. хим. Соц. 2017, 28, 2500–2505. [Перекрестная ссылка]
35. Порту, Иса; Сантос Нето, Дж. Х.; душ Сантос, Луизиана; Гомес, А.А.; Феррейра, SLC-определение аскорбиновой кислоты в натуральных фруктовых соках с использованием колориметрии цифровых изображений. Микрохим. J. 2019, 149, 104031. [CrossRef]
36. Конг, Л.; Ган, Ю.; Лян, Т.; Чжун, Л.; Пан, Ю.; Кирсанов Д.; Легин А.; Ван, Х.; Ван, П. Новый CD-спектрометр на базе смартфона для высокочувствительного и экономичного колориметрического обнаружения аскорбиновой кислоты. Анальный. Хим. Акта 2020, 1093, 150–159. [Перекрестная ссылка]
37. Ли, К.; Сюй, Х.; Ван, Ф.; Чжао, Ю.; Ши, Ю.; Чжао, Х.; Лю, Дж. Портативная платформа для смартфонов, интегрированная с датчиком флуоресценции на бумажной полоске для сверхчувствительного и визуального количественного определения аскорбиновой кислоты. Пищевая хим. 2023, 402, 134222. [CrossRef]
38. Чжаоа, В.; Каоа, П.; Чжуа, Ю.; Люа, С.; Гаоб, Х.-В.; Хуанг, К. Быстрое обнаружение содержания витамина С во фруктах и овощах Использование цифровой камеры и цветовой реакции. Ким. Ноябрь 2020 г., 43, 14:21–14:30. [Перекрестная ссылка]

Machine Translated by Google

39. Думанкас, Г.Г.; Рамасахаям, С.; Белло, Г.; Хьюз, Дж.; Крамер, Р. Методы хемометрической регрессии как новые и эффективные Инструменты исследования генетических ассоциаций. Анализ тенденций TrAC. хим. 2015, 74, 79–88. [Перекрестная ссылка]

40. Уолд, С.; Сьёстрем, М.; Эрикссон, Л. PLS-регрессия: основной инструмент хемометрики. Чемом. Интел. Лаб. Сист. 2001, 58, 109–130. [Перекрестная ссылка]

41. Ли, Б.; Моррис, Дж.; Мартин, Э.Б. Выбор модели для частичной регрессии наименьших квадратов. Чемом. Интел. Лаб. Сист. 2002, 64, 79–89. [Перекрестная ссылка]

42. Смола, А.Дж.; Шокопф, Б. Учебное пособие по регрессии опорных векторов. Стат. Вычислить. 2004, 14, 199–222. [Перекрестная ссылка]

43. Гудфеллоу И.; Бенджио, Ю.; Курвиль, А. Глубокое обучение; MIT Press: Кембридж, Массачусетс, США, 2016 г.; ISBN 9780262035613.

44. Латуильер, С.; Месехо, П.; Аламеда-Пинеда, Х.; Хоро, Р. Комплексный анализ глубокой регрессии. IEEE Транс. Шаблон Анальный. Мах. Интел. 2020, 42, 2065–2081. [Перекрестная ссылка]

45. Паскуаль, Л.; Гра, М.; Видаль-Бротонс, Д.; Альканьис, М.; Мартинес-Маньес, Р.; Рос-Лис, СП. Вольтамперометрический инструмент с электронным языком для эмуляции сенсорного анализа и дискриминации растительного молока. Сенсорные актуаторы B Chem. 2018, 270, 231–238. [Перекрестная ссылка]

46. Войчик, С.; Цепиела, Ф.; Якубовска, М. Анализ цветов образцов с помощью компьютерного зрения по сравнению с четырехдисковым иридий-платиновым вольтамперометрическим электронным языком для распознавания фальсификации натурального меда. Измер. Дж. Межд. Измер. Признался. 2023, 209, 112514. [CrossRef]

47. Ли, М. Математический анализ и оценка производительности функции активации GELU в глубоком обучении. Дж. Математика. 2023, 1, 4229924. [CrossRef]

48. Сикандар, С.; Махум, Р.; Алсалман, А.М. Новый гибридный подход к поиску изображений на основе контента с использованием Feature Fusion. Прил. наук. 2023, 13, 4581. [CrossRef]

49. Хасан, Массачусетс; Хак, Ф.; Сабудж, СР; Саркер, Х.; Гони, Министерство финансов; Рахман, Ф.; Рашид, М.М. Сквозная облегченная многомасштабная CNN для классификации рака легких и толстой кишки с интеграцией XAI. Технологии 2024, 12, 56. [CrossRef]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Подводные камни метаэвристического решения на основе стехиометрии Модели оптимизации метаболических сетей

Моника Фабиола Брионес-Баес 1,*, Лучано Агилера-Васкес 1, Нельсон Ранхель-Вальдес 1, Кристал Зунига 2,*, Ана Лидия Мартинес-Саласар 1 и Клаудия Гомес-Сантильян 1

1 Отдел исследований и исследований, Технологического института Сьюдад-Мадеро (TECNM), Лос-Мангос 89440, Мексика; luciano.av@cdmadero.tecnm.mx (Лос-Анджелес-Вирджиния); nelson.rv@cdmadero.tecnm.mx (NR-V.); ana.ms@cdmadero.tecnm.mx (АЛМ-С.); claudia.gs@cdmadero.tecnm.mx (CG-S.)

2 Факультет биологии, Государственный университет Сан-Диего, 5500 Campanile Drive, Сан-Диего, Калифорния 92182, США

* Переписка: d00070469@cdmadero.tecnm.mx (MFB-B.); czuniga2@sdsu.edu (Чехия)

Аннотация: Анализ баланса потоков (FBA) — это метод, основанный на ограничениях, который обычно используется для направления метаболитов через ограничивающие пути, которые часто включают такие состояния, как анаплеротические циклы, такие как циклы Кальвина, обратимые или необратимые реакции и узлы, где метаболические пути разветвляются. Этот метод может определить наилучшие условия для одного курса, но не работает при изучении путей прохождения нескольких представляющих интерес метаболитов. Недавние исследования метаболизма считают более естественным оптимизировать несколько метаболитов одновременно, а не только один; более того, они указывают на использование метаэвристики как на привлекательную альтернативу, которая расширяет возможности FBA для решения множества задач. Однако литература также предупреждает, что использование таких методов не должно быть диким. Вместо этого для достижения желаемых результатов необходимо провести тщательную настройку и отбор. В этой работе анализируется влияние на качество путей, построенных с использованием алгоритмов NSGAII и MOEA/D, а также нескольких новых моделей оптимизации; он проводит исследование двух тематических исследований: биосинтеза пигмента и узла метаболизма глутамата микроводорослей Chlorella vulgaris в трех условиях культивирования (автотрофных, гетеротрофных и миксотрофных), одновременно оптимизируя три метаболических посредника как независимые целевые функции. Результаты показывают разные характеристики NSGAII и MOEA/D, демонстрируя, что выбор модели оптимизации может сильно повлиять на прогнозируемые фенотипы.

Ключевые слова: клеточный метаболизм; ФБА; многокритериальная оптимизация; НСГАИИ; МОЭ/Д

check for updates

Цитирование: Брионес-Баэз, М.Ф.; Агилера-Васкес, Л.; Ранхель-Вальдес, Н.; Зуньига, К.; Мартинес-Салазар, Алабама; Гомес-Сантильян, К. Подводные камни метаэвристического решения стехиометрической оптимизации Модели метаболических сетей. Алгоритмы 2024, 17, 336. https://doi.org/10.3390/a17080336 .

Академические редакторы: Массимилиано Карамия и Фрэнк Вернер

Поступила: 30 марта 2024 г.

Пересмотрено: 1 июля 2024 г.

Принято: 16 июля 2024 г.

Опубликовано: 1 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Микроводоросли – это фотосинтезирующие клеточные микроорганизмы, известные с незапамятных времен. Их можно выращивать как в сточных водах, так и в чистых или соленых водах; некоторые штаммы, такие как Dunaliella salina, можно выращивать в соленой воде, а другие штаммы, такие как Chlorella vulgaris, выращиваются в пресной воде и могут выдерживать высокие температуры выращива Микроводорослям необходим источник углерода для осуществления фотосинтеза и производства биомассы. Углерод можно получить в виде CO2 из загрязняющих источников, и они преобразуют его в кислород, тем самым помогая уменьшить глобальное потепление [1].

Микроводоросли являются крупными производителями биомассы; внутри находятся такие метаболиты, как липиды , которые в будущем можно будет использовать в качестве биотоплива, аминокислоты и пигменты, которые в настоящее время используются в фармацевтической и косметической промышленности, а также белки, которые используются в качестве пищевых добавок [2,3]. Кроме того, микроводоросли находят биотехнологическое применение в качестве источника биовосстановления качества воды и используются в качестве альтернативы для удаления тяжелых металлов, поскольку некоторые штаммы, такие как Chlorella vulgaris и Scenedesmus obliquos, способны поглощать тяжелые металлы, такие как кадмий. Cd) и свинец (Pb) [4,5].

Все перечисленные характеристики делают изучение метаболизма микроводорослей привлекательным для метаболической инженерии. Эта дисциплина посвящена изучению топографии

Machine Translated by Google

сети, регуляция метаболического пути, выявление узких мест, определение метаболических потоков и устранение побочных реакций путем делеции гена [6].

В частности, различные методы метаболической инженерии использовались для анализа путей и оптимизации потоков для управления метаболизмом и модификации потоков для получения желаемого продукта и, таким образом, для увеличения коммерческой ценности. Однако метаболические потоки невозможно измерить in vivo; по этой причине для их измерения или прогнозирования необходимы подходы моделирования [7]. Среди них есть однокритериальные подходы, основанные на ограничениях, такие как FBA (анализ баланса потоков), точные математические многокритериальные подходы и эвристические подходы.

FBA — один из наиболее часто используемых методов изучения клеточного метаболизма. Это одноцелевой подход FBA, основанный на ограничениях. Этот подход широко используется при анализе потоков метаболических сетей, поскольку его можно использовать, даже если кинетические данные недоступны, но он требует информации о стехиометрических данных реакций, присутствующих в сети, требованиях роста и конкретных параметрах. методы измерения биологической системы, в частности реконструкция метаболической сети в масштабе генома [8] , включающая все известные реакции, присутствующие в изучаемых организмах, и гены, кодирующие каждый фермент [9].

Клеточный метаболизм в метаболическом моделировании описывается как совокупность химических реакций, присутствующих в организме. Математически это представляется стехиометрической матрицей S размером (m × n), где n — реакции, а m — метаболиты, участвующие в каждой реакции, с присвоением отрицательного коэффициента, если он представляет собой реагент, и положительного коэффициента , если он представляет собой реагент. продукт, а нулевой коэффициент означает, что метаболит не участвует в реакции; каждая реакция будет иметь нижнюю и верхнюю границы, ограничивающие пространство решений или максимальное и минимальное значение разрешенного потока. ФБА ищет линейная оптимизация целевой функции; эта функция представляет собой линейную комбинацию потоков, которая обычно представляет производство биомассы [10].

$$\begin{aligned} &\text{FBA } \max F(v) = v_{\text{биомасса}} \\ &\quad \text{С учетом} \\ &\qquad C \cdot v = 0 \\ &\qquad LB_j \quad v_j \quad UB_j, \quad j \quad \{1, \ldots, н\} \end{aligned} \tag{1}$$

Уравнение (1) определяет соответствующую задачу линейной оптимизации FBA [11], где v — вектор потока через реакции. Стехиометрическая матрица Sm×n представляет метаболическую сеть, где существует метаболит в каждой строке и реакция в каждом столбце. Величина ячейки Sij представляет собой стехиометрический коэффициент метаболита i, участвующего в реакции j [9], а LBj и UBj — нижнюю и верхнюю границы разрешенных потоков в метаболической системе. Стационарное предположение устанавливается при Sv = 0 [12].

Его универсальность привела к тому, что ФБА широко использовался в различных организмах, включая микроводоросли, бактерии, консорциумы микроорганизмов и т. д. Примером этого является предсказание роста клеток цианобактерий Synechocystis в [13]; его также использовали при разложении глюкозы путем анаэробного расщепления для прогнозирования распределения метаболитов и выявления трансформации углерода с целью оценки превращения этанола, пропионовой кислоты и масляной кислоты в уксусную кислоту [14]. FBA послужил исследованием в медицине, в котором активность потока рассчитывалась для изучения различий в метаболических путях и сравнения подтипов рака молочной железы [15].

Методика FBA была использована для оценки метаболических потоков в различных штаммах микроводорослей с использованием трех гетеротрофных и миксотрофных автотрофных условий роста. Первыми микроводорослями, использовавшими ФБА, были микроводоросли Chlamydomonas Reinhardtii [16], с помощью которых была получена первая метаболическая карта. Позднее микроводоросль Chlorella vulgaris была использована для изучения продукции липидов [17]; в обоих случаях единственной целевой функцией было производство биомассы .

Хотя этот подход широко используется, распределение метаболитов по путям обусловлено метаболитами, присутствующими в уравнении целевой функции и в экспериментальных параметрах; блокирование некоторых метаболических путей приводит к

Machine Translated by Google

распределение потоков с нулевыми значениями. Более того, FBA широко используется в поисках максимизации продукции интересующих соединений, но клеточный метаболизм в его естественном состоянии не направляет метаболические пути к производству конкретного метаболита . В поисках лучшего понимания метаболизма микроводорослей появились новые методы оптимизации, такие как метаэвристика для многокритериальной оптимизации [18].
Эти методы направлены на более равномерное распределение и ближе к реальности того, что происходит на метаболическом уровне, за счет одновременной оптимизации различных функций с противоречивыми целями.

Многокритериальная оптимизация обычно основана на поиске решений различных конфликтующих задач, которые необходимо оптимизировать одновременно. Многокритериальная оптимизация имеет большое значение и осуществляется на технологическом и научном уровне.
Сообщалось о некоторых примерах в химической промышленности по оптимизации процессов действующих установок, биопереработке, реакционной технике, профилактике и контроле и т. д. [19].
Он также используется в биологии и медицине [20] и в метаболической инженерии [18].

Многокритериальная оптимизация контрастирует с Cobrapy FBA с открытым доступом, который максимизирует или минимизирует только одну целевую функцию и позволяет получить только одно решение. При многокритериальной оптимизации получается набор решений. Полученные решения называются недоминируемыми, поскольку ни одно другое решение в пространстве поиска не является лучшим, чем другие, когда все цели рассматриваются одновременно. Этот набор решений известен как оптимальные по Парето решения [21].

Среди методов, используемых для многокритериальной оптимизации микроорганизмов, выделяются эволюционные алгоритмы, такие как NSGAII [18], MOMO, основанные на биообъективной модели, и точные математические методы, требующие большого количества вычислительных ресурсов [22]. .

Метаэвристические алгоритмы возникают в результате естественной эволюции биологических групп; они являются частью искусственного интеллекта и рождаются из естественных вычислительных и эвристических методов (алгоритмов частичного поиска). По сравнению с математическими методами, требующими больших вычислительных затрат, эти методы обеспечивают достаточно хорошие решения алгоритма оптимизации с приемлемым вычислительным временем и пространством [23].

Многоцелевые эволюционные алгоритмы (MOEA) широко признаны в научном сообществе как подход к решению задач многокритериальной оптимизации.
В частности, NSGAII (многоцелевой ЭА на основе недоминируемой классификации), предложенный [24], оказался весьма эффективным при решении двух или трех задач [25,26].
Эволюционный алгоритм MOEA/D, основанный на декомпозиции, тестируется, поскольку его особенностью является правильная работа при наличии более трех целевых функций.

Ранее метаэвристические методы исследования метаболизма применялись в работе [18].
Алгоритм NS- GAII был разработан для оптимизации трех целевых функций (белков, углеводов и CO2) в метаболической сети микроводоросли Chlamydomonas Reinhardtii с использованием алгоритма NSGAII, схемы кодирования, основанной на анализе баланса потоков (FBA). Однако алгоритмическое решение может быть разным для каждой задачи оптимизации, и эта разница может увеличиваться в зависимости от того, насколько хорошо используемая математическая модель объясняет изучаемые явления. Метаболические сети не могут быть освобождены от таких проблем, и в этих работах анализируются отдельные алгоритмы (NSGAII и MOEA/D) и отдельные модели оптимизации метаболических сетей (четыре подхода многокритериальной оптимизации) с точки зрения качества решений, которые могут быть достигнуты. ими, а также удобство предоставляемой информации.
Исследование проводится на двух тематических исследованиях, которые включают сложные условия, включающие циклы, бифуркации, а также обратимые и необратимые реакции. Основной вклад этой работы включает разработку трех новых многокритериальных моделей оптимизации метаболических сетей, одного нового алгоритмического решения, основанного на декомпозиции, а также анализ, который помогает правильно идентифицировать метаэвристику и модели для решения процесса оптимизации метаболизма.

Остальная часть документа структурирована следующим образом. В разделе 2 описываются метаэвристики, используемые для анализа в этом исследовании; в частности, в нем представлены общее определение и составляющие. В разделе 3 подробно описаны новые модели оптимизации для

Machine Translated by Google

метаболические сети, предложенные в этой работе. В разделе 4 обсуждаются оригинальные функции, включенные в конструкцию NSGAII и MOEA/D, используемые для решения предложенных оптимизационных моделей. В разделах 5 и 6 описывается план экспериментов, проводимых для проверки предложенных оптимизационных моделей и их метаэвристических решений; он содержит определения случаев исследований, а также экспериментов и результатов, завершаясь кратким обсуждением наблюдаемых данных.

2. Материалы и методы.

В этой работе алгоритмы NSGAII и MOEA/D рассматриваются как рассматриваемая метаэвристика для решения проблемы оптимизации метаболических сетей. Общее представление о разработке таких алгоритмов представлено в оставшейся части этого раздела.

2.1. Алгоритм NSGAII

NSGAII — это многоцелевой эволюционный алгоритм (MOEA), который использует недоминируемую сортировку и дистанцию скученности для оказания селективного давления в сторону фронта Парето [24]. Метаэвристика развивает начальную популяцию родителей P0 , используя общие вычислимые генетические операторы, такие как мутация, скрещивание и турнирный отбор, которые создают новое потомство Qt в каждом поколении. Чтобы поддерживать элитарность, текущая популяция Pt объединяется с потомком Qt , а новая популяция Pt+1 выбирается на основе рангов недоминирования и дистанции скученности, чтобы диверсифицироваться и разорвать связи, когда это необходимо.

Тремя основными компонентами NSGAII являются быстрый метод сортировки без доминирования, быстрая процедура оценки расстояния с скоплением и простой оператор сравнения с скоплением. Общий метод, полученный из NSGAII, можно изобразить следующим образом:

1. Инициализация популяции Po размера N с использованием равномерного распределения.
2. Создайте потомскую популяцию Qt, используя бинарный турнирный выбор на основе оператора сравнения-скопления, скрещивания и мутации, выполняемой в родительской популяции (Pt), где индекс «t» обозначает количество поколений. Потомковая популяция и родительская популяция объединяются, образуя всю популяцию Rt , Rt = Pt + Qt . Популяция Rt будет иметь размер 2N.
3. Выполните быструю недоминируемую сортировку для всей совокупности Rt, чтобы идентифицировать различные фронты целевых функций. F = f ast недоминируемый sort(Rt), где F = (F0, F1, F2, ...) будет иметь в F0 недоминируемый набор решений Rt , который лучше всего аппроксимирует границу Парето.
4. Построить новую родительскую популяцию (Pt+1) размера N из полученных фронтов (Fi). Эта популяция размера N теперь используется для отбора, скрещивания и мутации для создания новой популяции (Qt+i) размера N.
5. Процесс необходимо повторять до тех пор, пока не будет достигнуто максимальное количество итераций.

2.2. МОЭ/Д

MOEA/D — это стратегия, основанная на разложении задачи многокритериальной оптимизации MOP (как определено в [27]) на определенное количество подзадач скалярной оптимизации, которые оптимизируются одновременно. Каждая подзадача оптимизируется с использованием информации, поступающей исключительно из соседних подзадач, что обеспечивает меньшую вычислительную сложность в каждом поколении. Существует несколько подходов к преобразованию многокритериальной задачи в скалярное число оптимизационных задач. Одним из наиболее популярных подходов является MOEA/D, предложенный в [28], где задачи скалярной оптимизации можно сформулировать следующим образом:

$$\text{г}^{\text{тэ}}(y \mid wi, z) = \max\{wi \mid fk(x) \quad zk \mid\}, \quad \text{При условии: } x \quad \Omega \tag{2}$$

где w = (w1, w2, ... , wN) – вектор весов и wi 0 для всех i = 1, . . . , k и множество z = {z1, z2, . . . , zk} — точка отсчета, где zi = max{ fi(x) | × ω} для i = 1, . . . , м.

Machine Translated by Google

Для каждой оптимальной по Парето существует вектор весов w, где x является оптимальным точке x решение 2.

MOEA/D работает в течение определенного количества поколений, и в течение каждого поколения он также оказывает селективное давление в сторону фронта Парето, используя генетические операторы, такие как мутация и скрещивание. Ключевым элементом этой стратегии является то, что потомство заменяет родителей на основе скалярной функции и их близости на каждой итерации, пока не будет достигнут последний фронт. доставляется по алгоритму.

3. Предлагаемые подходы. В

этой работе предлагаются три новые модели оптимизации метаболических сетей, которые расширяют FBA до многоцелевой задачи оптимизации. Модели, получившие названия MOFBA2, MOFBA3 и MOFBA4, представляют собой усовершенствования по сравнению с MOFBA1, предложенной в [18] и изображенной в уравнении (3). MOFBA1 одновременно оптимизирует набор биопродуктов {vb1 , . . . vbm} вместо одного, удерживает потоки реакций в пределах границ и удовлетворяет условию установившегося состояния, т. е. обеспечивают Sv = 0 (где S — стехиометрическая матрица , а v — вектор потоков).

$$\begin{aligned} &\text{MOFBA } \max F(v) = \{vb1 , \ldots , вбм \} \\ &\quad \text{С учетом} \\ &\qquad C \cdot v = 0 \\ &\qquad LBj \quad vj \quad UBj , \quad j \quad \{1, \ldots , н\} \end{aligned} \tag{3}$$

MOFBA1 — это задача оптимизации, возникающая в результате непосредственной реализации задачи, определенной в уравнении (3). Он учитывает столько целевых функций, сколько наборов интересующих метаболитов. Аналогичным образом, он учитывает столько переменных решения, сколько потоков реакций необходимо для определения метаболической системы. Обратите внимание, что для подхода к оптимизации пространство поиска зависит от переменных решения, и на основе этого определения существует по одному для каждого возможного потока, т. е. метаэвристика должна искать правильные значения потока в пределах предоставленных границ n различных переменных решения.

Условия, описанные в предыдущем абзаце, характеризуют распространенную ошибку при разработке стратегий решения задач оптимизации. Трудность возникает из- за того, что метаэвристика может потребовать большего времени для поиска возможных решений, когда число переменных решения велико. Эта работа рассматривает эту ситуацию и предлагает три новые модели оптимизации метаболических сетей, которые уменьшают пространство поиска (т.е. количество переменных решения, которые метаэвристика использует в поиске). Эти модели интегрируются в соответствующий экспериментальный план, чтобы продемонстрировать, что для целей качества имеет значение, какую модель выбрать для решения определенных проблем.

Хотя подробности экспериментов представлены в дальнейших разделах, оставшаяся часть этого раздела содержит подробное описание трех новых MOFBA, предложенных в этой работе, с кратким описанием их актуальности и влияния в конце.

3.1. МФБА2

В MOFBA2, как и в MOFBA1, целевыми функциями, подлежащими оптимизации, являются наборы интересующих метаболитов; следовательно, количество целей m одинаково. С другой стороны, MOFBA2 рассматривает сокращенный набор переменных решения, состоящий только из потоков реакций {vb1 , . . . vbm} , связанный с теми же интересующими метаболитами, присутствующими в целевой функции, и дополнительный v, указывающий, какой из интересующих метаболитов ведет поиск. Другими словами, MOFBA2 имеет m + 1 переменных решения вместо n. Уравнение (4) формально определяет MOFBA2.

$$\begin{aligned} &\text{MOFBA } \max F(v) = \{vb1 , \ldots , вбм \} \\ &\quad \text{С учетом} \\ &\qquad FBA(vbk, v) \text{ осуществима} \\ &\qquad LBj \quad vb\,i \quad UBj , 1 \quad i \quad \{1, \ldots , м\} \\ &\qquad\quad k \quad m \end{aligned} \tag{4}$$

Machine Translated by Google

MOFBA2 описывает двухуровневую модель оптимизации, в которой внутренняя модель оптимизирует ведущий интересующий метаболит v с использованием FBA и ограничивает границы интересующих метаболитов теми, которые определены во внешней модели. Границы остальных реакций предполагается известными и фиксированными в соответствии с анализируемой метаболической сетью.

3.2. МФБА3

MOFBA3 предлагает суррогатную модель для оптимизации интересующих метаболитов. Суррогатная модель направлена на улучшение двух известных показателей: гиперобъема (HV) и дистанции поколений (GD). Эти индикаторы отражают, насколько хорошо решение сходится к фронту Парето. В то время как HV необходимо максимизировать, GD необходимо минимизировать.

MOFBA3 имеет m + 1 переменную решения, такую же, как и MOFBA2, т. е. потоки реакций {vb1 , . . . vbm} , связанный с теми же метаболитами, представляющими интерес, которые присутствуют в целевой функции, и потоке ведущих метаболитов v. С другой стороны, количество целей всегда равно 2, независимо от того, сколько метаболитов, представляющих интерес, рассматривается. Отличительной особенностью этой модели является ее суррогатность; вместо прямого поиска правильных значений потока интересующих метаболитов он использует показатели эффективности в многокритериальном контексте (т. е. показатели HV и GD). Уравнение (5) формально определяет MOFBA3.

$$\begin{aligned} \text{MOFBA min } F(v) = {} & 1HV(\{vb1 , \ldots , vbm \}, R), DG(\{vb1 , \ldots , vbm \}, Zr) \\ & \text{С учетом} \\ & \text{FBA}(vbk, v) \text{ осуществима} \\ & LBj \quad vb\,i \quad UBj , \quad i \quad \{1, \ldots , м\}\ 1 \\ & k \quad m \end{aligned} \tag{5}$$

Эта задача оптимизации предполагает, что существует контрольная точка R и набор ссылок Zr . Учитывая, что управление FBA осуществлялось в рамках пакета COBRApy, с учетом его ограничений, ориентиром для данной работы является R = {1000, 1000, . . . , 1000}. Также, учитывая доступность реализации ФБА благодаря тому же пакету, эталонный набор Zr формируется набором оптимальных решений, образованных решениями, полученными при решении ФБА до оптимальности, имея каждый представляющий интерес метаболит в качестве оптимизированной биомассы.

3.3. МФБА4

MOFBA4 — последняя предложенная задача оптимизации, сочетающая в себе идеи MOFBA2 и MOFBA3. То есть предлагается оптимизировать не только интересующие метаболиты, но и показатели XB и ГД. Число переменных решения для этой модели остается равным m + 1, а количество целей — m + 3.

Уравнение (6) формально определяет эту задачу оптимизации.

$$\begin{aligned} \text{MOFBA min } F(v) = {} & 1HV(\{vb1 ,..., vbm \}, R), DG(\{vb1 ,..., vbm \}, Zr)(vb1 ,..., vbm) \\ & \text{С учетом} \\ & \text{FBA}(vbk, v) \text{ осуществима} \\ & LBj \quad vb\,i \quad UBj , 1 \quad i \quad \{1, \ldots , м\} \\ & k \quad m \end{aligned} \tag{6}$$

3.4. В таблице

анализа 1 суммированы наиболее примечательные особенности моделей оптимизации, предложенных в этой работе, и сравнивается их с MOFBA1, предложенным в [18]. В новых моделях пространство поиска значительно сокращено, и некоторые из них используют в своих определениях информацию о конвергенции. Уникальные характеристики демонстрируют богатство моделей, которые можно разработать для решения конкретной задачи оптимизации.

Machine Translated by Google

Таблица 1. Соответствующие особенности моделей MOFBA для метаболических сетей.

Модель	№ Переменные решения	№ Цели	Суррогатное материнство
МФБА1	н	м	Нет
МФБА2	м + 1	м	Нет
МФБА3	м + 1	2	Да
МФБА4	м + 1	м + 2	Да

Предлагаемые задачи MOFBA невозможно решить с помощью традиционных линейных решателей, таких как FBA. Альтернативой является использование перечислительных схем или приближенных подходов, позволяющих один для получения решений, принадлежащих оптимальной границе Парето. В этом смысле данное исследование проанализировали использование метаэвристики, которая интегрирует FBA в процесс поиска, как подходящий подход к решению, поскольку они улучшают свою аппроксимацию к фронту Парето в каждую итерацию.

4. Метаэвристические планы

В этом разделе представлены конкретные детали, необходимые в этом исследовании для реализации метаэвристики NSGAII и MOEA/D для решения четырех задач оптимизации. проблемы MOFBA1, MOFBA2, MOFBA3 и MOFBA4. Эти метаэвристики основаны на структуры NSGAII и MOEA/D.

Рассматриваемые метаэвристики требуют определения следующих характеристик: (1) схемы кодирования; (2) функция оценки пригодности; (3) генетические операторы; и (4) ограничение стратегия управления. Метод инициализации популяции для обеих стратегий (NSGAII и MOEA/D) является случайным. Предлагаемая конструкция для остальных этих компонентов для обработки новые MOFBA подробно описаны в оставшейся части этого раздела. Адаптации романа для структуры NSGAII и MOEA/D включают умное вычислимое представление решения, связанные со схемами кодирования.

4.1. Схемы кодирования

В этой работе предлагается использовать отдельные наборы кодировок решений для каждого MOFBA (как определенные в уравнениях (3)–(6). Схемы кодирования включают определение структуры данных. это представляет собой решение метаболической сети. Скрипт, разработанный для экспериментов, находится в репозитории Github (https://github.com/multiobjectiveoptimization2). /МОФБА, доступ 24 июня 2024 г.).

Для MOFBA1 структура данных представляет собой вектор с действительным знаком V. Схема кодирования рассматривала метаболическую сеть MN, состоящую из набора реакций V и двух подмножеств. $B \subseteq M$, $B^{б} \subseteq V$, где $V^{M}_{Vb} =$, которые представляют собой реакции метаболитов интерес для лица, принимающего решения. Далее, пусть v = (v1, ... , vn) — вектор потока для V и предположим, что существуют начальные нижняя и верхняя границы LBi , UBi для каждого vi , $1 \le i \le n$. Затем схема кодирования W предлагает переопределить границы каждого связанного vi. с реакцией в V $^{M}_{Vb}$ с использованием двух значений (Ii , i). Новые лимиты рассчитываются как $LB^{новый}_{я} = Ii$ и $UB^{new}_{я} = (UBi - Ii)\ i + Ii$. Все остальные потоки сохранят свои пределы. без изменений. Другими словами, решение кодирует изменения границ для FBA для решения MN. с использованием заранее определенного биопродукта, который в этой работе предполагается $v1^{б}$. В результате кодирующий вектор W имеет размер O(n), асимптотический по числу реакций.

Для MOFBA2, MOFBA3 и MOFBA4 вектор размера m + 1 считается схема кодирования, где m — количество целевых функций. Первые m элементов вектора являются действительными переменными, значение которых представляет собой верхний предел потока для каждого метаболит, представляющий интерес в связанных реакциях. Дополнительный элемент — одноцелевой селектор оптимизации, который принимает значения от 1 до m, указывая, какой метаболит является будет оптимизирован в свою очередь. Если это обратимая реакция, значение в указанном переменная будет такой же для нижней границы (но отрицательной).

Machine Translated by Google

4.2. Функция оценки пригодности

Во всех задачах оптимизации MOFBA функции оценки пригодности (или FEA) считаются производным подмножеством набора, состоящего из значений интересующих потоков метаболитов, таких как метрики гиперобъема и расстояния между поколениями. Поскольку необходимая информация о биопродуктах связана со специфическими реакциями, пригодность решения, полученного метаэвристическим методом на MOFBA, оценивается с учетом значений их потоков. В MOFBA1 и MOFBA2 критериями или целевыми функциями, подлежащими оптимизации, будут b и de- потоки реакций, соответствующие интересующим биопродуктам $v_{b_1}, \ldots, v_{b_m}$). В MOFBA3 полученные гиперобъем и расстояние генерации определяются выбранным в V , отмеченным как (v из раствора, контрольная точка R и следующим образом: эталонный набор Zr , оптимизированная точка R — это наихудшее возможное экстремальное значение потоков, равное 1000 для любого представляющего интерес метаболита, учитывая его определение для FBA, в широко используемых платформах, таких как CobraPy. Набор Zr состоит из трех точек, которые включают в себя. оптимальные потоки, полученные при решении рассматриваемого случая методом FBA путем индивидуальной оптимизации каждого интересующего метаболита, поэтому при наличии n целевых функций Zr будет иметь мощность n. Стоит отметить, что при необходимости ведущего биопродукта; в MOFBA2 – MOFBA4 это значение выбирается исходя из значения одной из рассматриваемых переменных решения, как описано р

4.3. Генетические

операторы. Эти операторы создают новые решения путем динамического и случайного изменения значений переменных решения в существующих решениях. Этот выбор был обусловлен его успехом в решении задач, включающих переменные решения с действительными значениями [29]. Для NSGAII выбраны операторы мутации, скрещивания и простого, но надежного случайного выбора соответственно. Конкретные значения этих параметров взяты из литературы и приведены в таблице 2.

Таблица 2. Параметры реализации NSGAII.

Параметр	Ценить
Полиномиальная мутация	Вероятность = 1,0/d, где d — количество переменных решения. Индекс распределения = 20
SBXКроссовер	Вероятность = 100% Индекс распределения = 20
Критерий остановки	до достижения 100 000 оценок
Численность населения	100

Операторами, выбранными для MOEA/D для мутации и кроссовера, были полиномиальные мутации [30] с кроссовером путем дифференциальной эволюции. Стратегия отбора проста, но надежна, а в качестве функции агрегирования использовалось расстояние Чебышева. Более подробную информацию об операторах см. в [28]. Конкретные значения этих параметров взяты из литературы и приведены в таблице 3.

Таблица 3. Параметры реализации MOEA/D.

Параметр	Ценности
Полиномиальная мутация	Вероятность = 1,0/d, где d — количество переменных решения. Индекс распределения = 20
Дифференциальная эволюция	CR = 1 Ф = 0,5 K = 0,5
Критерий остановки	После завершения 100 000 оценок
Численность населения	100

Machine Translated by Google

4.4. Стратегия управления ограничениями.

В этой работе используется метод управления ограничениями, предложенный в [31], для создания избирательного давления в сторону возможных решений. По мере развития поколений в обеих метаэвристиках конкуренция между решениями всегда будет отдавать предпочтение выполнимому решению, несмотря на состояние недоминирования. В долгосрочной перспективе такая стратегия имеет тенденцию исключать неосуществимые решения из окончательного отчета об алгоритме. Многоцелевая оптимизация используется, когда одновременно необходимо оптимизировать несколько целей. Существует несколько многокритериальных эволюционных алгоритмов (MOEA), таких как NSGAII и MOEA/D. Хотя они используются для оптимизации многокритериальных задач, они существенно отличаются. NSGAII — это алгоритм классификации без доминирования , тогда как алгоритм MOEA/D основан на декомпозиции

5. План экспериментов В этом

подразделе представлен набор экспериментов, проведенных с целью проверки применения предлагаемых моделей оптимизации в качестве инструментов для улучшения понимания метаболизма микроводорослей. В области исследований эффективных решений многокритериальных задач были проведены эксперименты на двух сетях микроводорослей Chlorella vulgaris для оценки эффективности различных алгоритмов и соответствующих MOFBA. В подразделах представлены примеры исследований, план эксперимента и детали программного обеспечения, использованного при реализации, с целью проверки следующих гипотез.

Гипотезы 0 (H0). Это не имеет отношения к выбору модели и/или алгоритма решения для оптимизации потоков в метаболической сети.

5.1. Случаи

исследования По сравнению с предыдущим исследованием [18], в два тематических исследования были включены две сети: метаболизм глутамата и распределение потока пигментов микроводорослей Chlorella vulgaris ; были добавлены обратимые и необратимые реакции, представление обратимой реакции с использованием интервалов нижней границы потоков -1000 и верхней границы 1000, а необратимых - с интервалами от нижней границы 0 до верхней границы 1000. Кроме того, были включены узлы, где метаболиты разделяются на разные пути и циклы, которые часто присутствуют в метаболизме клеток.

5.1.1. Пример исследования 1: Метаболическая сеть Chlorella vulgaris

Метаболическую сеть микроводоросли Chlorella vulgaris [17] изучали с использованием алгоритмов NS-GAII и MOEA/D для трех различных условий культивирования: фотоавтотрофия (свет + компоненты), гетеротрофия (компонент) и миксотрофия (CO2 + свет + компонент) .). Среди соединений, которые использовались в качестве питательных веществ при выращивании, были добавки источников азота, таких как NO3 и NH4, а также сульфатов, таких как SO4, Fe2 и магний. Различные источники сельскохозяйственных культур влияют на выработку метаболитов. На следующем рисунке показано распределение пигментов в микроводоросли C. vulgaris.

В этом случае будет изучена часть сети распределения пигментов, поскольку распределение у микроводорослей, таких как C. vulgaris, имеет большое значение при изучении синтеза пигментов. Реакции, участвующие в обмене веществ, представлены в Приложении A.1, Таблице A1. Эта сеть включает в себя сложность обратимых и необратимых (FRDPth, GRDPth) реакций, узлов и циклов и показана на рисунке 1.

Machine Translated by Google

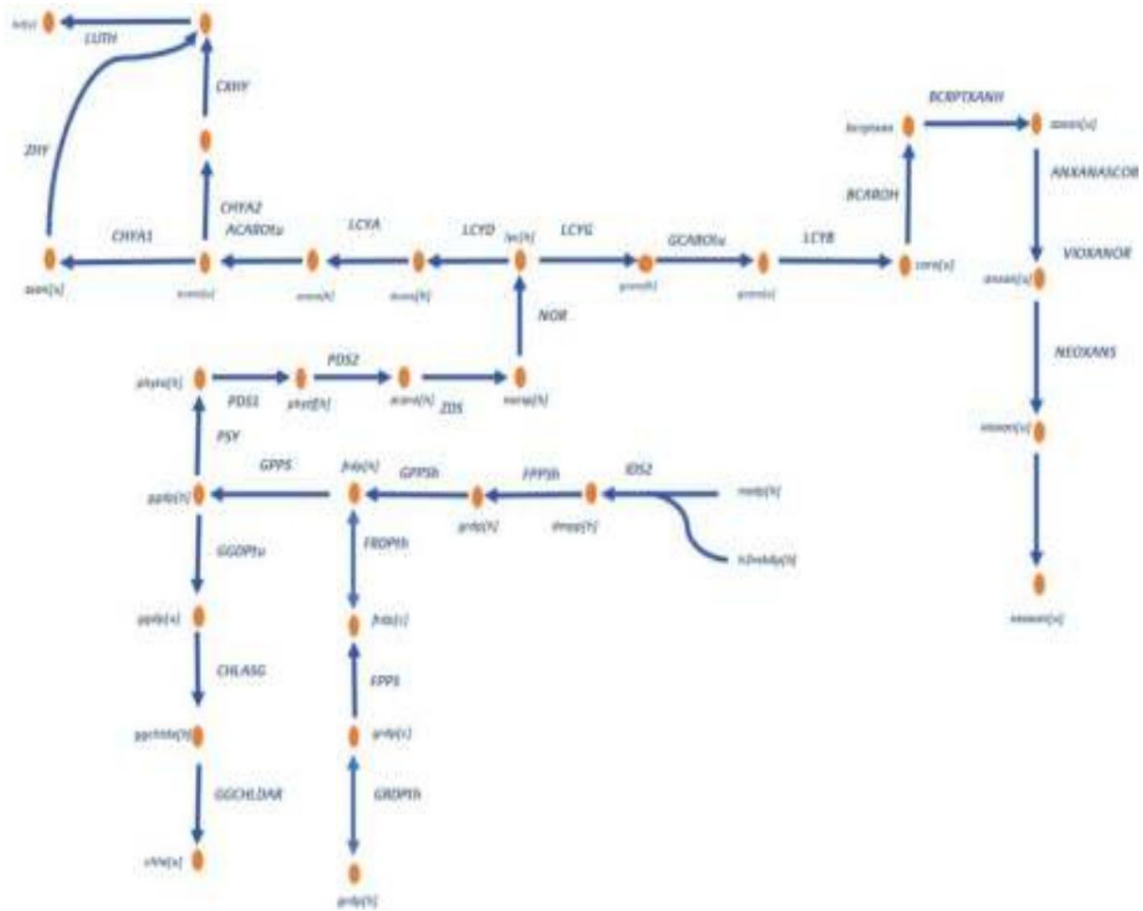


Рисунок 1. Распределение потоков путей биосинтеза пигментов [17].

5.1.2. Пример исследования 2: Многоцелевая оптимизация метаболической сети метаболизма Глутамат микроводорослей Chlorella vulgaris.

Метаболизм состоит из различных метаболических путей, которые переплетаются, образуя более сложный путь. Одним из таких путей является распределение потоков метаболизма глутамата, выполняющего различные функции, в том числе синтез аминокислот.

Эта метаболическая сеть представляет собой большую сложность из-за количества метаболитов , которые разветвляются в центральном узле, и наличия обратимых реакций, таких как ASPATh и ASPNA1Th. Эту сеть оценивали в трех различных условиях роста: автотрофии, гетеротрофии и миксотрофии, с использованием алгоритмов NSGAII и MOEA/D с четырьмя MOFBA. Этот метаболизм имеет большое значение, поскольку задействованы пути производства различных представляющих интерес продуктов, таких аминокислот, как тирозин, валин, лейцин и т. д., которые в дальнейшем могут быть использованы для производства белков. Реакции, участвующие в обмене веществ, представлены в Приложении A.1, Таблице A2. На рис. 2 представлено распределение потоков, связанных с метаболизмом глутамата, в хлоропластах и цитоплазм

Machine Translated by Google

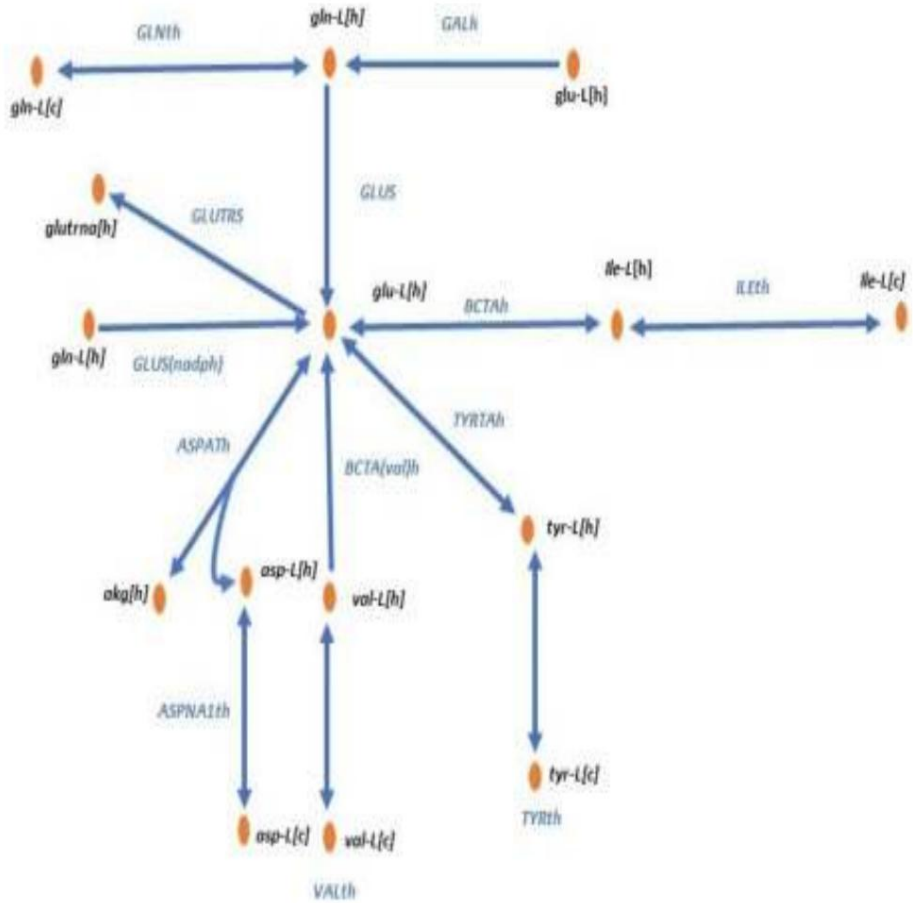


Рисунок 2. Распределение потоков, связанных с метаболизмом глутамата в хлоропластах и цитоплазме [17].

5.2. Определение эксперимента.

Возможно определение методологии, основанной на метаэвристике, которая улучшает информацию о потоках метаболической сети, предоставляемую методом FBA, путем переопределения анализа баланса потоков как многокритериальной задачи оптимизации; было предложено четыре эксперимента, которые обобщены в следующей Таблице 4.

Таблица 4. План экспериментов для демонстрации метаэвристической поддержки понимания метаболизма микроводорослей.

Цель эксперимента		Задействованные переменные
Эксперимент1	Использование разных моделей оптимизации дает разные результаты.	C. vulgaris, NSGAII, MOFA1 , MOFBA2 , MOFBA3 , MOFBA4 .
Эксперимент2	Продемонстрируйте, что использование метаэвристики способствует пониманию метаболизма микроводорослей.	C. vulgaris, ФБА, НСГАИИ, МОФБА4
Эксперимент3	Существуют алгоритмы оптимизации, более подходящие для решения конкретных задач.	C. vulgaris, NSGAII, MOEA/D, MOFBA4 .
Эксперимент 4. Убедитесь, что случайного выбора недостаточно.		C. vulgaris, НСГАII, ФБА, случайный.

Machine Translated by Google

Эксперимент 1 продемонстрировал, что разные модели многокритериальной оптимизации дают разные результаты. Алгоритм был установлен на NSGAII, микроводоросли были модифицированы, чтобы продемонстрировать универсальность подхода, и, наконец, были проанализированы четыре предложенные модели оптимизации. В изученных случаях было замечено, что неизменно лучшей моделью была MOFBA4, которая использовалась в последующих экспериментах.

В эксперименте 2 алгоритм, микроводоросли и однокритериальная модель FBA сравнивались с многокритериальной моделью, определенной MOFBA4. Было замечено, что метаболизм микроводорослей можно описать с помощью разных потоков, а не одного, и ими можно управлять для получения информации о различных представляющих интерес метаболитах. Это подтверждает легкость адаптации предложенных методов к разным типам метаболических сетей с учетом разных конфигураций.

Эксперимент 3 оценивал производительность алгоритмов NSGAII и MOEA/D при решении одной и той же задачи оптимизации. Было замечено, что для трех рассматриваемых целей (т.е. трех представляющих интерес метаболитов) NSGAII был лучшим. Этот результат согласуется с литературными данными, учитывая, что для двух или трех целей NSGAII показывает лучшие результаты, чем MOEA/D. Это оставляет открытым вопрос о том, улучшится ли MOEA/D для большего числа целей, что является открытым направлением исследований для его применения при изучении метаболических сетей микроводорослей.

Эксперимент 4 показывает, что простой случайной выборки недостаточно для получения лучшего результата. распределение решений, что возможно за счет использования метаэвристики.

Метаэвристики NSGAII и MOEA/D были реализованы с помощью фреймворка jMetalPy [32]. Модели оптимизации были разработаны на Python и использовались как часть реализации FBA, предоставляемой пакетом COBRApy [33]. Графика была воссоздана с использованием интерфейса pyplot библиотеки matplotlib [34]. Компьютер, используемый для проведения экспериментов, имеет 64-битный процессор с частотой 2,6 ГГц и 32 ОЗУ.

6. Результаты

В этом разделе обобщены данные, полученные в результате реализации экспериментов, описанных в разделе 3.2. В конце проводится обсуждение достигнутых целей исследования. Для визуализации результатов на рисунках 3–6 была импортирована библиотека matplotlib .

6.1. Эксперимент 1

En [18] показал, что NSGAII предоставляет решения лучшего качества, чем классический FBA для трех целевых функций. На рис. 3 представлены результаты эксперимента, в котором тестируются четыре описанных выше варианта MOFBA1, MOFBA2, MOFBA3 и MOFBA4 для алгоритма NSGAII, с оптимизацией трех целевых функций распределения потока пигментов микроводоросли Chlorella vulgaris. Можно заметить, что разные MOFBA ведут себя по-разному по отношению друг к другу; MOFBA1, MOFBA3 и MOFBA4 улучшают FBA. Однако лучшим поведением решения был MOFBA4 на рисунке 3d, поскольку он обеспечивает больше недоминируемых решений и поддерживает хорошее разнообразие популяции для среды с тем же размером популяции.

На рис . 3 показано распределение потока пигментов в микроводорослях C. vulgaris; видно , что каждый вариант MOFBA предлагает разное поведение, и все они улучшают FBA на рисунках 3a–c, но лучшее поведение решений можно наблюдать в варианте MOFBA4 на рисунке 3d.

Machine Translated by Google

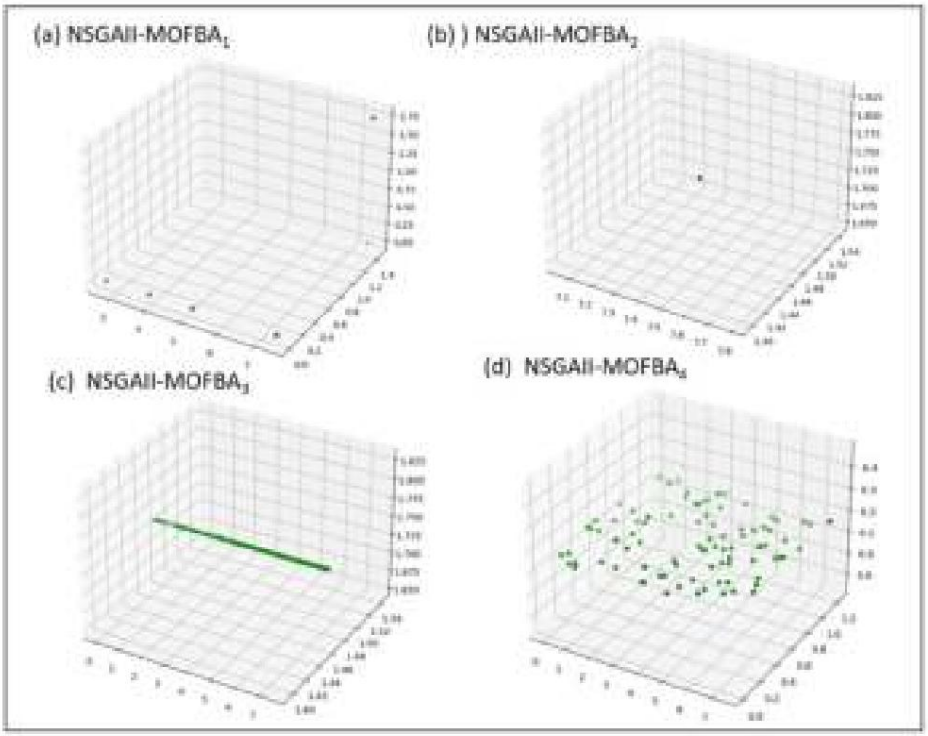


Рис. 3. Сравнение алгоритма NSGAII с вариантами MOFBA1 (а) , MOFBA2 (б) , MOFBA3 (в) и MOFBA4 (г) в распределении потоков пигментов.

6.2. Эксперимент

2. При сравнении производительности алгоритма NSGAII с вариантом MOFBA4 (рис. 4a) и классическим однокритериальным оптимизационным FBA (рис. 4b) можно заметить, что алгоритм NSGAII-MOFBA4 демонстрирует превосходство, демонстрируя , что предоставляемая информация улучшен за счет предоставления большего количества решений и, что немаловажно, значительно улучшенного распространения. Это усиленное распределение особенно очевидно на рисунках 3d и 4a, где изучены две различные метаболические сети, такие как метаболизм глутамата и распределение потоков пигментов в микроводорослях C. vulgaris.

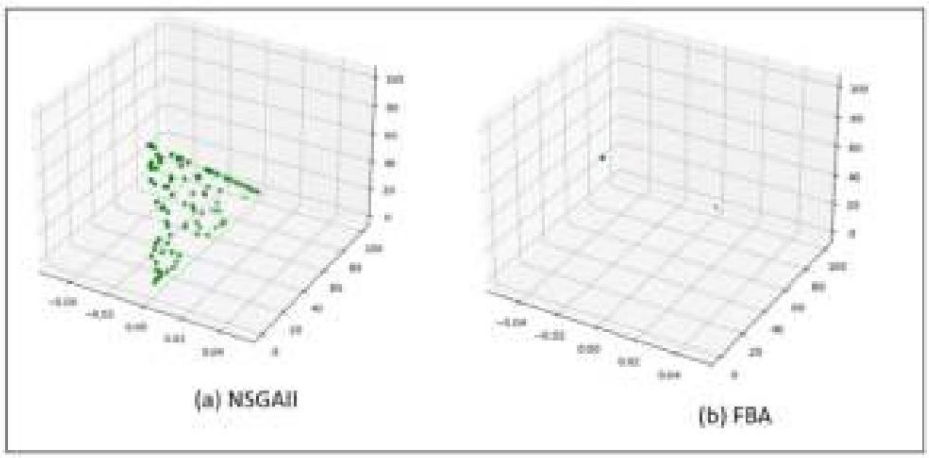


Рисунок 4. Сравнение NSGAII-MOFBA4 (а) и FBA (б) в распределении потоков, связанных с метаболизмом глутамата микроводорослей Chlorella vulgaris.

6.3. Эксперимент

3. На рисунке 5 представлена оценка алгоритмов NSGAII, рис. 5a,c, и MOEA/D, рис. 5b,d, в распределении потоков, связанных с метаболизмом глутамата.

Machine Translated by Google

а сеть распределения и потоки пигментов с помощью NSGAII, рисунок 5c, и MOEA/D, рисунок 5d, от микроводорослей C. vulgaris с вариантом алгоритма MOFBA4. Было показано, что алгоритм NSGAII имеет больше решений и лучшее разнообразие населения по сравнению с MOEA/D. Метаэвристика важна; в данном случае лучшим является NSGAII, что согласуется с литературными данными [35], поскольку этот алгоритм хорошо работает с 2 и 3 целями.

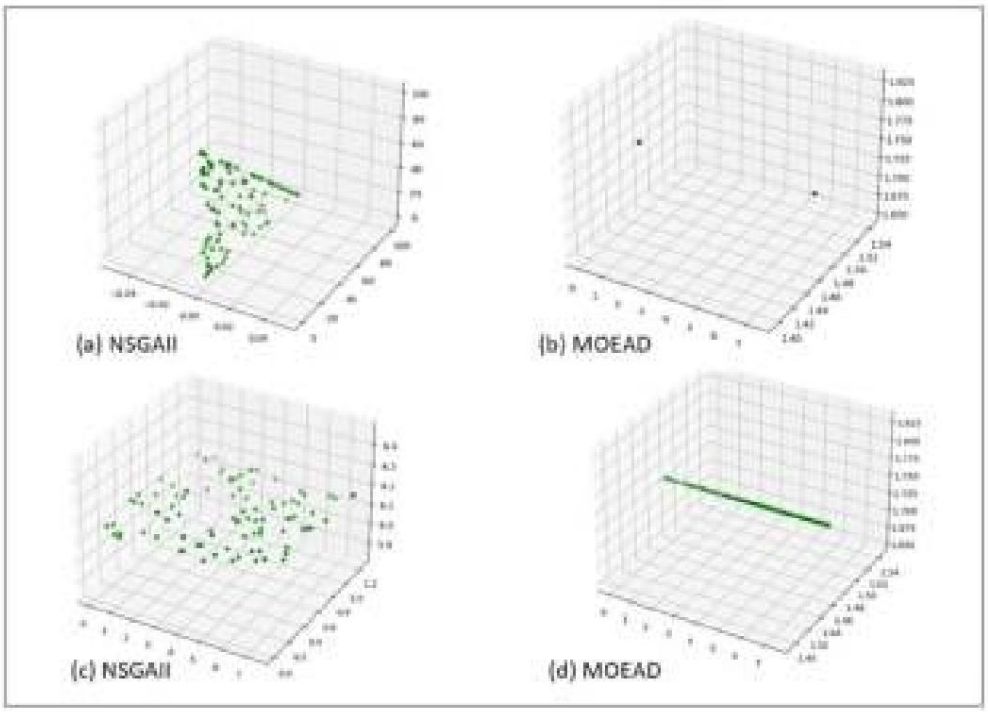


Рисунок 5. Сравнение алгоритмов NSGAII (a,c) и MOEA/D (b,d) в сети распределения пигментов и в распределении потоков, связанных с метаболизмом глутамата микроводорослей Chlorella vulgaris.

6.4. Эксперимент

4. Помимо тестирования различных тематических исследований с использованием подходов FBA, NSGAII и MOEA/D, был проведен эксперимент с использованием быстрого случайного подхода, который мы называем случайным, по распределению потока пигментов в микроводорослях C. vulgaris. . На рисунке 6 показано сравнение FBA, рисунок 6a, случайный вариант, рисунок 6c, и NSGAII, рисунок 6 Случайный метод, несмотря на свою скорость, не смог дать лучших результатов (рис. 6b) по сравнению с тем, что представлено на рис . 6в, NSGAII.

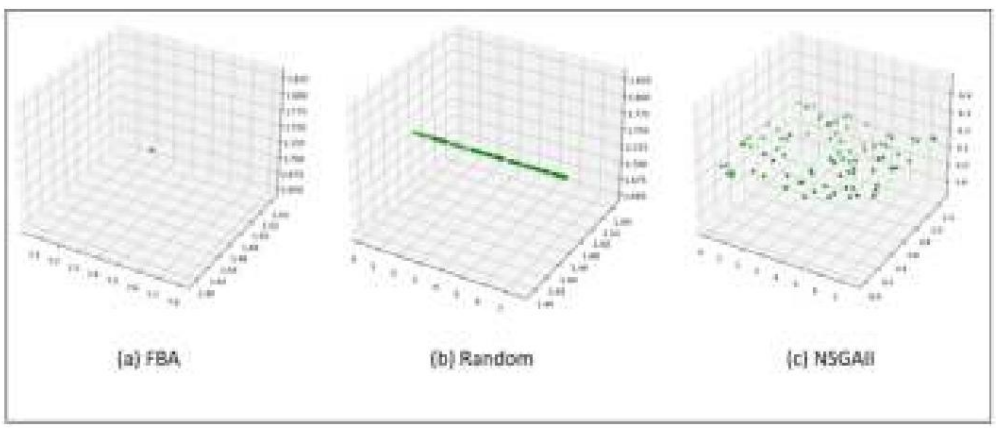


Рисунок 6. Сравнение вариантов (a) FBA, (б) случайного и (в) NSGAII в микроводорослях C. vulgaris.

Machine Translated by Google

6.5. Статистический анализ

Таблицы 5 и 6 показывают, что предложенные методы дают возможные решения; показано, что они удовлетворяют условиям, установленным в [17] и FBA, тем самым демонстрируя корреляцию решения in silico и его способность имитировать результаты в различных условиях культивирования. В Таблице 6 можно увидеть сравнение потоков в ммоль ч-1 , полученных с помощью FBA и NSGAII-MOFBA4 , и евклидова расстояния, представленного между ними. Таблица 5 показывает, что NSGAII обеспечивает большую универсальность для ограничения параметров условий выращивания за счет нижнюю и верхнюю границы значений, а также возможность моделировать циклы и бифуркации между метаболическими сетями. Аналогично, некоторые возможные решения, соответствующие Представлены НСГАИИ с вариантом MOFBA4 и решения, полученные из классического FBA.

Таблица 5. Распределение потоков в ммоль ч-1 , полученных методом NSGA II, связанных с синтезом пигменты в метаболизме микроводорослей C. vulgaris.

Реакция	фунт	УБ	С1	С2	S3	С4
НАДФН	0,000487	0,000487	0,000487	0,000487	0,000487	0,000487
IDS2	0	0,000487	$7,42 \times 10^{3}$	4,72 8	2,58 × 10–5	4,33 × 10 7
ФППШ	0	0,000487	7,42 × 10 5	4,72 × 10 5	2,58 × 10–5	4,33 × 10 7
ГППШ	0	0,000487	7,42 × 10 5	4,72 × 10 5	2,58 × 10–5	4,33 × 10 7
ФРДПт	0	0	0	0	0	0
ФПС	0	0	0	0	0	0
GRDPth	0	0	0	0	0	0
ГРДПХ	0	0	0	0	0	0
ГГППС	0	0,000487	7,42 × 10 5	4,72 × 10 5	2,58 × 10–5	4,33 × 10 7
v1	0	0,000402078	0	0	0	0
GGDPtu	-0,0000849	0,0000849	7,42 × 10 5	4,72 × 10 5	2,58 × 10–5	4,33 × 10 7
ЧЛАСГ	0	0,0000742	7,42 × 10 5	4,72 × 10 5	2,58 × 10–5	4,33 × 10 7
ГГЧЛДАР 0		0,0000742	7,42 × 10 5	4,72 × 10 5	2,58 × 10–5	4,33 × 10 7
КХЛАУ	0	0,0000742	7,42 × 10 5	4,72 × 10 5	2,58 × 10–5	4,33 × 10 7
PSY	0	2,24 × 10–8	0	0	0	0
ПДС1	2,24 × 10 8	2,24 × 10–8	0	0	0	0
ПДС2	2,24 × 10 8	2,24 × 10–8	0	0	0	0
ЗДС	2,24 × 10 8	2,24 × 10–8	1,59 × 10 8	1,54 × 10 8	1,06 × 10 8	3,58 × 10 9
НИ	2,24 × 10 8	2,24 × 10–8	1,59 × 10 8	1,54 × 10 8	1,06 × 10 8	3,58 × 10 9
v2	0,00	8,00 × 10 11	0	0	0	0
ЛКИГ	1,59 × 10 8	1,59 × 10 8	1,59 × 10 8	1,54 × 10 8	1,06 × 10 8	3,58 × 10 9
GCAROtu	1,59 × 10 8	1,59 × 10 8	1,59 × 10 8	1,54 × 10 8	1,06 × 10 8	3,58 × 10 9
ЛКИБ	1,59 × 10 8	1,59 × 10 8	1,59 × 10 8	1,54 × 10 8	1,06 × 10 8	3,58 × 10 9
v3	0,00	1,24 × 10 8	1,24 × 10 8	1,20 × 10 8	7,10 × 10 9	1,00 × 10 10
БКАРО	3,48 × 10 9	3,48 × 10 9	3,48 × 10 9	3,48 × 10 9	3,48 × 10 9	3,48 × 10 9
BCRPTXANH -3,48 × 10-9		3,48 × 10 9	3,48 × 10 9	3,48 × 10 9	3,48 × 10 9	3,48 × 10 9
v4	0,00	6,16 × 10 9	6,16 × 10 9	6,16 × 10 9	6,16 × 10 9	6,16 × 10 9
АНКСАНАС	2,68 × 10 9	2,68 × 10–9	2,68 × 10 9	2,68 × 10 9	2,68 × 10 9	2,68 × 10 9
v5	0,00	3,88 × 10 9	3,88 × 10 9	3,88 × 10 9	3,88 × 10 9	3,88 × 10 9
ВИОКСАНОР 2,41 × 10 9		2,41 × 10 9	2,41 × 10 9	2,41 × 10 9	2,41 × 10 9	2,41 × 10 9
v6	0,00	2,70 × 10 10	2,70 × 10 10	2,70 × 10 10	2,70 × 10 10	2,70 × 10 10
НЕОКСАНЫ	1,47 × 10 9	1,47 × 10 9	1,47 × 10 9	1,47 × 10 9	1,47 × 10 9	1,47 × 10 9
НЕОКСАНУ 0		1,47 × 10 9	1,47 × 10 9	1,47 × 10 9	1,47 × 10 9	1,47 × 10 9
ЛКИД	6,42 × 10 9	6,42 × 10 9	0	0	0	0
LCYA	6,42 × 10 9	6,42 × 10 9	0	0	0	0

Machine Translated by Google

Таблица 5. Продолжение.

Реакция	фунт	УБ	С1	С2	S3	С4
v7	0,00	1,33 × 10 9	0	0	0	0
АКАРОту	5,09 × 10 9	5,09 × 10 9	0	0	0	0
ЧЯ1	0	0	0	0	0	0
ЖЫ	0	0	0	0	0	0
ЧЯ2	0	5,09 × 10 9	0	0	0	0
СХНУ	0	5,09 × 10 9	0	0	0	0
v8	0	3,35 × 10 9	1,74 × 10 9	1,74 × 10 9	1,74 × 10 9	1,74 × 10 9
ЛУТ	1,74 × 10 9	1,74 × 10 9	1,74 × 10 9	1,74 × 10 9	1,74 × 10 9	1,74 × 10 9
ЛОРОКСАНУ 0		1,74 × 10 9	1,74 × 10 9	1,74 × 10 9	1,74 × 10 9	1,74 × 10 9
ФРДПт	0	1	0	0	0	0

Таблица 6. Распределение потоков в ммоль ч-1 , полученное с помощью FBA и алгоритма NSGAII-MOFBA4 .

	Евклидово расстояние между FBA и NSGAII-MOFBA4			
Реакция.	ФБА	МФБА4	МФБА4	МФБА4
АКАРОту'	5,09 × 10 9	0	0	0
АНКСАНАСКОР	2,68 × 10 9	2,68 × 10 9	2,68 × 10 9	2,68 × 10 9
БКАРО'	3,48 × 10 9	3,48 × 10 9	3,48 × 10 9	3,48 × 10 9
BCRPTXANH'	3,48 × 10 9	3,48 × 10 9	3,48 × 10 9	3,48 × 10 9
ХЛАСГ'	0	1,36106 × 10 9	2,35109 × 10 5	0,0000742
ЧЯА1'	0	0	0	0
ЧЯ2'	5,09 × 10 9	0	0	0
СХНУ'	5,09 × 10 9	0	0	0
ФПС'	0	0	0	0
ФРДПт'	0	0	0	0
GCAROtu'	1,59 × 10 8	7,94367 × 10 9	1,35944 × 10 8	1,59 × 10 8
ГГЧЛДАРЬ	0	1,36106 × 10 9	2,35109 × 10 5	0,0000742
GGDPtu'	0,000148102	1,36106 × 10 9	2,35109 × 10 5	0,0000742
GGPS'	0,000486596	1,36106 × 10 9	2,35109 × 10 5	0,0000742
GRDPth'	0	0	0	0
IDS1'	0,001459788	1,36106 × 10 9	2,35109 × 10 5	0,0000742
LCYA'	6,42 × 10 9	0	0	0
LCYB'	1,59 × 10 8	7,94367 × 10 9	1,35944 × 10 8	1,59 × 10 8
ЛКИД'	6,42 × 10 9	0	0	0
LCYG'	1,59 × 10 8	7,94367 × 10 9	1,35944 × 10 8	1,59 × 10 8
ЛУТ'	1,74 × 10 9	1,74 × 10 9	1,74 × 10 9	1,74 × 10 9
НЕОКСАНЫ	1,47 × 10 9	1,47 × 10 9	1,47 × 10 9	1,47 × 10 9
НИ'	2,24 × 10–8	7,94367 × 10 9	1,35944 × 10 8	1,59 × 10 8
ПДС1'	2,24 × 10–8	0	0	0
ПДС2'	2,24 × 10–8	0	0	0
PSY'	2,24 × 10–8	0	0	0
ВИОКСАНОР	2,41 × 10 9	2,41 × 10 9	2,41 × 10 9	2,41 × 10 9
ЗДС'	2,24 × 10–8	7,94367 × 10 9	1,35944 × 10 8	1,59 × 10 8
ЖЫ'	0	0	0	0
Евклидово расстояние		0,001545861	0,001514586	0,001451344

Machine Translated by Google

В этом разделе статистически подтверждается наличие разницы при использовании разных задачи оптимизации или алгоритмы, чтобы показать, что выбор актуален. Делать при этом он суммирует результаты путем сравнения по Гиперобъему (индикатор близости к фронт оптимального по Парето), чтобы увидеть, существует ли значительная разница между оптимизационные модели MOFBA3 и MOFBA4 , а также алгоритмы NSGAII и MOEA/D.

Первый анализ рассматривает модели MOFBA3 и MOFBA4, устанавливает решение алгоритм для NSGAII и оценивает все сети. Для анализа был использован Hypervolume. полученный в результате каждого из 30 прогонов алгоритма для каждой задачи. Используя непараметрический Знак рангового теста Уилкоксона, с уровнем достоверности 95% был подтвержден H0 , который указывает, что невозможно определить методологию, основанную на метаэвристике, которая улучшает информация о потоках метаболической сети, предоставляемая методом FBA путем переопределения Анализ баланса потоков как задача многокритериальной оптимизации. Таблица 7 суммирует результаты, показывающие значение гиперобъема в логарифмическом масштабе за прогон для каждой сети. и статус приемки H0 в последней строке. Видно, что гипотеза отвергается почти во всех проанализированных метаболических сетях. За исключением /textitChlorella, он может Следует отметить, что лучшей моделью оптимизации является MOFBA4.

Таблица 7. Данные были статистически проанализированы для подтверждения различий между MOFBA3 и MOFBA4 . Нулевая гипотеза H0 принималась, когда полученное значение p было меньше 0,05.

Метаболизм глутамата у C. vulgaris		Пигментная сеть у C. vulgaris	
MOFBA4 MOFBA3		МОФБА4 МОФБА3	
34,539	20,723	20.65	20,647
34,539	20,723	20.65	20,648
34,539	20,723	20.65	20,649
34,539	20,723	20.65	20,649
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20,652
34,539	20,723	20.65	20,648
34,539	20,723	20.65	20.650
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20,648
34,539	20,723	20.65	20,653
34,539	20,723	20.65	20,654
34,539	20,723	20.65	20,652
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20,649
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20.650
34,539	20,723	20.65	20.650
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20.650
34,539	20,723	20.65	20.650
34,539	20,723	20.65	20,647
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20,651
34,539	20,723	20.65	20,652
34,539	20,723	20.66	20,655
H0	ПРИНЯЛ	H0	ОТКЛОНЕННЫЙ

Machine Translated by Google

Второй анализ рассматривает алгоритмы NSGAII и MOEA/D, устанавливает модель оптимизации MOFBA4 и оценивает все сети. Для анализа был использован Hypervolume. полученное от каждого из 30 исполнений каждого алгоритма по решённой задаче. С использованием непараметрический знаковый ранговый критерий Уилкоксона с уровнем достоверности 95%, H0 был проверено, что указывает на то, что разница в средних значениях между выборками одинакова. В таблице 8 суммированы результаты и показано значение гиперобъема каждой сети за прогон при логарифмическая шкала и статус принятия H0 в последней строке. Видно, что гипотеза отвергается во всех проанализированных метаболических сетях. Эти результаты обоих анализ подтверждает то, что ожидалось, что важно рассмотреть, какая модель оптимизации использовать и какой алгоритм, поскольку их производительность при получении наборов решений может быть другим.

Таблица 8. Данные были статистически проанализированы для установления различий между использованием NSGAII и МОЭ/Д. Нулевая гипотеза H0 принималась, когда полученное значение p было меньше 0,05.

Метаболизм глутамата у C. vulgaris		Пигментная сеть у C. vulgaris	
НСГАИИ	МОЭ/Д	НСГАИИ	МОЭ/Д
20,723	20,723	20,647	20,723
20,723	20,723	20,648	20,723
20,723	20,723	20,649	20,723
20,723	20,723	20,649	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20,652	20,723
20,723	20,723	20,648	20,723
20,723	20,723	20.650	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20,648	20,723
20,723	20,723	20,653	20,723
20,723	20,723	20,654	20,723
20,723	20,723	20,652	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20,649	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20.650	20,723
20,723	20,723	20.650	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20.650	20,723
20,723	20,723	20.650	20,723
20,723	20,723	20,647	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20,651	20,723
20,723	20,723	20,652	20,723
20,723	20,723	20,655	20,723
H0	ОТКЛОНЕННЫЙ	H0	ОТКЛОНЕННЫЙ

Судя по графическим результатам, наблюдаемым, главным образом, из-за объема и дисперсии решений, полученных во всех рассмотренных метаболических сетях водорослей, показано что предложенный метод, основанный на многокритериальной оптимизации, решенной с помощью метаэвристики, обеспечивает лучшую поддержку анализа. С другой стороны, статистический анализ представленные в этом разделе, демонстрируют актуальность рассмотрения оптимизации модель и алгоритм, поскольку они могут способствовать различным типам улучшений.

Machine Translated by Google

Представленный здесь статистический анализ демонстрирует, что между оптимизационными моделями и метаэвристическими алгоритмами может быть значительная разница .

6.6. Обсуждение

Ранее практически не проводилось экспериментов с метаболизмом микроводорослей, за исключением [18]. Хотя существуют точные методологии, эволюционные подходы требуют меньше вычислительных ресурсов для решения множества задач; например, это дает преимущество использования меньшего количества времени и памяти. Такие подходы, как NSGAII и MOEA/D, предоставляют большую свободу выбора в процессе принятия решений благодаря разнообразию и количеству решений, а также возможности более легкого распознавания наиболее важных потоков в сети, их влияния и воздействия, вместо того, чтобы не игнорировать их. наличие методологии.

Из приведенных выше результатов можно сделать некоторые дополнительные выводы. Эксперимент 2 демонстрирует универсальность NSGAII для адаптации к различным обстоятельствам и его способность улучшать анализ метаболической сети, учитывая большее количество решений, которые он производит для каждого из них. Как показано в экспериментах с 1 по 4, аналитическая способность метаболической сети улучшается за счет введения алгоритма NSGAII.

Проблемы многокритериальной оптимизации, представленные в литературе, в настоящее время рассматривают различные метрики решения. В эксперименте 1 сравнивается использование алгоритма NSGAII с четырьмя вариантами задач оптимизации, причем задача оптимизации MOFBA4 является наиболее многообещающей за счет введения различных стратегий оптимизации, таких как оптимизация не только интересующих метаболитов, но также гиперобъема и расстояния между поколениями. По сравнению с MOFBA3, который минимизирует гиперобъем, в MOFBA2 переменными решения являются только потоки интересующих реакций.

Хотя в тематических исследованиях NSGAII имел лучшую графически наблюдаемую производительность, чем MOEA/D, как это произошло в эксперименте 3, поскольку тематические исследования имели три целевые функции и, согласно литературе, NSGAII лучше, чем MOEA/D, когда есть цель трех функций, открывается возможность использования MOEA/D в сетях, где необходимо оптимизировать более трех целевых функций. С помощью эксперимента 4 также можно заметить, что простого использования решения случайной выборки недостаточно для получения хорошего набора решений, например, при использовании метаэвристики. Однако необходимо принять особые меры, чтобы обеспечить соблюдение ограничений или информации о контроле, желаемой заинтересованным лицом.

Алгоритмы были протестированы на различных штаммах микроводорослей, как показано в [18] на C.rainhardti, а в этом исследовании с использованием C. vulgaris в сложных метаболических сетях, содержащих циклы, бифуркации и обратимые реакции, проверяется их жизнеспособность в различных метаболических процессах. сетей, подтверждая, что их можно использовать не только в отношении отдельных микроводорослей. Это оставляет открытой возможность его использования у других видов животных, где есть необходимость оптимизации более чем одной целевой функции.

7. Выводы

В настоящей научной работе изучены метаболические потоки зеленых микроводорослей. Цель была сосредоточена на проверке пригодности методов in silico в качестве вспомогательных стратегий для улучшения анализа метаболизма микроводорослей. В результате проведенных экспериментов были получены доказательства, подтверждающие следующие выводы:

Изучение метаболических потоков микроводорослей совершенствуется за счет увеличения количества растворов, удовлетворяющих условиям существования микроводоросли и позволяющих ей жить. Это заметно, потому что, в отличие от традиционных методов, таких как FBA, которые предлагают только решение, которое ограниченно расширяется за счет анализа чувствительности, ему очень предпочтительна интеграция его в методологию, основанную на метаэвристике и задачах многокритериальной оптимизации ; он увеличивает количество потоков, которые удовлетворяют искомым условиям в метаболической сети, а также одновременно позволяет оптимизировать несколько метаб интереса.

Существует несколько альтернатив анализу метаболической сети путем оптимизации нескольких представляющих интерес метаболитов, при этом метод FBA является основой процесса оптимизации.

Machine Translated by Google

В настоящей работе предложены четыре модели оптимизации, демонстрирующие этот результат, каждая из которых предлагает углы анализа, отличные от тех, которые предлагает FBA.

Задачи оптимизации, поддерживающие метаболические исследования, можно решить, рассматривая различные эволюционные метаэвристики и получая значимые результаты для анализа. Это демонстрируется с использованием NSGAII и MOEA/D для решения предложенных задач оптимизации. В этом исследовании NSGAII показал лучшую производительность в целом, что согласуется с литературными данными, за счет исключительно одновременной оптимизации трех целей. Это показывает, что для будущей работы анализ лучшей метаэвристики должен быть проведен до исследования.

Параметры поиска решения можно контролировать в ходе анализа метаболической сети, регулируя границы реакции. Это способствует дальнейшему совершенствованию изучения микроводорослей, поскольку становится возможным определение контролируемых сред. Это наблюдается в процессе валидации, когда параметры генерации решений были ограничены значениями, найденными в работе с образцами in vivo.

Выбор одного алгоритма или модели для оптимизации конкретной метаболической сети может быть затруднительным и требует тонкой настройки для определения конфигурации, которая лучше всего соответствует исследовательским интересам проводимой метаболической инженерии. Это очевидно, учитывая различия в производительности алгоритмов и оптимизационных моделей или различных комбинаций, протестированных в этой исследовательской работе.

Лицо, принимающее решения, например исследователь в области метаболической инженерии, улучшает свою способность принимать решения, визуализируя набор метаболических потоков, которые удовлетворяют условиям, указанным для изучаемой метаболической сети.

Вклад автора: Концептуализация, MFB-B. и ЛА-В.; методика, МФБ-Б. и НР-В.; программное обеспечение, МФБ-Б. и НР-В.; валидация, ЛА-В., АЛМ-С. и Чехия; формальный анализ, МФБ-Б., ЛА-В. и НР-В.; расследование, МФБ-Б.; ресурсы, МФБ-Б. и ЛА-В.; курирование данных, ЛА-В., АЛМ-С. и Чехия; письмо — подготовка первоначального черновика, МФБ-Б., НР-В. и КГ-С.; написание-рецензирование и редактирование, ЛА-В., АЛМ-С. и Чехия; визуализация, МФБ-Б.; надзор, ЛА-В. и АЛМ-С.; администрация проекта, МФБ-Б. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: CZ поддержан Национальным научным фондом (грант № 2313313). Любые мнения, выводы и выводы или рекомендации, выраженные в этом материале, принадлежат автору (ам) и не обязательно отражают точку зрения Национального научного фонда.

Заявление о доступности данных: сценарий модифицированных метаэвристических алгоритмов NSGAII и MOEA/D с соответствующими MOFBA хранится в репозитории GitHub: https://github. com/multiobjectiveoptimization2/MOFBAs (по состоянию на 24 июня 2024 г.).

Благодарности: Авторы выражают признательность за поддержку со стороны проектов CONAHCYT №. 3058, A1-S-11012 и Национальная лаборатория технологий информации (LANTI). Они также выражают признательность за поддержку со стороны ITCM и проекта TECNM 14612.22-P. Моника Фабиола Брионес Баес благодарит за получение стипендии № 784358 от CONAHCYT для продолжения обучения в аспирантуре.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Приложение А

Приложение А.1

Как упоминалось ранее, метаболическая сеть – это совокупность реакций, присутствующих в микроорганизме. Следующий список реакций представляет собой метаболическую сеть синтеза пигментов в микроводорослях Chlorella vulgaris .

Machine Translated by Google

Таблица A1. Реакции синтеза пигментов в метаболизме микроводорослей.

C. vulgaris [17].

Имя	Формула
ФРДПт	frdp[c] ⇌ frdp[c]
GRDPth	grdp[c] ⇌ grdp[c]
АКАРОту	akaro[c] ⇌ akaro[y]
GCAROtu	gkaro[c] ⇌ gkaro[y]
GGDPtu	ggdp[h] ⇌ ggdp[u]
ФПС	grdp[c] + ipdp[c] ⇌ frdp[c] + ppi[c]
ФППШ	grdp[h] + ipdp[h] ⇌ frdp[h] + h[h] + ppi[h]
ГГПС	frdp[h] + ipdp[h] ⇌ ggdp[h] + h[h] + ppi[h]
ГППШ	dmpp[h] + ipdp[h] ⇌ grdp[h] + h[h] + ppi[h]
IDS2	h[h] + h2mb4p[h] + nadf[h] ⇌ dmpp[h] + h2o[h] + nadf[h]
АНКСАНАСКОР	anxan[u] + ascb-L[u] ⇌ dhdascb[u] + h2o[u] + заксан[u]
БКАРО	caro[u] + h[u] + nadph[u] + o2[u] ⇌ bcrptxan[u] + h2o[u] + nadp[u]
BCRPTXANH	bcrptxan[u] + h[u] + nadf[u] + o2[u] ⇌ h2o[u] + nadp[u] + заксан[u]
ЧЯ1	akaro[u] + h[u] + nadf[u] + o2[u] ⇌ h2o[u] + nadp[u] + zxan[u]
ЧЯ2	akaro[u] + h[u] + nadf[u] + o2[u] ⇌ crpxan[u] + h2o[u] + nadf[u]
СХНУ	crpxan[u] + h[u] + nadf[u] + o2[u] ⇌ h2o[u] + lut[u] + nadf[u]
ЛКИБ	gkaro[y] ⇌ karo[y]
LCYA	lyc[h] ⇌ dcaro[h]
ЛКИГ	lyc[h] ⇌ gcaro[h]
НЕОКСАНЫ	виоксан[y] ⇌ неоксан[y]
НИ	norsp[h] + o2[h] + pqh2[h] ⇌ 2 h2o[h] + lyc[h] + pq[h]
ПДС1	фито[h] + pq[h] ⇌ фитофл[h] + pqh2[h]
ПДС2	phytfl[h] + pq[h] ⇌ pqh2[h] + zcaro[h]
PSY	2 гддп[ч] ⇌ 2 ч[ч] + фито[ч] + 2 ppi[ч]
ВИОКСАНОР	ascb-L[u] + виоксан[u] ⇌ anxan[u] + dhdascb[u] + h2o[u]
ЗДС	o2[h] + pqh2[h] + zcaro[h] ⇌ 2 h2o[h] + norsp[h] + pq[h]
ЖЫ	h[u] + nadph[u] + o2[u] + zxan[u] ⇌ h2o[u] + lut[u] + nadp[u]
ЧЛАСГ	chlda[u] + ggdp[u] ⇌ ggchlda[u] + h[u] + ppi[u]
ГГЧЛДАР	ggchlda[u] + 3 h[u] + 3 nadf[u] ⇌ chla[u] + 3 nadf[u]
ГГДР	ggdp[h] + 3 h[h] + 3 nadf[h] ⇌ 3 nadf[h] + pdp[h]
КХЛБСГ	chldb[u] + ggdp[u] ⇌ ggchldb[u] + h[u] + ppi[u]

Таблица A2. Реакции, полученные из распределения потоков, связаны с метаболизмом глутамата [17].

Имя	Формула
ГЛН-й	gln-L[c] + h[c] ⇌ gln-L[h] + h[h]
ГАЛх	atp[h] + glu-L[h] + nh4[h] ⇌ adp[h] + gln-L[h] + h[h] + pi[h]
ГЛЮС	glu-L[h] + h2o[h] + nad[h] ⇌ akg[h] + h[h] + nadh[h] + nh4[h]
переизбыток	atp[h] + glu-L[h] + trnaglu[h] ⇌ amp[h] + glutrna[h] + h[h] + ppi[h]
ГЛЮС(надф)	akg[h] + gln-L[h] + h[h] + nadf[h] ⇌ 2 glu-L[h] + nadf[h]
АСПАТх	akg[h] + asp-L[h] ⇌ glu-L[h] + oaa[h]
АСПНА1-й	asp-L[c] + na1[c] ⇌ asp-L[h] + na1[h]
VALth	h[c] + val-L[c] ⇌ h[h] + val-L[h]
ВСТА(val)ч	akg[h] + val-L[h] ⇌ 3mob[h] + glu-L[h]
ТИРТА	34hpp[h] + glu-L[h] ⇌ akg[h] + tyr-L[h]
ТИРт	h[c] + tyr-L[c] ⇌ h[h] + tyr-L[h]
ВСТАh	3mop[h] + glu-L[h] ⇌ akg[h] + ile-L[h]
ILEth	h[c] + ile-L[c] ⇌ h[h] + ile-L[h]

Ссылки

1. Эстевес, А.Ф.; Соарес, ОС; Вилар, виджей; Пирес, Ж.К.; Гонсалвес, Алабама Влияние длины волны света на улавливание CO2, биомасса Производство и поглощение питательных веществ зелеными микроводорослями: шаг вперед в интеграции и оптимизации процессов. Энергии 2020, 13, 333. [CrossRef]
2. Ван Ю.; Тиббетс, С.М.; Макгинн, П.Дж. Микроводоросли как источники высококачественного белка для пищевых продуктов и белковых добавок человека. Продукты питания 2021, 10, 3002. [CrossRef] [ПабМед]
3. Браун, MR; Джеффри, С. Биохимический состав микроводорослей классов зеленых водорослей Chlorophyceae и Prasinophyceae. 1. Аминокислоты, сахара и пигменты. Дж. Эксп. Мар Биол. Экол. 1992, 161, 91–113. [Перекрестная ссылка]
4. Холси, Р.; Рамос, ПВ; Маркс, Е.А.; Монтеро, О.; Рад, К. 2. Биотехнологическое использование микроводорослей: обзор современного уровня техники. и проблемы для экономики замкнутого цикла. Биокатал. Сельское хозяйство. Биотехнология. 2021, 36, 102114. [CrossRef]
5. Ромеро, DCV; Кардосо, АР; Монтес, В.Д. Альтернативное использование микроводорослей для удаления металлических песков. RIAA 2022, 13, 10.
6. Вулстон, Б.М.; Эдгар, С.; Стефанопулос, Г. Метаболическая инженерия: прошлое и будущее. Анну. Преподобный Хим. Биомол. англ. 2013, 4, 259–288. [Перекрестная ссылка] [ПабМед]
7. Касте, Дж.А.; Шачар-Хилл, Ю. Проверка и выбор модели при анализе метаболических потоков и анализе баланса потоков. Биотехнология. Прог. 2024, 40, e3413. [Перекрестная ссылка] [ПабМед]
8. Ананд С.; Мукерджи, К.; Падманабхан, П. Понимание анализа баланса потоков биохимических сетей. Биотехнология. Жене. англ. Ред. 2020 г., 36, 32–55. [Перекрестная ссылка] [ПабМед]
9. Орт, Дж. Д.; Тиле, И.; Палссон, Б.О. Что такое анализ баланса потоков? Нат. Биотехнология. 2010, 28, 245–248. [Перекрестная ссылка]
10. Лю, Ю.; Вестерхофф, Х.В. Конкурентный, многоцелевой и разделенный анализ баланса потоков для решения тканеспецифичных врожденные нарушения обмена веществ. Дж. Наследовать. Метаб. Дис. 2023, 46, 573–585. [Перекрестная ссылка]

Machine Translated by Google

11. Раман, К.; Чандра, Н. Анализ баланса потоков биологических систем: приложения и проблемы. Краткий. Биоинформ. 2009, 10, 435–449. [Перекрестная ссылка]

12. Льюис, Н.; Нагараджан, Х.; Палссон, Б. Ограничение метаболических отношений генотип-фенотип с использованием филогении in silico методы. Нат. Преподобный Микробиол. 2012, 10, 291–305. [Перекрестная ссылка]

13. Тоёсима, М.; Тойя, Ю.; Симидзу, Х. Анализ баланса потоков цианобактерий показывает избирательное использование фотосинтетических компонентов переноса электронов в различных спектральных условиях освещения. Фотосинт. Рез. 2020, 143, 31–43. [Перекрестная ссылка] [ПабМед]

14. Хуанг Дж.; Хоу, Дж.; Ли, Л.; Ван, Ю. Анализ баланса потоков деградации глюкозы путем анаэробного сбраживания при отрицательном давлении. Межд. Журнал Водородная энергетика 2020, 45, 26822–26830. [Перекрестная ссылка]

15. Трилла-Фуэртес, Л.; Гамес-Посо, А.; Диас-Альмирон, М.; Прадо-Васкес, Г.; Сапатер-Морос, А.; Лопес-Вакас, Р.; Нанни, П.; Самора, П.; Эспиноза, Э.; Фресно Вара, Дж. А. Компьютерное моделирование метаболизма предсказывает риск отдаленной безрецидивной выживаемости у больных раком молочной железы. Будущий Онкол. 2019, 15, 3483–3490. [Перекрестная ссылка]

16. Бойл, Северная Каролина; Морган, Дж. А. Анализ баланса потоков первичного метаболизма у Chlamydomonas Reinhardtii. Система BMC. Биол. 2009, 3, 4. [Перекрестная ссылка] [ПабМед]

17. Суньига, К.; Ли, Коннектикут; Хельсман, Т.; Леверинг, Дж.; Зелински, округ Колумбия; МакКоннелл, Бо; Лонг, КП; Кношауг, Е.П.; Гварниери, Монтана; Антоневич, MR; и др. Геномно-масштабная метаболическая модель зеленой водоросли Chlorella vulgaris UTEX 395 точно предсказывает фенотипы в автотрофных, гетеротрофных и миксотрофных условиях роста. Физиол растений. 2016, 172, 589–602. [Перекрестная ссылка] [ПабМед]

18. Брионес-Баэз, МФ; Агилера-Васкес, Л.; Ранхель-Вальдес, Н.; Мартинес-Салазар, Алабама; Зуньига, К. Многоцелевая оптимизация метаболизма микроводорослей: эволюционный алгоритм, основанный на FBA. Метаболиты 2022, 12, 603. [CrossRef] [ПабМед]

19. Рангаиа, врач общей практики; Петричиоле, А. Многоцелевая оптимизация в химической технологии: разработки и приложения; Рангаиа, врач общей практики, Бонилья-Петрисиолет, А., ред.; Уайли: Нью-Йорк, штат Нью-Йорк, США, 2013 г.

20. Лю, Х.; Тиан, Дж.; Дуань, П.; Ю, К.; Ван, Г.; Ван, Ю. ГрМоНАС: Многоцелевая структура NAS на основе детализации для эффективного медицинский диагноз. Вычислить. Биол. Мед. 2024, 171, 108118. [CrossRef]

21. PDE: подход дифференциальной эволюции на границе Парето для задач многокритериальной оптимизации. В материалах Конгресса по эволюционным вычислениям 2001 г. (номер по каталогу IEEE 01TH8546), Сеул, Республика Корея, 27–30 мая 2001 г.; Том 2, стр. 971–978.

22. Андраде, Р.; Дустмохаммади, М.; Сантос, Дж.; Саго, МФ; Мира, НП; Винга, С. MOMO – многоцелевой метаболический смешанный метод. Целочисленная оптимизация: применение к инженерии штаммов дрожжей. БМК Информ. 2020, 21, 69. [CrossRef]

23. Ван, Г.Г.; Чжао, Х.; Ли, К. Метаэвристические алгоритмы: теория и практика; CRC Press: Бока-Ратон, Флорида, США, 2024 г.

24. Деб, К.; Пратап, А.; Агарвал, С.; Мейариван, Т. Быстрый и элитарный многокритериальный генетический алгоритм: NSGA-II. IEEE Транс. Эвол. Вычислить. 2002, 6, 182–197. [Перекрестная ссылка]

25. Круз, Л.; Фернандес, Э.; Гомес, К.; Ривера, Г.; Перес, Ф. Многоцелевая оптимизация портфеля взаимозависимых проектов с «априорным» учетом предпочтений лиц, принимающих решения. Прил. Математика. Инф. наук. 2014, 8, 1517–1531. [Перекрестная ссылка]

26. Ривера, Г.; Флоренсия, Р.; Герреро, М.; Поррас, Р.; Санчес-Солис, Дж. Многокритериальный онлайн-анализ портфеля с помощью моделей компромиссного программирования, построенных на основных принципах нечеткого превосходства. Инф. наук. 2021, 580, 734–755. [Перекрестная ссылка]

27. Чанг, К. Х. Многокритериальная оптимизация и сложные темы. Теория проектирования и методы использования CAD/CAE; Эльзевир: Амстердам, Нидерланды, 2015 г.; стр. 325–406.

28. Чжан, Ц.; Ли, Х. MOEA/D: Многокритериальный эволюционный алгоритм, основанный на декомпозиции. IEEE Транс. Эвол. Вычислить. 2007, 11, 712–731. [Перекрестная ссылка]

29. Круз-Рейес, Л.; Фернандес, Э.; Ранхель-Вальдес, Н. Косвенное выявление параметров предпочтений на основе метаэвристической оптимизации. для решения многообъектных задач. Межд. Дж. Компьютер. Интел. Сист. 2017, 10, 56–77. [Перекрестная ссылка]

30. Деб, К.; Гоял, М. Комбинированный генетический адаптивный поиск (GeneAS) для инженерного проектирования. Вычислить. наук. Информ. 1996, 26, 30–45.

31. Деб, К. Эффективный метод обработки ограничений для генетических алгоритмов. Вычислить. Методы Прикл. Мех. англ. 2000, 186, 311–338. [Перекрестная ссылка]

32. Бенитес-Идальго, А.; Небро, Эй Джей; Гарсиа-Ньето, Дж.; Ореги, И.; Дель Сер, Дж. jMetalPy: Фреймворк Python для многоцелевых задач. оптимизация с помощью метаэвристики. Рой Эвол. Вычислить. 2017, 12, e0171744. [Перекрестная ссылка]

33. Ибрагим А.; Лерман, Дж. А.; Палссон, Бо; Хайдьюк, доктор COBRApy: реконструкция и анализ на основе ограничений для Python. Система BMC. Биол. 2013, 7, 74. [CrossRef]

34. Хантер, Дж. Д. Matplotlib: среда 2D-графики. Вычислить. наук. англ. 2007, 9, 90–95. [Перекрестная ссылка]

35. Лю, Д.; Хуан, К.; Ян, Ю.; Лю, Д.; Вэй, Х. Двухцелевой алгоритм на основе структуры NSGA-II для оптимизации эксплуатации резервуаров. Дж. Гидрол. 2020, 585, 124830. [CrossRef]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Гиперспектральный Python: HyPy

Вим Баккер *, Франк ван Рейтенбек, Харальд ван дер Верф *, Кристоф Хекер, Арьян Дейкстра и Фрик ван дер Меер

Факультет ITC, Университет Твенте, 7522 NH Энсхеде, Нидерланды; fjavanruitenbeek@utwente.nl (FvR); cahecker@utwente.nl (Швейцария); ahdijkstra@utwente.nl (AD); fdvandermeer@utwente.nl (FvdM)
* Корреспонденция: whbakker@utwente.nl (ВБ); hmavanderwerff@utwente.nl (HvdW)

Аннотация: В этой статье описывается проектирование, реализация и использование пакета Python под названием Hyperspectral Python (HyPy). Собственное программное обеспечение для обработки гиперспектральных изображений стоит дорого , а инструменты, разработанные с использованием этих пакетов, не могут распространяться свободно. Идея HyPy заключается в том, чтобы иметь возможность обрабатывать гиперспектральные изображения с помощью бесплатного программного обеспечения с открытым исходным кодом. HyPy был разработан с использованием Python и опирается на возможности таких пакетов, как NumPy и SciPy, по обработке массивов . HyPy был разработан с учетом практической спектрометрии изображений и реализовал ряд новых идей. Чтобы назвать некоторые из этих идей, HyPy имеет инструменты BandMath и SpectralMath для обработки изображений и спектров с использованием операторов Python, может обрабатывать спектральные библиотеки, как если бы они были изображениями, и может адресовать полосы по длине волны, а не по номеру полосы. Мы ожидаем, что HyPy будет полезен для исследований, образования и проектов, использующих гиперспектральные данные, поскольку он гибок и универсален.

Ключевые слова: гиперспектральная визуализация; обработка изображений; Питон; минералы

1. Введение

Пакеты обработки изображений существуют с момента создания первых компьютеров. Однако в начале этого столетия не многие пакеты были способны обрабатывать данные гиперспектральных изображений. Гиперспектральные изображения имеют тенденцию быть огромными, и стандартные операции, такие как классификация изображений, непригодны для обработки гиперспектральных данных высокой размерности. Традиционно для обработки гиперспектральных изображений используются фирменные пакеты, такие как ENVI (ENVI и IDL — торговые марки NV5 Geospatial Solutions, Inc.) (Environment for Visualizing Images) [1] и другие. Однако с появлением бесплатного программного обеспечения с открытым исходным кодом [2] возникла необходимость иметь возможность использовать и обрабатывать гиперспектральные изображения независимо от проприетарных пакетов.

Благодаря таким пакетам, как numpy [3], scipy [4] и matplotlib [5], язык Python [6] стал средой обработки данных и прототипирования, конкурируя с такими системами (собственными и бесплатными), как MATLAB (MatLab — зарегистрированная торговая марка The MathWorks, Inc.), IDL (ENVI и IDL являются торговыми марками NV5 Geospatial Solutions, Inc.), Octave (Octave является торговой маркой Octave Technologies, Inc.) и R.

Python, в дополнение к вышеупомянутым пакетам numpy и scipy, является бесплатным и имеет открытый исходный код. Пользователи могут получить доступ к исходному коду, вносить изменения и распространять свои собственные версии. В принципе, любой код, написанный на Python, можно распространять свободно.

Мы используем Python для реализации пакета Hyperspectral Python, HyPy (произносится [hip-ee], как и субкультура 1960-х годов), который представляет собой пакет программного обеспечения для чтения и записи, визуализации, обработки и анализа гиперспектральных данных.

HyPy начинался как ряд классов изображений и фабричных функций, позволяющих читать и записывать изображения в формате ENVI. Создание большего количества инструментов в HyPy получило импульс, когда наша исследовательская группа приняла участие в пилотном проекте по картографированию планет Марса [7] в 2009 и 2010 годах. Впоследствии были написаны различные общие инструменты и специальные инструменты для обработки гиперспектральных данных Марса. В то же время HyPy предложил

check for updates

Цитирование: Баккер, В.; ван Рейтенбек, Ф.; ван дер Верфф, Х.; Хекер, К.; Дейкстра, А.; ван дер Меер, Ф. Гиперспектральный Python: HyPy. Алгоритмы 2024, 17, 337. https://doi.org/10.3390/a17080337 .

Академические редакторы: Лаура Антонелли и Люсия Маддалена

Поступила: 27 июня 2024 г.
Пересмотрено: 22 июля 2024 г.
Принято: 30 июля 2024 г.
Опубликовано: 1 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария.
Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

Machine Translated by Google

платформа для реализации экспериментальных идей. Постепенно пакет был расширен до того, что есть сегодня. В этом документе описываются соображения проектирования и реализации HurPy с приведением ряда примеров.

2. Проектирование и реализация

В этом разделе мы стремимся показать, что HurPy обладает набором функций, которые отличают его от других пакетов и делают его универсальным и гибким инструментом для обработки гиперспектральных данных.

HurPy имеет следующие возможности: • HurPy поддерживает три уровня взаимодействия: графический интерфейс пользователя (GUI), интерфейс командной строки (CLI) и интерфейс программирования (API). Доступ к пользовательским интерфейсам HurPy можно получить из одного меню верхнего уровня. • Меню верхнего уровня настраивается и может быть адаптировано для различных целей. • HurPy имеет средства просмотра гиперспектральных изображений и спектральных библиотек. Эти средства просмотра имеют простой интерфейс, но при этом предлагают достаточную функциональность для изучения качества и информативности данных.

• HurPy предлагает возможность чтения и записи файлов данных ENVI. • Гиперспектральные изображения имеют тенденцию быть огромными, однако HurPy экономит память. • HurPy поддерживает три собственных формата изображений ENVI: BSQ, BIL и BIP. Внутри HurPy преобразует их в формат BIP, избавляя программиста от необходимости иметь дело с тремя разными форматами.

• HurPy предлагает функциональные возможности для работы с длинами волн вместо номеров диапазонов. Номера полос не имеют физического значения, тогда как длины волн имеют. • Необработанные данные изображения могут не иметь полос в порядке увеличения длины волны. Поэтому, HurPy способен сортировать длины волн в изображении «на лету».

• HurPy может использовать список плохих каналов (bbl) и пропускать полосы, помеченные как плохие. • HurPy может обрабатывать спектральные библиотеки ENVI, как если бы они были изображениями ENVI. • При необходимости спектры пересчитываются «на лету». Это устраняет необходимость спектральной повторной выборки перед объединением спектров из разных источников.

• HurPy — это бесплатное программное обеспечение с открытым исходным кодом, что делает его идеальным инструментом для образования. Студенты могут получить пакет бесплатно и использовать его дома.

В следующих разделах упомянутые выше моменты будут обсуждаться более подробно.

2.1. Уровни использования: GUI, CLI и API.

Пакеты обработки изображений поддерживают различные типы интерфейсов программного обеспечения. Например , пакет Khoros [8] поддерживает графический интерфейс пользователя, интерфейс командной строки и API. Кроме того, Хорос предлагает интерфейс для визуального программирования, который называется Cantata [9]. Создание среды визуального программирования сложна и выходит за рамки HurPy. Однако, как и в случае с Хоросом, HurPy предлагает графический интерфейс, интерфейс командной строки и API.

Преимущество этих трех уровней заключается в том, что они обеспечивают гибкость. Графические интерфейсы позволяют легко использовать инструменты даже новичкам. Из интерфейса понятно, какие входы и выходы могут понадобиться тому или иному инструменту. В дополнение к этому графический интерфейс можно использовать для одновременного вызова нескольких задач. Таким образом, две или более программы могут запускаться параллельно, что позволяет использовать возможности многопроцессорной обработки компьютера без необходимости параллельного кодирова

CLI можно использовать для запуска инструментов из командной строки. Однако реальная сила CLI заключается в том, что его можно использовать для создания сценариев, которые запускают последовательность инструментов для запуска полной цепочки обработки. Кроме того, сценарии также можно использовать для трудоемких задач, которые выполняются в течение длительного времени, или для автоматизации обработки больших наборов изображений.

Большую часть функций HurPy также можно импортировать в другие программы Python с помощью API.

Графический интерфейс инструмента HurPy состоит из отдельного модуля, который импортирует функцию HurPy, создает графический интерфейс и передает параметры функции HurPy. Один модуль содержит графический интерфейс, а другой — API и CLI. Например, модуль edgy.py содержит функции гиперспектральной фильтрации границ, а также команду:

Machine Translated by Google

линейный интерфейс. Графический интерфейс вызывается с помощью модуля tkEdgy.py. В данном случае три уровня следующие:

• Программа с графическим
интерфейсом: tkEdgy.py; • Программа
CLI: edgy.pyl • Функция API: edgy().

Меню верхнего уровня HyrPy можно использовать для запуска программы с графическим интерфейсом tkEdgy.py. Острый Фильтр можно найти как «Пространственно-спектральный фильтр границ» в меню «Фильтр».

2.1.1. графический интерфейс

Большинство инструментов HyrPy разделены на две программы: одна для графического интерфейса, другая для API и CLI. Большинство графических интерфейсов представляют собой простые интерфейсы для установки входных и выходных параметров, которые передаются в API. Пример графического пользовательского интерфейса можно найти в разделе 2.4.1, где обсуждается инструмент Band Math Tool.

Пользовательские интерфейсы для инструментов обработки графического пользовательского интерфейса создаются с использованием Tkinter, пакета GUI (графического пользовательского интерфейса), который является частью официального дистрибутива Python. Tkinter — это тонкий объектно-ориентированный слой поверх Tcl/Tk [10]. Существует множество пакетов, которые можно использовать для разработки графических интерфейсов на Python. Например, wxPython — это более обширный пакет (например, набор виджетов), чем Tkinter. Однако модуль Tkinter (интерфейс Tk) является стандартным интерфейсом Python для набора инструментов Tk GUI [11] и был выбран для HyrPy из-за его простоты и доступности. Все имена инструментов HyrPy начинаются с «tk», что указывает на то, что они имеют графический интерфейс.

2.1.2. CLI и сценарии Интерфейс

командной строки полезен для запуска процессов из командной строки.
Однако наиболее важной причиной использования CLI является возможность запускать цепочку обработки из сценария.
Скрипты удобны для итеративного повторения набора команд.
Таким образом, потенциально большой набор изображений может быть обработан точно таким же образом.
Запуск сценариев уменьшает количество ошибок и дает возможность запускать команды, требующие много времени на обработку, в ночное время.

HyrPy придерживается соглашений, используемых для предоставления параметров командной строки, а также ввода и вывода в командной строке. Для этой цели. он использует стандартный модуль Python под названием argparse, который предлагает удобный способ создания интерфейсов командной строки, которые можно использовать как в Unix, так и в оболочках Windows.

2.1.3. API

Интерфейс прикладного программирования (API) — это программный интерфейс, с помощью которого одна часть программного обеспечения предоставляет функциональность другому программному обеспечению. В Python функциональность одной программы можно импортировать в другую с помощью оператора импорта. Важной частью API является спецификация API, которая описывает функции и структуры данных, необходимые для использования доступных функций.

В настоящее время в HyrPy нет специального руководства по API, но основные классы, такие как Image и Spectrum, описаны в «Руководстве программиста» и «Руководстве по спектральной математике» соответственно. Кроме того, многие функции HyrPy описаны в Руководстве по написанию сценариев HyrPy (CLI) и Руководстве пользователя HyrPy (GUI), поскольку CLI и GUI часто действуют как тонкая оболочка вокруг этих функций.

Поскольку объекты данных являются важной частью API, заголовок и изображение HyrPy объекты обсуждаются ниже; см. раздел 2.2.

2.1.4. Меню верхнего уровня Меню

верхнего уровня HyrPy можно настроить с помощью файла конфигурации меню hyrpymenu.cfg. Функциями могут быть скрипты Python (обычно программы HyrPy GUI), файлы PDF или HTML или полные URL-адреса. Таким образом, верхнее меню можно использовать для запуска сценариев, а также для представления документов, таких как руководства или веб-страницы.

Machine Translated by Google

Инструменты выполняются как отдельные процессы; следовательно, несколько инструментов HypPy можно запускать и выполнять параллельно одновременно. На рисунке 1 показано верхнее меню HypPy.

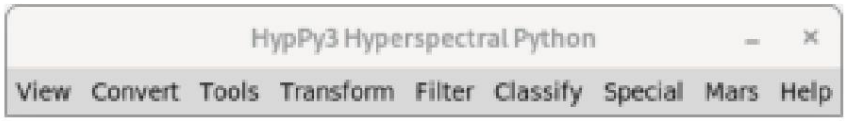


Рисунок 1. Меню верхнего уровня HypPy.

Скрипты Python выполняются Python как подпроцессы. PDF-файл, HTML или URL-адрес делегируются программе чтения файлов системы по умолчанию, которая запускает программу чтения PDF-файлов или веб- браузер в системе.

2.2. Форматы

изображений HypPy принял форматы ENVI в качестве своих собственных форматов. Форматы изображений ENVI и спектральной библиотеки ENVI представляют собой двоичные файлы с сопроводительным текстовым файлом заголовка. ENVI поддерживает три различных типа чередования в своем собственном формате: последовательное чередование полос (BSQ), чередование полос по пикселям (BIP) и чередование полос по строкам (BIL).

HypPy поддерживает все три формата ENVI: BSQ, BIP и BIL. Кроме того, HypPy может сопоставить спектральную библиотеку ENVI с изображением. Затем библиотека спектров представляется в виде изображения y x x x z , где y — количество спектров, хранящихся в библиотеке спектров, x всегда равно 1 (библиотека спектров видна как изображение только с одним столбцом пикселей), z — количество полос в спектральной библиотеке. Преимущество состоит в том, что спектральные библиотеки можно просматривать так же, как спектральные изображения. Кроме того, цепочку обработки, предназначенную для спектральных изображений, можно сначала протестировать в библиотеке спектров, чтобы увидеть, какое влияние цепочка обработки окажет на различные минералы, присутствующие в библиотеке спектров.

Для единообразия кодирования одноканальные изображения также отображаются в трех измерениях , и в этом случае размерность канала всегда равна 1. Из этого правила есть одно исключение: классификационные изображения отображаются как двумерные одноканальные изображения. Классификационные изображения имеют другое использование и семантику, чем гиперспектральные данные, и по этой причине классификационные изображения обрабатываются в HypPy по-другому.

Внутренне HypPy отображает все три формата: BSQ, BIP и BIL в массив с помощью BIP. структуру, тем самым избавляя программиста от необходимости иметь дело с тремя разными форматами.

2.2.1. Объекты заголовка и изображения

Модуль заголовка HypPy используется для чтения и записи файлов заголовков ENVI. Изображения ENVI и библиотеки спектров обычно состоят из двух файлов: данных и заголовка. Файлы данных содержат необработанные двоичные данные. Файлы заголовков содержат удобочитаемый текст.

Внутри HypPy заголовок ENVI представлен как объект Python класса Header. Объекты этого класса содержат атрибуты, которые отражают атрибуты в том виде, в котором они находятся в файлах заголовков ENVI. В коде Python есть несколько отличий. Во-первых, имена атрибутов в заголовке ENVI могут содержать пробелы, что неудобно в таком языке программирования, как Python, поскольку имена переменных не могут содержать пробелы. Было решено заменить пробелы в именах атрибутов ENVI подчеркиванием. Например, атрибут «тип данных» в заголовке файла ENVI будет переименован в « Во-вторых, некоторые значения внутри заголовка ENVI могут изменить свой внешний вид в Python. Например, записи IDL, как они появляются внутри заголовков ENVI, такие как значения атрибута длины волны, изменяются в стандартный тип данных списка Python. Перевод между записями IDL и списками Python выполняется внутри HypPy.

HypPy поддерживает все типы необработанных двоичных данных IDL/ENVI: от 8-битных целых чисел без знака до 64-битных целых чисел со знаком и от 32-битных чисел с плавающей запятой до 64-битных комплексных чисел двойной точности с парой вещественно-мнимого двойной точности. Кроме того, благодаря использованию имени поля «порядок байтов» HypPy поддерживает файлы как с прямым, так и с прямым порядком байтов.

Machine Translated by Google

HyperPy имеет две фабричные функции в модуле изображения: функцию Open() для открытия существующие изображения и функция New() для настройки новых (пустых) изображений.

Фабричная функция Open() возвращает один из объектов изображения класса ImageBIP, ImageBIL, ImageBSQ, Classification или ImageSL (спектральная библиотека, отображаемая как изображение). Изображения, к которым осуществляется доступ с помощью функции Open(), можно только читать, но не записывать. Чтобы защитить существующие изображения, они всегда открываются только для чтения.

Фабричная функция New() возвращает один из объектов изображения класса ImageBIP, ImageBIL , ImageBSQ или Classification. В дополнение к этим классам также поддерживается класс ImageSL , который представляет собой изображение спектральной библиотеки ENVI.

2.2.2. Карта памяти

HyperPy использует карту памяти Numpy, «numpy.memmap», для сопоставления файлов двоичных данных с массивами. Файлы, отображаемые в памяти, используются для доступа к большим файлам на диске без необходимости предварительного чтения всего файла в память. Объекты памяти NumPy представляют собой объекты, подобные массивам; ими можно манипулировать так же, как и массивами numpy. Поскольку в HyperPy был сделан выбор в пользу структуры BIP для внутреннего массива, оси двух других форматов, BSQ и BIL, необходимо поменять местами, чтобы они выглядели как BIP в возвращаемой карте памяти. Это достигается путем вызова функции транспонирования карты памяти. BSQ отображается с использованием транспонирования (1, 2, 0). BIL отображается с использованием транспонирования (0, 2, 1). Формат BIP не требует трансп

Можно было бы подумать, что использование функции транспонирования создаст копию данных с измененными местами осей данных. Однако это не так, поскольку для определенных операций numpy поддерживает так называемое «представление» данных. Представление создает ссылку на существующий массив с необходимой реорганизацией данных, добавляемых в метаданные. Это имеет два преимущества: во-первых, представление не создает копию данных в памяти, а во-вторых, когда представление открывается в режиме чтения-записи, любые изменения представления будут немедленно отражены в исходных данных.

Кроме того, порядок байтов и порядок байтов можно изменить, создав представление. Три формата (BSQ, BIP и BIL), порядок байтов (с прямым порядком байтов и с прямым порядком байтов) и тип данных (от 8-битных целых чисел без знака до 64-битных чисел с плавающей запятой и т. д.) можно обрабатывать путем создания представления данные в файле. Для входных и выходных данных изображения не требуется наличие копии в памяти компьютера. Таким образом, HyperPy может обрабатывать большие файлы, которые обычно не помещаются в памяти.

Для создания представлений применяются следующие функции: • Создание карты памяти: numpy.memmap(fname, mode = 'r', shape = shape, dtype = data_type); • Изменение порядка байтов: метод newbyteorder(); • Перестановка осей с BSQ на BIP: метод transpose(1, 2, 0); • Перестановка осей с BIL на BIP: метод transpose(0, 2, 1).

К сожалению, изменение типа данных входного файла создает копию в памяти, а не вид:

• Измените тип входных данных: метод astype(dtype).

Однако есть несколько предостережений при использовании numpy-карт памяти. Не все операции и функции, которые можно использовать с массивами numpy, можно использовать с картами памяти, и только ограниченное количество операций или функций на карте памяти способны возвращать представление.

Кроме того, само использование карты памяти имеет и недостатки. Настройка карты памяти требует немного дополнительной памяти, и вся реорганизация представления должна выполняться «на лету» при каждом доступе к данным в файле. Таким образом, карта памяти менее эффективна, чем копирование массива в памяти.

Преимущества карты памяти очевидны: она позволяет обрабатывать большие наборы данных , которые не могут поместиться в физическую память компьютера. Более того, внутри данные все время отображаются в одной и той же структуре BIP; в программе нет необходимости

Machine Translated by Google

чтобы отслеживать реальный формат чередования файла данных. Все это прозрачно обрабатывается с помощью карты памяти и ее представлений.

2.2.3. Форматы спектральных библиотек

НурРу поддерживает два основных типа спектральных библиотек: спектральные библиотеки ENVI и ASCII или текстовые спектральные библиотеки. Основным недостатком спектральных библиотек ENVI является то, что список длин волн должен быть одинаковым для всех спектров в спектральной библиотеке. Поэтому число полос и центральные длины волн всех спектров должны быть одинаковыми для всех включенных спектров. Преимущество этого формата заключается в том, что это двоичный формат, к которому можно обращаться как к изображению, что дает возможность обрабатывать библиотеки спектров так же, как и спектральные изображения. Это можно использовать для тестирования цепочек обработки спектральных библиотек перед их запуском на изображениях. Процесс обнаружения определенных минералов можно протестировать, например, в спектральной библиотеке Геологической службы США [12] , прежде чем применять его к изображению. Проблемы с ложными срабатываниями определенных минералов можно обнаружить на ранних этапах разработки технологической цепочки.

С другой стороны, спектральные библиотеки ASCII предлагают большую гибкость. Текстовые файлы можно импортировать в другие программы, при этом спектральные диапазоны отдельных спектров не обязательно должны быть одинаковыми. Сложность заключается в том, что перед использованием спектры необходимо повторно дискретизировать. НурРу выполняет повторную выборку спектров на лету, используя функцию interp1d() из модуля scipy.interpolate для объединения обоих наборов длин волн. Промежуточный этап повторной выборки спектров не требуется.

2.2.4. Списки плохих групп

Формат изображения ENVI поддерживает понятие так называемого списка плохих полос (BBL), обозначаемого в заголовочном файле ENVI ключевым словом BBL. В BBL должно быть столько элементов, сколько полос на изображении. Последовательно полосы отмечаются как плохие или хорошие.
Ноль (0) для определенного диапазона означает, что данным в этом диапазоне нельзя доверять из-за неисправности датчика или чрезмерного шума. Единица (1) для диапазона означает действительные данные.
НурРу может учитывать список плохих полос, так что плохие полосы не отображаются в данных массива. В большинстве инструментов НурРу это можно указать, установив для параметра use_bbl значение true. Если для этой опции установлено значение false, то в массиве видны все данные, включая плохие полосы. Такое виртуальное изображение, в котором для параметра use_bbl установлено значение true, может иметь меньше каналов, чем исходный файл данных.

Внутри это обрабатывается массивом индекс-индекс, который используется в качестве индексного массива для одновременного доступа к нескольким элементам. Недостаток заключается в том, что обработка таких виртуальных образов происходит медленнее, чем использование полного образа. Еще одним недостатком является то, что номера каналов в виртуальном изображении не совпадают с номерами каналов в исходном изображении. Однако одна из дизайнерских идей НурРу заключается в использовании длины волны, а не номера полосы для адресации спектральной полосы изображения (см. следующий раздел).

2.2.5. Сортировка длин волн Часто

датчики состоят из нескольких банков детекторов, каждый из которых охватывает определенный диапазон длин волн. Данные от этих детекторов могут иметь перекрывающиеся диапазоны длин волн или даже появляться в неожиданном порядке. Например, необработанные данные прибора OMEGA европейского Mars Express содержат данные банков детекторов SWIR1 (0,93–2,69 мкм) и SWIR2 (2,53–5,09 мкм) перед данными детектора VNIR (0,36 мкм). от мкм до 1,07 мкм) [13]. Другой известный гиперспектральный прибор AVIRIS состоит из четырех спектрометров с перекрывающимися диапазонами [14]. Обратите внимание, что избыточные полосы спектрального перекрытия спектрометров AVIRIS также можно удалить или поместить в список плохих полос; см., например, [15].

Очевидно, что переключенный порядок и перекрывающиеся диапазоны могут привести к проблемам при построении графика или обработке таких данных. Поэтому одним из критериев разработки НурРу была возможность «на лету» сортировать длины волн гиперспектрального куба данных. В компромиссе между использованием памяти и использованием ЦП НурРу не преобразует входное изображение в память; поэтому,

Machine Translated by Google

он должен конвертировать изображение «на лету». Многие инструменты HypPy предлагают опцию sort_wavelengths при открытии изображения.

HypPy сортирует длины волн, сохраняя таблицу индексов в качестве атрибута объектов изображений. Эта таблица преобразует номера каналов входного изображения в каналы, отсортированные по длине волны. Преимущество состоит в том, что внутри данные выглядят так, как будто их полосы отсортированы по длине волны. Очевидным недостатком является то, что номера каналов входного изображения могут не соответствовать номерам каналов исходного объекта изображения.

Однако если полосы рассматриваются по длинам волн, а не по номерам полос, то последний пункт не имеет значения. Длины волн имеют физический смысл и должны быть предпочтительным способом определения полос изображения. Номера диапазонов могут меняться и даже иметь разное значение для разных датчиков дистанционного управления. К сожалению, не все изображения имеют длины волн, либо потому, что длины волн отсутствуют, либо изображения не содержат спектральных данных, поэтому номера каналов все еще можно использовать в HypPy.

2.3. Зрители

2.3.1. Средство просмотра

изображений Средство просмотра изображений HypPy предлагает простой дисплей для просмотра гиперспектральных изображений и получения спектров. Средство просмотра изображений было разработано для быстрой интерактивной оценки качества изображения и получения спектров конечных элементов. Его простота облегчает использование начинающим пользователям гиперспектральных данных. Ползунки можно использовать для переключения отображения с одного диапазона на другой. В цветном режиме доступны три ползунка, по одному для каждого из каналов RGB дисплея . В строке состояния в нижней части окна представлена информация о длинах волн диапазонов, положении курсора и т. д. С помощью ползунков и левой кнопки мыши полосы и спектры входного изображения можно оценить по качеству и информативности. На рис . 2 показано гиперспектральное изображение образца горной породы. Этот конкретный образец породы 15-91-FE является частью комплекса Фен в Норвегии, который включает карбонатитовую интрузию [16].

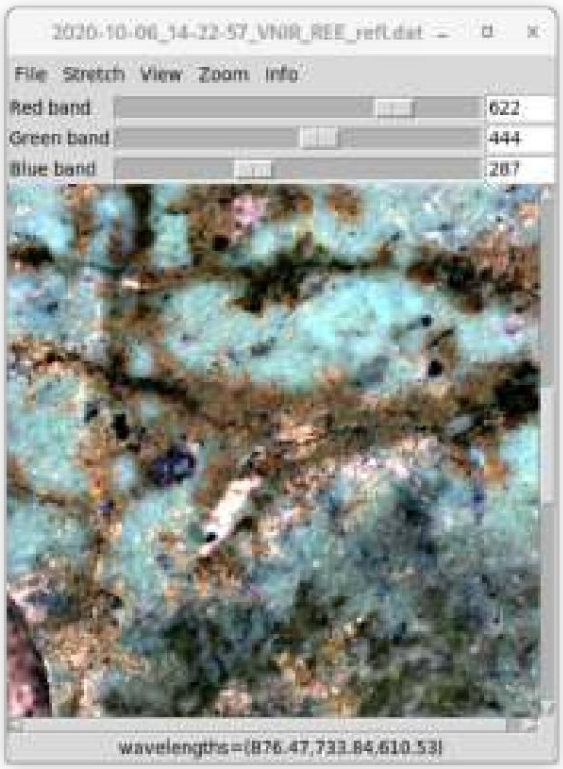


Рисунок 2. Средство просмотра изображений, показывающее композит искусственного цвета образца 15-91-FE. Красный канал 876 нм, зеленый 734 нм и синий 611 нм. Ползунки красного, зеленого и синего цветов можно использовать для изменения полос цветовых каналов RGB.

Machine Translated by Google

В меню «Файл» можно установить две опции: нужно ли сортировать полосы перед чтением или использовать BBL изображения.

Программа отображения может открывать два изображения одновременно. Первое изображение всегда видно в средстве просмотра. Второе изображение, называемое «теневым изображением», никогда не видно. Однако когда присутствует теневое изображение, спектры получаются на основе теневого изображения, а не видимого изображения. Это похоже на ГИС, в которой первый слой находится сверху, а второй слой скрыт под первым изображением. Эта опция полезна для загрузки изображения классификации или изображения маски в первый слой, в то время как спектры берутся из теневого изображения, что позволяет оценить, имеет ли смысл рассчитанная классификация или изображение маски. Кроме того, эта опция полезна для изучения карты длин волн вместе с гиперспектральным изображением, на основе которого была получена карта длин волн.

Меню «Растянуть» предлагает ряд стандартных методов растягивания изображения [17]. Реализованы такие методы, как 1–99% и выравнивание гистограммы. Кроме того, изображение можно оперативно инвертировать, повысить насыщенность или псевдоокрасить с помощью ряда доступных цветовых схем.

Щелчок левой кнопкой мыши в любом месте изображения откроет окно графика с графиком, представляющим спектр пикселя в этом месте, см. Рисунок 3. Дальнейший щелчок по изображению приведет к появлению дополнительных графиков на графике. При удалении окна графика процесс начинается заново с чистым окном графика. Таким образом, спектры разных точек изображения можно визуализировать вместе.

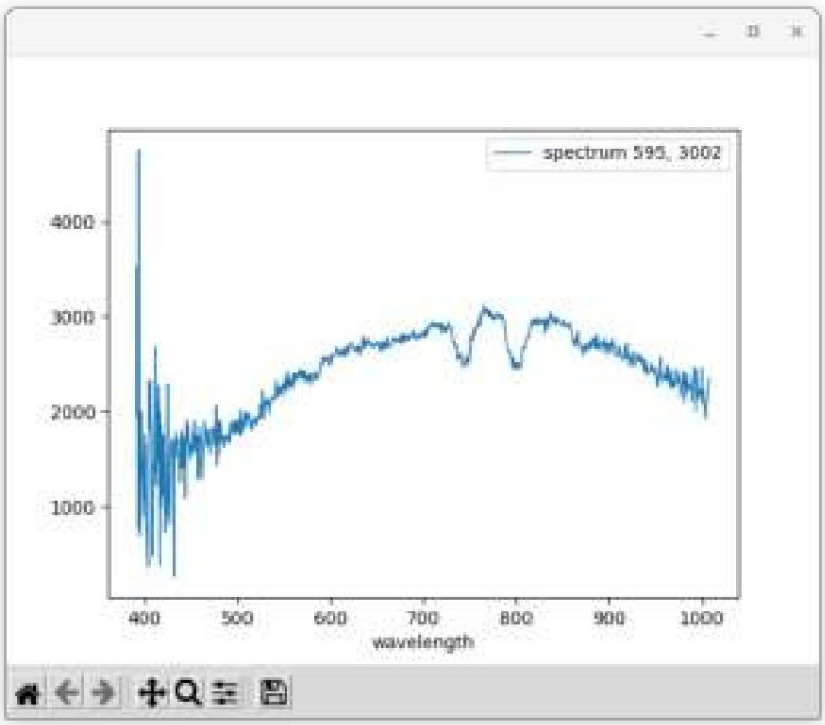


Рисунок 3. Окно графика, показывающее спектр местоположения пикселя 595, 3002. Спектр показывает две особенности поглощения при 740 и 800 нм. Эти особенности обусловлены редкоземельными элементами (в частности неодимом), содержащимися в кристаллах монацита. Спектр ниже 450 нм зашумлен из-за низкой чувствительности гиперспектральной VNIR-камеры в синем диапазоне. Панель инструментов в нижней части окна представляет собой стандартную панель инструментов для фигур matplotlib и имеет кнопки для навигации по фигуре: домой, назад, вперед, панорамирование/масштабирование, масштабирование до прямоугольника и настройка подграфика. Кнопку «Сохранить» справа можно использовать для сохранения рисунка в различных форматах, таких как PNG или PDF.

Используя «Окно значений», эти спектры также можно сохранить в виде текстового файла, конечных элементов или спектральных библиотек или импортировать в другое программное обеспечение для визуализации и дальнейшей обработки. Каталог спектров конечных членов можно использовать в качестве спектральной библиотеки в классификаторах HyPy, преобразователе спектральных углов и линейном несмешивании.

Machine Translated by Google

Программа отображения изображений HypPy использует модули Image и ImageTk из библиотеки изображений Python (PIL, теперь называемой Pillow [18]) для отображения изображений и matplotlib для создания двумерных графиков спектров.

2.3.2. Средство просмотра спектральной библиотеки

Средство просмотра спектральных библиотек — это средство просмотра спектральных библиотек типов ENVI и ASCII. После выбора входной библиотеки спектров имена спектров, содержащихся в библиотеке спектров, отображаются в списке. При двойном щелчке по названию появится окно графика, показывающее спектр выбранного минерала или материала. При выборе нескольких спектров эти спектры будут отображаться вместе в окне графика, что позволит сравнить спектры.

Программа просмотра спектральной библиотеки поддерживает различные форматы ASCII, формат спектральной библиотеки CRISM (PDS), версии 6 и 7 USGS SpecLib [12], формат ASD [19] и двухколоночные данные в виде простого текста . Средство просмотра спектральной библиотеки может рекурсивно читать все дерево каталогов. См. рисунки 4 и 5, на которых показан интерфейс программы просмотра спектральной библиотеки и окно графика, показывающее пример спектра монацита.

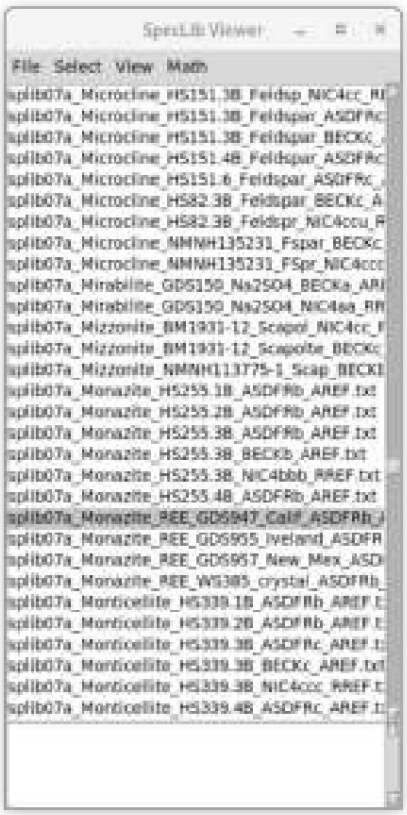


Рисунок 4. Интерфейс программы просмотра спектральной библиотеки, в данном случае загруженной спектральной библиотекой версии 7 из Геологической службы США [12].

Кроме того, средство просмотра спектральной библиотеки поддерживает манипулирование спектрами с использованием выражения «спектральной математики». К спектрам можно применять подмножество функций numpy. Индексы и срезы можно использовать в спектрах, а также доступно множество математических операторов и функций для построения операций обработки спектров и их тестирования в (выбранной) библиотеке спектров.

Спектры с разными длинами волн и диапазонами длин волн будут автоматически пересчитываться на лету. Все эти функции доступны в API и реализованы в спектральном классе HypPy (см. также раздел 2.4.2 ниже). Результаты спектральной математической операции в средстве просмотра спектральной библиотеки можно сохранить, предварительно установив выходной ка

Machine Translated by Google

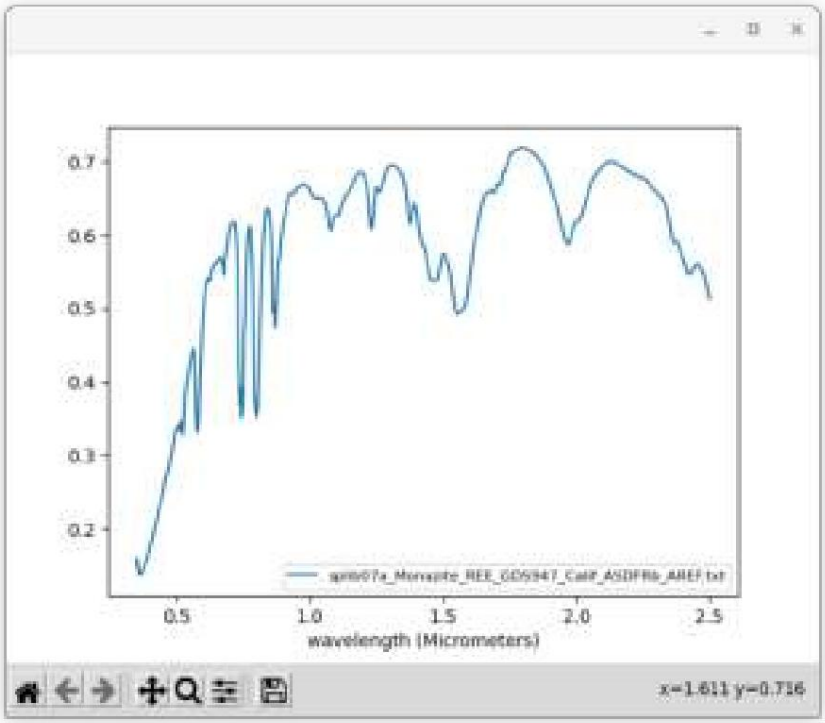


Рисунок 5. Окно графика, показывающее спектр Monazite_REE_GDS947_Calif_ASDFRb_AREF. Известно, что этот образец содержит Ce, La, Nd и Th. Обратите внимание на особенности поглощения Nd при 740 и 800 нм.

2.4. Инструменты

Далее подмножество инструментов HypPy будет обсуждаться более подробно.

2.4.1. Инструмент «Математический диапазон»

Мультиспектральные и гиперспектральные изображения можно рассматривать как набор спектральных диапазонов, и часто создаются функции для обработки таких данных подиапазонно. Инструмент Band Math реализует это.

Примером такой математической операции для полос является расчет NDVI, при котором разница между ближней инфракрасной полосой и красной полосой нормализуется путем деления на сумму ближней инфракрасной полосы и красной полосы. Для NDVI, обычно для красного диапазона, используется диапазон от 400 до 700 нм, а для инфракрасного диапазона — от 700 до 1100 нм. В следующем примере для красного цвета мы выбираем полосу около 675 нм, а для инфракрасного — полосу около 1000 нм. В HypPy NDVI изображения AVIRISNG можно рассчитать с помощью одного из следующих выражений: (1) или (2).

$$(i1[59] \quad i1[124])/(i1[59] + i1[124]), (i1(675,0) \tag{1}$$

$$i1(1000,0))/(i1(675,0) + i1(1000,0)). \tag{2}$$

В полосовой математике последовательность имен файлов изображений связана с переменными i1, i2 и так далее. В примере NDVI необходимо только одно входное изображение, связанное с переменной i1. Изображения нумеруются с 1, а номера каналов нумеруются с 0. В данном примере полоса 59 — это полоса с центральной длиной волны 676,9 нм, а полоса 124 — с центральной длиной волны 1002,4 нм. Обратите внимание, что в этом примере используется BBL, а первые семь полос, помеченные как плохие, отсутствуют.

В математике полос также возможно и даже предпочтительно использовать длины волн вместо номеров полос. Полоса с ближайшей центральной длиной волны будет выбрана автоматически. Обратите внимание, что в математике полос, когда вместо номеров полос используются длины волн, эти длины волн должны быть заключены в круглые, а не квадратные скобки. В

Machine Translated by Google

это реализуется с помощью «вызова» класса изображения. Объекты Python могут быть разработаны так, чтобы вести себя как списки или итераторы, имеющие индекс в квадратных скобках или функцию со списком параметров в круглых скобках. На рисунке 6 показан пример соотношения полос.

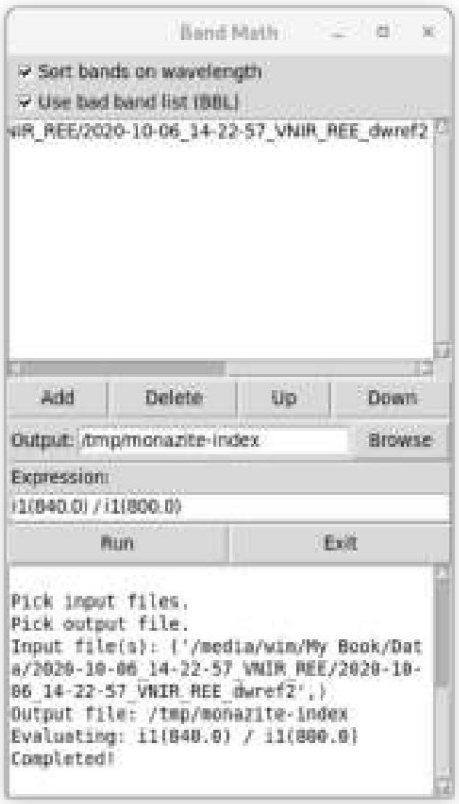


Рисунок 6. Интерфейс математического инструмента группы. Формула показывает пример соотношения полос для простого индекса монацита, основанного на плече и особенности поглощения Nd при 840 нм и 800 нм соответственно. В этом случае существует только один входной файл i1.

НурРу использует функцию Python eval() для выполнения выражений, подобных приведенному выше примеру NDVI. Групповые математические выражения — это, по сути, выражения Python. НурРу устанавливает переменные i1, i2 и т. д. Это объекты изображений НурРу, которые можно вызывать для возврата полос определенной длины волны. Связывание переменных изображения с файлами изображений является важным шагом в использовании математических выражений каналов. Обратите внимание, что синтаксис, используемый в полосовой математике, отличается от синтаксиса, используемого в спектральной математике. В полосовой математике используются объекты изображения, а в спектральной математике — объекты спектра; см. ниже.

2.4.2. Класс спектра и спектральная математика

Спектральная математика используется для применения математических выражений к спектрам изображений или библиотекам спектров. В групповой математике используется групповая парадигма; спектральная математика использует парадигму спектра . В спектральной математике изображение обрабатывается попиксельно. В полосовой математике изображение обрабатывается по полосам. Спектральная математика опирается на функциональность, реализованную в классе спектра. Интерфейс инструмента спектральной математики аналогичен инструменту математики диапазона (см. рисунок 6), но выражение должно следовать синтаксису класса спектра.

Класс спектра был создан для манипулирования спектрами. Идея состоит в том, что объект спектра содержит информацию о длинах волн, а также о значениях, будь то коэффициент отражения, яркость или другие данные. Класс спектра предлагает доступ к данным, а также функции для работы со спектральными данными.

В списке ниже приведены некоторые свойства объекта-контейнера спектра:

• Объект спектра содержит длины волн и значения. • Объект спектра может содержать имя и описание.

Machine Translated by Google

• Индекс можно использовать для выбора значений или создания подмножеств спектра. •
Целое число в качестве индекса обозначает номер диапазона; поплавок обозначает длину волны.
• Математические операторы, такие как +, , * и/или, могут использоваться в спектрах.
• Объект спектра содержит множество операторов и методов. В настоящее время реализовано около 150
функций и 20 операторов (см. руководство NurPy по спектральной математике). • При работе со спектрами с разными длинами волн спектры автоматически
пересчитывается на лету.

Как было показано ранее, спектральные индексы могут быть реализованы с использованием математики полос. Спектральный индекс должен иметь высокое значение, если определенный минерал или материал присутствует, и низкое значение, если он отсутствует. Пример такого индекса был приведен в предыдущем разделе, где мы представили пример NDVI.

Однако если изображение содержит шум, то результирующий NDVI также может содержать шум. В таком случае может быть полезно работать со средними значениями в диапазонах полос, а не с отдельными полосами. В математике диапазонов невозможно использовать диапазоны диапазонов, поскольку диапазоны диапазона должны обрабатываться один за другим. Однако в спектральной математике спектральный диапазон можно получить, используя срез спектра.

Спектральное математическое выражение для NDVI теперь может выглядеть следующим образом (3):

$$(S1[700.0 : 1100.0].mean() \quad S1[400.0 : 700.0].mean())/ (S1[700.0 : 1100.0].mean() + S1[400.0 : 700.0].mean()), \tag{3}$$

где S1 обозначает спектры первого изображения. Спектры последующих входных изображений имеют номера S2, S3 и т.д. Однако в этом примере используется только одно изображение. Квадратные скобки содержат то, что в Python называется срезами и обозначается двоеточием «:».
Обратите внимание, что в качестве индексов используются числа с плавающей запятой, которые обозначают длины волн. Целые числа как индексы интерпретируются как номера полос. Результатом среза спектра является новый объект спектра , содержащий подмножество исходного объекта спектра. В этом случае подмножество будет содержать спектральные значения примерно от 400 до 700 нм и от 700 до 1100 нм.

Длины волн являются приблизительными, поскольку эти длины волн соответствуют номерам ближайших полос, которые могут не совпадать с длинами волн центральных длин волн.
NurPy не генерирует ошибок или предупреждений при преобразовании длин волн в диапазоны, чтобы отдать приоритет скорости над проверками качества, поскольку проверки потребляют значительное время процессора. Однако пользователи и программисты могут проверить запрошенную и полученную длину волны, используя функции wave2index() и index2wavelength().

Впоследствии к этим подмножествам используется функция усреднения mean(). Другими словами, результатом будет NDVI, рассчитанный на основе среднего спектра между 400 нм и 700 нм и спектра между 700 нм и 1100 нм. Таким образом, можно построить спектральные индексы, которые работают с диапазонами длин волн, а не с отдельными полосами. Очевидным преимуществом является то, что это снижает шум на выходе. Обратной стороной является то, что такие выражения выполняются медленнее, чем пошаговые операции. Более того, в приведенном выше примере срезы и средние значения вычисляются дважды, поскольку инструмент спектральной математики поддерживает только одно выражение. Однако, умело используя оператор моржа ':=', который был введен в Python версии 3.8 [20], можно предотвратить двойные вычисления. Тогда выражение для NDVI будет выглядеть как выражение (4):

$$((ir := S1[700.0 : 1100.0].mean()) \quad (красный := S1[400.0 : 700.0].mean())))/ (ir + красный), \tag{4}$$

в котором среднее значение инфракрасного диапазона хранится в переменной ir, а среднее значение красного диапазона хранится в переменной red соответственно, и эти значения извлекаются позже в выражении для вычисления знаменателя.

Обратите внимание, что в уравнениях (3) и (4) мы используем широкий диапазон для красного и инфракрасного спектра , используя поглощение хлорофилла в видимой части спектра (от 400 до 700 нм) в качестве красного цвета и отражение от структуры клеток листа. в ближней инфракрасной части спектра

Machine Translated by Google

(от 700 до 1100 нм) как ИК. Конкретные диапазоны длин волн могут различаться в зависимости от используемого датчика дистанционного зондирования или технологии. Например, Такер [21] использовал диапазон от 630 до 690 нм для красного цвета и от 750 до 800 нм для инфракрасного диапазона, которые являются диапазонами спектральных полос Landsat MSS 5 и 7 соответственно.

HyrPy использует функцию Python eval() для выполнения выражений, подобных приведенному выше примеру усреднения NDVI. Спектральные математические выражения — это выражения Python. Однако HyrPy устанавливает переменные S1, S2 и т. д., которые представляют собой объекты спектра HyrPy, которые можно использовать в качестве класса контейнера Python. В этом смысле поведение объекта спектра похоже на объект списка Python, который можно индексировать, нарезать, перебирать и т. д.

В HyrPy спектральную математику можно использовать как в инструменте спектральной математики, так и в средстве просмотра спектральной библиотеки.

2.4.3. Пространственно-спектральные фильтры

Пространственно-спектральные фильтры учитывают как спектральную, так и пространственную информацию . HyrPy имеет три типа пространственно-спектральных фильтров: градиентные фильтры, средние фильтры и фильтры биннинга. Градиентные фильтры реализуются путем рассмотрения локального пространственного ядра спектров как векторов и расчета взвешенной разницы между этими векторами. Фильтры среднего вычисляют взвешенную сумму этих векторов. Фильтры биннинга уменьшают количество выборок за счет агрегирования значений.

В HyrPy есть инструмент для гиперспектральной фильтрации границ. Такие фильтры можно использовать для обнаружения спектральной изменчивости сцены. Области с высоким значением являются спектрально неоднородными, а области с низкими значениями — спектрально однородными. Результаты таких фильтров могут служить первым шагом в оценке качества изображения, классификации или сегментации. Результатом работы градиентных фильтров всегда является однополосное изображение с уровнем серого. Инструмент HyrPy, называемый гиперспектральным градиентом, реализует ряд направленных (вверх, вниз и две диагонали) и ненаправленных фильтров (острый и Собел). Был реализован ряд мер расстояния , которые можно использовать в сочетании с этими фильтрами.

Измерение расстояния спектрального угла (SA) учитывает только спектральную информацию входного изображения, поскольку спектральный угол нечувствителен к интенсивности входного спектра. С другой стороны, мера расстояния разности интенсивностей (ID) учитывает только информацию об интенсивности спектра, а не спектральную информацию. Мера евклидова расстояния (ED) чувствительна как к спектральной информации, так и к информации об интенсивности спектров, хотя большую часть времени кажется, что в ED доминирует интенсивность входных спектров. Помимо стандартных мер спектральных разностей, обсуждавшихся выше, доступны две другие меры расстояния, которые широко используются в гиперспектральном анализе : расстояние Брея-Кёртиса (BC) и дивергенция спектральной информации (SID) [22].

Все эти меры расстояния имеют свои свойства относительно спектрального сходства.

Если градиентные фильтры учитывают двумерный набор полных спектров 3 × 3, то фильтры среднего учитывают меньший трехмерный набор спектральных значений. Фильтры среднего значения применяют ядро 3 × 3 × 3 к кубу гиперспектральных данных. Фильтр может учитывать все значения ядра или ограниченный набор ядер. Текущие ядра в HyrPy вычисляют средневзвешенное значение 27, 19 или 7 значений ядра. Реализованы две функции усреднения: среднее и медиана. Средний фильтр является линейным фильтром и, следовательно, его влияние на изображение можно предсказать. Медианный фильтр уменьшает влияние выбросов. В этом отношении медианный фильтр может быть полезен для фильтрации шума перца и соли. Однако эффект медианного фильтра предсказать не так легко, как эффект среднего фильтра, что может привести к неожиданным эффектам на выходном изображении. В некоторых случаях медианные фильтры могут иметь эффект, сравнимый с морфологическими фильтрами, в том смысле, что области кажутся сжимающимися или увеличивающимися, что может сбивать с толку и, таким образом, усложнять интерпретацию полученного изображения.

Результатом пространственно-спектрального среднего фильтра всегда является гиперспектральный куб тех же размеров, что и входные данные. Однако, как и в случае с обычными двумерными пространственными фильтрами, следует учитывать краевые эффекты. Первое и последнее значения в любом из трех измерений (x, y и полоса) обрабатываются иначе, чем остальная часть гиперспектрального куба.

Machine Translated by Google

В дополнение к этим фильтрам HyrPy имеет ряд инструментов биннинга: пространственного, спектрального и пространственно-спектрального биннинга, чтобы уменьшить количество выборок в кубе изображения. Эти инструменты можно использовать для уменьшения объема данных, уменьшения шума за счет усреднения и ускорения последующих этапов обработки. Например, пространственно-спектральное бинирование xy = 3 × 3 и z = 11 уменьшит объём данных почти в 100 раз (3 × 3 × 11 = 99) и уменьшит стандартное отклонение шума почти в раз. 10 (99).

2.4.4. Устройство отображения

длин волн. Картирование физических свойств по гиперспектральным изображениям является сложным процессом и часто зависит от наличия спектральных библиотек целевых материалов. Многие существующие методы зависят от субъективных решений интерпретатора при выборе конечных членов и последующей обработке изображений правил. Если предварительная информация недоступна, ее следует извлечь путем характеристики характеристик спектрального поглощения. На мультиспектральных изображениях такие особенности можно найти с помощью индексов, основанных на соотношении полос и глубине полос; см., например, [23]. Однако гиперспектральные изображения показывают больше спектральных деталей особенностей, что открывает возможность использования более сложных методов.

Особенности поглощения спектральных кривых важны для анализа и интерпретации. Такие особенности могут быть параметризованы как положение центра длины волны, глубина, площадь и асимметрия [24]. Обычно наиболее важными характеристиками являются положение центра длины волны и глубина основной особенности поглощения. Положение характеристики поглощения по длине волны можно использовать для идентификации минералов, поскольку многие минералы имеют свои собственные уникальные длины волн поглощения. Глубина такой особенности указывает на обилие минерала .

Центральная длина волны и глубина самого глубокого объекта и последующих самых глубоких объектов могут быть определены автоматически. Эти параметры можно рассчитать для всех пикселей (спектров) гиперспектрального изображения и представить в виде изображений уровня серого, представляющих длину волны и глубину объекта.

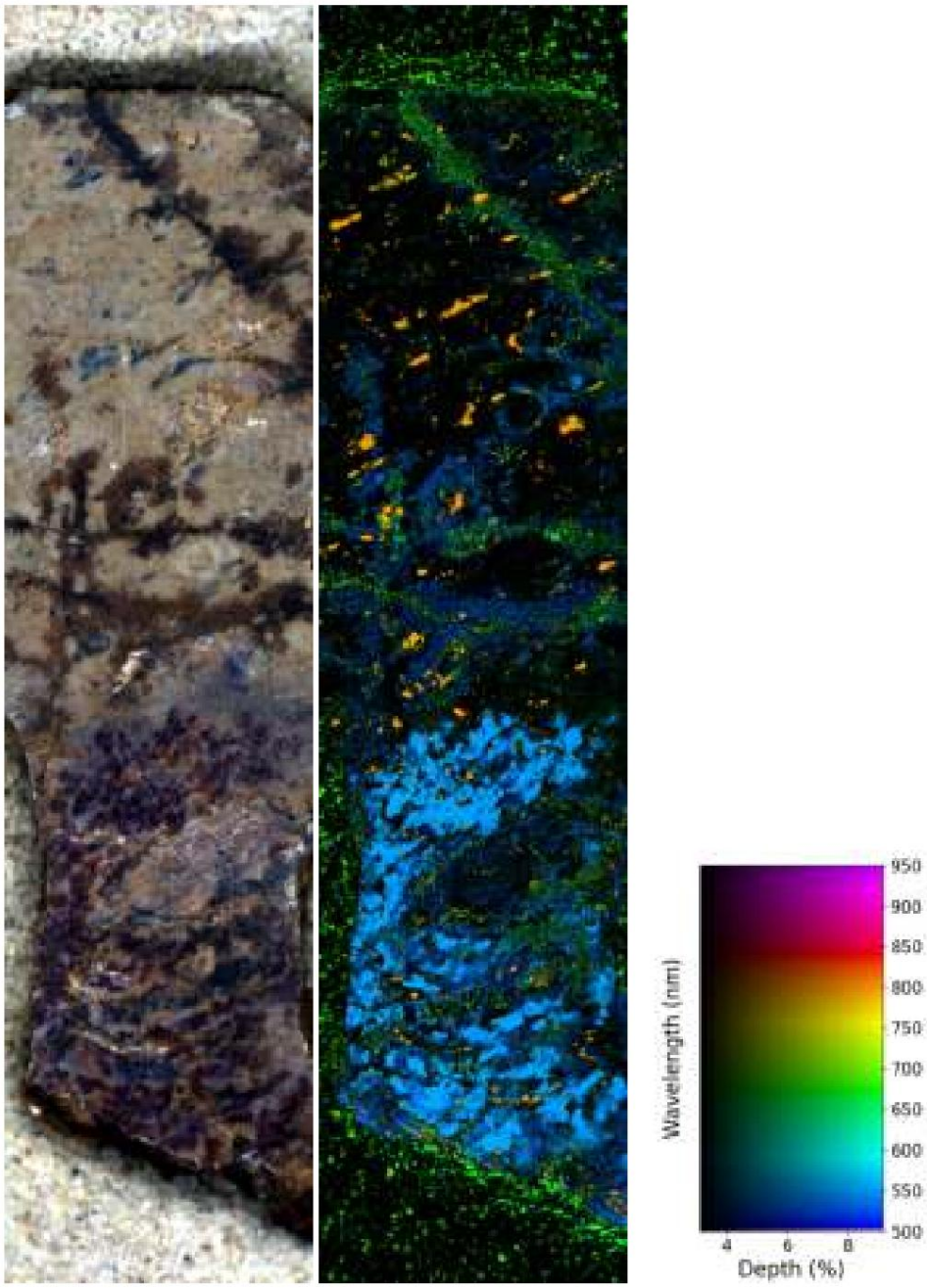
Впоследствии полученные изображения длины волны и глубины можно объединить с помощью преобразования HSI (оттенок-насыщенность-интенсивность) [17] таким образом, чтобы результирующее изображение отображало длину волны в цвете (оттенок). Интенсивность представляет глубину объекта. Для насыщенности установлено значение 1. Инструмент отображения длин волн HyrPy автоматически определяет длину волны и глубину самого глубокого поглощения всех пикселей гиперспектрального куба и объединяет их в одно цветное изображение.

На рис . 7a показан образец горной породы в композите естественного цвета. На рисунке 7b показан результат работы преобразователя длин волн; На рис . 7в показана легенда к карте длин волн. Монацит, содержащий РЗЭ, виден в виде оранжевых зерен. Зелёный и синий — это карбонаты. Дальнейшее обсуждение этого образца породы см. в [16].

Устройство отображения длин волн является неконтролируемым, простым и повторяемым; большие площади могут быть картированы сразу, а детали сохраняются в фокусе выбранного спектрального диапазона. Результат, называемый картой длин волн, может быть понятен пользователям с ограниченным опытом обработки изображений. Однако, хотя цвета могут указывать на присутствие определенных минералов, полученная карта длин волн не является картой минералов. Карту длин волн можно использовать в качестве этапа предварительной классификации, чтобы понять спектральную изменчивость сцены . Знания, полученные из карты длин волн, полезны для дополнительных этапов создания карты минералов.

Устройство отображения длин волн полагается только на два входных параметра: диапазон длин волн, на котором нужно сфокусироваться, и растяжение, которое будет использоваться для определения глубины объекта. Устройство отображения длин волн было впервые опубликовано в van Ruitenbeek et al. [25], а руководство по этому методу опубликовано в [24].

Machine Translated by Google



(а) Цветной композит (б) Карта длин волн (с) Легенда

Рис. 7. (а) Композиция естественного цвета образца породы 15-91-FE (красный 650 нм, зеленый 550 нм и синий 450 нм). (б) Карта длин волн в диапазоне от 500 до 950 нм показывает длину волны и глубину самой глубокой детали. Монацит виден оранжевым цветом (показывающим структуру Nd при длине волны 800 нм) или желто-зеленым, когда особенность при длине волны 740 нм оказывается глубже, чем особенность при длине волны 800 нм. (в) Легенда к карте длин волн.

2.4.5. Деревья решений

Дерево решений — это тип классификатора, который состоит из серии двоичных решений, используемых для определения класса для каждого входного пикселя. Решения основаны на характеристиках наборов данных. Каждое решение разделяет данные на один из двух потенциальных классов или групп классов. Решения — это выражения, применяемые к одному изображению или набору изобра Каждое из этих изображений представляет собой свойство входных пикселей.

Одним из таких свойств, например, может быть положение длины волны самого глубокого поглощения в спектре входного пикселя. Выражение может быть построено для проверки

Machine Translated by Google

может ли входной пиксель содержать глинистый минерал с самой глубокой поглощающей способностью около 2,2 мкм. Последующие решения могли бы проверить характеристики воды этого пикселя, чтобы увидеть, имеет ли материал высокий индекс кристалличности. После ряда решений пикселю будет присвоен класс.

Деревья решений полезны для построения классификаторов, основанных на физических свойствах, а не на статистических выводах. Конечно, это возможно только в том случае, если все свойства, необходимые для деревьев решений, могут быть установлены с достаточной уверенностью.

Такой пакет, как ENVI, поддерживает построение и визуализацию новых деревьев решений. Однако ENVI имеет ряд недостатков. Способ реализации деревьев решений в ENVI довольно жесткий, поскольку после построения узла в дереве его место в дереве решений фиксируется. В окончательном дереве решений ENVI невозможно перемещать элементы. В ENVI, если вы хотите изменить дерево решений, вам придется перестроить его с нуля.

HypPy можно использовать для выполнения деревьев решений ENVI. Единственное условие — выражения, используемые для принятия решений, могут быть переведены в выражения Python.

В дополнение к формату дерева решений ENVI, HypPy имеет собственный формат дерева решений. Этот формат представляет собой текстовый формат, который можно редактировать с помощью текстового редактора.

Для визуализации деревьев решений HypPy использует пакет Graphviz [26]. Для этого дерево решений ENVI или HypPy необходимо сначала преобразовать в «точечный» формат. Язык точек — это обширный язык описания графов. Двоичные деревья решений, используемые ENVI и HypPy, представляют собой крошечное подмножество всех возможных графиков и типов графиков, поддерживаемых пакетом Graphviz.

2.4.6. Зональная статистика

Инструмент зональной статистики вычисляет статистику по пикселям входного изображения в зонах, определенных изображением зоны. Входное изображение представляет собой обычное гиперспектральное или мультиспектральное изображение. Изображение зоны может быть результатом изображения классификации или изображения маски. Для каждого отдельного значения в изображении зоны на входном изображении запускается функция агрегирования для всех местоположений, имеющих это уникальное значение в изображении зоны. В настоящее время реализованы следующие функции агрегирования: среднее значение, медиана, минимум, максимум, стандартное отклонение и среднее значение плюс или минус двойное стандартное отклонение.

Результаты зональной статистики полезны для определения свойств областей входного изображения. Например, в случае шума во входном изображении средний спектр области может выявить больше спектральных деталей, чем спектр одного пикселя, в котором спектральные детали объектов могут быть заглушены шумом.

3. Области применения

Ниже показано применение HypPy в нескольких проектах в сфере образования, консалтинга и исследования будут рассмотрены более подробно.

3.1. Образование

HypPy — один из пакетов программного обеспечения, используемых на наших курсах спектральной геологии в Университете Твенте [27]. Ясность интерфейса HypPy делает его отличным инструментом для студентов, начинающих изучать спектральную геологию. Одним из примеров опубликованных студенческих исследований с использованием HypPy является [28]. Учебное пособие по анализу особенностей поглощения для поиска руды с помощью геологического дистанционного зондирования см. в [24].

HypPy можно использовать на краткосрочных курсах для изучения основных концепций гиперспектральной визуализации и выполнения специализированных операций по обработке и интерпретации гиперспектральных данных. Например, HypPy использовался в курсе «Дистанционное зондирование и минеральная спектроскопия для геологоразведочных и горнодобывающих геологов», предлагаемом в Стамбуле [29].

В декабре 2022 года HypPy использовался на спектральном семинаре во время ежегодной конференции GRSG [30]. Семинар включал обзор теории и практической реализации спектральных методов идентификации минералов, сопровождавшийся практическим лабораторным занятием, включающим применение аналитики спектральной обработки к недавно полученному набору аэроспектральных данных высокого разрешения, охватывающему видимую и около-

Machine Translated by Google

инфракрасные спектры (VNIR от 400 нм до 1 мкм), коротковолновый инфракрасный (SWIR от 1 до 2,5 мкм) и длинноволновый инфракрасный (LWIR от 8 до 15 мкм) спектральные диапазоны. В ходе лабораторного сеанса использовалось программное обеспечение гиперспектральной обработки HurPy, что обеспечивало беспрепятственный доступ к данным проб.

3.2. Консалтинг

Мы изучили возможность использования лабораторных гиперспектральных изображений в коротковолновом инфракрасном диапазоне (SWIR) для коммерческого партнера для идентификации и картирования минералогического состава конкретных образцов осадочного бурового керна из недр Нидерландов. Возможности написания сценариев HurPy оказались очень важными в этом проекте.

Наши результаты показывают, что использование лабораторных гиперспектральных изображений в коротковолновом инфракрасном диапазоне (SWIR) с пространственной дискретизацией 200 мкм предлагает ценный неразрушающий подход для идентификации и картирования белых слюдяных, глинистых и карбонатных минералов в осадочных бурах. ядра. Совместно с партнерами из Гронингенского университета с помощью HurPy нам удалось построить прототип цепочки обработки для количественного определения содержания минералов и получения ряда геофизических свойств буровых кернов.

3.3. Исследовать

Первая работа над HurPy началась с пилотного проекта по картографированию планет Марса для Европейского космического агентства (EKA). Пилотное исследование было направлено на перенос современных цифровых методов и рабочих процессов, используемых при геологическом картировании Земли, на планетарное картографирование. В центре внимания этого исследования были наборы данных EKA, полученные со стереокамеры высокого разрешения (HRSC) и гиперспектрального прибора видимого и инфракрасного минералогического картографического спектрометра (OMEGA) на борту Mars Express [31]. Первым шагом рабочего процесса была обработка необработанных данных, чтобы выделить интересующие особенности производных информацио В их число входили цифровые модели рельефа, извлеченные из стереоданных, и карты минералов, созданные с использованием гиперспектральных данных. Эти инструменты использовались для интерпретации данных и создания геологической карты района Нили Фоссэ. Это исследование проложило путь к использованию метода картирования длин волн для регулярного картирования планет Марс и Земля [7].

HurPy представляет методологию обработки, предназначенную для преобразования данных OMEGA из измерения яркости на датчике в коэффициент отражения поверхности. Для решения конкретных проблем, связанных с данными OMEGA, мы разработали инновационные подходы для определения спектра пропускания для каждой сцены , проведения атмосферной коррекции и фильтрации систематического и случайного шума. См., например, van Ruitenbeek et al. [25]. Эти фильтры теперь являются частью HurPy. Методология обработки генерирует изображения отражательной способности поверхности, что позволяет извлекать однопиксельные спектры для идентификации особенностей поверхности. В соответствии с различными предыдущими исследованиями, наши результаты предполагают, среди прочего, существование богатых железом глин на поверхности Марса.

Картирование физических свойств по гиперспектральным изображениям сложно и часто зависит от доступности спектральных библиотек целевых материалов. Многие существующие методы зависят от субъективных решений интерпретатора при выборе конечных членов и постобработке изображений правил. Если предварительная информация недоступна, ее следует извлечь путем характеристики характеристик спектрального поглощения. В Ван Рютенбеке и др. В [25] они описывают метод, называемый картографом длин волн (см. раздел 2.4.4), предназначенный для автоматического определения минимального коэффициента отражения в диапазоне длин волн. Мы пришли к выводу , что картограф длин волн оказывается полезным инструментом для объективного выбора конечных членов, картирования минералов и интерпретации областей, демонстрирующих зональность минеральных изменений, с использованием гиперспектральных изображений. Полезность картографа длин волн уже была признана в других исследованиях. Например, он был использован для картирования обнажений минералов литийсодержащих пегматитов в пилотном карьере, расположенном недалеко от Уиса, Нам

В другом исследовании к спектральному классу была добавлена функция энтропии, чтобы можно было рассчитать индекс сортировочных процессов в гидротермальных системах. Гидротермальные процессы влияют на состав горных пород посредством химических реакций и течения флюидов, и это

Machine Translated by Google

В статье показано, что остальные закономерности таких процессов можно визуализировать с помощью энтропии Шеннона. Применение энтропии полезно для изучения гидротермальных месторождений полезных ископаемых, а также может быть ценным при изучении среды ранней жизни.

Функции, которые уже реализованы в классе спектра HyrPy, можно использовать для добавления шума к данным. Это было использовано для моделирования бортовых гиперспектральных датчиков SWIR для картирования газа метана (CH4) [33]. Результаты исследования показали , что три действующих в настоящее время бортовых спектрометра визуализации не смогли обнаружить шлейф CH4 . В документе подчеркивается важность как спектральной выборки, так и радиометрической калибровки для успешного картирования метана.

Современные спектрометры визуализации компактны, легки, экономичны и применимы в лабораторных условиях. Точные гиперспектральные изображения требуют минимального шума, точной калибровки и постоянной однородности отклика. Периодические проверки желательны из-за старения оптических и электронных компонентов. В другом исследовании мы представляем метод количественной оценки геометрических искажений улыбки и трапецеидальных искажений на тестовых изображениях [34].
Измерения проводятся с помощью простых моделей улыбки и трапецеидального искажения, а также уже существующих инструментов HyrPy. Кроме того, для получения значений улыбки и трапецеидального искажения применяются функции интерполяции квадратичным сплайном, нормализованной взаимной корреляции и оптимизации.

Результаты показывают, что улыбка и краеугольный камень лабораторного спектрометра визуализации можно определить с точностью до субпикселя.

4. Обсуждение

Помимо HyrPy, несколько других пакетов предлагают функции обработки гиперспектральных изображений.

Например, QUANTools был реализован с использованием языка программирования ENVI/IDL для автоматического обнаружения нескольких параметров функции поглощения [35]. Однако для запуска QUANTools необходима рабочая версия ENVI/IDL, а пакет QUANTools предлагает ограниченную функциональность, аналогичную одному конкретному инструменту в HyrPy, который называется преобразователем длин волн.

PySptools — это модуль Python, реализующий спектральные и гиперспектральные алгоритмы [36]. Однако это пакет или библиотека, и вам придется написать код, прежде чем вы сможете его использовать. Фактически, HyrPy использует pysptools для своего линейного классификатора несмешивания. В отличие от pysptools, HyrPy предлагает пользовательский интерфейс для этой функциональности.

Дополнительный заслуживающий внимания набор инструментов известен как Spectrum Python (SPy) [37]. SPy — это модуль Python, предназначенный для обработки данных гиперспектральных изображений и содержащий функции для таких задач, как чтение, отображение, манипулирование и классификация гиперспектральных изображений. Примечательно, что SPy ограничен интерактивным использованием из командной строки Python или с помощью сценариев Python, в отличие от HyrPy.

Программные пакеты Orfeo и EnMAPBox являются мощными инструментами, используемыми в приложениях дистанционного зондирования и обработки изображений [38,39]. Они широко используются в области дистанционного зондирования и предоставляют широкие возможности для обработки и анализа спутниковых изображений. Хотя эти пакеты ориентированы на обработку спутниковых изображений, HyrPy также предназначен для обработки общих гиперспектральных изображений, таких как проксимальные изображения, записанные гиперспектральными камерами в лаборатории.

Как и сам язык Python [6], HyrPy был разработан с упором на простоту, а не на производительность [40]. Некоторые операции в HyrPy могут выполняться относительно медленно. Однако модульность функций и инструментов можно использовать для создания экспериментальных программ или даже прототипов цепочек обработки, работающих на любом оборудовании.

Что касается скорости, будущие версии Python могут быть быстрее. Если назвать один из текущих проектов, то язык Mojo представляет собой надмножество языка Python со скоростью языка C [41].

Machine Translated by Google

5. Выводы

В этой статье мы описали дизайн, реализацию и применение пакета Python под названием Hyperspectral Python, HyrPy. Идея HyrPy заключается в том, чтобы иметь возможность обрабатывать гиперспектральные изображения с помощью бесплатного программного обеспечения с открытым исходным кодом. HyrPy был разработан с использованием языка Python и в значительной степени опирается на возможности обработки массивов таких пакетов, как NumPy и SciPy. HyrPy можно использовать для раннего анализа данных и создания прототипов цепочек обработки. HyrPy был разработан с учетом простоты и практичности спектрометрии изображений. Одновременно он включает в себя несколько инновационных и мощных концепций.

HyrPy — это пакет, находящийся в разработке и для разработки, и он доказал свои достоинства. Простота и гибкая структура меню верхнего уровня делают HyrPy привлекательным для использования в сфере образования и краткосрочных курсов. Новые методы исследования можно быстро запрограммировать и протестировать. Примерами таких исследовательских проектов являются пилотный проект EKA по картированию планет Марса и исследовательский проект по определению краеугольного камня и улыбки гиперспектральных линейных камер.

В проектах его использовали для построения целых цепочек обработки от гиперспектральных данных до информации. Короче говоря, HyrPy можно использовать как платформу для генерации и тестирования новых идей по обработке гиперспектральных данных. Ядро чтения и записи гиперспектральных изображений, картограф длин волн, гиперспектральные градиентные фильтры, пространственно-спектральные фильтры, инструменты полосовой и спектральной математики с собственным языком выражений, инструменты дерева решений, использование зональной статистики, а также специальная библиотека изображений и спектров. зрители являются хорошим примером этого.

Мы ожидаем, что HyrPy также принесет большую пользу другим организациям и частным лицам, занимающимся исследованиями, образованием и проектами, использующими гиперспектральные данные.

Вклад авторов: концептуализация, Всемирный банк и FvR; методология, ВБ; программное обеспечение, ВБ; валидация, WB, FvR, HvdW и CH; ресурсы, AD; написание – подготовка первоначального проекта, ВБ; написание — обзор и редактирование, HvdW; визуализация, ББ; надзор, HvdW и FvdM; администрация проекта , ВБ; приобретение финансирования, FvdM и FvR. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Мы признательны за финансовую поддержку, полученную от Европейского космического агентства (проект планетарного картографирования Марса) и Вейнанда ван Эйндховена из Deep Atlas. Финансирование, предоставленное этими организациями, сыграло решающую роль в успешном выполнении нашего исследовательского проекта.

Заявление о доступности данных: HyrPy можно загрузить с Github и Zenodo, см. [42]; Имя: HyrPy (Гиперспектральный Python); Лицензия: Стандартная общественная лицензия на графический процессор версии 3; Контакт: whbakker@utwente.nl; Требования к оборудованию: обычный ноутбук или настольный компьютер с ОС Windows, Mac или Linux; Язык программы: Python версии 3; Минимальные требования к программному обеспечению: numpy, scipy, Pillow, Matplotlib, pysptools; Размер программы: 12,2 МБ.

Благодарности: Мы благодарны Университету Твенте за предоставление необходимых ресурсов и оборудования, которые в значительной степени способствовали бесперебойному развитию наших исследований. Кроме того, мы благодарны всем студентам, которые прямо или косвенно внесли свой вклад в эту работу, используя HyrPy во время учебы. Их сотрудничество и поддержка сыграли важную роль в формировании HyrPy.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Сокращения

В данной рукописи использованы следующие сокращения:

API Интерфейс программирования
Список плохих групп BBL
БИЛ Полоса, чередующаяся по линии
Последовательный диапазон частот BSQ
Интерфейс командной строки CLI
Графический пользовательский интерфейс GUI
SWIR Коротковолновый инфракрасный порт
VNIR видимый и ближний инфракрасный диапазон

Machine Translated by Google

Ссылки

1. NV5 Geospatial Solutions Inc. ENVI — среда для визуализации изображений. 2023. Доступно онлайн: https://www.nv5.geospatialsoftware.com/docs/using_envi_Home.html (по состоянию на 24 ноября 2023 г.).
2. Прокакис Э. Бесплатное программное обеспечение с открытым исходным кодом: свобода, прозрачность и эффективность в эпоху цифровизации. J. Политическая этика Новая Технол. АИ 2022, 1, e31230. [Перекрестная ссылка]
3. Харрис, ЧР; Миллман, К.Дж.; ван дер Уолт, SJ; Гоммерс, Р.; Виртанен, П.; Курнапо, Д.; Визер, Э.; Тейлор, Дж.; Берг, С.; Смит, Нью-Джерси; и др. Программирование массивов с помощью NumPy. Природа 2020, 585, 357–362. [Перекрестная ссылка] [ПабМед]
4. Виртанен П.; Гоммерс, Р.; Олифант, ТЕ; Хаберланд, М.; Редди, Т.; Курнапо, Д.; Буровский, Э.; Петерсон, П.; Векессер, В.; Брайт, Дж.; и др. SciPy 1.0: Фундаментальные алгоритмы научных вычислений на Python. Нат. Методы 2020, 17, 261–272. [Перекрестная ссылка] [ПабМед]
5. Хантер, Дж. Д. Matplotlib: среда 2D-графики. Вычислить. наук. англ. 2007, 9, 90–95. [Перекрестная ссылка] 6. ван Россум, Г.; Дрейк, Флорида Справочное руководство по Python 3; CreateSpace: Скоттс-Вэлли, Калифорния, США, 2009 г. Доступно онлайн: https://dl.acm.org/doi/book/10.5555/1593511 (по состоянию на 31 июля 2024 г.).
7. Трагхайм, Д.Г.; Марш, Ш.; Педли, Колорадо; Нэпьер, BR; Бейтсон, Л.; Смит, АГ; Маршан, АР; Ганнинк, Дж.Л.; Остук, JHP; Мюллер, Дж. П. Пилотный проект планетарного картирования Марса; Европейское космическое агентство (ЕКА): Париж, Франция, 2010 г.
8. Константинидес, К.; Расуре, Дж. Р. Среда разработки программного обеспечения «Хорос» для обработки изображений и сигналов. IEEE Транс. Процесс изображения. 1994, 3, 243–252. [Перекрестная ссылка] [ПабМед]
9. Янг, М.; Аргиро, Д.; Кубица, С. Кантата. ACM SIGGRAPH Comput. Граф. 1995, 29, 22–24. [Перекрестная ссылка]
10. Оустерхаут, Дж. К.; Джонс, К.; Фостер-Джонсон, Э.; Товарищи, Д.; Гриффин, Б.; Велтон, Д. Tcl и набор инструментов Tk, 2-е изд.; Аддисион-Уэсли Серия профессиональных вычислений; Аддисон-Уэсли: река Аппер-Сэддл, Нью-Джерси, США, 2009 г.
11. Роузман, М. Современный Tkinter для занятых разработчиков Python: быстро научитесь создавать великолепные пользовательские интерфейсы для Windows, Mac и Linux с использованием стандартного набора инструментов графического интерфейса Python; Вечерняя пресса: Виктория, Британская Колумбия, Канада, 2020.
12. Кокалы, РФ; Кларк, Р.Н.; Суэйзи, Джорджия; Ливо, Кентукки; Хофен, ТМ; Пирсон, Северная Каролина; Мудрый, РА; Бензель, ВМ; Понижает, ГА; Дрисколл, РЛ; и др. Спектральная библиотека Геологической службы США, версия 7; Технический отчет; Геологическая служба США: Рестон, Вирджиния, США, 2017 г. [CrossRef]
13. Бибринг, Дж. П.; Ланжевен, Ю.; Альтьери, Ф.; Арвидсон, Р.; Бейлюд, Г.; Бертель, М.; Дуте, С.; Дроссарт, П.; Энкреназ, Т.; Забудь, Ф.; и др. ОМЕГА: Обсерватория минералогии, воды, глаз и активности; Европейское космическое агентство, ЕКА (специальное издание): Париж, Франция, 2009 г.; СП-1291; стр. 75–95. Доступно онлайн: https://sci.esa.int/s/WnjXOYW . (по состоянию на 31 июля 2024 г.).
14. Вейн, Г.; Грин, Род-Айленд; Крис, Т.Г.; Энмарк, ХТ; Хансен, Э.Г.; Портер, В.М. Воздушный видимый/инфракрасный спектрометр (AVIRIS). Дистанционный датчик окружающей среды. 1993, 44, 127–143. [Перекрестная ссылка]
15. Цзя, Г.; Хуэни, А.; Тао, Д.; Гэн, Р.; Шепман, Мэн; Чжао, Х. Восстановление спектральной отражательной способности сверхвысокого разрешения по данным спектрометра дистанционного зондирования. Опция Экспресс 2016, 24, 19905–19919. [Перекрестная ссылка] [ПабМед]
16. Мариен, К.; Дейкстра, АХ; Уилкинс, К. Гидротермальные изменения карбонатита в комплексе Фен, Норвегия: минералогия, геохимия и значение для формирования ресурсов редкоземельных элементов. Минерал. Маг. 2018, 82, C115–C131. [Перекрестная ссылка]
17. Ричардс, Дж.А.; Цзя, Х. Анализ цифровых изображений дистанционного зондирования; Springer: Берлин/Гейдельберг, Германия, 1999. [CrossRef]
18. Дрисколл, М. Пиллоу: Обработка изображений с помощью Python; Leanpub: Виктория, Британская Колумбия, Канада, 2021 г. Доступно онлайн: https://leanpub.com/подушка (по состоянию на 31 июля 2024 г.).
19. Формат файла ASD Malvern Panalytical Ltd. — версия 8. Технический отчет, Malvern Panalytical. 2017. Доступно онлайн: https: //www.malvernpanalytical.com/en/learn/knowledge-center/user-manuals/asd-file-format-v8 (по состоянию на 31 июля 2024 г.).
20. Хеттингер Р. Что нового в Python 3.8. 2019. Доступно онлайн: https://docs.python.org/3/whatsnew/3.8.html. (доступ на 24 ноября 2023 г.).
21. Такер, К.Дж. Красные и фотографические инфракрасные линейные комбинации для мониторинга растительности. Дистанционный датчик окружающей среды. 1979, 8, 127–150. [Перекрестная ссылка]
22. Чейн-И Чанг. Расхождение спектральной информации для анализа гиперспектральных изображений. В материалах Международного симпозиума по геонаукам и дистанционному зондированию IEEE 1999 г., IGARSS'99 (кат. № 99CH36293), Гамбург, Германия, 28 июня – 2 июля 1999 г.; Том 1, стр. 509–511. [Перекрестная ссылка]
23. Хьюсон, Р.Д.; Кудахи, Ти Джей; Мизухико, С.; Уэда, К.; Могер, А.Дж. Создание бесшовной геологической карты с использованием ASTER в Сломанном мире Провинция Хилл-Курнамона в Австралии. Дистанционный датчик окружающей среды. 2005, 99, 159–172. [Перекрестная ссылка]
24. Хекер, К.; ван Рейтенбек, FJA; ван дер Верфф, HMA; Баккер, Вашингтон; Хьюсон, РД; ван дер Меер, Ф.Д. Анализ особенностей спектрального поглощения при поиске руды: учебное пособие по использованию этого метода в геологическом дистанционном зондировании. IEEE Geosci. Дистанционный датчик Mag. 2019, 7, 51–71. [Перекрестная ссылка]
25. ван Рейтенбек, FJA; Баккер, Вашингтон; ван дер Верфф, HMA; Зегерс, Т.Э.; Остук, JHP; Омер, З.А.; Марш, Ш.; ван дер Меер, Ф.Д. Картирование положения длин волн самых глубоких поглощений для изучения минерального разнообразия на гиперспектральных изображениях. Планета. Космическая наука. 2014, 101, 108–117. [Перекрестная ссылка]
26. Эллсон Дж.; Ганснер, скорая помощь; Куцофиос, Э.; Норт, Южная Каролина; Вудхалл, Г. Графвиз и Dynagraph — построение статического и динамического графика Инструменты. В программном обеспечении для рисования графиков; Springer Link: Нью-Йорк, штат Нью-Йорк, США, 2004 г.; стр. 127–148. [Перекрестная ссылка]
27. ИТЦ. Магистерская программа в области геоинформатики и наблюдения Земли — 2023–2024 учебный год. 2023. Доступно онлайн: https://studyguide.itc.nl/m-geo . (по состоянию на 24 ноября 2023 г.).

Machine Translated by Google

28. Портела, Б.; Сепп, доктор медицины; ван Рейтенбек, FJA; Хекер, К.; Диллес, Дж. Х. Использование гиперспектральных изображений для идентификации сростков пирофиллита и мусковита и алунита в неглубокой эпитермальной среде медно-порфирового района Йерингтон . Рудная геология. Ред. 2021, 131, 104012. [CrossRef]

29. Портела, Б. Краткий курс — дистанционное зондирование и спектроскопия минералов для геологоразведочных и горнодобывающих геологов, 7–11 ноября 2022 г., Стамбул. 2022. Доступно онлайн: https://www.linkedin.com/posts/brunobvportela_geologicalremotesensing-istanbul-turkey-activity-7001884993555693568-bkXG . (по состоянию на 24 ноября 2023 г.).

30. Геологическая группа дистанционного зондирования. Ежегодная конференция GRSG и годовое общее собрание акционеров 2022: от орбиты до обнажения породы. 2022. Доступно онлайн: https://www.grsg.org.uk/grsg-agm-conference-2022 (по состоянию на 24 ноября 2023 г.).

31. Чикарро, А.; Мартин, доктор медицинских наук; Траутнер, Р. Миссия «Марс-Экспресс»: обзор. В «Марсианском экспрессе: европейская миссия на Красную планету»; ЕКА: Нордвейк, Нидерланды, 2004 г.; Том СП-1240. Доступно онлайн: https://sci.esa.int/documents/33745/35957/. 1567254632829-ОбзорWeb.pdf (по состоянию на 31 июля 2024 г.).

32. Бойсен, Р.; Лоренц, С.; Тиле, СТ; Фукслох, WC; Марэ, Т.; Некс, ПАМ; Глоаген, Р. Точная гиперспектральная визуализация минерализованных обнажений: пример литийсодержащих пегматитов в Уисе, Намибия. Дистанционный датчик окружающей среды. 2022, 269, 112790. [Перекрестная ссылка]

33. Скафутто, РДМ; ван дер Верфф, Х.; Баккер, Вашингтон; ван дер Меер, Ф.; де Соуза Фильо, CR. Оценка бортовых спектрометров SWIR для картирования CH4: последствия позиционирования полос, спектральной выборки и шума. Межд. Дж. Прил. Земля Обс. Геоинф. 2021, 94, 102233. [CrossRef]

34. Баккер, В.; ван дер Верфф, Х.; Ван дер Меер, Ф. Определение улыбки и краеугольного камня лабораторных гиперспектральных линейных камер. В материалах 10-го семинара по гиперспектральной визуализации и обработке сигналов: эволюция в области дистанционного зондирования (WHISPERS) 2019 г., Амстердам, Нидерланды, 24–26 сентября 2019 г.; стр. 1–5. [Перекрестная ссылка]

35. Копачкова, В.; Коука, Л. Картирование минералов на основе автоматического обнаружения множественных особенностей поглощения. EARSeL EProc. 2014, 13, 95–99. [Перекрестная ссылка]

36. Териен, К. Добро пожаловать в документацию PySptools. 2022. Доступно онлайн: https://pysptools.sourceforge.io/. (доступ на 24 ноября 2023 г.).

37. Боггс, Т. Руководство пользователя Spectral Python (SPy). 2020. Доступно онлайн: https://www.spectralpython.net/user_guide.html. (доступ 24 ноября 2023 г.).

38. Гризонне, М.; Мишель, Дж.; Пугон, В.; Инглада, Дж.; Савино, М.; Крессон, Р. Orfeo ToolBox: Обработка удаленных данных с открытым исходным кодом. восприятие изображений. Откройте Геоспат. Программное обеспечение для обработки данных. Стоять. 2017, 2, 15. [CrossRef]

39. ван дер Линден, С.; Рабе, А.; Якимов, Б.; Тиль, Ф.; Купер, С.; Окужени, А.; Хостерт, П. Интеграция спектроскопии изображений и ГИС — бесплатный анализ изображений с открытым исходным кодом в QGIS с помощью EnMAP-Box 3. В материалах OSA Optical Sensors and Sensing Congress 2021 (AIS, FTS, HISE, SENSORS, ES), Вашингтон, Округ Колумбия, США, 19–23 июля 2021 г.; Издательская группа Optica: Вашингтон, округ Колумбия, США, 2021 г.; п. HF4E.2. [Перекрестная ссылка]

40. ван Россум, Г.; Фридман Л. Почему Python 3.11 такой быстрый. 2022. Интервью на YouTube. Доступно онлайн: https://youtu.be/TLhRuZ9cJWc (по состоянию на 31 июля 2024 г.).

41. Modular Inc. Mojo — новый язык программирования. 2023. Доступно онлайн: https://www.modular.com/mojo. (доступ 24 ноябрь 2023 г.).

42. Баккер, WH, wimhbakker/hyppy: HyPPy3. Зенодо 2024. [CrossRef]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Проектирование многоканальных систем спектрального интеллекта с использованием Приближенный алгоритм дискретного преобразования Фурье для антенны Приложения для восприятия спектра на основе массивов

Арджуна Маданаяке [1], Кеттура Лоуренс 1,* , Бопаж Умеша Кумарасири 3 и Ренато [1], Сивакумар Сивасанкар [1], Тушара Гунаратне [2], Edussooriya Дж. Синтра Чамира, США [4]

1 Факультет электротехники и вычислительной техники, Международный университет Флориды, Майами, Флорида 33174, США; amadanay@fiu.edu (AM); ukuma003@fiu.edu (БУК); ssiva011@fiu.edu (CC)
2 Центр астрономии и астрофизики Герцберга, Национальный исследовательский совет Канады, Пентиктон, Британская Колумбия V0H 1K0, Канада; таким образомhara.gunaratne@nrc-cnrc.gc.ca
3 Кафедра электронной и телекоммуникационной техники Университета Моратувы, Моратува 10400, Шри-Ланка; chamira@uom.lk
4 Технологический факультет Федерального университета Пернамбуку, Каруару 55015, PE, Бразилия; rjdsc@de.ufpe.br
* Переписка: klawr041@fiu.edu

Аннотация: Радиоспектр – дефицитный и чрезвычайно ценный ресурс, требующий бережного обращения. Мониторинг в реальном времени и динамическое распределение ресурсов. Динамический доступ к спектру (DSA) — это новый парадигма управления радиоспектром, которая требует алгоритмов на основе искусственного интеллекта и машинного обучения для оптимального производительность в условиях быстро меняющихся условий канала и возможных кибератак в электромагнитной области. Быстрое обнаружение в нескольких направлениях с использованием матричных процессоров с последующим Алгоритмы на основе искусственного интеллекта и машинного обучения для обнаружения и восприятия сигналов, измеряемых с помощью Среда имеет решающее значение для обеспечения поддержки принятия решений в DSA. В рамках направленного и широкополосного Для восприятия спектра крайне необходима способность точно распределять широкополосные входные сигналы по каналам с использованием эффективного анализа Фурье. Однако мелкозернистое быстрое преобразование Фурье (БПФ) для большого числа направлений требует больших вычислительных ресурсов и приводит к увеличению площади кристалла и энергопотреблению. Мы решаем эту проблему, используя недавно предложенное приближенное дискретное преобразование Фурье. (ADFT), который имеет собственную разреженную факторизацию для реализации в реальном времени с низкой сложностью. и энергопотребление. ADFT используется для создания широкополосного многолучевого цифрового формирователя луча. и блок внимания на основе временного спектра, который отслеживает 32 дискретных направления в 32 поддиапазонах. в режиме реального времени с использованием безмножительного алгоритма с низкой вычислительной сложностью. Выход эта спектральная единица внимания применяется в качестве переменной решения для интеллектуального приемника, который адаптирует свой центральная частота и разрешение по частоте с помощью канализаторов БПФ, специально разработанных для работы в режиме реального времени мониторинг в высоком разрешении. Этот двухэтапный процесс позволяет применять БПФ с точным усилением только к направления и полосы интереса, определенные двухмерным пространством-временем низкой сложности на основе ADFT. блок внимания. Мелкозернистое БПФ обеспечивает спектральную сигнатуру, которая может найти будущие варианты использования в механизмы нейронных сетей для распознавания модуляции, идентификации устройств Интернета вещей и RFI идентификация. Алгоритмы формирования луча и спектрального распределения каналов, архитектура цифрового компьютера, и ранние прототипы, использующие 32-элементный полностью цифровой многоканальный приемник и программируемый пользователем Представлена высокоскоростная программно-определяемая радиосвязь (SDR) на базе вентильной матрицы (FPGA).

Ключевые слова: приближенное дискретное преобразование Фурье; широкополосное восприятие спектра; широкополосный многолучевое цифровое формирование диаграммы направленности; алгоритмы низкой сложности

check for updates

Цитирование: Маданаяке, А.; Лоуренс, К.; Кумарасири, Будапешт; Сивасанкар, С.; Гунаратне, Т.; Эдуссурия, США; Синтра, Р.Дж. Проектирование многоканальных каналов Использование систем Spectrum Intelligence Приблизительный дискретный Фурье Алгоритм преобразования для антенны Восприятие спектра на основе массивов Приложения. Алгоритмы 2024, 17, 338. https://doi.org/10.3390/a17080338

Академический редактор: Фрэнк Вернер

Поступила: 3 июля 2024 г.
Пересмотрено: 24 июля 2024 г.
Принято: 26 июля 2024 г.
Опубликовано: 1 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://Creativecommons.org/licenses/by/4.0/).

1. Введение

В классическом произведении Толкина «Властелин колец» вездесущее «Всевидящее око» чувствует и до сих пор вдохновляет жителей Средиземья действовать исходя из их близости к объектам, находящимся под влиянием Лорда Саурона [1]. В нашем мире мы обеспечим постоянную и

Machine Translated by Google

вездесущее знание электромагнитного спектра посредством восприятия в пространстве, времени и форме волны посредством восприятия радиочастот (РЧ) и, таким образом, обеспечивает интеллектуальные действия для DSA на основе искусственного интеллекта. Нам нужно радиочастотное всевидящее око для спектра. Сочетание обнаружения радиочастотного спектра с искусственным интеллектом и машинным обучением (ML) обеспечивает полное восприятие спектра. Вездесущность, подразумевающее повсюду, и постоянство, подразумевающее постоянное получение информации о том, что может происходить в электромагнитной среде, приведут к невообразимо мощным возможностям интеллектуальных беспроводных систем завтрашнего дня. Спектр — чрезвычайно ценный и дефицитный природный ресурс; однако экспоненциальный рост беспроводной связи неизбежно приводит к исчерпанию спектра. Проблему нехватки спектра теоретически можно смягчить за счет перевода сетей связи на миллиметровый диапазон волн в диапазоне от 30 до 300 ГГц . Тем не менее, промышленность недавно усвоила горький урок, связанный с миллиметровыми волнами. Действительно, технология мм-волн работает в лабораторных условиях; тем не менее, существует ряд проблем: (i) трудно заставить его работать в реальных условиях, поскольку острые радиочастотные лучи, необходимые для поддержания связи, склонны к блокировкам; (ii) поиск луча представляет собой трудную задачу, особенно при рассмотрении аспектов мобильности; (iii) усилители мощности (УМ) неэффективны и тратят много энергии; и (iv) время когерентности канала очень мало, что требует более быстрого обновления информации о состоянии канала (CSI), что приводит к высокой вычислительной нагрузке.

1.1. Группа FR3: Спектр Златовласки

С одной стороны, диапазоны частот FR1/менее 6 ГГц (устаревший спектр) просто недоступны для дальнейшего роста. Действительно, поддиапазон 100 МГц на частоте 3,5 ГГц был продан за ошеломляющую цену в 29 миллиардов долларов США, что ясно показывает спрос со стороны беспроводной индустрии. Однако все деньги мира не могут обеспечить вас ресурсами, которых нет. Учитывая, что миллиметровые волны вряд ли приведут к созданию практических систем в ближайшем будущем, отрасль обращается к золотой середине. Эта спектральная золотая середина находится в диапазоне FR3 (от 6 ГГц до 24 ГГц), который игроки отрасли ласково называют верхним средним диапазоном . В нашей недавней работе мы показали, что верхний-средний диапазон не слишком низок, чтобы быть таким же переполненным, как FR1 (устаревший диапазон), и не слишком высок, чтобы создание практически масштабируемой инфраструктуры в настоящее время нежизнеспособно. Полоса FR3 находится «в самый раз», поскольку находится в диапазоне, благоприятном для масштабируемых систем. Кроме того, в FR3 доступна полоса пропускания для беспроводных технологий следующего поколения при условии, что можно ориентироваться в запутанной паутине политики использования спектра [2].

1.2. Экономика общего спектра

Существует острая проблема с использованием верхнего среднего диапазона для беспроводной связи. Диапазон от 6 ГГц до 24 ГГц уже выделен для критически важных приложений, а его лицензии принадлежат Министерству обороны (DoD) и другим государственным организациям. Большинство критически важных датчиков Министерства обороны США, включая радары X-диапазона, сети связи, научно-экспериментальные диапазоны (например, радиоастрономия) и датчики метеорологических радаров, также существуют в диапазоне FR3. Первичные пользователи (PU), скорее всего, никогда не откажутся от своего «драгоценного» спектра навсегда; однако в будущем, скорее всего, потребуется DSA, где вторичные пользователи (SU), такие как коммерческие беспроводные сети, смогут использовать диапазоны FR3 таким образом, чтобы практически не создавать угрозы для критически важных приложений, реализуемых PU. Хотя DSA может принести обществу огромные выгоды и экономический рост, такая великая сила сопряжена с огромной ответственностью, поскольку соблюдение соглашений о разделе должно быть обеспечено при условии неотказуемости и нулевого доверия.

1.3. Потребность в спектральном восприятии ИИ в режиме

реального времени «Министерство обороны определило, что совместное использование спектра возможно, если будут внедрены определенные передовые функции подавления помех и структура координации для облегчения совместного использования спектра», — провозглашает Национальная стратегия по использованию спектра на 2023 год, принятая Белым домом [3]. Раннее обнаружение военных действий занимает более важное место в списке приоритетов, чем качество обслуживания коммерческих беспроводных сетей. Хотя вероятность катастрофического события невелика, она не равна нулю. Единственный способ гарантировать, что SU не посягают на распределение спектра PU в пространстве, времени

Machine Translated by Google

и домены формы сигналов осуществляются посредством независимых измерений, предполагая нулевое доверие, чтобы полностью гарантировать соответствие соглашениям о совместном использовании спектра. Помимо реальных ошибок и непреднамеренных вредных радиочастотных помех, существуют надвигающиеся киберугрозы со стороны неизвестных до сих пор злоумышленников. К возможности создания помех критически важной беспроводной и оборонной инфраструктуре не следует относиться легкомысленно. Надежные, высокочувствительные измерения с постоянным обнаружением радиочастотных сигналов повсюду и в любое время позволяют повысить устойчивость к таким будущим угрозам с точки зрения ведения войны за спектр.

1.4. Ожидается , что спектральные датчики с вездесущим

восприятием РЧ автономия благодаря системам искусственного интеллекта, созданным с использованием глубокого обучения (DL) и обучения с подкреплением (RL), будут генерировать глубокие знания — то есть «восприятие» — о воспринимаемой электромагнитной среде [4]. Национальная стратегия в области спектра (Белый дом, 11 ноября 2023 г.) призывает к начальным исследованиям в диапазонах от 3,1 до 3,45 ГГц, от 5,030 до 5,091 ГГц, от 7,125 до 8,400 ГГц и от 18,1 до 18,6 ГГц в целях обеспечения DSA между коммерческими предприятиями и Министерством обороны. сущности. Восприятие с поддержкой искусственного интеллекта требует распознавания радиочастотного спектра и обнаружения параметров формы сигнала (например, параметров модуляции и, возможно, даже кодирования [5]), частей агрегации несущих и полосы пропускания или аналогичной информации, направлений распространения, уровней мощности и поведения канала, такого как многолучевое распространение. метрики , доплеровский сдвиг и межсимвольная интерференция (ISI) [6]. Типичные датчики спектра, вероятно, не предоставляют действенной информации о спектре, за исключением статистики использования [7–10], где обнаружение «дыр в спектре» определяет поддиапазон(ы), возможно, доступные для SU [11–17]. Мошенничество со спектром является серьезной проблемой [18]. Наличие вредных радиочастотных помех (RFI) можно доказать только путем независимого измерения и восприятия третьей стороной. Чтобы усвоить модель спектрального ландшафта, нам потребуется хорошее понимание измеряемой радиочастотной среды; простое объединение энергии с использованием БПФ слишком ограничивает возможности будущего использования. Постоянное подавление событий RFI повсюду является сложной задачей и требует обнаружения помех, импульсного оружия и мошеннич
Нам необходимо восприятие в реальном времени с использованием машинного интеллекта, поскольку события происходят повсюду и постоянно со скоростью тысячи событий в секунду, что исключает любые подходы с участием человека . Восприятие радиочастотной среды с полной поддержкой искусственного интеллекта имеет решающее значение и служит интеллектуальным источником информации для управления автоматизированным DSA [19].

1,5. Вклад статьи

В этой статье мы предлагаем вычислительно эффективную интеллектуальную систему измерения спектра . Мы используем многолучевой формирователь луча на основе ДПФ для направленного обнаружения радиочастотных сигналов. Чтобы уменьшить вычислительную сложность многолучевого формирователя луча, мы используем приближенное дискретное преобразование Фурье (ADFT) вместо DFT. Цифровая обработка предлагаемого многолучевого формирователя луча реализована на программируемой пользователем вентильной матрице (FPGA). Серверная часть цифровой обработки, подключенная к антенной решетке, состоящей из 32 патч-антенн, работающих на частоте 5,7 ГГц, подтверждает измерение спектра в реальном времени . Кроме того, экспериментальная установка демонстрирует применение многолучевого формирования луча на основе ADFT для измерения спектра, предлагая многообещающий подход для несложного и эффективного направленного восприятия спектра в реальном времени.

1.6. Организация статьи

Оставшаяся часть статьи организована следующим образом. В разделе 2 представлен всесторонний обзор, который начинается с изучения динамического доступа к спектру с упором как на методы обработки сигналов, так и на подходы машинного обучения/глубокого обучения, за которым следует обсуждение радиоастрономии и радиочастотных помех. Раздел 3 углубляется в приближенное дискретное преобразование Фурье, предлагая понимание теоретической основы и конкретной реализации ADFT. В разделе 4 мы представляем подробный обзор системы, охватывающий предлагаемую архитектуру, аналоговый интерфейс, ядро ADFT и цифровую спектральную оценку ADFT, цифровой спектрометр БПФ и цифровую высокоскор
В разделе 5 представлены экспериментальные результаты, включая калибровку и измерения луча.

Machine Translated by Google

демонстрация практической эффективности предлагаемого метода. Наконец, в разделе 6 обсуждаются последствия наших выводов и излагаются потенциальные направления будущих исследований.

2. Обзор

В этом разделе мы рассматриваем соответствующие работы по доступу к динамическому спектру. Сначала мы рассмотрим методы обработки сигналов, используемые для измерения спектра. Далее в первом подразделе мы рассмотрим машинное обучение и недавно предложенные методы глубокого обучения. Далее мы рассмотрим радиоастрономию с использованием радиотелескопов нового поколения, в частности, спектральное распознавание радиочастотных помех.

2.1. Динамический доступ к спектру: обработка сигналов и подходы ML/DL DSA впервые

был рассмотрен два десятилетия назад на основе измерений фактического использования спектра, выполненных Целевой группой по политике в области спектра Федеральной комиссии по связи (FCC) [20]. Вопреки мнению большинства о насыщенности спектра , фактические измерения показали, что многие части спектра простаивают в данное время и в данном месте, что приводит к появлению спектральных дыр. Это является результатом политики доступа к фиксированному спектру. В соответствии с DSA вторичным пользователям (т. е. нелицензированным пользователям) разрешено использовать спектральные дыры первичных пользователей (т. е. лицензированных пользователей), не мешая работе первичных пользователей посредством когнитивного радио [21–24], которое представляет собой интеллектуальную систему, созданную на программно определяемом радио. Однако существует верхний предел пропускной способности канала, которого можно достичь с помощью DSA [25]. Здесь мы кратко рассматриваем методы измерения спектра, связанные с DSA, а по другим аспектам DSA, таким как управление использованием спектра и политика совместного использования спектра, читатель отсылается к [26–28] .

Было предложено несколько методов как для узкополосного, так и для широкополосного зондирования спектра . Для узкополосного зондирования основными методами, не связанными с ML, являются обнаружение энергии [29–32], обнаружение на основе согласованного фильтра [33–35], обнаружение циклостационарных признаков [36,37] и обнаружение на основе ковариации [38–40]. Подходы обнаружения энергии и обнаружения на основе ковариации не требуют предварительного знания сигнала основного пользователя; однако первый вариант ненадежен при низких значениях отношения сигнал/шум (SNR). Обнаружение на основе согласованного фильтра обеспечивает оптимальное распознавание и лучшее обнаружение при низких значениях отношения сигнал/шум, но этот метод требует предварительного знания сигнала основного пользователя, которое не всегда может быт Обнаружение циклостационарных особенностей, как правило, устойчиво к неопределенности шума и хорошо работает при низких значениях SNR; однако он страдает от большого времени обнаружения. При широкополосном зондировании в качестве простого подхода широкополосный сигнал можно разделить на набор узкополосных сигналов посредством временного ДПФ или банка фильтров [41–44] и считывать последовательно или параллельно с использованием методов узкополосного зондирования. Однако последовательное зондирование требует больше времени, тогда как параллельное зондирование требует больше ресурсов. Поскольку широкополосный сигнал со спектром можно рассматривать как разреженный сигнал в частотной области, для широкополосного зондирования было предложено несколько методов, основанных на компрессионном зондировании [45–48]. Эти подходы требуют более низкой частоты дискретизации и более низкого энергопотребления в дополнение к быстрому измерению . Таким образом, подходы, основанные на компрессионном зондировании, устраняют большинство недостатков последовательного или параллельного узкополосного зондирования широкополосного сигнала.

За последнее десятилетие для измерения спектра было предложено значительное количество методов машинного и глубокого обучения. Алгоритмы ML в основном использовались для определения шаблонов занятости канала используемого первичного канала и для классификации канала как свободного или занятого. В [49–53] модели кластеризации K-средних и модели гауссовой смеси впервые использовались для определения присутствия основных пользователей, а затем использовался такой классификатор, как машина опорных векторов и K-ближайший сосед, чтобы определить, является ли канал свободен или занят. Методы DL, такие как сверточные нейронные сети и длинная кратковременная память, использовались для восприятия спектра в [54–57], которые превзошли традиционные методы. Недавно в [58] была предложена архитектура DL на основе трансформатора . Этот подход DL превосходит ранее предложенные подходы, основанные на сверточных нейронных сетях. Другой подход DL, используемый для совместного восприятия спектра, — это глубокое обучение с подкреплением [59–62]. Это глубокое подкрепление

Machine Translated by Google

Подходы к обучению повышают надежность системы измерения спектра и позволяют ей принимать более точные решения в динамичных средах. Отсылаем читателя к [63–65] за всесторонним обзором методов измерения спектра.

Пространственно-временное моделирование систем когнитивного радио с использованием концепций многомерной обработки сигналов было представлено в [66]. Здесь в качестве формирователей луча использовалось направленное зондирование с использованием трехмерных фильтров с бесконечной импульсной характеристикой (БИХ). Подобные подходы к направленному зондированию с использованием многомерных фильтров были представлены в [67–69] Кроме того, в [70–72] для направленного зондирования использовались системы антенных решеток .
В [73] для измерения направления в реальном времени использовался мультилучевой формирователь на основе DFT. По сравнению с мультилучевым формирователем, использованным с ADFT в этом исследовании, в [73] применялось точное DFT с 16 одновременными лучами на частоте 2,4 ГГц. Отметим, что верхний предел пропускной способности канала, которого можно достичь с помощью DSA [25], не изменяется при использовании ADFT вместо DFT.

2.2. Радиоастрономия и радиочастотные помехи

Радиоастрономия для наблюдения за Вселенной полагается на минимально загрязненный радиоспектр [74]. Обнаружение и оценка слабых небесных сигналов, приходящих из далеких галактик, особенно необходимы для достижения ключевых научных целей радиотелескопов следующего поколения, а именно решетки квадратного километра (SKA) (https://www.skao.int/ доступ осуществлен 24 июля 2024 г.) [75], Большая миллиметровая/субмиллиметровая решётка Атакамы — модернизация широкополосной чувствительности (ALMA-WSU) (https://science.nrao.edu/facilities/alma/ science_sustainability/широкополосная чувствительность-обновление доступ осуществлен 24 июля 2024 г.) [76] и ngVLA Very Large Array следующего поколения) (https://ngvla.nrao.edu/ доступ: 24 июля 2024 г.) [77]. Несмотря на то, что они расположены в отдаленных малонаселенных регионах мира, растущая интенсивность, частота, полоса пропускания и возникновение радиочастотных помех ставят под угрозу полезность радиотелескопов следующего поколения [78]. Радиочастотные помехи от глобальной навигационной спутниковой системы (GNSS) [79], наземных и бортовых систем беспроводной связи, спутниковых мегагруппировок [80] и других технологий вторгаются в спектры, которые используются исключительно радиоастрономами, что затрудняет различение слабые небесные сигналы от RFI.

Международный союз электросвязи (МСЭ) (https://www.itu.int/en/Pages/default.aspx доступ осуществлен 24 июля 2024 г.) и FCC (https://www.fcc.gov/ доступ осуществлен 24 июля 2024 г.) играют важную роль в защите спектра для радиоастрономов и других пассивных пользователей. МСЭ через свой сектор радиосвязи (ITU-R) отвечает за разработку международных правил и рекомендаций по испол
Он признает важность защиты спектра для пассивных служб, таких как радиоастрономия, спутниковые службы исследования Земли и метеорологические спутниковые службы.
МСЭ-R определяет конкретные полосы частот для этих пассивных служб и устанавливает нормативные положения, обеспечивающие их защиту от вредных помех. Эти положения включают процедуры координации, ограничения мощности и требования к разносу частот для защиты чувствительных наблюдений и измерений, проводимых радиоастрономами и другими пассивными пользователями [81,82]. Аналогичным образом, Федеральная комиссия связи США признает важность защиты спектра для радиоастрономических и других пассивных служб.

Управление инженерии и технологий (OET) FCC разрабатывает политику и правила для предотвращения вредного вмешательства в эти услуги. Он устанавливает технические стандарты и условия лицензирования, учитывающие потребности пассивных пользователей, включая радиоастрономов . Кроме того, FCC тесно сотрудничает с Национальным научным фондом (NSF) и другими соответствующими агентствами для координации использования спектра и защиты целостности научных наблюдений. Однако обе организации находятся под постоянным давлением со стороны отрасли и правительств с требованием более коммерческого использования спектра, который ранее был выделен некоммерческим пользователям.

Поэтому в идеале знание ключевых характеристик радиочастотных помех (т.е. частотного диапазона, схемы модуляции, местоположения источника, интенсивности и т. д.) могло бы помочь операторам радиотелескопов при планировании и составлении графика наблюдений. Тем не менее, это не практично для всех.

Machine Translated by Google

случаи RFI; следовательно, записи мощности радиоспектра с точки зрения мощности, направления, продолжительности и времени происшествия также помогут радиоастрономам пометить и исключить загрязненные наблюдения [83]. В этом случае вездесущее восприятие ЭМ-среды было бы огромным благом на этапе постобработки радиоастрономических наблюдений [84].

3. Приближенное ДПФ.

В этом разделе мы рассмотрим основы аппроксимации ДПФ, описывая его техническое обоснование и математический формализм с точки зрения задачи оптимизации. Далее мы сосредоточимся на современном 32-точечном приближенном ДПФ, которое описывается численно вместе с соответствующим быстрым алгоритмом. Также представлен анализ арифметической сложности обсуждаемого 32-точечного ADFT со сравнением с эталонными методами.

3.1. Теоретическая основа

Дискретное преобразование Фурье представляет собой линейный оператор, который связывает входной сигнал N-точки с выходным сигналом x = x[0] x[1] . . . x[N 1] согласно X = X[0] X[1] . . . X[N - 1] следующему выражению:

$$X[\kappa] = \sum_{n=0}^{N-1} x[n] \cdot \omega_N^{-nk} \qquad \kappa = 0, 1, \ldots, N \quad 1, \tag{1}$$

где $\omega_N = \exp(2\pi j/N)$ — корень N-й степени из единицы, а j 1. Вычисленное по определению ДПФ представляет собой вычислительную сложность в O(N2), которая непомерно высока для приложений реального времени.

На практике ДПФ вычисляется с помощью одной из многих эффективных численных процедур — быстрых алгоритмов, которые вместе называются «БПФ». Особенно популярным выбором БПФ являются алгоритмы на основе системы счисления 2, такие как БПФ Кули-Тьюки, которые могут рекурсивно разлагать блок ДПФ с N точками на два подблока ДПФ с N/2 точками [85].

В общем, БПФ способны снизить вычислительную сложность вычисления ДПФ с O(N2) до O(N log N) [86]. Во многих случаях [87–92] такое снижение вычислительных затрат позволяет достичь теоретической минимальной мультипликативной сложности ДПФ [93].
Как следствие, дальнейшее снижение мультипликативной сложности невозможно. Поскольку это достаточно зрелая область исследований, предложение новых БПФ, способных обеспечить значительное снижение затрат на вычисления, является маловероятным событием.

Однако заметного сокращения можно добиться в рамках отказоустойчивой парадигмы. Если вычислению ДПФ разрешен заданный уровень ошибок, то вычислительные усилия можно скорректировать для достижения более низкого, но приемлемого уровня точности. Как следствие, для оценки ДПФ требуется меньше арифметических операций [94].

3.2. Проблема оптимизации .

Систематически способ получения таких несложных оценок ДПФ заключается в использовании теории матричной аппроксимации. В матричном формализме ДПФ описывается следующей операцией:

$$X = \Phi_N \cdot x, \tag{2}$$

где N-точечная матрица ДПФ, обозначаемая FN, представляет собой матрицу размера N × N с элементами, заданными формулами $f_{k,n} = \omega_N^{-nk}$, H - 1.

n = 0, 1, . . . , В контексте аппроксимации дискретных преобразований качественно аппроксимированное F^ дискретное преобразование, то матрица преобразования таковы, что точные и приблизительные сигналы области преобразования «близки» в некотором смысле, поддающемся количественному определению. Обычно аппроксимационное преобразование должно удовлетворять следующим условиям: (i) оно может в значительной степени сохранять значимые свойства точного преобразования; (ii) существует математическая близость между точной и приблизительной матрицей преобразования в соответствии с контекстуально значимой метрической функцией — обычно показателем производительности

Machine Translated by Google

заслуга; и (iii) он требует значительно меньших вычислительных затрат по сравнению с точным вычисления с помощью БПФ. В [95] представлен обзор теории.

Обычно, чтобы сохранить физическую интерпретацию приближенного спектра, приближенные матрицы преобразования стремятся быть близкими в евклидовом смысле к связать точную матрицу преобразования. Этого можно добиться за счет минимизации нормы Фробениуса разности между точной и кандидатской матрицами для

аппроксимация с учетом ограничения низкой сложности. Поэтому низкая сложность аппроксимацию можно получить, ограничив ее элементы числовыми наборами тривиальных множимые, например {0, ±1} или {0, ±1, ±2}.

В символах возможная формулировка задачи получения аппроксимации ДПФ.

определяется выражением [95]

$$\min_{x_3,\,n} \quad FN \quad S \cdot \hat{F}\,N \quad F \tag{3}$$

$$\text{при условии} \quad f_{k,n} \quad \{0, \pm 1\} \quad ^{2}, \tag{4}$$

$\hat{}$

1, . . . , N 1, обозначает элементы приближенной матрицы $\hat{F}$, где $f_{k,n}$, где k, n = 0, H,

· F обозначает матричную норму Фробениуса, а S — матрица нормализации, используемая для обеспечения что базисные векторы $S \cdot \hat{F}$ H имеют энергию, равную или близкую к единице.

3.3. 32-точечный ADFT

В этой работе мы приняли 32-точечное ADFT, введенное математически в [96] и дальнейшее развитие в теории и аппаратуре представлено в [97,98]. Явная числовая форма выбранный 32-точечный ADFT приведен в Приложении A. На рисунке 1 показано сравнение амплитудных откликов для 32-точечного точного DFT и 32-точечного ADFT. Фазовые отклики по максимумам, т . е . лучам, остаются неизменными, и любое отклонение фазы лишь происходит в глубоких областях стоп-зоны. Поэтому влияние характера аппроксимации ADFT на фазах диаграммообразных сигналов пренебрежимо мал.

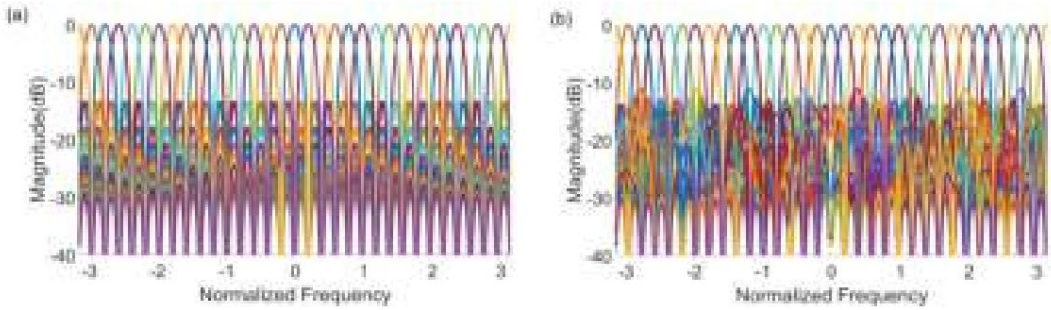


Рисунок 1. Смоделированная амплитудная характеристика (а) идеального 32-точечного ДПФ и (б) 32-точечного ДПФ низкой сложности АДФТ [97,98].

3.4. Быстрый алгоритм и анализ сложности

Представленная матрицей низкой сложности $\hat{F}$ 32, выбранный ADFT допускает следующая факторизация:

$$\hat{F}_{32} = W8 \cdot W7 \cdot W6 \cdot W5 \cdot W4 \cdot W3 \cdot W2 \cdot W1, \tag{5}$$

где Wi , i = 1, 2, . . . , 8 — разреженные матрицы низкой сложности. В Приложении A мы дайте явное определение матриц Wi , i = 1, 2, . . . , 8.

Для сравнения в таблице 1 указано количество арифметических операций для точного 32-точечного ДПФ. вычисляется в соответствии с его определением и с помощью БПФ Кули – Тьюки по системе счисления 2. Для этой операции при подсчете мы предполагаем, что (i) входной сигнал является чисто вещественным; (ii) тривиальные умножения на 0, ±1 или ±j не учитываются; и (iii) умножение действительного числа на комплексное. число выполняется как два действительных умножения, а не комплексное умножение [99]. Из-за (i) и (iii), 704 нетривиальных мультипликатора $\hat{F}$ 32 эквивалентны 1408 действительным умножениям.

Machine Translated by Google

При оценке количества дополнений учитывалась сумма числа реальных и дополнения мнимой части. Также была представлена арифметическая сложность обсуждаемого 32-точечный ADFT по его определению и с помощью быстрого алгоритма (факторизации). Анализ сложности БПФ Кули–Тьюки по основанию 2 подробно описан в [99].

Таблица 1. Производительность ДПФ, БПФ Кули-Тьюки и обсуждаемого 32-точечного ADFT. Тривиальный умножения на ±1 или ±j не учитываются; предполагаются только действительные входные сигналы.

Метод	Реальные умножения	Реальные дополнения
Точный 32-точечный ДПФ (F32)	1408	1666 г.
Radix-2 БПФ Кули – Тьюки (N = 32) [99]	88	408
Приблизительное ДПФ (Fˆ 32) [96]	0	1282
Быстрый алгоритм для Fˆ32 ($\prod_{i=1}^{8}$ Ви) [97]	0	144

В таблице 2 показана арифметическая сложность каждого отдельного матричного члена Fˆ 32 факторизация. Матрица W8 не представляет арифметических затрат, поскольку состоит только из Операции смены знака и обмена данными (замена вещественных и мнимых частей). Обмен данными является следствием умножения на j (поворота на π/2).

Таблица 2. Производительность матричных факторов быстрого алгоритма ADFT. Тривиальные умножения на ±1 или ±j не учитывается; предполагается только действительный входной сигнал.

Операция	П1	П2	W3	W4	W5	W6	W7	W8
Реальные умножения	0	0	0	0	0	0	0	0
Реальные дополнения	30	30	14	14	30	14	12	0

4. Обзор системы

В этом разделе мы описываем конструкцию и реализацию антенн, микроволновых приемников , схем преобразователей смешанных сигналов и цифровых реализаций на основе FPGA, которые составляют предлагаемая система восприятия пространственно-временного спектра. Антенны и микроволновка системы были спроектированы и реализованы собственными силами, а система FPGA была закуплена у Сообщество разработчиков приборов для радиотелескопов с открытым исходным кодом (CASPER), которое находится в заряд основных радиоинструментов, таких как массив квадратных километров (SKA), Мерчисон Широкоугольная решетка, телескопическая решетка Аллена и другие.

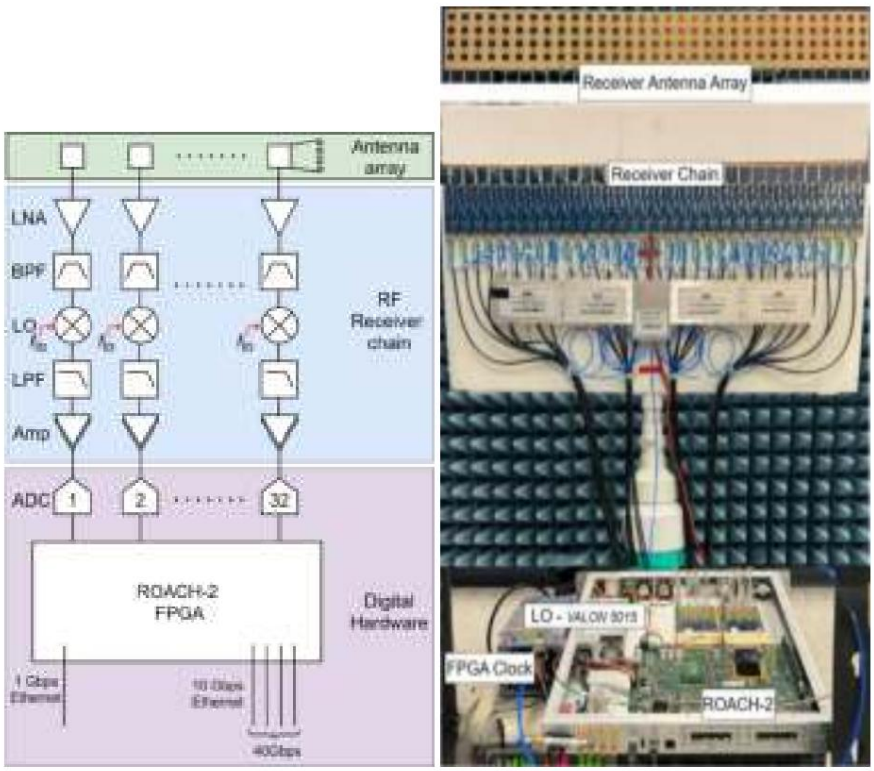
4.1. Предлагаемая архитектура

Экспериментально проверить и измерить лучи, генерируемые предлагаемым ADFT. алгоритмов, радиочастотная система 5,7 ГГц с цифровой обработкой была тщательно спроектировано и реализовано. На рисунке 2a показана общая архитектура системы на основе антенная решетка из 32 элементов, включающая ключевые подсистемы, такие как сама антенная решетка, цепь радиочастотного приемника и цифровой аппаратный блок. На рис . 2б показана экспериментальная установка. предлагаемой архитектуры. Помимо цифровых ядер, отвечающих за выполнение пространственное DFT дискретизированных сигналов, в рамках FPGA была разработана дополнительная схема. Эта дополнительная схема облегчает манипулирование данными в реальном времени и измерения луча. под разными углами, повышая гибкость и возможности системы.

Реконфигурируемое цифровое вычислительное оборудование с открытой архитектурой (ROACH-2). Платформа обработки сигналов (DSP) показана на рисунке 3 и играет решающую роль, поскольку выступая в качестве промежуточного уровня, обеспечивая связь между оборудованием FPGA (точнее его память) и бортовой компьютер. Плата ROACH -2 взаимодействует с главный хост-сервер Linux через соединение Ethernet 1 Гбит/с . Выполняется процедура Python на хост-сервере запрограммирован на взаимодействие с платформой РОАЧ-2 , выполняя функции управления и извлечения данных из памяти ПЛИС при итеративном сканировании под разными углами. Эта уникальная архитектура системы не только обеспечивает точный луч

Machine Translated by Google

измерения, но также повышает общую эффективность и автоматизацию эксперимента, обеспечивая надежную платформу для проверки передовых алгоритмов формирования луча.



(а) (б)

Рис. 2. (a) Архитектура системы; (b) экспериментальная установка 32-лучевого приемника с решеткой работает в диапазоне частот от 5,700 до 5,800 ГГц , за которой следует цифровая серверная часть системы ROACH-2 FPGA (Xilinx Virtex-6 sx475T FPGA).



Рис. 3. Платформа DSP на базе ROACH-2 на базе ПЛИС Xilinx Virtex-6 Sx475 и 32-канальной карты АЦП. Мы благодарим доктора Дэна Вертимера из Калифорнийского университета в Беркли и сообщество CASPER за их вклад в инструменты ROACH-2 и CASPER.

Machine Translated by Google

4.2. Аналоговые входные

каскады РЧ-интерфейс формирователя луча в режиме приема был сконструирован с использованием 32-элементной однородной линейной матрицы (ULA), работающей на частоте 5,7 ГГц, в сочетании с 32 цепями радиочастотных приемников прямого преобразования на печатных платах (PCB). Элементы массива расположены на расстоянии 0,6λ, что соответствует примерно 32 мм на частоте 5,7 ГГц. Каждый антенный элемент выполнен в виде вертикальной подрешетки, состоящей из патч-антенн 4 × 1, что обеспечивает пассивное формирование диаграммы направленности в вертикальной плоскости. Это аналоговое формирование луча работает независимо от алгоритмов формирования луча на основе пространственного БПФ, обсуждаемых в этой работе.

Выходы антенных элементов напрямую подключены к 32 гетеродинным приемникам . В этих приемниках используется система централизованного гетеродина (LO), которая включает в себя генератор с низким фазовым шумом, подключенный к сети делителя мощности с 32 выходами. Начальный каскад каждого приемника оснащен малошумящим усилителем (МШУ), который обеспечивает усиление 16 дБ на частоте 5,8 ГГц при коэффициенте шума 2,4 дБ. После усиления сигналы фильтруются через полосовой фильтр в диапазоне от 4,7 до 6 ГГц для устранения внеполосного шума и помех. Отфильтрованные сигналы затем микшируются и пропускаются через фильтры нижних частот для получения входных сигналов низкой промежуточной частоты (ПЧ) с понижением частоты.

Впоследствии эти 32 сигнала с низкой ПЧ усиливаются примерно на 30 дБ и оцифровываются с помощью двух плат аналого-цифрового преобразователя (АЦП) ADC16x250-8 . Каждая плата АЦП поддерживает 16 несимметричных входных каналов, обеспечивает 8-битное разрешение и работает с частотой дискретизации до 250 млн отсчетов/с на канал. Такая установка обеспечивает высококачественную оцифровку полученных сигналов для дальнейшей обработки [97].

4.3. Ядра ADFT и цифровая спектральная оценка ADFT

Обсуждаемый 32-точечный ADFT был реализован с использованием инструментов Xilinx. Разреженные матрицы из факторизации, показанной в (5), были структурированы как параллельные блоки ввода-вывода для оптимизации производительности [97,98]. Проверка этих блоков проводилась с помощью MATLAB Simulink 8.5, обеспечивая надежность проекта перед развертыванием оборудования. Впоследствии проверенная конструкция ADFT была преобразована в FPGA Xilinx Virtex-6 sx475T.

Для интеграции ядра ADFT в платформу ROACH-2 FPGA мы использовали две библиотеки сообщества Collaboration for Astronomy Signal Processing and Electronics Research (CASPER) [100,101]. Такая интеграция позволила системе параллельно генерировать 32 одновременных радиочастотных луча. Каждый луч соответствует дискретному направлению, рассчитанному с помощью функции θk = arcsin(k/32), где k находится в диапазоне от 16 до 15. Успешная реализация ADFT на ПЛИС ROACH-2 не только подтвердила эффективность алгоритма , но и продемонстрировал практическое применение передовых методов проектирования цифрового оборудования. Эта работа подчеркивает потенциал ADFT в различных приложениях, требующих эффективного формирования радиочастотного луча.

Мы реализовали метод пространственного окна для решения проблемы высоких боковых лепестков в ADFT, где боковые лепестки в худшем случае были на 1,5 дБ выше, чем у (точного) DFT. Были протестированы четыре конфигурации: три оконные функции (Баттерворта, Хэмминга и Хэннинга) и базовая версия без оконной функции [85]. Выбор между этими оконными функциями осуществлялся с помощью мультиплексора, оснащенного штырем селектора управления.

На рисунке 4 показан блок пространственной обработки окон, связанный с разреженными матрицами ADFT. С математической точки зрения оконное управление представляет собой диагональную матрицу. Спектральную оценку на основе ADFT можно описать путем объединения оконной диагональной матрицы (Dwindowing), ядра ADFT низкой сложности (Fˆ 32) и диагональной матрицы , нормирующей степень (Dnormalizer). Математическое представление предлагаемой приближенной оценки спектра имеет вид

$$\text{ADFTestimator} = \text{Dnormalizer} \cdot \text{Fˆ } 32 \cdot \text{Уменьшение окон.} \quad (6)$$

Machine Translated by Google

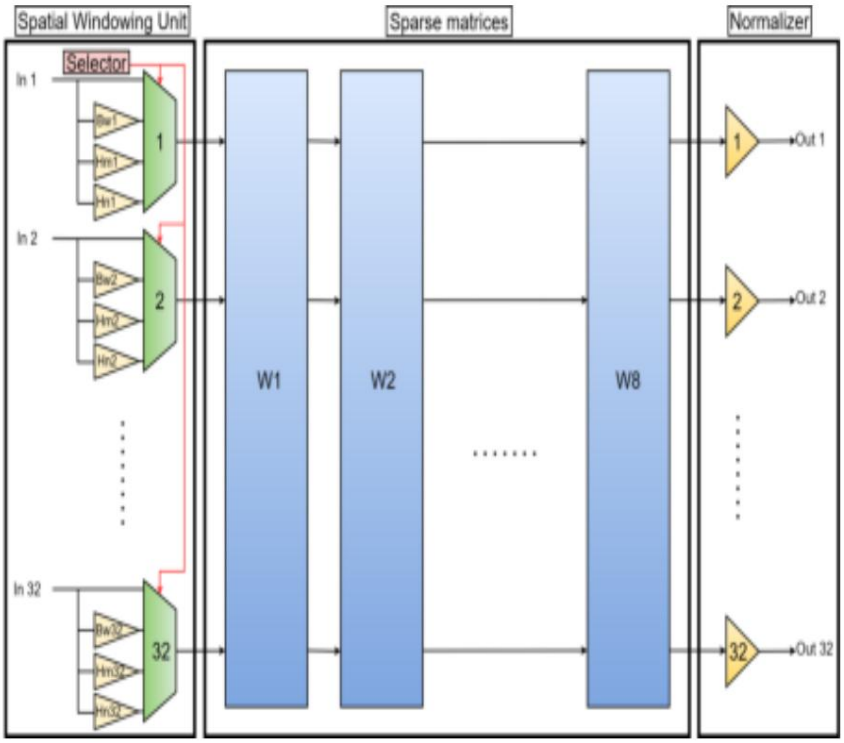


Рисунок 4. Архитектура ADFT с пространственным управлением окнами и блоками нормализации мощности.

Рисунок 5 иллюстрирует ответ ADFT и его модификации при использовании оконных методов. Примечательно, что введение оконных функций привело к расширению основных лепестков, что является компромиссом для подавления боковых лепестков. Этот компромисс между шириной основного лепестка и подавлением боковых лепестков необходимо учитывать при применении методов пространственного окна. Однако реализация пространственной обработки окон увеличивает сложность аппаратного обеспечения из-за мультиплексоров, вводимых блоком обработки окон. Поэтому пользователь должен выбрать подходящую конфигурацию, исходя из своих конкретных требований и ограничений.

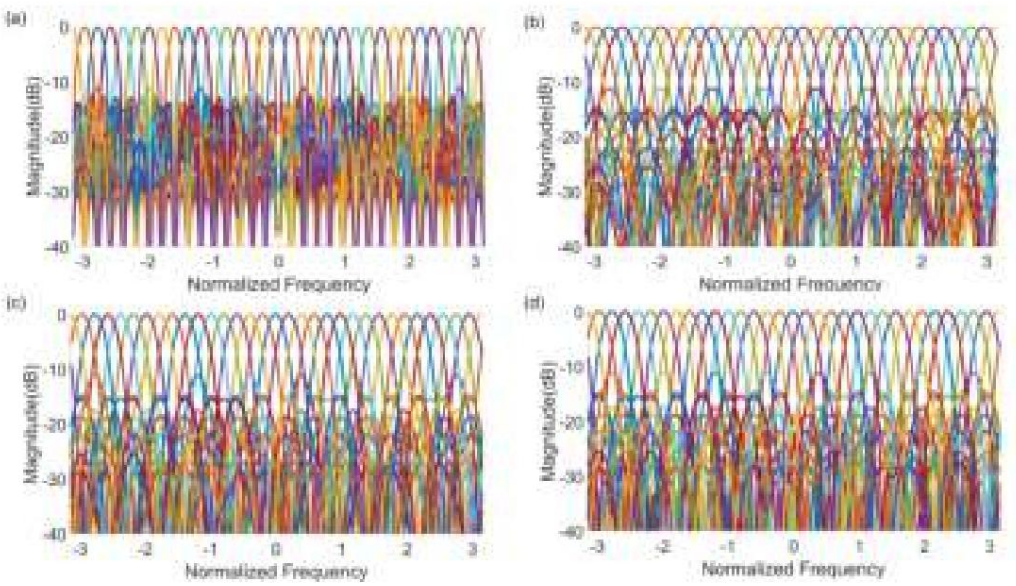


Рисунок 5. Смоделированная амплитудная реакция 32-точечного ADFT и его модификаций при трех различных методах оконного моделирования. Подграфик (a) показывает базовый ответ без какого-либо окна. Подграфики (b – d) демонстрируют ответы ADFT после применения окон Баттерворта, Хумминга и Хэннинга соответственно.

Machine Translated by Google

4.4. Цифровые спектрометры БПФ

Описанная система включает в себя реализацию пространственного БПФ, которое генерирует 32 параллельных выходных сигнала, каждый из которых соответствует отдельному пространственному радиочастотному лучу. После этого пространственного БПФ к каждому из этих 32 пространственных радиочастотных лучей применяется временной спектрометр БПФ, обеспечивая канализацию во временной частотной области. Каждое временное БПФ вычисляется по 1024 точкам, что требует использования 32 параллельных ядер БПФ. Эта параллельная обработка облегчается за счет использования биплексных блоков БПФ с реальной выборкой, доступных в FPGA Xilinx Virtex-6 SX475T.

Биплексные блоки БПФ с реальной выборкой используют метод эрмитова сопряжения, позволяющий вычислить БПФ с реальной выборкой. Этот метод использует сложное ядро БПФ для одновременного преобразования двух потоков вещественных данных. В результате операция БПФ производит все частоты, как положительные, так и отрицательные, причем отрицательные частоты являются зеркальным отражением своих положительных аналогов [100].

Выходные данные временного БПФ для каждого луча дают 1024 комплексных значения, соответствующих 1024 дискретным элементам разрешения по частоте. Спектральная плотность мощности (PSD) оценивается путем возведения в квадрат величины каждого элемента разрешения по частоте. Эта оценка получается путем усреднения 1024 временных оценок БПФ, что приводит к новому измерению PSD каждые 1 миллион тактовых циклов. Блок векторного аккумулятора блока оперативной памяти (BRAM) собирает значения PSD и передает их в программный процесс, поддерживаемый потоком проектир На рисунке 6 показан общий поток ADFT и временного БПФ, включая интеграцию энергии и регистры BRAM. Библиотека Python casperfpga используется для взаимодействия с оборудованием CASPER, извлекая значения из общей памяти BRAM.

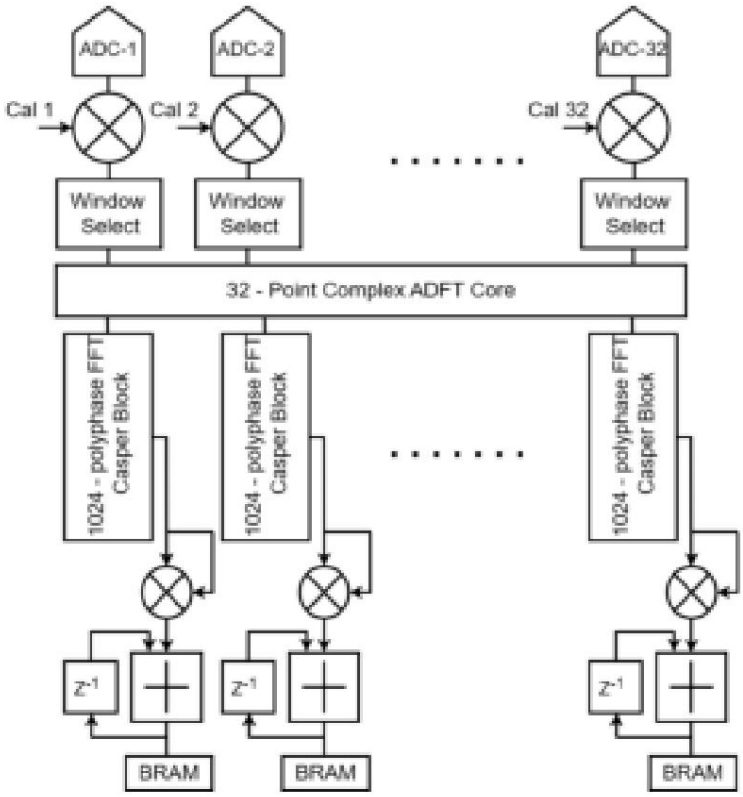


Рисунок 6. Калибровка ADFT и FFT, интеграция энергии и общий системный блок, включая регистры BRAM.

FPGA работает с тактовой частотой, установленной на 200 МГц. Учитывая эту конфигурацию, разрешение по частоте системы для каждого луча определяется как F = 100/1024 МГц, что составляет примерно 100 кГц. Таким образом, полоса пропускания Найквиста системы составляет 100 МГц.

Такая конфигурация обеспечивает точное распределение каналов и частотный анализ с высоким разрешением для каждого из 32 пространственных радиочастотных лучей.

Machine Translated by Google

4.5. Высокоскоростное цифровое

соединение Платы ROACH -2 оснащены четырьмя портами SFP+ на одной мезонинной карте 10 GbE , где интерфейс Ethernet управляется встроенным кварцевым генератором с частотой 156,25 МГц . Эта тактовая частота внутренне умножается на коэффициент 66 внутри FPGA, в результате чего эффективная скорость передачи данных составляет 10,3125 Гбит/с.

Ядро CASPER 10 GbE Simulink облегчает передачу и прием пакетов UDP через IPv4, инкапсулированных в кадры Ethernet. Каждый кадр Ethernet требует 38 байтов заголовка кадра, 20 байтов заголовка IPv4 и 16 байтов заголовка UDP, что в сумме составляет не менее 74 байтов служебных данных на пакет. Чтобы соответствовать 64-битным границам, некоторые заголовки могут также включать заполнение нулями, что еще больше влияет на эффективную пропускную способность. Максимальный размер полезной нагрузки для ядра CASPER 10 GbE составляет 8192 байта, реализованного в BRAM, с дополнительными 512 байтами, доступными в распределенной оперативной памяти для заголовков, специфичных для приложения [100].

5. Результаты экспериментов

5.1. Калибровка

Калибровка системы РЧ-матриц необходима для оптимального формирования луча и проводилась в два отдельных этапа. Первоначально калибровка была сосредоточена на АЦП с использованием процедур с открытым исходным кодом, ранее разработанных членами группы CASPER для этого оборудования. Второй этап был посвящен цифровой коррекции несоответствий в СВЧ-интерфейсе. Относительное усиление и рассогласование фаз выходов ПЧ для каждой цепи определялись по отношению к эталонной цепи с использованием входа эталонной несущей 5,71 ГГц . Относительные различия в коэффициентах усиления компенсировались с помощью цифровых умножителей, а масштабные коэффициенты усиления рассчитывались путем измерения средних значений мощности каждого выхода АЦП. В идеале для компенсации разности фаз требуются фильтры дробной задержки, поскольку радиочастотные цепи могут иметь задержки меньше, чем значение периода выборки. В качестве временного решения были разработаны и реализованы в аппаратном обеспечении целочисленные фильтры задержки для уменьшения разности фаз.

5.2. Измерения луча

Диаграммы мощности были получены путем передачи непрерывного (CW) сигнала на частоте fRF = 5,71 ГГц. Частота сигнала LO fLO определяет ПЧ как fRF fLO. Для измерений в этом исследовании частота fLO была установлена на уровне 5,7 ГГц, в результате чего ПЧ составила 10 МГц, а выходные сигналы, преобразованные с понижением частоты, были оцифрованы на частоте fCLK = 200 МГц.

Цифровые интеграторы на каждом выходе бункера ADFT использовались для расчета полученной энергии в течение фиксированного периода времени. Пространственное формирование диаграммы направленности происходит на частоте 200 МГц, а PSD рассчитывается в реальном времени путем возведения в квадрат величины каждого элемента разрешения по частоте. Эта оценка получена путем усреднения 1024 временных вычислений БПФ, выполненных на частоте 200 МГц. Геометрия измерительной установки существенно влияет на измеряемый коэффициент решетки лучей. В идеале передатчик и приемник должны находиться на достаточном расстоянии друг от друга, чтобы волны, падающие на решетку приемника, можно было аппроксимировать как плоские волны. Согласно стандартным рекомендациям [102], расстояние между передатчиком и приемником должно превышать 20 м на частоте 5,7 ГГц , чтобы апертура приемника находилась в дальней зоне. Лучи измерялись на открытой парковке с расстоянием между передатчиком и приемником около 20 м.

На рисунке 7 показаны измеренные диаграммы коэффициентов массива для лучей от 0 до 16. Благодаря вводу действительных сигналов ПЧ в ADFT результирующие диаграммы направленности демонстрируют симметрию, создавая зеркальные изображения для остальных элементов разрешения. Кроме того, в пределах интервалов от 0 до 16 наблюдается наличие двух зеркальных пиков. Измеренные лучи точно соответствуют ожидаемым диаграммам направленности, сохраняя расстояние более 10 дБ от другого лу

Каждый выходной элемент ADFT соответствует радиочастотному лучу в дальней зоне, направленному под дискретным углом, заданным выражением θk = arcsin(k/N), где k = -16, -15, . . . , 14, 15 и N = 32 (количество элементов массива). Такая конфигурация обеспечивает разделение лучей 180 /32 = 5,625 , обеспечивая угловое разрешение ±2,8125 .

Machine Translated by Google

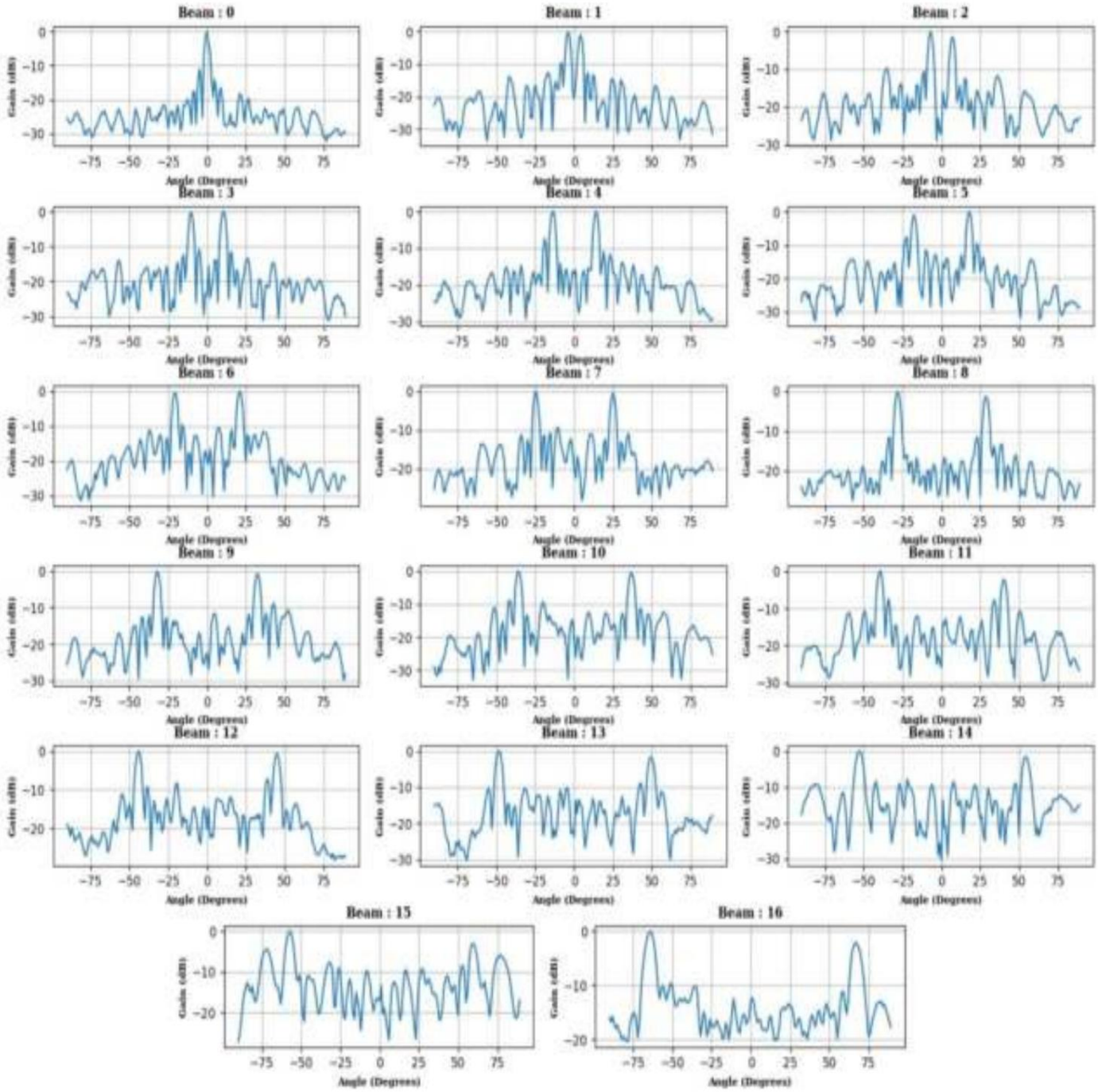


Рисунок 7. Шестнадцать лучей, измеренные от антенной решетки 5,7 ГГц (вертикальная ось — в децибелах, а горизонтальная ось — азимутальный угол [90 , 90]). Каждый подрисунок содержит измеренные диаграммы коэффициентов массива луча с использованием реальных входных данных ядра ADFT. Мнимая составляющая каждого входа ADFT здесь установлена на ноль, что приводит к ожидаемым симметричным основным лепесткам.

На рисунке 8 показано измеренное временное распределение PSD соответствующих лучей под углом несколько частот, когда источник сигнала расположен под углом 15 и 50 . , 35 , Соответствующая экспериментальная установка с относительным расположением источников и радиочастотами, излучаемыми в целях тестирования, показана на рисунке 9.

Machine Translated by Google

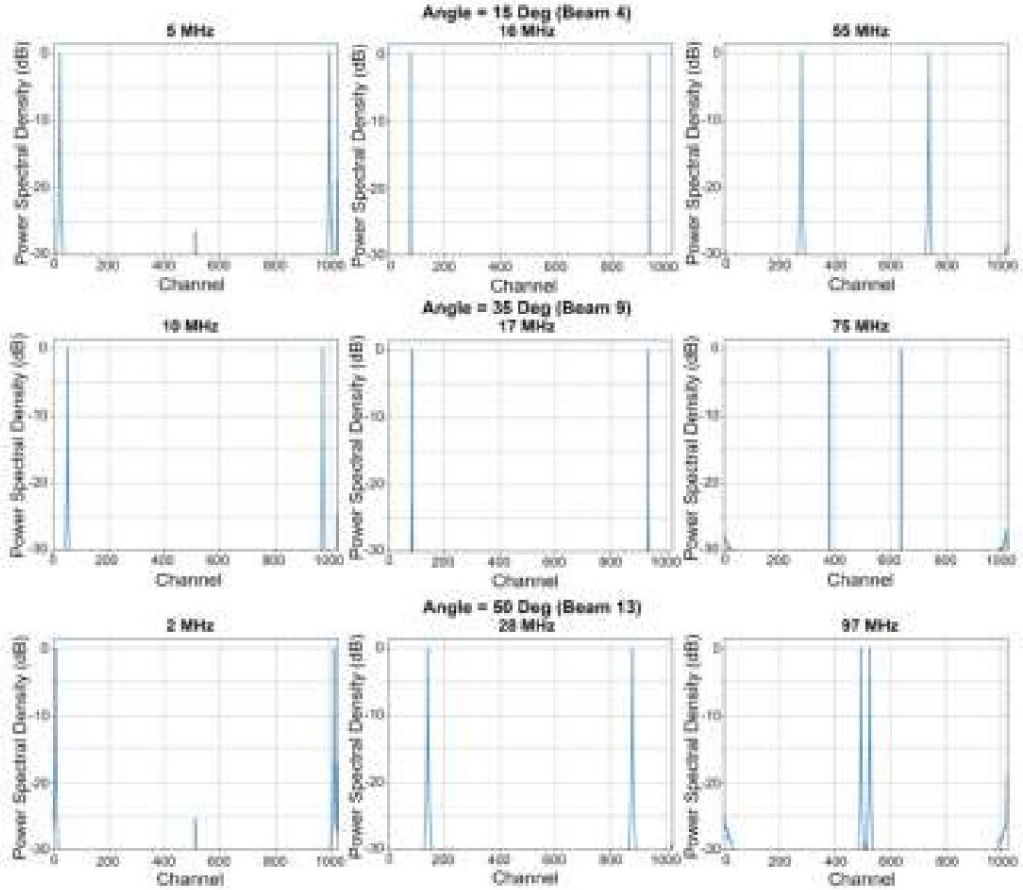


Рисунок 8. Временная PSD по 1024 дискретным элементам разрешения по частоте для конкретного радиочастотного луча при кратных частоты, когда источник расположен под углом 15, 35 , и углы поперечного приёмника 50 .

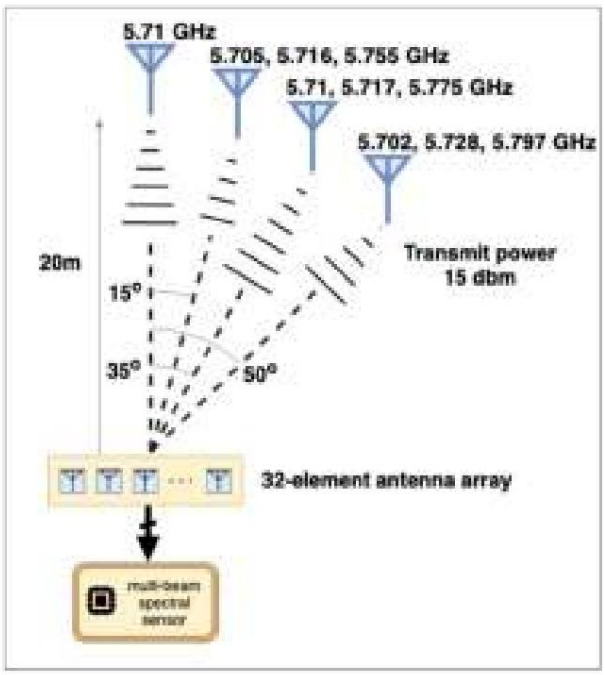


Рисунок 9. Иллюстрация общей установки, показывающая передающие антенны и приемную антенную решетку. и многолучевой спектральный датчик. Приемник антенной решетки из 32 элементов улавливает сигналы от в разных направлениях, при этом передатчики расположены на расстоянии 20 м от приемника.

Machine Translated by Google

6. Обсуждение

В этой статье представлена несложная архитектура цифрового формирования луча, предназначенная для генерировать 32 одновременных радиочастотных луча, используя 32-точечную аппроксимацию ДПФ, эффективно исключая операции умножения. Мы проверили предлагаемый метод на полностью работоспособном 32-элементная цифровая одномерная приемная решетка, работающая на частоте 5,7 ГГц. Одномерный массив использует 32 параллельных АЦП для выборки выходных сигналов антенны и использование ADFT, реализованного на FPGA Xilinx. для расчета 32 радиочастотных лучей в реальном времени. Кроме того, в этой статье обсуждается улучшение нашу существующую 32-канальную систему измерения спектра 5,7 ГГц за счет внедрения 32 параллельных 1024-битных временных БПФ на аппаратной платформе ROACH-2 . Этот экспериментальный проверка с помощью измерений в реальном времени в диапазоне частот и углов падения демонстрирует способность архитектуры воспринимать информацию от нескольких направлениях с частотным разрешением 100 кГц и разрешением по направлению около 6 .

Авторы планируют расширить ADFT до большего количества элементов и создать многомерные формирователи луча ADFT для управления лучом как по углу места, так и по азимуту самолеты. Например, используя массив 32 × 8, мы могли бы расширить набор лучей до прямоугольной формы. сетка размером 32 × 8, в которой как 32-точечные, так и 8-точечные БПФ заменены соответствующими Алгоритмы ADFT, основанные на факторизации разреженной матрицы [103–106].

Вклад автора: концептуализация и проектирование системы AM Theory и проектирование ADFT. и быстрый алгоритм от RJC Обзор многомерной обработки сигналов от CUSE Обзор радио проблемы астрономических помех от TG Digital, реализация ядра ADFT от KL Experimental настройка, цифровое проектирование инструментов CASPER, калибровка и эксперименты в реальном времени со стороны SS и BUK Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Работа частично поддержана ННФ (по грантам FuSE 2329012, SWIFT 2229471, ICORPS 2243346, SpectrumX), NTIA, CNPq (Бразилия) в рамках гранта 315047/23-2 и исследований Сената комитета Университета Мортувы в рамках гранта SRC/LT/2020/08.

Заявление о доступности данных. Все исходные материалы этого исследования документированы в статья. За более подробной информацией обращайтесь к соответствующим авторам.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Сокращения

В данной рукописи использованы следующие сокращения:

АЦП	Аналого-цифровой преобразователь;
ADFT	Приближенное дискретное преобразование Фурье;
ИИ	Искусственный интеллект;
	Большая миллиметровая решетка ALMA Atacama;
BRAM	Блочная оперативная память;
CASPER	Сотрудничество в области обработки астрономических сигналов и исследований в области электроники;
CSI	Информация о состоянии канала;
CW	Непрерывная волна;
ДПФ	Дискретное преобразование Фурье;
ДЛ	Глубокое обучение;
ДПО	Динамический доступ к спектру;
ЦСП	Цифровая обработка сигналов;
ФКС	Федеральная комиссия по связи;
БПФ	Быстрое преобразование Фурье;
FPGA	Программируемая вентильная матрица;
ГНСС	Глобальная навигационная спутниковая система;
ЕСЛИ	Промежуточная частота;

Machine Translated by Google

Интернет вещей	Интернет вещей;
ИСИ	Межсимвольная интерференция;
МСЭ	Международный союз электросвязи;
МШУ	Малошумящий усилитель;
LO	Гетеродин;
МЛ	машинное обучение;
ngVLA	Очень большой массив нового поколения;
	Национальный научный фонд ННФ;
	Управление инженерии и технологий OET;
печатная плата	Печатная плата;
PSD	Спектральная плотность мощности;
ПУ	Основной пользователь;
РФ	Радиочастота;
RFI	Радиочастотные помехи;
РЛ	Обучение с подкреплением;
ROACH	Реконфигурируемое вычислительное оборудование с открытой архитектурой;
SDR	Программно-определяемая радиосвязь;
SFP	Компактный сменный форм-фактор;
СКА	Массив квадратных километров;
ОСШ	Отношение сигнал/шум;
ВС	Вторичный пользователь;
ULA	Единая линейная матрица;
	Обновление чувствительности широкополосного доступа WSU.

Приложение А

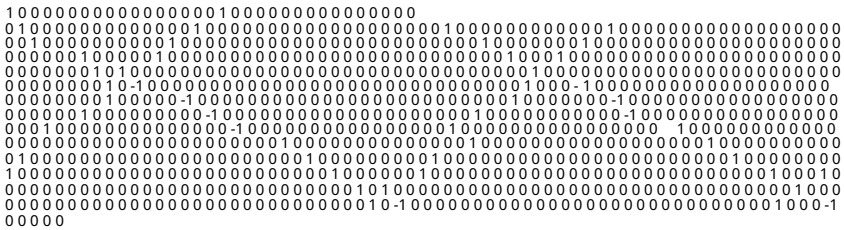
Приложение А.1. 32-точечный быстрый алгоритм ADFT

Быстрый алгоритм для обсуждаемого 32-точечного приближенного ДПФ $F\hat{}_{32}$ состоит из факторизации в разреженных матрицах низкой сложности, как показано в (5). Приближенная матрица и ее условия факторизации явно приведены ниже:

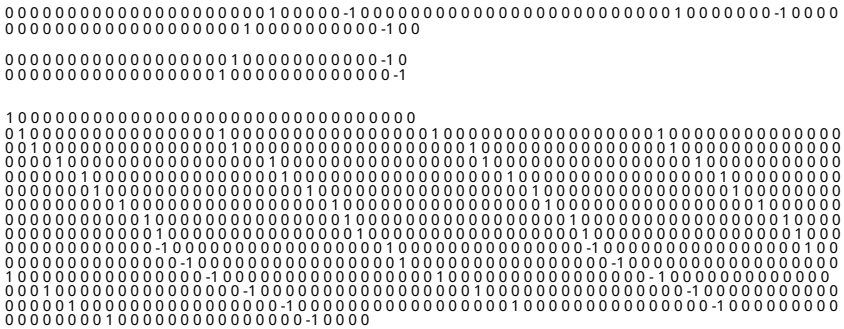
$$F\bar{}_{32} = \text{[sparse matrix]}, \quad \text{(A1)}$$

Machine Translated by Google

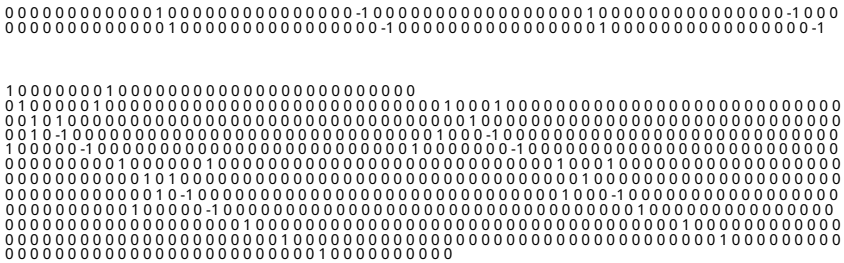
где y = 1 + j, и обозначает комплексное сопряжение:



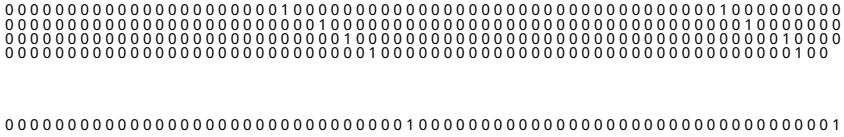
W1 = , (A2)



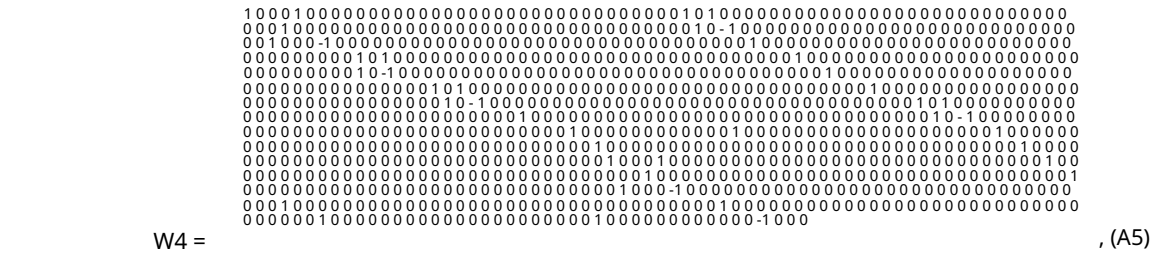
W2 = , (A3)

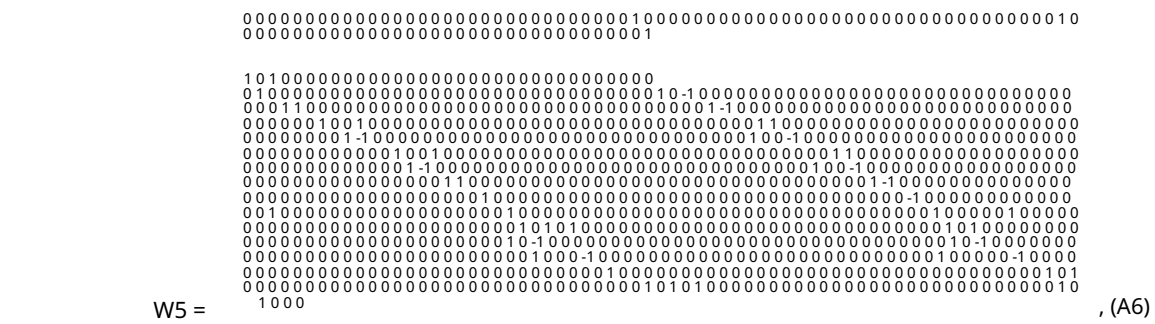


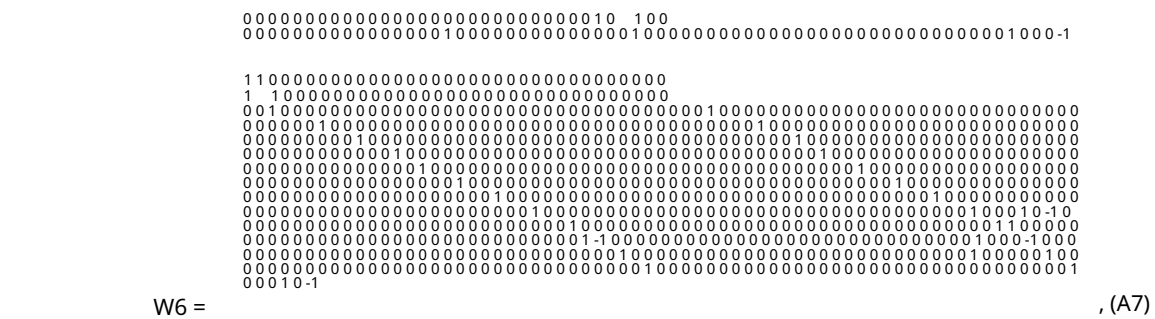
W3 = , (A4)



Machine Translated by Google

W4 =  , (A5)

W5 =  , (A6)

W6 =  , (A7)

Machine Translated by Google

W7 = , (A8)

и

W8 = . (A9)

Ссылки

1. Толкин, Дж. «Хоббит и Властелин колец»; Хоутон Миффлин Харкорт: Бостон, Массачусетс, США, 1954 год.
2. Канг, С.; Меззавилла, М.; Ранган, С.; Маданаяке, А.; Венкатакришнан, С.Б.; Хеллбург, Г.; Гош, М.; Рахмани, Х.; Дхананджай, А. Сотовые беспроводные сети в верхнем среднем диапазоне. IEEE Open J. Commun. Соц. 2024, 5, 2058–2075. [Перекрестная ссылка]
3. Белый дом. Национальная стратегия использования спектра; Белый дом: Вашингтон, округ Колумбия, США, 2023 г.
4. Рестучча, Ф.; Мелодия, Т. Глубокое обучение на физическом уровне: системные проблемы и приложения для 5G и за его пределами. IEEE Коммун. Маг. 2020, 58, 58–64. [Перекрестная ссылка]
5. Триаридис, К.; Думанидис, К.; Хацидиамантис, Северная Дакота; Карагианнидис, ГК MM-Net: мультимодальный подход к автоматической классификации модуляции. IEEE-сообщество. Летт. 2023, 28, 328–331. [Перекрестная ссылка]
6. Балдези, Л.; Рестучча, Ф.; Мелодия, Т. ChARM: Совместное использование спектра NextG посредством динамического управления O-RAN в реальном времени на основе данных . В материалах Международной конференции IEEE по компьютерным коммуникациям (INFOCOM), Лондон, Великобритания, 2–5 мая 2022 г.; IEEE: Пискатауэй, Нью-Джерси, США, 2022 г.; стр. 240–249.
7. Чжан Л.; Сяо, М.; Ву, Г.; Алам, М.; Лян, ЮК; Ли, С. Обзор передовых методов совместного использования спектра в сетях 5G. IEEE Вирел. Коммун. 2017, 24, 44–51. [Перекрестная ссылка]
8. Ху, Ф.; Чен, Б.; Чжу, К. Совместное использование полного спектра в сетях когнитивного радио в направлении 5G: обзор. Доступ IEEE 2018, 6, 15754–15776. [Перекрестная ссылка]
9. Шокри-Гадиколаи, Х.; Боккарди, Ф.; Фишионе, К.; Фодор, Г.; Зорзи, М. Совместное использование спектра в сотовых сетях миллиметрового диапазона посредством ассоциации ячеек, координации и формирования луча. IEEE Дж. Сел. Районы Коммун. 2016, 34, 2902–2917. [Перекрестная ссылка]
10. Льв, Л.; Чен, Дж.; Ни, К.; Дин, З.; Цзян, Х. Когнитивный неортогональный множественный доступ с кооперативной ретрансляцией: новый рубеж беспроводной связи для совместного использования спектра 5G. IEEE-сообщество. Маг. 2018, 56, 188–195. [Перекрестная ссылка]
11. Агарвал, С.; Де, С. eDSA: Энергоэффективные протоколы доступа к динамическому спектру для сетей когнитивной радиосвязи. IEEE Транс. Моб. Вычислить. 2016, 15, 3057–3071. [Перекрестная ссылка]

Machine Translated by Google

12. Чиневе, ТМ; Ханке, Г.П. Кооперативный доступ к динамическому спектру быстрой конвергенции для когнитивных радиосетей. IEEE Пер. Инди. Информ. 2017, 14, 3386–3394. [Перекрестная ссылка]
13. Чанг, Х.Х.; Сонг, Х.; Йи, Ю.; Чжан, Дж.; Он, Х.; Лю, Л. Распределенный динамический доступ к спектру посредством глубокого обучения с подкреплением: подход, основанный на резервуарных вычислениях. IEEE Internet Things J. 2018, 6, 1938–1948. [Перекрестная ссылка]
14. Нгуен, штаб-квартира; Нгуен, Британская Колумбия; Донг, TQ; Нго, Д.Т.; Нгуен, Т.А. Глубокое Q-обучение с многополосным зондированием для динамического доступа к спектру. В материалах Международного симпозиума IEEE по сетям динамического доступа к спектру (DySPAN), Сеул, Республика Корея, 22–25 октября 2018 г.; стр. 1–5. [Перекрестная ссылка]
15. Напарстек О.; Коэн, К. Глубокое многопользовательское обучение с подкреплением для распределенного доступа к динамическому спектру. IEEE Транс. Вирел. Коммун. 2019, 18, 310–323. [Перекрестная ссылка]
16. Ван, С.; Лю, Х.; Гомес, РН; Кришнамачари, Б. Глубокое обучение с подкреплением для динамического многоканального доступа в беспроводной сети Сети. IEEE Транс. Когн. Коммун. Сеть. 2018, 4, 257–265. [Перекрестная ссылка]
17. Ю, Ю.; Ван, Т.; Лью, С.С. Множественный доступ с глубоким обучением для гетерогенных беспроводных сетей. IEEE Дж. Сел. Районы Коммун. 2019, 37, 1277–1290. [Перекрестная ссылка]
18. Рестон, В.А. Ligado Networks подает в суд на правительство США за незаконное и безвозмездное изъятие у Ligado эксклюзивного спектра, лицензированного FCC. 2023. Доступно онлайн: https://ligado.com/press/ligado-networks-sues-us-government-for-unlawful-and-uncompensated-takeing-of-ligados-exclusive-fcc-licensed-spectrum/ . (по состоянию на 28 мая 2024 г.).
19. Барден, С.; Дьюдни, П.; Фризен, Р.; Муровинский Р.; Саванандам, С. ATD Отчет о стратегическом обзоре исследований и разработок - портфель обработки сигналов; Центр исследований астрономии и астрофизики Герцберга - Национальный исследовательский совет Канады: Виктория, Британская Колумбия, Канада, 2022 г.
20. Целевая группа FCC по политике в области спектра. Отчет Рабочей группы по эффективности использования спектра, ноябрь 2002 г.; FCC: Вашингтон, округ Колумбия, США, 2024 г.
21. Митола Дж.; Магуайр, Дж. Когнитивное радио: сделать программные радиоприемники более персонализированными. IEEE Перс. Коммун. 1999, 6, 13–18. [Перекрестная ссылка]
22. Хайкин С. Когнитивное радио: Беспроводная связь, управляемая мозгом. IEEE Дж. Сел. Районы Коммун. 2005, 23, 201–220. [Перекрестная ссылка]
23. Лян, Ю.К.; Чен, КЦ; Ли, Дж.Я.; Махонен, П. Когнитивные радиосети и коммуникации: обзор. IEEE Транс. Вех. Технол. 2011, 60, 3386–3407. [Перекрестная ссылка]
24. Ван, Б.; Лю, К.Р. Достижения в области когнитивных радиосетей: обзор. IEEE Дж. Сел. Вершина. Сигнальный процесс. 2011, 5, 5–23. [Перекрестная ссылка]
25. Талеби С.П. Отказы первичных услуг и производительность вторичных услуг в сетях когнитивного радио. Вирел. Коммун. Моб. Вычислить. 2015, 15, 1982–1990. [Перекрестная ссылка]
26. Чжао, Ц.; Сэдлер, Б.М. Обзор динамического доступа к спектру. Сигнальный процесс IEEE. Маг. 2007, 24, 79–89. [Перекрестная ссылка]
27. Джи, З.; Лю, К.Р. Динамическое совместное использование спектра: теоретический обзор игр. IEEE-сообщество. Маг. 2007, 45, 88–94. [Перекрестная ссылка]
28. Бхаттараи, С.; Пак, JMJ; Гао, Б.; Биан, К.; Лер, В. Обзор динамического совместного использования спектра: текущие инициативы, проблемы, и дорожная карта будущих исследований. IEEE Транс. Когн. Коммун. Сеть. 2016, 2, 110–128. [Перекрестная ссылка]
29. Цзэн Ю.; Лян, ЮК; Чжан Р. Слепое комбинированное обнаружение энергии для измерения спектра в когнитивном радио. IEEE-сигнал Процесс. Летт. 2008, 15, 649–652. [Перекрестная ссылка]
30. Линг, Х.; Ву, Б.; Вэнь, Х.; Хо, РН; Бао, З.; Пан, Л. Адаптивное пороговое управление для измерения спектра на основе обнаружения энергии в Когнитивные радио. IEEE Вирел. Коммун. Летт. 2012, 1, 448–451. [Перекрестная ссылка]
31. Ранджан, А.; Анураг; Сингх, Б. Разработка и анализ спектрального восприятия в когнитивном радио на основе обнаружения энергии. В материалах Международной конференции по обработке сигналов и информации, Нандед, Индия, 6–8 октября 2016 г.; стр. 1–5.
32. Арджун, Ю.; Мрабет, ZE; Гази, НЕ; Тамтауи, А. Зондирование спектра: усовершенствованная методика обнаружения энергии, основанная на измерении шума. В материалах ежегодного семинара и конференции IEEE по вычислительной технике и связи, Лас-Вегас, Невада, США, 8–10 января 2018 г.; стр. 828–834.
33. Цзян, К.; Ли, Ю.; Бай, В.; Ян, Ю.; Ху, Дж. Устойчивое определение спектра на основе статистического согласованного фильтра в условиях неопределенности шума . В материалах Международной конференции по коммуникационным технологиям, Чэнду, Китай, 9–11 ноября 2012 г.; стр. 1209–1213.
34. Чжан Х.; Стул.; Гао, Ф. Определение спектра на основе согласованного фильтра и определение уровня мощности для сети когнитивного радио. В материалах Глобальной конференции IEEE по обработке сигналов и информации, Атланта, Джорджия, США, 3–5 декабря 2014 г.; стр. 1267–1270.
35. Льв, К.; Гао, Ф. Определение спектра на основе согласованного фильтра и распознавание уровня мощности с помощью нескольких антенн. В материалах китайского саммита IEEE и Международной конференции по обработке сигналов и информации, Чэнду, Китай, 12–15 июля 2015 г.; стр. 305–309.
36. Ильяс И.; Пол, С.; Рахман, А.; Кунду, Р.К. Сравнительная оценка восприятия когнитивного спектра на основе циклостационарного обнаружения. В материалах ежегодной конференции IEEE по универсальным вычислениям, электронике и мобильной связи, Нью-Йорк, штат Нью-Йорк, США, 20–22 октября 2016 г.; стр. 1–7.
37. Коэн, Д.; Эльдар, Ю.К. Сжатое циклостационарное обнаружение для когнитивного радио. В материалах Международной конференции IEEE по акустике, речи и обработке сигналов (ICASSP) 2017 г., Новый Орлеан, Луизиана, США, 5–9 марта 2017 г.; стр. 3509–3513.
38. Цзэн Ю.; Лян, Ю.К. Алгоритмы распознавания спектра для когнитивного радио, основанные на статистических ковариациях. IEEE Транс. Вех. Технол. 2009, 58, 1804–1815. [Перекрестная ссылка]
39. Цзэн Ю.; Лян, Ю.К. Алгоритмы распознавания спектра на основе собственных значений для когнитивного радио. IEEE Транс. Коммун. 2009, 57, 1784–1793. [Перекрестная ссылка]
40. Зайен, Б.; Хаяр, А.; Кансанен, К. Слепое зондирование спектра для когнитивного радио на основе оценки размерности пространства сигналов. В материалах Международной конференции IEEE по коммуникациям 2009 г., Дрезден, Германия, 14–18 июня 2009 г.; стр. 1–5.

Machine Translated by Google

41. Фарханг-Бороужени, Б. Банк фильтров для измерения спектра для когнитивных радиостанций. IEEE Транс. Сигнальный процесс. 2008, 56, 1801–1811. [Перекрестная ссылка]

42. Цюань, З.; Кюи, С.; Саид, АХ; Плохое, оптимальное многополосное обнаружение высоковольтного соединения для измерения спектра в сетях когнитивного радио. IEEE Транс. Сигнальный процесс. 2009, 57, 1128–1140. [Перекрестная ссылка]

43. Рагху, И.; Чоудари, СС; Элиас, Э. Эффективное распознавание спектра для когнитивного радио с использованием наборов косинусно-модулированных фильтров. В Материалы конференции IEEE Region 10, Сингапур, 22–25 ноября 2016 г.; стр. 2086–2089.

44. Кумар А.; Саха, С.; Бхаттачарья, Р. Улучшенное обнаружение границ на основе вейвлет-преобразования для распознавания широкополосного спектра в когнитивном радио. В материалах конференции USNC-URSI по радионауке, Фахардо, Пуэрто-Рико, 26 июня – 1 июля 2016 г.; стр. 21–22.

45. Поло, ЮЛ; Ван, Ю.; Пандхарипанде, А.; Леус, Г. Сжатие широкополосного спектра. В материалах Международной конференции IEEE по акустике, речи и обработке сигналов, Тайбэй, Тайвань, 19–24 апреля 2009 г.; стр. 2337–2340.

46. Арджун, Ю.; Каабуш, Н. Широкополосное зондирование спектра: байесовский подход к компрессионному зондированию. Датчики 2018, 18, 1839. [Перекрестная ссылка] [ПабМед]

47. Ли, З.; Чанг, Б.; Ван, С.; Лю, А.; Цзэн, Ф.; Луо, Г. Динамическое компрессионное широкополосное зондирование на основе реконструкции энергии канала в когнитивном Интернете вещей. IEEE Транс. Инди. Информ. 2018, 14, 2598–2607. [Перекрестная ссылка]

48. Ян, Л.; Фанг, Дж.; Дуань, Х.; Ли, Х. Оценка спектра мощности с быстрым сжатием: к практическому решению для широкополосной связи Спектральное зондирование. IEEE Транс. Вирел. Коммун. 2020, 19, 520–532. [Перекрестная ссылка]

49. Дин, Г.; Ву, К.; Яо, Ю.Д.; Ван, Дж.; Чен, Ю. Обучение на основе ядра для статистической обработки сигналов в когнитивных радиосетях : теоретические основы, примеры приложений и будущие направления. Сигнальный процесс IEEE. Маг. 2013, 30, 126–136. [Перекрестная ссылка]

50. Тилина, К.М.; Чой, КВ; Сакиб, Н.; Хоссейн, Э. Методы машинного обучения для совместного восприятия спектра в когнитивной сфере. Радиосети. IEEE Дж. Сел. Районы Коммун. 2013, 31, 2209–2221. [Перекрестная ссылка]

51. Микаил, АМ; Го, Б.; Ван, З. Подход машинного обучения к объединению данных для совместного измерения спектра. В материалах Международной конференции по киберраспределенным вычислениям и открытию знаний, Шанхай, Китай, 13–15 октября 2014 г.; стр. 429–434.

52. Газизаде, Э.; Никпур, Б.; Могадам, Д.А.; Незамабади-пур, Х. Метод взвешивания на основе PSO для улучшения методов машинного обучения для совместного обнаружения спектра в сетях CR. В материалах конференции по роевому интеллекту и эволюционным вычислениям, Бам, Иран, 9–11 марта 2016 г.; стр. 113–118.

53. Халфи, Б.; Заид, А.; Хамдауи, Б. Когда машинное обучение сочетается со сжатием выборки для измерения широкополосного спектра. В материалах Международной конференции по беспроводной связи и мобильным вычислениям, Валенсия, Испания, 26–30 июня 2017 г.; стр. 1120–1125.

54. Ли, В.; Ким, М.; Чо, Д.Х. Глубокое совместное зондирование: совместное зондирование спектра на основе сверточных нейронных сетей. IEEE Транс. Вех. Технол. 2019, 68, 3005–3009. [Перекрестная ссылка]

55. Лю, К.; Ван, Дж.; Лю, Х.; Лян, YC Deep CM-CNN для зондирования спектра в когнитивном радио. IEEE Дж. Сел. Районы Коммун. 2019, 37, 2306–2321. [Перекрестная ссылка]

56. Сони, Б.; Патель, ДК; Лопес-Бенитес, М. Схема восприятия спектра на основе долгосрочной кратковременной памяти для использования когнитивного радио Статистика первичной деятельности. IEEE Access 2020, 8, 97437–97451. [Перекрестная ссылка]

57. Пэн, К.; Гилман, А.; Васконселос, Н.; Косман, ПК; Мильштейн, Л.Б. Надежное глубокое зондирование посредством трансферного обучения в когнитивном радио. IEEE Вирел. Коммун. Летт. 2020, 9, 38–41. [Перекрестная ссылка]

58. Чжан, В.; Ван, Ю.; Чен, Х.; Кай, З.; Тиан, З. Трансформатор спектра: широкополосный детектор спектра, основанный на внимании. IEEE Пер. Вирел. Коммун. 2024, 1–1. [Перекрестная ссылка]

59. Сарихани, Р.; Кейния, Ф. Совместное зондирование спектра и машинное обучение: подход к глубокому обучению с подкреплением. IEEE Коммун. Летт. 2020, 24, 1459–1462. [Перекрестная ссылка]

60. Цай, П.; Чжан, Ю.; Пан, К. Глубокое обучение с подкреплением на основе координационных графов для совместного восприятия спектра при Коррелированное затухание. IEEE Вирел. Коммун. Летт. 2020, 9, 1778–1781. [Перекрестная ссылка]

61. Гао, А.; Ду, К.; Нг, SX; Лян, В. Совместное зондирование спектра с использованием многоагентного подхода к обучению с подкреплением в когнитивных радиосетях. IEEE-сообщество. Летт. 2021, 25, 2604–2608. [Перекрестная ссылка]

62. Нго, QT; Джаявикрама, бакалавр; Привет.; Дуткевич, Э. Мультиагентное зондирование спектра на основе DRL с помощью RIS и когнитивных функций. Спутниково-наземные сети. IEEE Вирел. Коммун. Летт. 2023, 12, 2213–2217. [Перекрестная ссылка]

63. Али, А.; Хамуда, В. Достижения в области определения спектра для сетей когнитивного радио: теория и приложения. IEEE-сообщество. Выж. Репетитор. 2017, 19, 1277–1304. [Перекрестная ссылка]

64. Арджун, Ю.; Каабуш, Н. Комплексное исследование по распознаванию спектра в сетях когнитивного радио: последние достижения, новые Проблемы и будущие направления исследований. Датчики 2019, 19, 126. [CrossRef]

65. Чжан Ю.; Луо, З. Обзор исследований по распознаванию спектра на основе глубокого обучения. Электроника 2023, 12, 4514. [CrossRef]

66. Видженаяке, К.; Маданаяке, А.; Кота, Дж.; Брутон, Л. Спектральные белые пространства пространства-времени в когнитивном радио: теория, алгоритмы, и Цепи. IEEE J. Emerg. Сел. Вершина. Сист. цепей. 2013, 3, 640–653. [Перекрестная ссылка]

67. Маданаяке, А.; Видженаяке, К.; Тран, Н.; Куклев, Т.; Хум, С.; Брутон, Л.Т. Направленное зондирование спектра с использованием настраиваемых многомерных пространственно-временных дискретных фильтров. В материалах Международного симпозиума IEEE «Мир беспроводных, мобильных и мультимедийных сетей», Сан-Франциско, Калифорния, США, 25–28 июня 2012 г.; стр. 1–6.

Machine Translated by Google

68. Видженаяке, К.; Маданаяке, А.; Брутон, Лейтенант; Девабхактуни, В. Оценка DOA и локализация источника в CR-сетях с использованием управляемых двумерных БИХ-фильтров. В материалах Международного симпозиума IEEE по схемам и системам, Пекин, Китай, 19–23 мая 2013 г.; стр. 65–68.

69. Гунаратне, Т.К.; Брутон, Л.Т. Адаптивные двухмерные трапециевидные КИХ-фильтры с комплексными коэффициентами для широкополосного формирования луча в когнитивных радиосистемы. Сист. цепей. Сигнальный процесс. 2011, 30, 587–608. [Перекрестная ссылка]

70. Уилкокс, Д.; Цакалаки, Э.; Кортун А.; Ратнараджа, Т.; Пападиас, СВ; Селлатурай, М. Об использовании когнитивного радио в пространственной области Однорадиопаразитарные антенные решетки. IEEE Дж. Сел. Районы Коммун. 2013, 31, 571–580. [Перекрестная ссылка]

71. Цянь, Р.; Селлатурай, М.; Ратнараджа, Т. Направленное спектральное зондирование для когнитивного радио с использованием массивов ESPAR с одной радиочастотной цепью. В материалах Европейской конференции по сетям и коммуникациям 2014 г. (EuCNC), Болонья, Италия, 23–26 июня 2014 г.; стр. 1–5.

72. Яздани, Х.; Восуги А. О системах когнитивного радио с направленными антеннами и несовершенным спектральным зондированием. В материалах Международной конференции IEEE по акустике, речи и обработке сигналов (ICASSP) 2017 г., Новый Орлеан, Луизиана, США, 5–9 марта 2017 г.; стр. 3589–3593.

73. Пулипати, С.; Арияратна, В.; Эдуссурия, США; Видженаяке, К.; Ван, Х.; Маданаяке, А. Многолучевое направленное зондирование источников ISM 2,4 ГГц в реальном времени на основе FPGA. В материалах конференции инженерных исследований Моратувы 2019 г. (MERCon), Моратува, Шри-Ланка, 3–5 июля 2019 г.; стр. 129–134.

74. Ричард Томпсон, А.; Гергели, Т.Э.; Ванден Бут, Пенсильвания Помехи и радиоастрономия. Физ. Сегодня 1991, 44, 41–49. [Перекрестная ссылка]

75. Обсерватория «Научные цели – квадратный километр». Доступно онлайн: https://www.skao.int/en/explore/science-goals . (доступ 27 мая 2024 г.).

76. Карпентер Дж.; Броган, К.; Ионо, Д.; Мрочковски, Т. Обновление широкополосной чувствительности ALMA2030. arXiv 2022, arXiv:2211.00195.

77. Наука с очень большим массивом нового поколения. Доступно онлайн: https://ngvla.nrao.edu/page/science . (доступ 27 май 2024 г.).

78. Сихлангу, И.; Узир, Н.; Бассетт, Б.А. Природа и эволюция радиочастотных помех УВЧ и L-диапазона на радиотелескопе MeerKAT . В материалах семинара RFI 2022, Европейский центр среднесрочных прогнозов погоды (ECMRWF), онлайн, 14–18 февраля 2022 г.

79. Галт Дж. Загрязнение со спутников. Природа 1990, 345, 483–483. [Перекрестная ссылка]

80. Де Пре, CG; Андерсон, ЧР; Желева, М. 2023. Астрономии угрожают радиопомехи от спутников. Что можно с этим сделать. Доступно онлайн: https://www.weforum.org/agenda/2023/03/astronomy-radio-interference-satellites-technology/ . (по состоянию на 27 мая 2024 г.).

81. Предпочтительные полосы частот для радиоастрономических измерений ниже 1 ТГц. Рекомендация МСЭ-R RA.314-11, Международный союз электросвязи — Сектор радиосвязи. 2023. Доступно онлайн: https://www.itu.int/dms_pubrec/itu-r/rec/. ra/R-REC-RA.314-11-202312-I!!PDF-E.pdf (по состоянию на 27 мая 2024 г.).

82. Критерии защиты, используемые при радиоастрономических измерениях. Рекомендация МСЭ-R RA.769-2, Международный союз электросвязи — Сектор радиосвязи. 2003. Доступно в Интернете: https://www.itu.int/dms_pubrec/itu-r/rec/ra/R-REC-RA.769-2-200305-I!!PDF-E.pdf (по состоянию на 27 мая 2024 г.).

83. Фридман, П.А.; Баан, Вашингтон. Методы подавления радиопомех в радиоастрономии. Астрон. Астрофиз. 2001, 378, 327–344. [Перекрестная ссылка]

84. Чжан Л.; Джин, Р.; Чжан, К.; Ван, Р.; Чжан, Х.; Вэнь, З. Объединенный метод обнаружения, локализации и подавления радиопомех путем объединения одномерных и двумерных радиометров с синтетической апертурой. Remote Sens. 2024, 16, 667. [CrossRef]

85. Проакис, Дж.Г.; Манолакис, генеральный директор по цифровой обработке сигналов; Пирсон Прентис Холл: Аппер-Сэддл-Ривер, Нью-Джерси, США, 2007 г.

86. Оппенгейм, А.В.; Шафер, РВ; Бак, Дж. Обработка сигналов в дискретном времени; Международные издания Прентис Холл; Прентис-Холл: Аппер- Сэддл-Ривер, Нью-Джерси, США, 1999 г.

87. Багай, Р.; Димитров В. Алгоритм DHT, основанный на кодировании целых алгебраических чисел. Электрон. Летт. 1999, 35, 1303–1305. [Перекрестная ссылка]

88. Британак, В.; Йип, П.; Рао, К.Р. Дискретные косинусные и синусоидальные преобразования; Академическое издательство: Кембридж, Массачусетс, США, 2007.

89. Фейг, Э.; Виноград С. О мультипликативной сложности дискретных косинусных преобразований. IEEE Транс. Инф. Теория 1992, 38, 1387–1391. [Перекрестная ссылка]

90. Малвар, Х.С.; Халлапуро, А.; Карцевич, М.; Керофски, Л. Преобразование низкой сложности и квантование в H.264/AVC. IEEE Транс. Сист. цепей. Видео Технол. 2003, 13, 598–603. [Перекрестная ссылка]

91. Виноград С. Арифметическая сложность вычислений; Серия региональных конференций CBMS-NSF по прикладной математике; СИАМ: Филадельфия, Пенсильвания, США, 1980 г.; Том 33.

92. Виноград С. О вычислении дискретного преобразования Фурье. Математика. Вычислить. 1978, 32, 175–199. [Перекрестная ссылка]

93. Хайдеман, Мультипликативная сложность MT, свертка и ДПФ; Шпрингер: Гейдельберг, Германия, 1988.

94. Оппенгейм, А.В.; Вергезе, « Сигналы, системы и вывод ГХ»; Серия обработки сигналов Prentice-Hall; Пирсон: Верхнее седло Ривер, Нью-Джерси, США, 2016 г.

95. Синтра, Р. Дж. Метод целочисленной аппроксимации дискретных синусоидальных преобразований. Сист. цепей. Сигнальный процесс. 2011, 30, 1481–1501. [Перекрестная ссылка]

96. Суарес Вильягран, Д.М. Аппроксимации дискретного преобразования Фурье с приложениями в обнаружении и оценке. Магистр Диссертация, Федеральный университет Пернамбуку, Ресифи, Бразилия, 2015 г.

Machine Translated by Google

97. Маданаяке, А.; Арияратна, В.; Мадишетти, С.; Пулипати, С.; Синтра, Р.Дж.; Коэльо, Д.; Оливейра, Р.; Байер, FM; Белостоцкий Л.; Мандал, С.; и др. На пути к 1024-лучевой цифровой матрице с низким SWaP: 32-лучевая подсистема на частоте 5,8 ГГц. IEEE Транс. Антенны Пропаг. 2020, 68, 900–912. [Перекрестная ссылка]

98. Маданаяке, А.; Синтра, Р.Дж.; Акрам, Н.; Арияратна, В.; Мандал, С.; Коутиньо, Вирджиния; Байер, FM; Коэльо, Д.; Раппапорт, ТС Приблизительные ДПФ Fast Radix-32 для формирования цифрового радиочастотного луча с 1024 лучами. IEEE Access 2020, 8, 96613–96627. [Перекрестная ссылка]

99. Блахут Р.Э. Быстрые алгоритмы обработки сигналов; Издательство Кембриджского университета: Кембридж, Великобритания, 2010 г. [CrossRef]

100. Сотрудничество в области астрономической обработки сигналов и исследований в области электроники. 2024. https://casper.berkeley.edu/ (доступ на 23 мая 2024 г.).

101. Инструментарий CASPER. Гитхаб. 2024. https://github.com/casper-astro/mlib_devel/ (по состоянию на 24 мая 2024 г.).

102. Штуцман, В.Л.; Тиле, Г. А. Теория и проектирование антенн; Уайли: Хобокен, Нью-Джерси, США, 2012 г.

103. Куласекера, С.; Маданаяке, А.; Суарес, Д.; Синтра, Р.Дж.; Байер, FM Апертуры многолучевого приемника с использованием 8-точечного аппроксимированного ДПФ без множителя. В материалах конференции IEEE Radar Conference 2015 (RadarCon), Арлингтон, Вирджиния, США, 10–15 мая 2015 г.; стр. 1244–1249. [Перекрестная ссылка]

104. Чжао, Х.; Маданаяке, А.; Синтра, Р.Дж.; Мандал, С. Аналоговый 8-точечный многолучевой формирователь с приближенным ДПФ в токовом режиме и коэффициентом усиления 4,7 Пропускная способность канала Гбит/с. IEEE Access 2023, 11, 53716–53735. [Перекрестная ссылка]

105. Куласекера, С.; Маданаяке, А.; Видженаяке, К.; Байер, FM; Суарес, Д.; Синтра, Р.Дж. Многолучевые цифровые формирователи луча с радиочастотной апертурой 8 × 8, использующие двумерные аппроксимации БПФ без множителей. В материалах конференции инженерных исследований Моратувы 2015 г. (MERCon), Моратува, Шри-Ланка, 7–8 апреля 2015 г.; стр. 260–264. [Перекрестная ссылка]

106. Арияратна, В.; Маданаяке, А.; Тан, Х.; Коэльо, Д.; Синтра, Р.Дж.; Белостоцкий Л.; Мандал, С.; Раппапорт, TS 8/16-лучевые алгоритмы аналогового приближенного БПФ, архитектуры и КМОП-схемы для приемопередатчиков MIMO с формированием диаграммы направленности 5G. ИИЭР Дж. Экстрен. Сел. Вершина. Сист. цепей. 2018, 8, 466–479. [Перекрестная ссылка]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Полная проекция субхеджа для автоматов пошагового хеджирования †

Антонио Аль Серхали * и Йоахим Нирен *

Центр Инрия, Университет Лилля, 59000 Лилль, Франция
* Адрес для переписки: antonio.al-serhali@inria.fr (AAS); joachim.niehren@inria.fr (JN)
† Эта статья представляет собой расширенную версию нашей статьи, опубликованной на Международном симпозиуме по основам теории вычислений, Трир, Германия, 18–21 сентября 2023 г.

Аннотация: Мы демонстрируем, как оценивать автоматы пошагового хеджирования (SHA) с проецированием субхеджирования, при этом полностью проецируя нерелевантные субхеджирования. Поскольку для этого требуется передача конечной информации о состоянии сверху вниз, мы вводим понятие нисходящего пошагового хедж-автомата. Мы используем их для определения оценщиков в памяти и потоковой передачи с полной проекцией подхеджирования для SHA. Затем мы настраиваем оценщиков так, чтобы они могли принять решение о членстве на самом раннем этапе. Мы применяем наши алгоритмы к проблеме ответа на обычные запросы XPath к потокам XML . Наши эксперименты показывают, что полное субхеджирование SHA действительно может ускорить получение ответов на самые ранние запросы к потокам XML и стать конкурентоспособным по сравнению с лучшими существующими инструментами потоковой передачи для запросов XPath.

Ключевые слова: автоматы; алгоритм проецирования; потоковый алгоритм; XML

1. Введение

Живые изгороди — это последовательности букв алфавита и деревьев h , где h — снова живая изгородь. Хеджи абстрагируются от синтаксических деталей документов XML или JSON , но при этом могут их представлять. Линеаризация хеджа — это вложенное слово, которое можно записать в файл. В этой статье мы изучаем проблему сопоставления шаблонов хеджирования, т. е. принадлежит ли данный хедж h заданному регулярному хедж-языку L. Регулярные хедж- языки могут быть определены, в частности, в XPath или JSONPath.

Проекция необходима для эффективности многих алгоритмов на словах, деревьях, изгородях или вложенных словах. Интуитивно понятно, что алгоритм является проектирующим, если он посещает только фрагмент входной структуры: в лучшем случае ту часть, которая имеет отношение к рассматриваемой проблеме. Актуальность проекции для обработки XML уже отмечалась в [1–4]. Например, встроенный в память оценщик Saxon проецирует входные XML- документы относительно программы XSLT , которая содержит набор запросов XPath , на которые необходимо ответить одновременно [5]. Инструмент QuiXPath [4] оценивает запросы XPath в потоковом режиме с проекцией поддерева и потомка. Проецирование во время оценки запросов JSONPath на документы JSON в потоковом режиме называется быстрой перемоткой вперед [6].

Проецирование оценки в памяти предполагает, что полный граф входного хеджа построен заранее. Тем не менее, проекция все равно может сэкономить время, если необходимо сопоставить несколько шаблонов на одном и том же входном хедже или если график все равно был построен по разным причинам. В потоковом режиме ситуация аналогичная: необходимо проанализировать весь входной хедж в потоке, но оценщикам нужно проверять только те узлы, которые не проецируются. Учитывая, что чистый синтаксический анализ выполняется на два или три порядка быстрее, чем оценка шаблона, выигрыш во времени при проецировании может быть значительным.

Отправной точкой этой статьи является понятие нерелевантности позиций хеджа h по отношению к хедж-языку L , который мы вводим. Это позиции h , в которых для всех возможных продолжений вставка какого-либо субхеджа не влияет на принадлежность к L. Мы предлагаем алгоритм сопоставления шаблонов хеджирования с полной проекцией субхеджа. Наш алгоритм определяет соответствие шаблона хеджирования, полностью проецируя субхеджи, расположенные в нерелевантных позициях субхеджирования. Другими словами

check for updates

Цитирование: Аль Серхали, А.; Нирен, Дж. Полная проекция субхеджа для автоматов пошагового хеджирования. Алгоритмы 2024, 17, 339. https://doi.org/10.3390/a17080339 .

Академический редактор: Хеннинг Фернау и Клаус Янсен

Поступила: 29 января 2024 г.
Пересмотрено: 23 июля 2024 г.
Принято: 23 июля 2024 г.
Опубликовано: 2 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

Machine Translated by Google

соответствует ли изгородь h шаблону L без посещения какой-либо части изгороди h , расположенной в позиции, нерелевантной по отношению к L.

Регулярные языки хеджирования могут быть представлены вложенными регулярными выражениями, которые могут быть получены из обычных запросов XPath [7] , или какими-либо автоматами хеджирования, которые могут быть скомпилированы из вложенных регулярных выражений. Мы используем пошаговые хедж-автоматы (SHA) [8], недавний вариант стандартных хедж-автоматов, которые, в свою очередь, появились в 1960-х годах [9,10 SHA смешивают обработку стандартных древовидных автоматов снизу вверх с обработкой автоматов с конечным словом (DFA) слева направо . Они не поддерживают обработку сверху вниз и не имеют явного стека, в отличие от вложенных словных автоматов NWA [11–15]. SHA имеют хорошее представление о детерминизме снизу вверх и слева направо, избегая проблематичного понятия детерминизма для стандартных хедж-автоматов и слишком дорогостоящего понятия детерминизма для NWA. Кроме того, доступны компиляторы из вложенных регулярных выражений в SHA [8], а алгоритмы детерминизации [16] создают небольшие SHA для всех обычных запросов XPath в существующих практических тестах [7,17].

Целью настоящей работы является добавление проекции подхеджирования к недавнему алгоритму потоковой передачи для детерминированных SHA , который может обрабатывать обычные запросы XPath самым ранним способом. Большинство альтернативных подходов потоковой передачи избегают самого раннего ответа на запрос, рассматривая подъязыки регулярных запросов без задержек, так что выбор узла зависит только от прошлого узла, но не от будущего [18,19] или допуская поздний выбор [20].
В частности, недавно было продемонстрировано, что самый ранний ответ на запрос для обычных монадических запросов, определенных детерминированными SHA [21], имеет меньшую сложность в худшем случае, чем для детерминированных NWA [22]. Таким образом, впервые на практике стал возможен самый ранний ответ на регулярные запросы . С другой стороны, экспериментально он по-прежнему медленнее, чем лучшие существующие не самые ранние подходы для общих регулярных монадических запросов к потокам XML (см. сравнение в [23]), поскольку последний поддерживает проецирован То, как можно делать прогнозы для CC3, не изучалось до версии настоящей статьи, представленной на конференции [24].

Рассмотрим пример фильтра XPath [self::list/child::item] , которому удовлетворяет XML - документ, если его корнем является элемент списка , имеющий дочерний элемент . При кодировке XML - документа в качестве хеджей, выбранных в настоящей статье, удовлетворяющие хеджи имеют вид list · h1 · item · h2 · h3 · h4, а для некоторых хеджей h1, h2, h3, h4. При оценке этого фильтра на хедже достаточно проверить первое поддерево на наличие списка корневых меток, а затем все его дочерние элементы, пока не будет найдено одно с элементом корневой метки . Изгороди этих дочерних элементов, т. е. изгородь h3 и подизгороди деревьев верхнего уровня в h1 и h3, можно спроецировать, так же как изгородь h4: их не нужно проверять для оценки этого фильтра. Однако необходимо запомнить, равен ли уровень текущего узла 0, 1 или выш Этот уровень информации может естественным образом обновляться сверху вниз. Однако оценщики SHA действуют исключительно снизу вверх и слева направо. Следовательно, проектирующие оценщики для SHA должны основываться на более общих машинах. Мы предлагаем нисходящие пошаговые хедж-автоматы (SHA), вариант SHA , который поддерживает обработку сверху вниз. По сути, это автоматы с выталкивающим лесом Неймана и Зейдля [25], за исключением того, что они применяются к немаркированным изгородям, а не к помеченным лесам. Кроме того, известно, что NWA действуют в основном таким же образом с вложенными словами [26], допуская при этом несколько более общие видимые нажатия вниз. Затем мы различаем состояния проекции субхеджирования для SHA и покажем, как использовать их для оценки SHA с проекцией субхеджирования как в памяти, так и в потоковом режиме.

В качестве первого вклада мы представляем безопасный компилятор без изменений из SHA в SHA , который может различать соответствующие состояния проекции субхеджа. Идея состоит в том, что безопасный без изменений SHA может различать контексты, в которых состояния SHA не будут безопасно меняться. Например, фильтр XPath [self::list/child::item] может быть определен детерминированным SHA , работающим снизу вверх, который наш компилятор сопоставляет с SHA , работающим сверху вниз. Контекстная информация, ставшая явной в результате нисходящей обработки, касается уровней состояний. Это позволяет нам различать состояния проекции, начиная с уровня 2, в которых субхеджи можно игнорировать. Мы доказываем корректность нашего компилятора на основе нетривиального инварианта:

Machine Translated by Google

устанавливаем (заметим, что доказательство потребовало адаптации оригинального компилятора из FCT [24]). Мы также представляем контрпример, показывающий, что прогноз «без изменений» не является полным для прогноза субхеджирования. Это показывает, что субхедж может быть неактуальным, даже если его состояние все еще может меняться.

В качестве второго вклада мы предлагаем алгоритм конгруэнтного проецирования. Он снова компилирует SHA в SHA , но опирается на отношения конгруэнтности состояний автомата. Затем мы доказываем, что конгруэнтное проецирование дает не только надежный алгоритм, но и то, что этот алгоритм также является полным для проецирования субхеджирования, т. е. все сильно нерелевантные субхеджы (см. определение 9) переводятся в состояние цикла. Конгруэнтная проекция начинается на верхнем уровне изгородей с конгруэнтности Майхилла – Нероде, хорошо известной из минимизации автоматов.

Для языков со словами L это сравнение идентифицирует префиксы v , имеющие одинаковые (L) = {w | v · оставшийся язык v ^{1}w L}. Префикс v тогда является нерелевантным суффиксом, если его (L) либо остаточный язык v для ^{1}универсален, либо пуст. В случае живых изгородей, которые простираются слов путем добавления иерархической вложенности через конструктор дерева, соответствие должно быть адаптировано при переходе к поддеревьям.

Мы показываем, что оба компилятора могут увеличивать размер автоматов в геометрической прогрессии, поскольку они должны иметь возможность преобразовывать восходящие детерминированные автоматы на монадических деревьях в детерминированные автоматы сверху вниз. Это то же самое, что обращение детерминированных автоматов над словами, которое, как известно, приводит к экспоненциальному разрушению в худшем случае (например, для семейства языков (a + b) , где n · a ^{n}(a + b)^{n} минимальный детерминированный лево- автомат, работающий справа налево , имеет 2n+1 состояний, тогда как минимальный детерминированный автомат, работающий справа налево, имеет n + 1 состояний).

Экспоненциального взрыва можно избежать, если вы заинтересованы в оценке хеджирования автоматом с проекцией субхеджирования, не создавая полные SHA статически. Вместо этого только необходимая часть SHA может быть построена на лету при оценке хеджирования.

Наш третий вклад — усовершенствование двух предыдущих компиляторов, чтобы они также могли различать безопасные состояния для выбора в самой ранней позиции. Для этого мы объединили эти компиляторы с вариантом самого раннего тестера членства из [21] , который работает в памяти, компилируя SHA вместо NWA. Более того, отказ от членства также обнаруживается на самой ранней позиции. Таким образом, мы получаем самый ранний тестер членства в памяти для детерминированных SHA.

Наш четвертый вклад — показать, как запускать предыдущие оценщики в памяти в потоковом режиме, одновременно считывая линеаризацию хеджирования во вложенное слово в качестве входных данных. Таким образом, мы можем улучшить недавний самый ранний тестер членства в потоковой передаче, лежащий в основе инструмента AStream [21] , добавив к нему проекцию подхеджирования.

Наш пятый и последний вклад — это реализация и экспериментальная оценка потокового алгоритма ответа на самые ранние запросы для dSHA путем введения субхедж- проекции в инструмент AStream [21]. Мы реализовали как проекцию «безопасно без изменений», так и проекцию конгруэнтности. Для этого мы заменили наш самый ранний тестер членства с субхедж-проекцией на самый ранний алгоритм ответа на запросы для монадических запросов. Для экспериментов мы начали с детерминированных SHA , созданных с помощью компилятора из [8] для прямых прямых запросов XPath теста XPathMark [27] и реальных запросов XPath [7]. Оказывается, что проекция конгруэнтности проявляется гораздо сильнее, чем проекция без изменений, по крайней мере, для половины эталонных запросов. Это значительно сокращает время выполнения всех обычных запросов XPath , поскольку большие части входных хеджирования могут быть спроецированы . В нашем тесте прогнозируемый процент варьируется от 75,7% до 100% входного потока. Для запросов XPath , которые содержат только дочерние оси, самый ранний алгоритм ответа на запросы AStream с проекцией конгруэнтности становится конкурентоспособным по эффективности с лучшим существующим инструментом потоковой передачи под названием QuiXPath [23], который не всегда является самым ранним для некоторых запросов. Наша текущая реализация самой ранней проекции конгруэнтности в AStream в тестовых запросах в 1,3–2,9 раза медленнее, чем QuiXPath. Улучшение меньше для запросов XPath с осью потомков, где меньшая проекция субхеджа.

Machine Translated by Google

возможный. Вместо этого потребуется какая-то проекция потомков. Тем не менее, даже в худшем случае в нашем тесте самая ранняя проекция конгруэнтности с AStream в настоящее время всего в 13,8 раза медленнее, чем QuiXPath.

Контур. Мы начнем с предварительных сведений о хеджах и вложенных словах в разделе 2. В разделе 3 мы напомним о проблеме сопоставления шаблонов хеджирования и формально определим нерелевантные субхеджи. В разделе 4 мы напоминаем определение SHA, знакомим с SHA и обсуждаем их оценщики в памяти. В разделе 5 мы вводим понятие состояний проекции субхеджирования, определяем, когда SHA является полным для проекции субхеджирования, и представляем оценщик в памяти для SHA с проекцией субхеджирования. В разделе 6 мы вводим проекцию «безопасно-без изменений», утверждаем ее обоснованность и показываем ее неполноту. В разделе 7 мы вводим конгруэнтную проекцию и доказываем, что она корректна и полна для субхедж-проекции. Самые ранние оценщики в памяти для SHA с проекцией субхеджирования описаны в разделе 8. Потоковые варианты наших оценщиков в памяти систематически выводятся в разделе 9. В разделе 10 обсуждается, как перенести наши алгоритмы с тестирования членства на монадические запросы. В разделе 11 обсуждаются наши практические эксперименты. В разделе 12 обсуждается дальнейшая работа над понятиями автоматов, связанными с SHA и SHA . В Приложении A представлена переформулировка самого раннего алгоритма запроса-ответа из [21], основанного на SHA и с ограничениями схемы. Приложение B дает убедительное доказательство прогноза без изменений .

Комментарии к публикации. Эта оригинальная версия журнальной статьи была опубликована на конференции FCT [24]. По сравнению с предыдущей публикацией, следующие материалы новые:

- Определение субхеджирования нерелевантных префиксов вложенных слов и определение полнота для проекции субхеджа (раздел 3).
- Добавление ограничений схемы на протяжении всей статьи и понятие схемы. полнота (раздел 4.3). • Понятие
полноты проекции субхеджа (раздел 5.2). • Доказательство надежности алгоритма
прогнозирования без изменений в разделе 6.2 перенесено к приложению Б.
- Алгоритм конгруэнтного проецирования и доказательство его корректности и полноты для подхедж-проецирования (раздел 7). • Систематический
метод добавления прогноза субхеджирования к самому раннему SHA (раздел 8). • Систематический
метод сведения корректности вычислителей потоковых автоматов к соответствующим вычислителям в памяти (раздел 9). • Обсуждение того, как справляться с
монадическими запросами, используя при этом доступность ограничения схемы (раздел 10).
- Реализация проекции конгруэнтности с экспериментальными результатами для оценки XPath в потоках XML (раздел 11). • Более продолжительное
обсуждение соответствующей работы (Раздел 12). • В Приложении
A мы также показываем, как получить самые ранние dSHA из dSHA, адаптировав компилятор из [21] , сопоставляющий dSHA с самыми ранними dNWA.

2. Предварительные сведения

Пусть A и B — множества. Если A является подмножеством B, то дополнение A в B обозначается как $\overline{A} = B \setminus A$, сохраняя при этом множество, относительно которого дополнение берется неявно. Областью определения бинарного отношения $r \subseteq A \times B$ является $\mathrm{dom}(r) = \{a \in A \mid \exists b \in B.\ (a, b) \in r\}$. Частичная функция $f : A \to B$ — это бинарное отношение $f \subseteq A \times B$, которое является функциональным. Полная функция $f : A \to B$ — это частичная функция $f : A \to B$ с такой, что $\mathrm{dom}(f) = A$.

Слова. Пусть N — множество натуральных чисел, включая 0. Пусть алфавит Σ — множество. Множество слов над Σ — это Σ^*. Слово — это $(a_1, \ldots, a_n) \in \bigcup_{n \in \mathbb{N}} \Sigma^n$, где $n \in \mathbb{N}$ записано как $a_1 \ldots a_n$. Пустое слово длины 0 обозначим через $\varepsilon \in \Sigma^0$, и по $v_1 \cdot v_2 \in \Sigma^*$ тот двух слов $v_1, v_2 \in \Sigma^*$.

Machine Translated by Google

Если v = u · v' · w — слово, то мы называем u префиксом слов v и v ' множитель v и w суффикс, мы из v. Для любого подмножества L , обозначаем набор префиксов слов в L через prefs(L) и Σ множество суффиксов слов из L с помощью suffs(L).

Для любого подалфавита ' Σ — проекция слова v Σ) , к Σ ' это слово projΣ ′(v) Σ в 'полученная из v удалением всех букв из Σ \ Σ '.

(Σ Hedges. Живые изгороди — это последовательности букв и деревьев h с изгородью h. Более формально, изгородь h НΣ имеет следующий абстрактный синтаксис:

x, x ' НΣ ::= ε | a | ч | час · час ' где a € Σ

Предположим, что ε · h = h · ε = h и (h · h1) · h2 = h · (h1 · h2). Поэтому рассмотрим любое слово aab = a в Σ как хедж в НΣ , т.е. Σ · a · b НΣ . Любой хедж можно преобразовать или сохранить в виде графика. Для подписи Σ = {list, item, A, . . . , Z}, граф примера хеджа в НΣ показан на рисунке 1. Он кодирует последовательность XML- документов на рисунке 2:

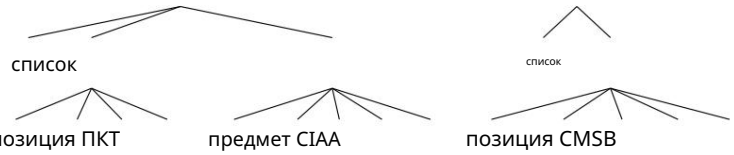


Рисунок 1. Граф хеджа список · item · F · C · T · item · C · I · A · A · list · item · C · M · S · B равен последовательность деревьев.

```
<list>
    <item>FCT</item>
    <item>CIAA</item> </
list>
<список>
    <item>CMSB</item> </
list>
```

Рисунок 2. Последовательность XML- документов, закодированных хеджем на рисунке 1.

Вложенные слова. Вложенное слово над Σ — это слово в алфавите Σ ^ = Σ { , }, в котором все круглые скобки правильно вложены. Интуитивно это означает, что каждой открывающей скобке соответствует соответствующая закрывающая скобка, и наоборот.

Множество вложенных слов над Σ можно определить как подмножества слов над Σ ^ , которые линеаризация хеджа, где функция линеаризации nw : НΣ (Σ { , }) определяется следующим образом:

nw(ε) = ε, nw(h) = · nw(h) · , nw(a) = a, nw(h · h ') = nw(h) · nw(h ').

Итак, множество всех вложенных слов над Σ есть nw(НΣ). Пусть hdg — обратная инъективная функция nw, ограниченная ее образом, поэтому отображение вложенных слов в хеджирование с hdg(nw(h)) = h для всех h НΣ .

Вложенные префиксы слов. Префиксы, суффиксы и факторы вложенных слов сами по себе не могут быть вложенными словами. Например, изгородь h = a · b · c имеет линеаризацию nw(h) = a bc . Его префикс a b плохо вложен, так как у него есть висячая открывающая скобка. Его суффикс c также плохо вложен, поскольку у него есть висячая закрывающая скобка.

Любой алгоритм, который пересекает изгородь h сверху вниз и слева направо, проверяет все префиксы nw(h). Любой префикс v в nw(h) не только различает позицию хеджа h , но также указывает часть хеджа, которая находится перед этой позицией в линеаризации nw(h). Пример графически показан на рисунке 3. Это особенно справедливо для алгоритмов потоковой передачи, которые получают вложенное слово nw(h) в качестве входных данных в потоке.

Machine Translated by Google

и может быть проверен только один раз слева направо. Но это в равной степени справедливо и для алгоритмов в памяти , которые получают входной хедж h как иерархическую структуру, график которой хранится в памяти, а затем обходит его сверху вниз и слева направо. Любой узел будет посещен дважды: один раз при чтении открывающей скобки и переходе в подхедж, и другой раз при переходе к закрывающей скобке после обработки подхедж.

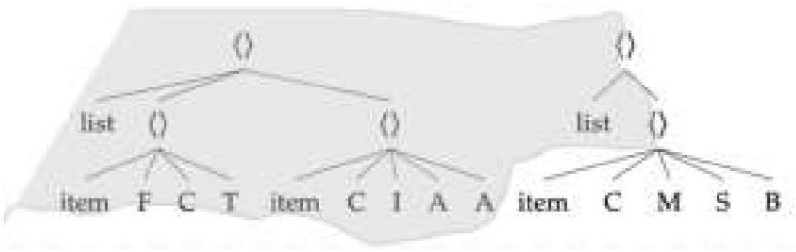


Рисунок 3. Часть хеджа, отличающаяся вложенным префиксом слов list · item · F · C · T · item · C · I · A · A · list · .

3. Проблема

Мы формально определяем задачу сопоставления шаблонов для хеджирования и понятие нерелевантности субхеджирования, которое позволяет нам охарактеризовать те части входного хеджа, которые не нужно посещать во время сопоставления с образцом.

3.1. Сопоставление шаблонов

хеджирования Нас интересуют алгоритмы, которые сопоставляют шаблон хеджирования с хеджированием. Начнем с алгоритмов, которые проверяют, существует ли такое совпадение. Иногда это называют фильтрацией или, альтернативно, логическим ответом на запрос. В разделе 10 мы рассматриваем более общую проблему возврата набора паросочетаний. Эту проблему иногда называют монадическим ответом на запрос. В наших экспериментах в разделе 11 применяется ответ на монадический запрос для обычных запросов XPath.

Мы начинаем с шаблонов хеджирования, которые представляют собой вложенные регулярные выражения с буквами в сигнатуре Σ . Для этого мы фиксируем набор V рекурсивных переменных и предполагаем, что он не пересекается с Σ .

e, e nRegExpΣ ::= ε | a | r | e · e | e + e | e * | e | μz.e, где a Σ , z V

Каждому вложенному регулярному выражению e nRegExpΣ удовлетворяет подмножество хеджей из e HΣ V. Напомним определение этого множества лишь неформально. Выражению ε удовлетворяет только пустое слово, выражению a Σ — только хедж a , а выражению z V — только однобуквенный хедж z. Выражение e · e с оператором конкатенации и соответственно. e,e со звездным оператором Клини и e nRegExpΣ удовлетворяется повторениями. Выражение e из слов, e. nRegExpΣ удовлетворяется конкатенацией хеджей, удовлетворяющих e и e удовлетворяющее удовлетворяющим обеспечивает горизонтальную рекурсию на живых изгородях. Выражению μ μz.e , где z V и e nRegExpΣ , хеджи без рекурсивной переменной x , в которых z неоднократно заменяется хеджем, удовлетворяющим e , пока он не исчезнет. Мы принимаем обычное ограничение [28] , согласно которому в любом вложенном регулярном выражении μz.e все вхождения z в e охраняются конструктором дерева . . Таким образом, μ-оператор обеспечивает правильную вертикальную рекурсию. Также обратите внимание, что для определения подъязыков HΣ нам нужны только такие выражения в nRegExpΣ , в которых все вхождения переменных рекурсии z связаны в области действия связывателя μz.

Хорошо известно, что вложенные регулярные выражения в nRegExpΣ могут определять все регулярные подъязыки хеджей в HΣ , т.е. они имеют ту же выразительность, что и хедж-автоматы с алфавитом Σ . Аналогичные результаты для регулярных древесных выражений и древесных автоматов являются фольклором [29].

Machine Translated by Google

Пример 1 (Вложенные регулярные выражения обычных фильтров XPath). Например, мы можем определить подмножества хедж-кодировок XML- документов с двумя типами элементов в Σ = {list, item} с помощью выражений в nRegExpΣ . В этой статье мы будем использовать кодировку XML- документа, которая создает хеджи, удовлетворяющие следующему регулярному выражению doc nRegExpΣ , где z V:

$$\text{дерево}^* \text{ где дерево doc} =_{\text{def}} =_{\text{def}} \mu z. \quad (\text{список} + \text{элемент}) \cdot z^*$$

В нашем приложении кодировки немного богаче, чтобы различать различные типы узлов XML , такие как элементы, атрибуты, комментарии и тексты. Фильтр XPath , такой как [self::list/child::item], затем может быть выражен с помощью вложенного регулярного выражения:

$$\text{filter} =_{\text{def}} \quad \text{list} \cdot \text{doc} \cdot \quad \text{item} \cdot \text{doc} \quad \cdot \text{doc} \quad \cdot \text{doc}$$

Общие компиляторы от обычных запросов нисходящего фильтра XPath к вложенным регулярным выражениям были предложены в [8]. Реализация была разработана в контексте инструмента AStream [21].

Мы будем использовать схемы S НΣ , чтобы ограничить входные данные нашей задачи сопоставления с образцом. Поскольку нас в основном интересуют регулярные схемы, мы не предполагаем регулярность в глобальном масштабе, поскольку проекция без изменений не полагается на нее. Задача сопоставления с образцом для вложенных регулярных выражений относительно схемы S НΣ получает следующие входные данные:

- вложенное регулярное выражение e nRegExpΣ и хедж h
- S.

Затем он возвращает истинностное значение суждения h e. Входное хеджирование h S может быть задано либо графом, который находится в памяти, либо потоком, содержащим линеаризацию nw(h) входного хеджа, который можно прочитать только один раз слева направо.

3.2. Нерелевантные субхеджи

Мы определяем концепцию нерелевантных вхождений субхеджей относительно заданного шаблона хеджирования. Что это означает, зависит от типа алгоритма, который мы будем использовать для сопоставления с образцом. Мы будем использовать алгоритмы, которые работают с входным хеджем сверху вниз, слева направо и снизу вверх. В частности, это справедливо для потоковых алгоритмов.

Интуитивно понятно, что когда алгоритм сопоставления с образцом достигает узла сверху вниз или слева направо, последующий фрагмент которого не имеет значения, он может перепрыгнуть через него, не проверяя его структуру. Что означает прыжок, должно быть ясно, если изгородь задается графом, хранящимся в памяти. Обратите внимание, что полный график должен быть построен заранее, даже если некоторые его части могут оказаться нерелевантными. Тем не менее, можно выиграть много времени, перепрыгивая через ненужные части, если либо граф уже построен по другим причинам, либо если на одной и той же изгороди необходимо решить множество задач сопоставления с образцом.

В случае потоков нерелевантный подхедж все равно необходимо проанализировать, но в противном случае он не будет анализироваться. Чаще всего возможный анализ осуществляется автоматами, что может занять на два порядка больше времени, чем необходимо для синтаксического анализа. Таким образом, отсутствие необходимости проведения какого-либо анализа может значительно ускорить алгоритм потоковой передачи.

Определение 1. Пусть S НΣ — схема и L S — язык, удовлетворяющий этой схеме. Определим diff prefs(NΣ): $\overset{л}{С}$ как наименее симметричное отношение на префиксах вложенных слов u, u′

$$\text{разница}\overset{л}{С}(\text{ты, ты}') \quad w \quad \text{suffs}(N\Sigma). u \cdot w \quad \text{nw}(L) \quad u' \cdot w \quad \text{nw}(S \setminus L).$$

Префикс вложенного слова u называется субхеджем, релевантным для L со схемой S , если существуют вложенные слова). В $_{v, v L}$ NΣ такой, что diff $\overset{л}{С}(u \cdot v, u \cdot v$ ′противном случае префикс u называется субхеджем, не имеющим значения для со схемой S.

Итак, разница$\overset{л}{С}$(ты, ты′) утверждает, что ты и ты ′ имеют продолжения, которые ведут себя по-разному по отношению к L и S. Более того, префикс u не имеет значения для субхеджа, если принадлежность к языку не зависит от вставки хеджа в u при условии, что членство в схеме

Machine Translated by Google

гарантировано. Случай без ограничений схемы подпадает под приведенное выше определение путем выбора S = H_{Σ}^{L}. В этом случае дополнительным diff является сравнение Майхилла-Нерода H_{Σ} для L, которое хорошо известно теории формального языка благодаря минимизации ДКА (см. любой стандартный учебник по теории формального языка или Википедию (теорема Майхилла-Нерода: https: //en.wikipedia.org/wiki/Myhill-Nerode_theorem, доступ 23 июля 2024 г.)).

Это соответствие служит основой для изучения обычных языков в стиле Голда на положительных и отрицательных примерах (см. [30] или Википедию (обучение в стиле Голда в пределе https://en.wikipedia.org/wiki/Language_identification_in_the_limit, доступ 23 июля 2024 г.)).

Ограничения схемы необходимы для того, чтобы наше приложение использовало обычные шаблоны XPath.

Пример 2 (обычные фильтры XPath). Рассмотрим схему S = doc для хеджирования, представляющую XML- документы с сигнатурой Σ = {list, item}, и обычный язык хеджирования L = filter, где

фильтр = список · дерево · элемент · дерево · дерево · дерево

— это вложенное регулярное выражение из примера 1 для фильтра XPath [self::list/child::item].

Напомним, что это вложенное регулярное выражение можно применять только к хеджированию, представляющему только документы XML . Префикс вложенного слова u = list · item неактуален для языка L со схемой S. Заметим, что его продолжение list · item с суффиксом w = принадлежит L. Следовательно, для любого h H_{Σ}, если продолжение list · item · h принадлежит S = doc, то оно принадлежит и L. Тем не менее, для хеджа h1 = продолжение list · item · h1 не принадлежат L, поскольку не удовлетворяют схеме S = doc.

Префикс item также не имеет значения для языка L, даже независимо от схемы. Это связано с тем, что L не содержит никакого продолжения этого префикса к какому-либо хеджированию. Префикс list не является несущественным для языка L относительно S. Это можно увидеть с помощью суффикса w = , поскольку list не принадлежит L , а продолжение list · h с h = item принадлежит в L и оба продолжения удовлетворяют схеме S = doc.

Однако ограничения схемы могут также иметь неожиданные последствия, которые делают задачу более утомительной.

Пример 3 (Неожиданные последствия ограничений схемы). Рассмотрим сигнатуру Σ = {a}, образец L = { a } и схему S = L { · a}. Префикс действительно не имеет отношения к L и S. Чтобы убедиться в этом, мы рассмотрим все возможные закрывающие суффиксы один за другим. Самый маленький закрывающий префикс — . Заметим, что изгородь h L тогда и только тогда, когда h = a. Таким образом, принадлежность к L , по-видимому, зависит от подграницы h. Однако также обратите внимание, что h S тогда и только тогда, когда h = a. Следовательно, если предположить, что h принадлежит схеме S, он также должен принадлежать L. Таким образом, шаблон должен совпадать при предположении о членстве в схеме. Следующий закрывающий суффикс — · a. Заметим, что изгородь h · a S тогда и только тогда, когда h = ε. Однако h · a/ L, поэтому шаблон не будет совпадать ни для одного изгородей h с этим префиксом, если предположить, что входное хеджирование удовлетворяет схеме. Ситуация аналогична для более крупных суффиксов, поэтому принадлежность к языку ни в коем случае не зависит от субхеджа в префиксе , если предположить , что полный хедж удовлетворяет схеме S.

3.3. Основные свойства

Сначала мы покажем, что префиксы, не относящиеся к подхеджированию, остаются нерелевантными, если они расширены вложенным словом любого хеджа. Это свойство следует ожидать при любом разумном понятии неуместности субхеджирования.

Лемма 1. Для любого вложенного префикса слова u, языка L, S H_{Σ} и вложенного слова v N_{Σ}, если префикс u является нерелевантным для L со схемой S, то префикс u · v тоже является таковым.

Доказательство. Пусть u — субхедж, нерелевантный для L со схемой S , а v N_{Σ} — вложенное слово. Мы N_{Σ} и суффикс вложенные слова v $'_{,в}$ " вложенного слова w suffs(N_{Σ}) такие, что фиксируют произвольные

Machine Translated by Google

′ты · v · v · w nw(S) и u · v · v · w nw(S). Поскольку v · v нерелевантность ' и v · v « также являются вложенными словами, подхеджа u дает следующее:

′ты · v · v · w nw(L) u · v · v « · w nw(L).

Следовательно, префикс вложенного слова u · v не имеет значения для L со схемой S. □

Определение 2. Бинарное отношение D на префиксах вложенных слов назовем разностным отношением на префиксах , если для всех префиксов вложенных слов u, u ' prefs(NΣ) и вложенные слова v NΣ :

(и · v, и ' · v) D (u, u ') Д.

Лемма 2. разница $\overline{C}^{л}$ является разностным отношением.

Доказательство. Мы должны показать, что для всех ' prefs(NΣ) и вложенные слова v NΣ , которые являются префиксов ' u, u (u · v, u · v) $\overline{C}^{л}$ diff , следует (u, u ' разница S . Итак, позволь тебе, тебе ' префиксами вложенных слов, а v вложенное слово такое, что (u · v, u ' · v) $\text{diff}^{л}_{C}$. Тогда существует вложенный суффикс w такой, что

u · v · w nw(L) u ' · v · w nw(S \ L).

Суффикс w˜ = v · w тогда удовлетворяет условию u · w˜ nw(L) u , ' · w˜ nw(S \ L). Следовательно, (u, u ') $\text{diff}^{л}_{C}$ что и требовалось. □

В случае слов без схем дополнение $\overline{\text{diff}^{л}_{\Sigma}}$ — это отношение эквивалентности, известное как сравнение Майхилла–Нерода. Однако в случае схем дополнение $\overline{\text{diff}^{л}}$ S не может быть транзитивным; таким образом, это может даже не быть отношением эквивалентности. Это может показаться удивительным на первый взгляд.

Пример 4. Пусть Σ = {a, b}, L = {ε, b, aa} и S = L {aab}. Тогда (ε, a) diff не существует продолжения, $\overline{C}^{L}$ с тех пор продолжающего ε и a в схему. По той же причине имеем (a, aa) diff дополнение diff $\overline{C}^{л}$. Однако разница $C^{л}$ (ε, aa), поскольку ε · b L и aa · b S \ L. Таким образом, (ε, aa)/ $\overline{\text{diff}^{л}_{C}}$, так $\overline{C}^{L}$ не транзитивно.

Следует отметить, что для всех языков L diff рефлексивен и транзитивен, поэтому $\overline{\text{diff}^{л}_{H\Sigma}}$ отношение эквивалентности и, следовательно, конгруэнтность. Для общих схем S diff S может не быть $\overline{\text{diff}^{л}}$ транзитивным, как показано в примере 4, и, следовательно, может быть неконгруэнтным.

Это указывает на то, что схемы могут затруднить принятие решения о нерелевантности субхеджа. Действительно, естественный способ исключения схем, основанных на дополнениях, не работает, что подтверждается следующей леммой (в частности, контрпримером в доказательстве 1./ 2.):

Лемма 3. Для любого вложенного префикса слова u suffs(NΣ) и L, S HΣ , рассмотрим следующие два свойства:

1. u не имеет значения для L со схемой S. 2. u не имеет значения для L S со схемой $\overline{H\Sigma}$.

Тогда свойство 2 влечет за собой свойство 1, но не всегда наоборот.

Доказательство. 1./ 2.: Здесь представлен контрпример. Пусть L = {ε} и S = {ε, a}. Тогда префикс a неактуален для L со схемой S. Однако префикс a актуален для L S = HΣ \ {a} со схемой $\overline{H\Sigma}$, поскольку для вложенных слов v = ε и v = a имеем a · v/ L S и a · v ' $\overline{L}$ S.

Machine Translated by Google

2.: Далее мы предполагаем свойство 2. Рассмотрим произвольные вложенные слова v, v · w ' NΣ и 1. вложенный суффикс слова w suffs(NΣ) такой, что u · v · w, u · v, чтобы ' S. Достаточно показать, что u · v · w S и u · v ' · w S подразумевает u · v · w L u · v ' · ш € L. Это импликация тривиально выполняется, если u · v · w S или u · vu · ' · w S. В противном случае имеем v · w S и u · v ' · w S. Из свойства 2., как и предполагалось, следует u · v · w L ′ты · v · w L. Итак, свойство 1 выполнено. □

4. Автоматы хеджирования.

Мы напоминаем понятие автоматов пошагового хеджирования (SHA), вводим их нисходящее расширение (SHA), обсуждаем полноту схемы для обоих типов автоматов, определяем состояния суб-хедж-проекции для SHA и покажем, как используйте их для оценки SHA с проекцией субхеджа. В этом разделе мы ограничиваемся оценками в памяти, но обсудим потоковую оценку в разделе 9. Связь с несколькими тесно связанными понятиями автоматов, связанных с изгородями, лесами или вложенными словами, обсуждается более подробно в разделе 12.

4.1. Пошаговые хедж-автоматы

SHA — это недавнее понятие автоматов для хеджирования [8] , которые естественным образом смешивают восходящие деревья и словесные автоматы с письмом слева направо. Они расширяют пошаговые древесные автоматы [31] тем, что работают с изгородями, а не над деревьями без ранга, т. е. с последовательностями букв и деревьями, содержащими изгороди. Они отличаются от автоматов с вложенными словами (NWA) [11,15] тем, что обрабатывают хеджирование напрямую, а не их линеаризацию во вложенные слова.

Определение 3. Автоматом пошагового хеджирования (SHA) называется набор A = (Σ , Q, , I, F) , где Σ и Q — конечные множества, I, F Q и = ((a)a Σ , , @), где a Q × Q, Q и @ (Q × Q) × Q. SHA является детерминированным или, что то же самое, dSHA , если I и содержат не более одного элемента, и все отношения (a)a Σ и @ являются частичными функциями.

Множество состояний Q включает в себя подмножество I начального состояния, подмножество F конечных состояний подмножество начальных состояний дерева. Правила перехода в имеют три формы: если (q, q then y ' и) a , нас есть буквенное правило, которое мы пишем как q a q) ' @в , Если (q, q a в . Наконец, если q Q, записываем как q@p q ' то у нас есть начальное правило применения дерева, которое мы правило, которое мы обозначаем как q в .

На рисунке 4 мы иллюстрируем график dSHA для фильтра XPath [self::list/child::item].

Граф SHA получается и рисуется как обычно для автомата на словах. Состояния — это узлы графа; в примере Q = {0, 1, 2, 3, 4, 5}. Правила перехода определяют края. Ребро буквенного правила q a q в помечено буквой a Σ . В примере Σ = {список, элемент}. Любое применимое правило q@p q , помеченное состоянием p Q. Начальное состояние 0 I обозначено входящей ' р в рисуется как ребро q q серой стрелкой, а начальное состояние дерева 0 входящей серой стрелкой, помеченной символом . Конечные состояния, такие как 4 F, обозначены двойным кружком.

Machine Translated by Google

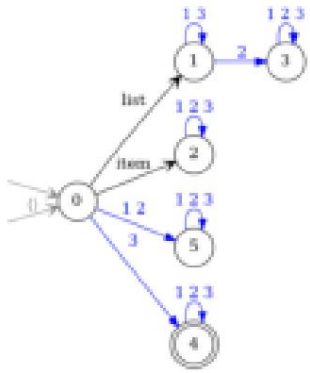


Рис. 4. dSHA для фильтра XPath [self::list/child::item], который является полной схемой для схемы. док.

4.1.1. Семантика

Затем мы определяем семантику SHA, т. е. переходы в хеджах, которые он определяет, и язык хеджирования, который он принимает. Для любого хеджа h НΣ определим шаги перехода Q, a Σ и h, h

qq час р относительно такой, что для всех q, ', п, п ' НΣ

q a ' в

qq a q ' относительно

ч q q ' относительно q ' ч ' q « относительно

ч.ч qq « относительно

q Q q

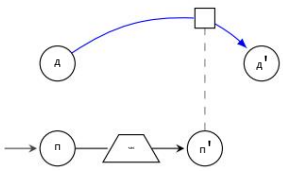
ε q относительно

h р в р

p д@п ' д ' в

qq ' относительно

Переходы можно использовать для недетерминированной оценки хеджирования снизу вверх и слева направо с использованием SHA. Первые три правила определяют, как оценивать последовательности деревьев и букв, такие как обычный конечный автомат, предполагая, что деревья уже были оценены по состояниям. Последнее правило гласит, как перевести дерево h из некоторого состояния q в некоторое состояние Это можно визуализировать следующим образом:



Для любого начального состояния дерева р необходимо недетерминированно оценить подизгородь h , чтобы вернуть в '. Для каждого п' получить его, затем недетерминированно ' такой, что q@p' д ' в состояние q некоторое состояние р .

Хедж принимается A , если он допускает переход от некоторого начального состояния к некоторому конечному. состояние. Язык L(A) представляет собой множество всех принятых хеджей:

L(A) = {h НΣ | q q час ' относительно , q I, q ' € F}.

4.1.2. Бежит

Прогоны могут представлять всю историю одного выбора недетерминированного оценщика в хедже. Например, единственный успешный запуск dSHA на рисунке 4 на хедже h = h = список · item ' — это хедж 0 с изгородью h 0 · list · 1 · 0 · item · 2 · 3 · 4 вычисление которого показано на рисунке 5. На верхнем уровне прогон оправдывает переход 4 относительно из начального состояния 0 час 0 в конечное состояние 4. При запуске в состоянии 0 сначала необходимо оценить субхедж, час ' поэтому запуск на верхнем уровне необходимо приостановить до этого

Machine Translated by Google

сделано. Подпробег на h ' в конечном итоге оправдывает переход 0 ′ч 3 относительно . В этот момент запуск верхнего уровня может быть возобновлен и продолжен в состоянии 4, поскольку 0@3 4 в .

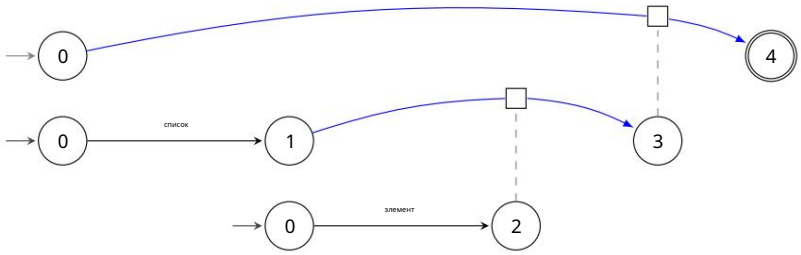


Рисунок 5. Уникальный успешный запуск dSHA на рисунке 4 на хедж-списке list · item — это хедж 0 · 0 · list · 1 · 0 · item · 2 · 3 · 4, вычисленный, как проиллюстрировано выше.

При запуске рисования мы иллюстрируем любой механизм приостановки/возобновления с помощью рамки, например, рамки на краю 0 4. Рамка обозначает будущее значение, вычисленное нижним уровнем, что приводит к приостановке работы верхнего уровня до его прибытия. В конце концов, поле заполняется состоянием 3, как показано цифрой 3, так что вычисления можно возобновить на верхнем уровне.

Далее мы формально определим серии SHA на хеджах. Является ли хедж с буквами и состояниями R НΣ Q - пробегом или простым прогоном на хедже h НΣ , записанном R run (h), определяется следующими правилами:

$$\frac{q\ a\ {}'в}{д \cdot а \cdot д'\ q\ \ run\ \ (a)} \qquad \frac{д \cdot р \cdot д'\ \ run\ \ (h)\ q'{}' \cdot P'{}' \cdot д«\ \ run\ \ (h')}{д \cdot р \cdot д'{}' \cdot P'{}' \cdot д«\ \ run\ \ (h \cdot h\ {}')}$$

$$\frac{\text{истинный}}{q\ \ run\ \ (\varepsilon)\ q \cdot \ R\ \ \cdot q} \qquad \frac{р\ в\ \ р \cdot R \cdot р\ \ run\ \ (h)\ q@р\ \ '\ \ д'{}' в}{}$$

Заметим, что если R ' run (h) run (h), то h можно получить удалением всех состояний из R, т. е. projΣ (R) = h. Любую серию R run (h) можно идентифицировать с помощью отображения префиксов вложенного слова nw(h) в состояния в Q. Заметим, что nw(h) = projΣ (nw(R)). Для любого префикса v слова nw(h) существует уникальный префикс r вложенного слова nw(R), который заканчивается некоторым состоянием q Q и удовлетворяет условию projΣ (r) = v. Затем мы называем q состоянием R в точке префикс v. Например, серия 0 · 0 · list · 1 · 0 · item · 2 · 3 · 4 на хедже list item из рисунка 5 присваивает состояние 0 вложенным префиксам слов ε и и укажите 1 для префикса вложенного слова · список и т. д.

-серия называется серией A = (Σ , Q, , I, F) , если она начинается с некоторого состояния из I. Серия A считается успешным, если оно заканчивается некоторым состоянием в F.

Лемма 4. h L(A) тогда и только тогда, когда существует успешный прогон R run (h) A.

Доказательство. Непосредственно с индукцией по h. □

4.1.3. Определение

SHA всегда можно определить, используя конструкцию подмножества, известную из детерминизация конечных автоматов на словах или древесных автоматах.

Учитывая SHA A = (Σ , Q, , I, F), набор состояний его детерминизации равен Qdet = правила перехода 2 Q. Детерминизация представлена на рисунке 6.

Machine Translated by Google

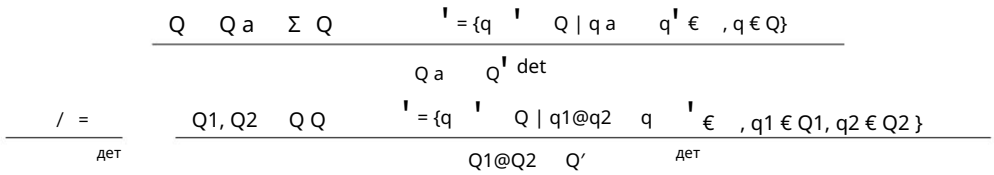


Рис. 6. Детерминизация ^{det} правил перехода SHA .

4.1.4. Тестирование членства в памяти

Проверка принадлежности h L(A) для хеджа h, график которого хранится в памяти, может быть выполнена путем вычисления уникального прогона определения A восходящим способом и слева направо, а также проверки того, является ли это успешно. В качестве альтернативы можно вычислить единственное множество P Q такое, что ^{час} I P относительно детерминизации A , и проверить , является ли P F/ = . Вычисления по сути те же, но полная история вычислений теряется при возврате только набора P достигнутых состояний, а не прогона.

Для определения SHA мы можем положиться на вариант обычной конструкции подмножества, хорошо известный из конечных автоматов на словах или стандартных древесных автоматов (см., например, [8,16]). Когда дело доходит до проверки членства, достаточно определения на лету, при котором создается только та часть детерминированного автомата, которая используется при запуске на h. Эта часть имеет линейный размер O(|h|), тогда как полная детерминизация A может иметь экспоненциальный размер O(2 |A|). Таким образом, принадлежность h L(A) можно проверить за время O(|h| |A|).

4.1.5. Доступность хеджирования.

Для любого множества P Q мы определяем набор состояний, доступных из P через хеджирование, следующим образом:

$$\text{акк}\quad (P) = \{q \;'\; ^{\text{час}}\quad Q \mid \quad q \quad P.\quad h \quad H\Sigma\ .\ q \qquad q \qquad '\ ^{\text{относительно}}\ \}.$$

Отметим, что acc (P) можно вычислить из и P за линейное время в размере . Состояние является доступным для хеджирования из P тогда и только тогда, когда оно доступно для P на графике . Или путем вычисления наименьшей фиксированной точки следующей наземной программы Datalog в зависимости от P и , которая имеет линейный размер в A:

$$\frac{q \quad P}{\text{acc}(p).} \qquad \frac{q\ a\ \ '\text{в}}{\text{соотв}(q\ 'q):\quad \text{acc}(q).} \qquad \frac{q@p \quad q\ '\text{в}}{\text{acc}(q\ '):\quad \text{акк}(q),\ \text{акк}(p).}$$

Ясно, что L(A) = тогда и только тогда, когда F acc (I) = . Следовательно, для SHA можно определить пустоту за линейное время . Это контрастирует с более общим понятием SHA , которое мы представить следующий.

4.2. Нисходящие пошаговые автоматы хеджирования

Оценка SHA осуществляется снизу вверх и слева направо, но не может передавать какую-либо информацию сверху вниз. Далее мы предлагаем расширение SHA до SHA , добавив возможность передачи информации о конечном состоянии сверху вниз.

SHA по сути эквивалентны нисходящим лесным автоматам Неймана и Зейдля [25], но переносятся из (маркированных) лесов в (немаркированные) живые изгороди. См. раздел 12 для более подробного обсуждения взаимосвязи SHA с другими понятиями автоматов и, в частности, с NWA. лар.

Определение 4. Автоматом пошагового хеджирования вниз (SHA) называется набор A = (Σ , Q, , I, F), где Σ и Q — конечные множества, I, F Q и = ((a)aεΣ , , @). Кроме того, а Q × Q и @ Q × Q, (Q × Q) × Q. SHA является детерминированным или, что то же самое, dSHA , если I содержит не более одного элемента и все отношения, , а и @ являются частичными функциями.

Machine Translated by Google

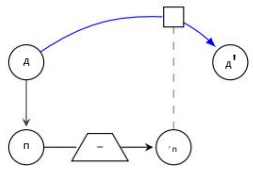
Единственное отличие от SHA — это форма правил открытия дерева. Если (q, q ')
Q, то у нас есть начальное правило дерева, которое мы обозначим ' в .
Итак, укажите q . следующим образом: q q , где начинается вычисление подхеджа, зависит от состояния q
родительский элемент.

4.2.1. Семантика

Определение шагов перехода для SHA отличается от определения для SHA только использованием правил открытия в следующем правиле вывода:

$$\frac{pq \quad h \quad p \text{ в } p \quad ' \text{ относительно } q@p' \quad д' \text{ в}}{dq \quad ' \text{ относительно}}$$

Это означает, что вычисление подрези h начинается в состоянии p , таком что q p в .



Таким образом, выбранное состояние перезапуска p теперь может зависеть от состояния q, указанного выше. Именно так информация о конечном состоянии передается сверху вниз с помощью SHA . SHA, напротив, работают исключительно снизу вверх и слева направо.

4.2.2. Бежит

Понятие запусков можно напрямую адаптировать от SHA к SHA . Находясь в состоянии q, достаточно перезапустить вычисления в субхеджах в состоянии p, так что

p . Таким образом, информация о конечном состоянии передается вниз (в то время как для SHA необходимо выбрать некоторое начальное дерево q , которое не зависит от q). Единственное правило прогонов, которое следует изменить, заключается в следующем:

$$\frac{p \text{ в } p \cdot R \cdot pq' \quad \text{run} \quad (h)\ q@p \quad ' \quad д' \text{ в}}{q \cdot \quad R \quad \cdot q' \quad \text{run} \quad (\quad h \quad)}$$

Пример выполнения dSHA на рисунке 7 показан на рисунке 8. Здесь информация об уровне узлов передается вниз. Оно представлено количеством простых чисел. Например, при открытии первого поддерева, помеченного элементом, мы используем правило перехода 1' « заявляя
0
что человек переходит из состояния 1 на первом уровне в состояние 0 на втором уровне.

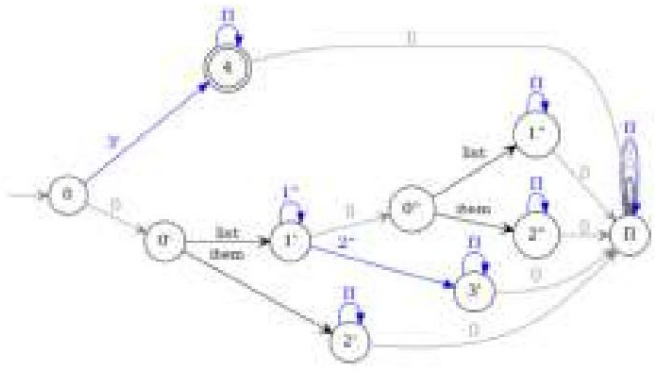


Рисунок 7. Детерминированный SHA с состоянием субхедж-проекции П, полученный с помощью безопасного проецирования без изменений .

Machine Translated by Google

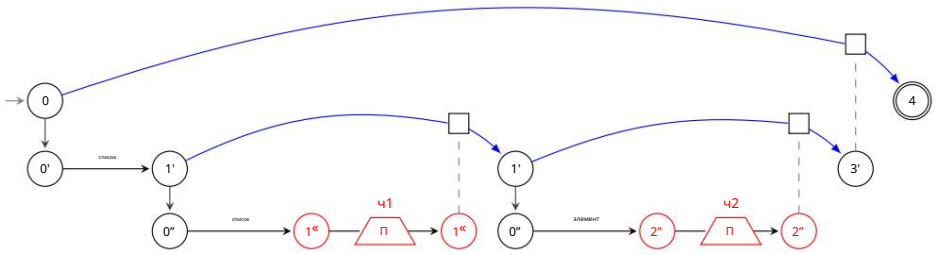


Рисунок 8. Успешное выполнение SHA на рисунке 7 на хедже list · list · h1 · item · h2 .

Видно, что все узлы поддеревьев h1 и h2 оцениваются в состояние проекции П. Таким образом, можно перепрыгивать эти поддеревья, не читая их. Это оправдано тем , что они находятся на уровне 2, информация, которую передавал оценщик этого SHA вниз.

4.2.3. Тестирование членства в памяти

Проверка членства h L(A) для SHA в памяти снова может быть выполнена путем вычисления уникального цикла определения A. Как и раньше, тестер членства может быть основан на определении «на лету». Однако процедура определения для SHA сложнее, чем для SHA , поскольку SHA может передавать информацию сверху вниз, а не только снизу вверх и слева направо. Детерминизация не только опирается на построение чистого подмножества, но также использует пары состояний в подмножествах, в основном так же, как и для вложенных словных автоматов (NWA) [14,15]. Это необходимо для работы со стеком состояний, который был рассмотрен выше. Следовательно, каждый шаг детерминизации может стоить времени O(|A|) , как указано, например, в [23]. Таким образом , верхняя граница времени проверки принадлежности для SHA s составляет O(|h| |A| ^{5}) и больше не является линейным, как в случае с SHA.

4.2.4. Связь с SHA Любой

SHA A = (Σ , Q, , I, F) может быть сопоставлен с SHA A , вниз = (Σ , Q, вниз, ЕСЛИ) сохраняя при этом прогоны и детерминизм. Единственное изменение вниз по сравнению с описывается следующим правилом:

$$\frac{p \text{ в } q \quad Q}{q \qquad p \text{ в } \text{вниз}}$$

Независимо от текущего состояния q Q, SHA Подпреграда вниз может начать оценку поддерева в любом открытом состоянии дерева p . Например, dSHA A dSHA A на вниз для рисунке 4 из введения представлен на рисунке 9. И наоборот, можно преобразовать dSHA A в нисходящей эквивалентный SHA, введя недетерминизм, чтобы исключить преобразование любой обработки. См. раздел 12.2 для получения информации о конструкции.

Machine Translated by Google

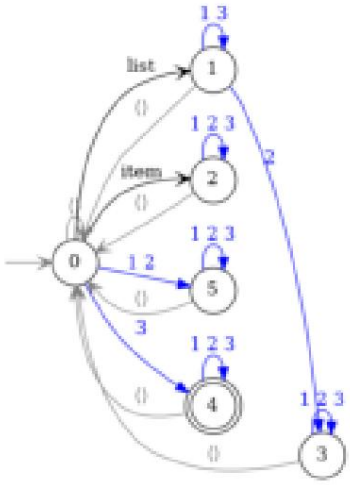


Рисунок 9. SHA A $^{\text{вниз}}$ для dSHA A для фильтра [self::list/child::item] на рисунке 4.

4.2.5. Доступность хеджирования

Доступность хеджирования может быть определена для SHA так же, как и для SHA. Однако набор acc (P) больше не может быть вычислен за линейное время размером , в отличие от SHA. Это все еще можно сделать в кубическом времени, как и для NWA [22]. Проблема в том, что доступность хеджирования для SHA больше не совпадает с понятием доступности графа автомата.

Например, в SHA на рисунке 7 имеем acc ({0}) = {4}. Обратите внимание, что 0 ' выполняет в , в принадлежит этому множеству, даже несмотря на то, что существует правило результате чего узел 0' не перехода 0 0 , доступное из узла 0 в графе автомата. Это связано с тем, что 0′ недоступен ни через какое хеджирование из 0; то есть не существует хеджа h такого, что 0 относительно . Обратите внимание, что 0 · · 0 является фактором вложенного слова некоторой серии . Это показывает, что 0', тем не менее, доступен из 0 через вложенный префикс слова. Для любого SHA A доступность хеджирования может быть вычислена с помощью следующей наземной программы Datalog кубического размера в зависимости от количества состояний A :

$$\frac{\text{п@к}' \quad q\ \text{в} \quad p \quad p'\ \text{в}}{\text{acc}(p, q) : \quad \text{acc}(p', q').} \qquad \frac{p \quad Q}{\text{acc}(p, p).} \qquad \frac{p\ \text{a} \quad q\ \text{в}}{\text{acc}(p, q).}$$

$$\frac{p, q, r \quad Q}{\text{acc}(p, q) : \quad \text{acc}(p, r), \text{acc}(r, q).}$$

Из-за кубического времени доступности хеджирования для SHA мы будем использовать только доступность хеджирования. для SHA, где это происходит в линейном времени.

4.3. Полнота схемы. Схемы

— это входные данные проблемы принадлежности, которую мы рассматриваем, а полнота — фундаментальное свойство автоматов. Чтобы правильно объединить оба аспекта, мы предлагаем понятие полноты схемы, которое мы определяем единообразно как для SHA , так и для SHA .

Набор правил - переходов SHA назовем полным, если все его отношения (a)aεΣ , и @ тотальны. Для SHA мы требуем, чтобы было непустым множеством (а не полным отношением).

SHA или SHA A называется полным, если полно и I/ = . Мы можем завершить любой SHA или SHA , добавив новое состояние приемника, а также правила перехода в состояние приемника для всех левых частей, для которых других правил перехода не существует. Состояние приемника становится начальным тогда и только тогда, когда ранее не было начального состояния. Хотя это не окончательно. Обозначим пополнение A через compl(A).

Определение 5. Частичная серия A на изгороди h — это префикс r серии compl(A) на h такой, что

- r заканчивается некоторым состоянием Q, и

Machine Translated by Google

- r не содержит состояние стока compl(A).

Частичный прогон r из A на h называется блокирующим, если не существует ни одного частичного прогона r, ' A на h, такой который является строгим префиксом r' .

Например, рассмотрим хедж h = list · item . dSHA на рисунке 4 имеет уникальную серию R = 0 · · 0 · list · 1 · · 0 · item · 2 · 3 · · 4 на h , что показано на рисунке 5.
Таким образом, все частичные серии h являются префиксами nw(R) , которые заканчиваются некоторым состоянием, т. е. 0, 0 · · 0, 0 · · 0 · list · 1 и т. д. Ни одна из этих частичных серий не является блокирующей.

Определение 6. Схемой автомата A называется хедж-язык S HΣ такой, что L(A) S.
Автомат A будем называть схемополным для схемы S , если ни один из хедж h S не имеет блокирующего частичного пробега A.

Схемы S обычно используются для ограничения типа хеджирования, которое задача может принять в качестве входных данных. Задача сопоставления шаблонов dSHA для схемы S требует двух входных данных : хеджа h S и автомата A, а не вложенного регулярного выражения e. Затем автомат A выбирает те входные хеджи h S , которые соответствуют шаблону, т. е. такие, что h L(A). По этой причине мы предполагаем, что S является схемой для A, т. е. L(A) S. Мы часто будем предполагать, что A является схемно-полной для S, так что никакой частичный прогон A на любом входном хедже h S может когда-нибудь заблокировать. Этого всегда можно добиться на основе следующей леммы 5.

Пример 5. dSHA на рисунке 4 для фильтра XPath [self::list/child::item] имеет сигнатуру Σ = {item, list}. Это полная схема для документа схемы. Чтобы это произошло, мы добавили к этому автомату состояние 5 , которое не используется ни при одном успешном запуске. Тем не менее, этот SHA не является полным. Для его завершения нам пришлось бы запустить 5 в состояния стока, добавив в него множество переходов, таких как 0@4 5 и 0@5 5, а также добавить для всех остальных состояний q {1, 2, 3, 4, 5} правила перехода список 5, q элемент 5, q@4 5 и q@5 5. q

Как мы увидим, автоматное завершение сделало бы алгоритм «безопасного без изменений» неполным для проекции субхеджирования в примере 5. Однако без полноты схемы проецирование «безопасного без изменений» может быть ненадежным: оно может упустить из виду отказ от хеджирования внутри схема. Вот почему гораздо лучше предполагать только полноту схемы, а не полную полноту входных автоматов проекции без изменений. Для конгруэнтной проекции неявно предполагается полнота схемы.

Заметим, что полнота схемы четко определена даже для нерегулярных схем. Для безопасного проецирования без изменений нам действительно не нужно ограничивать схемы регулярными, тогда как для конгруэнтного проецирования рассматриваются только регулярные схемы.

Более того, если A является схемополым для универсальной схемы S = HΣ и не имеют недоступные состояния, то A полно.

Полноту схемы детерминированных автоматов для регулярных схем всегда можно обеспечить, как мы покажем далее. Для этого мы определяем, что A совместима со схемой S , если для любого хеджа h S существует серия A в run (h). Полнота схемы подразумевает совместимость, как показывает следующая лемма. Обратное верно только в предположении детерминизма.

Лемма 5. Пусть A — детерминированный SHA со схемой S. Тогда A совместим с S тогда и только тогда, когда A является схематически полным для S.

Доказательство. Предположим, что A схемно-полна для S Wlog., можно считать S/ = . Обратите внимание, что если I = , то ε будет блокирующим частичным пробегом для любого хеджа в S \ {ε}. Поскольку S/ = , то существует q0 I. Итак, q0 является частичным пробегом для любого хеджа h S \ {ε}. Поскольку не существует блокирующих частичных прогонов по полноте схемы, этот прогон можно шаг за шагом расширить до прогона по любому h S. Следовательно, для любого хеджа h S существует некоторый прогон A, что показывает совместимость.

Machine Translated by Google

Обратно, пусть A совместимо с S и детерминировано. Мы должны показать, что A является схемно полным для S. Пусть v — частичная серия A на h S. По совместимости существует серия R run (h) , которая начинается в некотором начальном состоянии A. Согласно детерминизм, v должен быть префиксом nw(R). Таким образом, v не является блокирующим. □

Предложение 1. Любой SHA A с регулярной схемой S можно сделать схематически полной для S.

Доказательство. По лемме 5 достаточно сделать A совместимым с S. Пусть B — dSHA с S = L(B). Автомат B совместим с S. Поскольку B детерминирован, из леммы 5 следует , что B схемно полон для S. Поскольку S — схема для A такая, что L(A) L(B). Затем мы вычисляем завершение compl(A). Произведение C = B × compl(A) с конечными состояниями является схемно полным для схемы S. Кроме того, поскольку L(A) S = L(B), оно
Из СВ F = F
следует, × Ф A

что L(C) = L(B) L(A) = L(A). □

Наконец, заметим, что схему S можно распознать по тому же произведению C = B × по множеству конечных compl(A) за исключением замены набора конечных состояний F C состояний схемы × (QA {sink}). Имеем S = Ф C = Ф Б L(C[F C/FS]), где C[F C/FS] — dSHA .

получено из C заменой множества конечных состояний F C с Ф S .

5. Проекция субхеджа

Далее мы вводим понятие состояний проекции субхеджа для SHA , чтобы различать префиксы хеджей, которые не имеют отношения к субхеджированию, и оценивать SHA с помощью субхедж-проекции.

5.1. Состояния проекции субхеджа

Интуитивно понятно, что состояние проекции субхеджа может зацикливаться только до тех пор, пока хедж не закроется (а затем оно применяется). При переходе в поддерево состояние проекции субхеджа все равно может быть изменено на какое-либо другое состояние цикла. Однако при выходе из поддерева необходимо вернуться в исходное состояние проекции подизгороди. Очевидно, что любой сток представляет собой состояние субхеджевой проекции. Однако еще более интересными случаями являются состояния субхеджевой проекции, которые не являются стоками.

Мы начинаем с SHA без каких-либо ограничений, но будем навязывать полноту схемы. и детерминизм позже.

Определение 7. Состояние q Q назовем состоянием субхедж-проекции , если существует q , ' Q называемое свидетелем q, такое, что набор правил перехода , содержащий q ' или q в крайнем левом позицию, включен в следующий набор:

{q q ', д@д ' д, д ' ' q ′@q ′ q ′}

q {q ′ a ', q a q | a € }

Множество всех состояний проекции субхеджа обозначается Q shp.

Для любого полного SHA приведенный выше набор правил перехода должен быть равен набору в только все правила перехода с q или q , ' самой левой позиции. Но если SHA представляет собой полной схемы для некоторой схемы S, то не все эти переходы должны присутствовать. Также обратите внимание , что состояние субхеджевой проекции q может быть равно своему '. Поэтому любой свидетель д ' свидетелю q некоторого состояния субхеджевой проекции само по себе является состоянием субхеджевой ' .

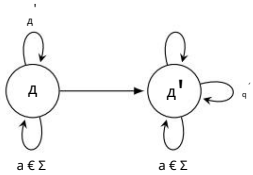
В примере dSHA A = (Σ , Q, , I, F) на рисунке 7 мы имеем Q = {П, 4, 2′ shp , проекции со свидетел

Свидетелем всех этих состояний проекции субхеджа является П. Обратите внимание, что не все возможные переходы присутствуют для состояний в Q \ {П}, учитывая, что этот автомат не является полным (но все еще является полным по схеме shp для схемы doc).

Лемма 6. Если q Q шп является состоянием проекции субхеджа и q р относительно , тогда q = p.

Machine Translated by Google

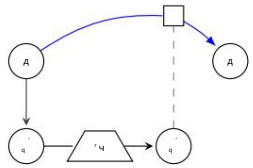
Доказательство. Предположим, что q p относительно и что q является состоянием субхедж-проекции со д . Итак, единственные возможные переходы с q или q свидетелями в левой части:



Предположим, что h = t1 · · · · · tn — последовательность деревьев или букв ti НΣ Σ , где 1 i n. Тогда существуют прогоны Ri run (ti) и прогоны R run (h) , имеющие вид h = q0 · R1 · q1 · · Rn · qn, где q0 = q и qn = p. Индукцией по i докажем для всех 0 i n , что qi = q . Для i = 0 это тривиально. Для i > 0 предположение индукции показывает, что qi 1 = q.

Случай ti = a Σ . Тогда qi 1 a qi в . Поскольку qi 1 = q является состоянием субхедж-проекции , отсюда следует, что qi = q тоже.

Случай ti = h для некоторого h НΣ . Поскольку qi 1 = q является состоянием субхедж-проекции с свидетель q , подпрогоном R , распознающим ti, это должно быть обосновано следующей диаграммой:



Итак , подпробег R h может содержать только свидетеля q и, кроме того, qi = q. □

5.2. Полнота проекции субхеджа

Прежде чем определить, когда SHA завершен для проекции субхеджирования, мы покажем, что состояния проекции субхеджирования в сериях детерминированных SHA могут возникать только в нерелевантных префиксах.

Предложение 2. Пусть A = (Σ , Q, , I, F) — dSHA со схемой S, R run (h) — пробег compl(A) на некотором хедже h S, a q — это состояние проекции субхеджа для . Тогда для любого префикса r · q из nw(R) префикс projΣ (r) из nw(h) не имеет отношения к подхеджированию для L(A) для схемы S.

Доказательство. Пусть R run (h) — единственный пробег compl(A) на хедже h S. Предположим, что r · q · s = nw(R) для некоторых состояний проекции субхеджа q из , и пусть v = projΣ (r) и w = projΣ (s). Поскольку v · w = nw(h) и h S, то v · w nw(S). Фиксируем) · w nw(S) произвольно. Пусть ˜h НΣ , такой, что некоторое хедж h nw(˜h) = v · nw(h) · ж. Для нерелевантности v для субхеджа мы должны показать, что

h L(A) ˜h L(A).

« » Мы предполагаем, что h L(A) и должны показать, что ˜h L(A). Для частичного пробега r · q из · p run (h A на ˜h должны существовать R и p такие, что q · R состояние ′). Так как q является подизгородью ,) · q · s проекции , то по лемме 6 следует , что q = p. Следовательно, r · q · nw(R прогон A на ˜h. является Поскольку R был успешным для A, s должно заканчиваться в F; таким образом, приведенный выше прогон на ˜h также успешен для A. Следовательно, ˜h L(A), как мы показали « » .

Мы предполагаем, что ˜h L(A) и должны показать, что h L(A). Пусть R˜ — единственный пробег compl(A) на ˜h. детерминизм, R˜ должен иметь префикс r · q. Итак, существуют.

Machine Translated by Google

P ', п ',s, такой, что p · R ' · q run (h ′) и r · q · R ' · p · s = R˜. Лемма 6 показывает, что p = q. Следовательно, r · q · s — единственная серия compl(A) на h , и эта серия является принимающей. Итак, h L(A).

□

Заметим, что предложение 2 не будет справедливым без предположения детерминизма. Чтобы увидеть это, мы можем добавить некоторый приемник к любому dSHA в качестве второго начального состояния. Тогда всегда можно перейти к этому стоку, который представляет собой состояние проекции субхеджа. Тем не менее, никакая приставка, идущая к раковине, может оказаться неактуальной для живой изгороди.

Предложение 2 показывает, что состояния проекции субхеджа позволяют различать префиксы , которые не имеют отношения к субхеджированию для детерминированных SHA . Интересный вопрос: можно ли найти таким образом все нерелевантные префиксы субхеджа? Тесно связанный с этим вопрос заключается в том , существуют ли для всех регулярных шаблонов dSHA , которые проецируют все нерелевантные субхеджи.

Пример 6. Пересмотрим пример 3 с Σ = {a}, регулярным шаблоном L = { a } и регулярной схемой S = L { · a}. Префикс не имеет значения для L и S, поскольку все его продолжения, оканчивающиеся в схеме, отвергаются: такое продолжение есть только одно — · a. Однако dSHA для L и S с максимальной проекцией субхеджа приведена на рисунке 17, но не переходит в состояние проекции субхеджирования в этом префиксе. Причина в том, что он перейдет в одно и то же состояние {3, 4}D0 для любого префикса в NΣ , игнорируя вложенное слово нерелевантного субхеджа. Следовательно, это состояние должно принять a . Но при чтении a в этом состоянии dSHA переходит в состояние отклонения {5}D0, показывая, что хеджирование из схемы · a S отклонено.

Пример показывает, что в конечном итоге мы рассуждаем с помощью наборов префиксов. Сначала мы поднимаем понятие нерелевантности субхеджа для наборов префиксов.

Определение 8. Пусть S HΣ — схема и L S — язык, удовлетворяющий этой схеме. Набор вложенных слов-префиксов U prefs(NΣ) называется субхеджем, релевантным для L со схемой S , если существуют вложенные слова v NΣ , такие, что для каждого префикса u · v »). В противном случае набор diff U (ты' « ты « ты называется субхеджем, нерелевантным для L относительно S.

Чтобы объяснить задачу примера 6, определим для каждого вложенного слова-префикса w prefs(NΣ) класс classL (w) подобных префиксов с заменой вложенного слова любого нерелевантного подизреза на NΣ слева направо. Для всех префиксов w prefs(NΣ), вложенных слов v NΣ , букв a Σ , подмножеств префиксов U prefs(NΣ) и l Σ ^ букв или круглых скобок применяется следующее:

$$\text{класс}L_C(ш) = \text{class_pr}\{\varepsilon\}(ш)$$

$$\text{class_prU}(\varepsilon) = \begin{cases} U \cdot N\Sigma, & \text{если U не имеет значения для L относительно S,} \\ ты & \text{иначе} \end{cases}$$

$$\text{class_prU}(w \cdot \cdot v) = \text{class_insU}'(, v) \text{ где } U' = \text{class_prU}(w)\ \text{class_prU}(a \cdot v) = \text{class_insU}'(a, v) \text{ где } U' = U \cdot a\ \text{class_prU}(v\ \text{class_insU}(l, v) = ' \cdot v) = \text{class_insU}'(, v) \text{ где } U' = \text{class_prU}(\cdot v '$$

$$\text{class_prU}(v) = \begin{cases} U \cdot N\Sigma & \text{, если U не имеет значения для L относительно S} \\ \text{class_prU} \cdot l(v) & \text{иначе} \end{cases}$$

Определение 9. Префикс u prefs(NΣ) является сильно нерелевантным для L относительно S , если множество classL (u) нерелевантно для L относительно S.

Пример 7. В нашем примере 6, где L = { a }, S = L { · a} и Σ = {a}, мы имеем classL ( ) = · NΣ , поэтому префикс не имеет особого значения для субхеджирования. Множество classL (a) = S NΣ релевантно для субхеджа, поэтому префикс a не является строго нерелевантным для субхеджа. Множество classL (· a) = NΣ · a · NΣ нерелевантно субхеджированию, поэтому префикс · a сильно нерелевантно субхеджированию.

Machine Translated by Google

Определение 10. dSHA A = (Σ , Q, , I, F) является полным для проекции субхеджа для схемы S , если A является схемно полным для S и для всех хеджей h S и всех префиксов v из nw(h) , которые сильно нерелевантны для , единственная серия A на h переходит в некоторое состояние проекции субхеджа в точке v.

Пример 8. dSHA на рисунке 17 является полным для проекции субхеджа для нашего примера 6, где L = { a } и S = L { · a}. Оно было получено методом конгруэнтной проекции. На всех префиксах нерелевантного класса субхеджа classL ( ) он переходит в состояние проекции субхеджа {0}D1. На всех префиксах класса classL (a), который не является нерелевантным для субхеджа, он переходит в состояние {3, 4}D0, которое не является состоянием проекции субхеджа. На всех префиксах нерелевантного класса субхеджа classL (· a) он переходит в состояние проекции субхеджа {5}D0.

Пример 9. dSHA на рисунке 7 завершен для проекции субхеджа с документацией схемы. Его можно получить путем безопасного проецирования без изменений. Однако в целом значения dSHA , полученные с помощью безопасного проецирования без изменений, могут быть неполными для проекции субхеджирования, как мы обсудим в разделе 6.3.

5.3. Вычислитель в памяти с проекцией субхеджирования

Далее мы покажем, как уточнить переходы для SHA с помощью проекции субхеджирования. Определим отношение перехода с помощью проекции субхеджа shp Q × Q относительно , такая, что для всех изгородей h, h ' НΣ и буквы a Σ ,

$$\frac{q \quad Q_{\text{шп}} \quad h \quad H\Sigma}{q \quad \text{shp } q \text{ относительно}} \qquad \frac{/ \quad Q \quad \text{shp } q\ a \quad qq'_{\text{в}}}{q\ a \quad \text{shp } q'_{\text{относительно}}} \qquad \frac{q/ \ \varepsilon\ Q_{\text{шп}}}{q\ \varepsilon \quad \text{shp } q \text{ относительно}}$$

$$\frac{q/ \quad Q \qquad \text{shp } q \ \text{shp } q'_{\text{относительно}} \quad q' \quad \text{шп } q \ «_{\text{относительно}}}{x{\cdot}ч_{\text{шп } qq} \ «_{\text{относительно}}}$$

$$\frac{q/_{q} \quad Q \ \text{shp } p \ p\ в \quad p \ \text{shp} \quad д@п' \quad д'_{в}}{h \ --'_{\text{относительно}}}$$

shp qq Первое правило гласит, что переходы проецирования субхеджа остаются в состояниях проекции субхеджа до тех пор, пока не будет достигнут конец текущего субхеджа. Это верно по лемме 6 при условии отсутствия блокирующих проходов. Другие правила гласят, что оценщик ведет себя без проекции субхеджа в других состояниях.

Предложение 3. Пусть A = (Σ , Q, , I, F) — SHA , схематически полная для S. Тогда для всех хеджей h S и состояний q, p Q справедливо следующее:

$$p \text{ относительно}_{\text{час}} \quad \text{тогда и только тогда, когда}_{\text{час}} \quad q \quad \text{shp } p \text{ относительно}_{\text{час}} \quad q$$

Доказательство. Мы различаем два случая следующим образом:

Случай q/ Q Тогда для любого p Q q p относительно тогда и только тогда, когда q shp p относительно по определению shp : проектирование переходов.

Случай q Q Тогда q shp q относительно . Поскольку h S и A схемно полны для S, существует shp. h существует некоторая серия A на h; таким образом, состоянием является q, такое, что q ' относительно . Лемма 6 показывает q , что q ' = q, следовательно, q q относительно .

□

Для оценки недетерминированного SHA A = (Σ , Q, , I, F) с проекцией субхеджа на некоторый изгородь h , график которого хранится в памяти, мы можем вычислить уникальное подмножество P Q, такой что I shp P относительно переходного соотношения дет детерминации

Machine Translated by Google

A и проверим, P F/ = . Достигнув при этом любого состояния проекции субхеджа P , оценщик может непосредственно перейти к концу текущего субхеджа, применив правило «пребывания» определения A:



Таким образом, он избежит посещения какого-либо субхеджа в состоянии проекции субхеджа определения A. Таким образом, время работы будет линейно зависеть от размера непроецируемой части h в предположении, что детерминизация A доступна.

Как и прежде, только необходимая часть определения A для оценки h должна производиться «на лету». В целом, это требует полиномиального времени по размеру необходимой части определения (но не обязательно линейного времени, поскольку определение SHA является более дорогостоящим, чем для SHA). Также обратите внимание, что всякий раз, когда мы обнаруживаем новое состояние P, детерминированное для A, мы должны проверить, является ли это состоянием проекции субхеджа. Для должен вычислить состояние P », так что правила P разрешены для состояний субхедж-проекции для P этого определим и проверим, только ли переход со свидетелем группы относительно .

6. Прогноз «безопасного без изменений»

Мы хотим решить обычную задачу сопоставления с образцом для живых изгородей с проекцией подизгороди. Мы предполагаем, что обычный шаблон задается dSHA , тогда как схема может быть произвольной, за исключением того, что dSHA должен быть полной схемы (см. раздел 4.3). Затем мы представляем алгоритм безопасного проецирования без изменений, который компилирует dSHA в некоторый dSHA , одновременно вводя состояния проекции субхеджирования.

Идея проекции «безопасно-без изменений» заключается в передаче информации сверху вниз, с помощью которой можно различать состояния цикла, которые больше не будут безопасно изменяться. Таким образом, создаются состояния субхедж-проекции, что мы проиллюстрируем примерами. Доказательство правильности проекции без изменений нетривиально и поучительно для доказательства обоснованности конгруэнтной проекции в разделе 7.3. Полноты прогноза субхеджирования ожидать нельзя, поскольку прогноз без изменений учитывает только простые циклы. Более общие циклы рассматривать невозможно, и неясно, как это можно сделать. Алгоритм конгруэнтного проецирования, описанный в разделе 7.2, в конечном итоге даст ответ на этот вопрос.

Мы сохраняем компилятор без изменений независимым от схемы и, следовательно, не должны предполагать ее регулярность. Но мы должны предположить полноту схемы входного dSHA , чтобы показать, что выходной dSHA распознает тот же язык внутри схемы. Обычную задачу сопоставления с образцом затем можно решить в памяти и с помощью проекции субхеджа, вычислив dSHA для входного хеджа в памяти и с помощью проекции субхеджа, как обсуждалось в разделе 5.3.

6.1. Алгоритм

Теперь мы опишем алгоритм безопасного прогнозирования без изменений. Пусть A = (Σ , Q, , I, F) — SHA со схемой S. Алгоритм принимает dSHA A в качестве входных данных и компилирует его в некоторый dSHA A snc. Мы объясним, почему A подходит для проекции субхеджа при условии , что A является полной схемы для схемы S. Нам даже не нужно предполагать, что схема S является регулярной. Доказательство занимает восемь страниц и поэтому вынесено в приложение.

Для любого множества P Q определим набор состояний, которые безопасно приводят к P:

безопасно (P) = {q Q | acc ({q}) P}.

Таким образом, состояние q безопасно для P , если любое хеджирование, прочитанное из q , на котором прогон с не блокируется, должно достичь некоторого состояния в P. Заметим, что Safe (P) может быть вычислено за линейное время в

Machine Translated by Google

размер с использованием обратной доступности хеджа из P. Мы определяем набор состояний, которые больше не будут меняться:

без изменений = {q | q Safe ({q})}.

Итак, q no-change тогда и только тогда, когда acc ({q}) {q}. Это означает, что все правила перехода, начинающиеся с q , должны зацикливаться на q. В примере автомата на рисунке 4 самоциклическими состояниями являются состояния без изменений = {2, 3, 4, 5}.

Лемма 7. Пусть A = (Σ , Q, , I, F) — SHA , схематически полная для схемы S. Для любого хеджа h S и состояния q I no-change тогда верно, что q q относительно .

Доказательство. Полнота схемы A для S, примененная к h S и q I, приводит к существованию acc (q). = q. Этот некоторого состояния q Q, такого что q q относительно . Пусть q быть любым таким состоянием. Поскольку q no-change , мы имеем q Safe ({q}), так что q Заметим, что q acc ({q}) влечет q . доказывает q q относительно □

Для любого состояния q Q и подмножества состояний Q Q определим следующее:

s-вниз (q, Q) = безопасно ({p Q | q@ p Q}) s-без изменений (q) = s-вниз (q, {q})

Состояние принадлежит s-down (q, Q) , если все состояния p acc (q) удовлетворяют условию q@ p Q. Итак, p — это состояние на нижнем уровне, которое безопасно перейдет в некоторое состояние в Q. Состояние p принадлежит s-no-change (q) , если оно безопасно не меняет q, т. е. если {q}@ acc ({p}) {q}.

Затем мы компилируем SHA A в SHA A = (Σ , Qsnc — новый символ и , , F снк). Для этого пусть П рассматриваем набор состояний следующим образом:

Qsnc = {П} (Q × 2 Q).

На практике мы ограничиваем пространство состояний теми состояниями, которые доступны, или очищаем dSHA, оставляя только те состояния, которые используются в каком-либо успешном запуске. Но в худшем случае построение действительно может быть экспоненциальным. Пример 15 можно адаптировать, чтобы продемонстрировать это.

Пара (q, P) означает, что состояние q больше не будет безопасно изменяться в текущем подхеже, если q P no-change . В этом случае подживую изгородь можно проектировать. Наборы начального и финального состояния определяются следующим образом:

= I × { } F знак равно F × { } .

Как сгенерировать правила перехода A из правил A, описано на рисунке 10. В состояниях, назначенных на верхнем уровне (q, P), множество P пусто, так что только состояния в состоянии без изменений безопасны ни для каких изменять. Вот почему при определении I snc и F вниз из используйте P = . Когда состояния (q, P) , для которого q безопасно не изменять, т. е. q P no-change , применяется идешь , то следующее правило:

q P без изменений

(q, P) П в .

Оценка на нижнем уровне переходит в дополнительное состояние П, где затем повторяется до тех пор, пока не вернется к q на верхнем уровне. При переходе из некоторого состояния (q, P), такого, что q/ ε P no-change , применяется следующее правило:

в q/ P без изменений

(q, P) (q q ′ ,s-без изменений (q)) в

Machine Translated by Google

Состояния в множестве s-no-change (q) на нижнем уровне безопасно не вызовут изменения q на верхнем уровне для любого последующего подизгорода (мы могли бы обнаружить больше нерелевантных подизделий, разрешив q изменяться, но только в уникальном Это можно получить с помощью альтернативного определения: s-no-change′ (q) = r Qs-down (q, {r}) Вычисление этого набора потребует квадратичного времени O(n |A|) .), а вычисление s-no-change (q) может осуществляться за линейное время O(|A|)).

$$\frac{q\ a \quad q' \text{ в } q/ \quad P \quad \text{без изменений}}{(q, P)\ a \quad (q', P) \text{ в } \text{CHC}} \qquad \frac{a \quad \Sigma \quad q \quad P \quad \text{без изменений}}{(q, P)\ a \quad (q, P) \text{ в } \text{CHC}}$$

$$\frac{' \text{ в } q/ \quad P \quad \text{без изменений}}{(q, P) \quad (q' \text{,s-без изменений } (q)) \text{ в } \text{CHC}} \qquad \frac{q \quad P \quad \text{без изменений}}{(q, P) \quad \Pi \text{ в } \text{CHC}}$$

$$\frac{q@p \quad q \quad ' \text{ в } q/ \quad P \quad \text{без изменений}}{(q, P)@(p,\text{s-без изменений } (q)) \quad (q \quad ', P) \text{ в } \text{CHC}} \qquad \frac{q \quad P \quad \text{без изменений}}{(q, P)@\Pi \quad (q, P) \text{ в } \text{CHC}}$$

$$\frac{a \in \Sigma}{\Pi a \quad \Pi \text{в} \quad \text{CHC}} \qquad \frac{\text{истинный}}{\Pi@\Pi \quad \Pi \text{в} \quad \text{CHC}} \qquad \frac{\text{истинный}}{\Pi \quad \Pi \text{в} \quad \text{CHC}}$$

Рисунок 10. Правила перехода SHA A CHC следует из данных SHA A.

При применении к SHA на рисунке 4 для [self::list/child::item] конструкция дает SHA на рисунке 11, который действительно равен SHA на рисунке 7 вплоть до переименования состояния. При запуске на хедже list · list · h1 · item · h2 , как показано на рисунке 8, ему не нужно посещать подхеджи h1 или h2 , поскольку все они будут достигнуты, начиная с состояния проекции П. .

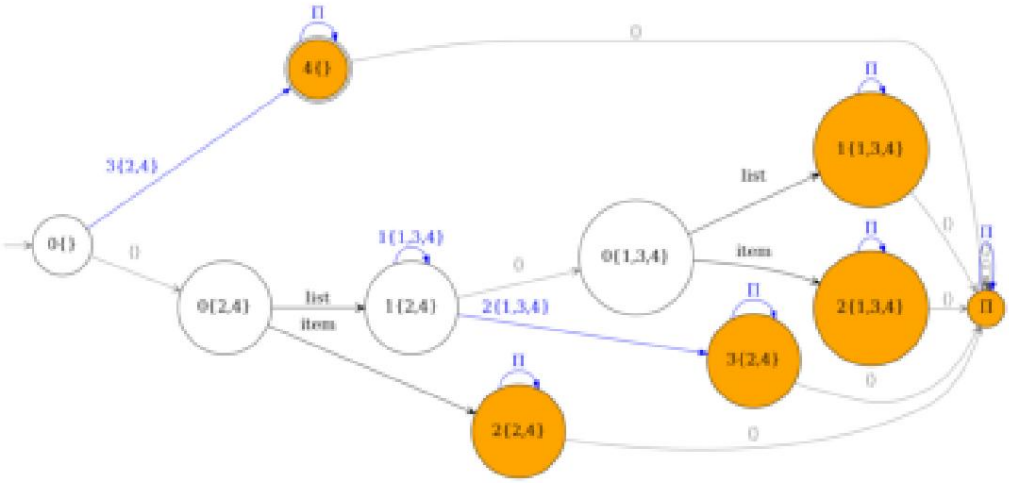


Рисунок 11. Проекция безопасности без изменений CHC построенный на основе dSHA A для запроса dSHA A [self::list/child::item] показана на рисунке 4. Состояния проекции субхеджа окрашены в оранжевый цвет. Бесполезные состояния и переходы, выходящие за пределы документа схемы , опущены. Состояние Пимеет переход, помеченный подстановочной буквой «-», которая обозначает либо список букв , либо элемент. Этот dSHA равен dSHA на рисунке 7 до переименования состояния 0 = (0, {}), 0′ = (0, 1' {2, 4}), 0′′ = (0, {1, 3). , 4}), = (1, {2, 4}), 1′′ = (1, {1, 3, 4}), 2′ = (2, {2, 4}), 2′′ = (2, {1, 3, 4}), 3′ = (3, {2, 4}) и 4 = (4, {}). Напомним, что без изменений = {2, 3, 4, 5}.

6.2. разумность

Далее мы сформулируем и докажем результат обоснованности прогноза без изменений.

Machine Translated by Google

Теорема 1 (Обоснованность проекции «безопасного отсутствия изменений»). Если SHA A является полной схемы для некоторой схемы S, то проекция без изменений для A сохраняет язык внутри этой схемы: L(A snc) S = L(A).

Проецирующий в памяти оценщик A snc будет более эффективным, чем непроецирующий оценщик A. Обратите внимание, что размер A из A. Следовательно, для CHC может быть экспоненциально больше, чем это оценки dSHA A с проекцией субхеджа на заданный хедж h мы можем предпочесть только необходимая часть A на лету. Эта часть имеет размер O(|h|) и может быть вычислена за время O(|A| |h|), что позволяет избежать CHC экспоненциального времени предварительной обработки.

6.3. Неполнота

Проекция «Безопасное отсутствие изменений» может быть неполной для проекции субхеджирования, поэтому не все префиксы, которые не имеют существенного значения для субхеджирования, сопоставляются с состояниями субхеджевой проекции. Это показано в противоположном примере dSHA A для фильтра XPath [child::item] на рисунке 12. Его проекция CHC приведен на рисунке 13. Обратите внимание, что префикс u = безопасности без изменений A item item имеет класс classL(A) (u) = item. item.NΣ , который не имеет отношения к субхеджированию, поэтому u сильно нерелевантно субхеджированию для фильтра XPath [child::item]. Тем не менее, оно приводит к состоянию (2, {1, 3, 5, 6}), которое не является состоянием субхедж-проекции, по Проблема в том, что это состояние все еще можно изменить на (4, {1, 3, 5, 6}). Это состояние каким-то образом эквивалентно фильтру, но не равно (2, {1, 3, 5, 6}).

Следует упомянуть еще одну проблему неполноты: проекция безопасного отсутствия изменений чувствительна к завершению автомата, как было отмечено ранее в примере 5. Это связано с тем, что состояние может принадлежать состоянию без изменений до завершения, но не более после добавления приемника. Тем не менее, такие состояния никогда не изменяются ни на одном дереве, удовлетворяющем схеме. Эту проблему можно применить, например, к примеру dSHA на рисунке 4 для запроса XPath [self::list/child::item]. Поэтому для безопасного проецирования без изменений было важно предполагать только полноту схемы, а не навязывать полную полноту.

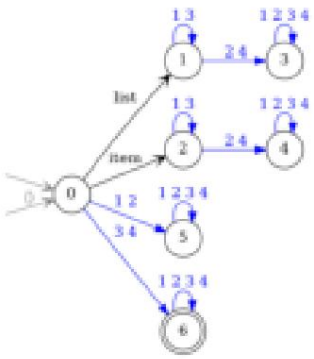


Рисунок 12. Полная схема dSHA для фильтра XPath [child::item]. Это противоположный пример полноты прогноза «безопасно без изменений», см. рисунок 13.

Machine Translated by Google

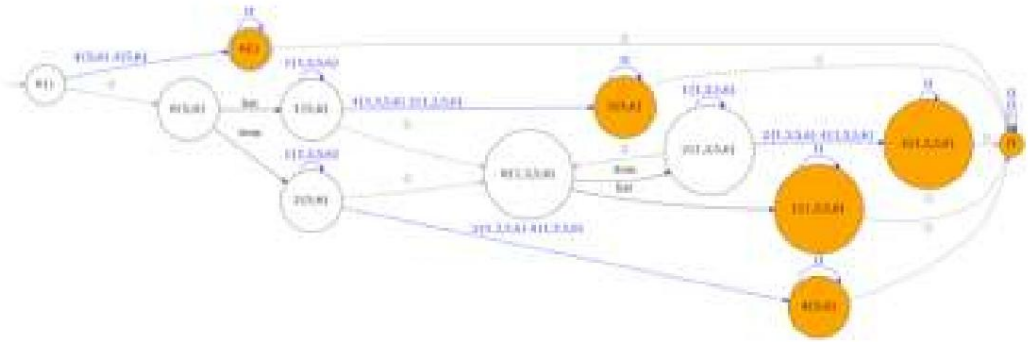


Рисунок 13. Проекция без изменений. Проекция ^{CHC} dSHA A на рисунке 12. Он неполный для субхеджирования с документом схемы в состоянии (2, {1, 3, 5, 6}): это состояние не является состоянием проекции субхеджирования, даже если префиксы list · item и item · item , ведущие к нему, не имеют значения для субхеджа.

7. Конгруэнтная проекция Мы

представляем алгоритм конгруэнтной проекции для обнаружения нерелевантных подхеджей для регулярных шаблонов хеджирования с регулярными ограничениями схемы. Мы доказываем, что конгруэнтная проекция является полной для субхедж-проекции, разрешая неполноту проекции без изменений. Для этого конгруэнтная проекция больше не может игнорировать схему, поэтому мы должны предположить, что входная схема тоже является регулярной.

7.1. Идеи

Исходная идея проекции конгруэнтности состоит в том, чтобы разрешить противоположный пример полноты проекции без изменений на рисунке 13. Здесь состояние изменяется при чтении нерелевантного подхеджа. Но изменение состояния переходит в эквивалентное состояние, поэтому это не имеет особого значения. Таким образом, идея состоит в том, чтобы определить, когда состояние всегда остается эквивалентным, а не неизменным при чтении произвольного субхеджа. В конечном итоге это делается с помощью конгруэнтной проекции на рисунке 14 , которую мы собираемся построить.

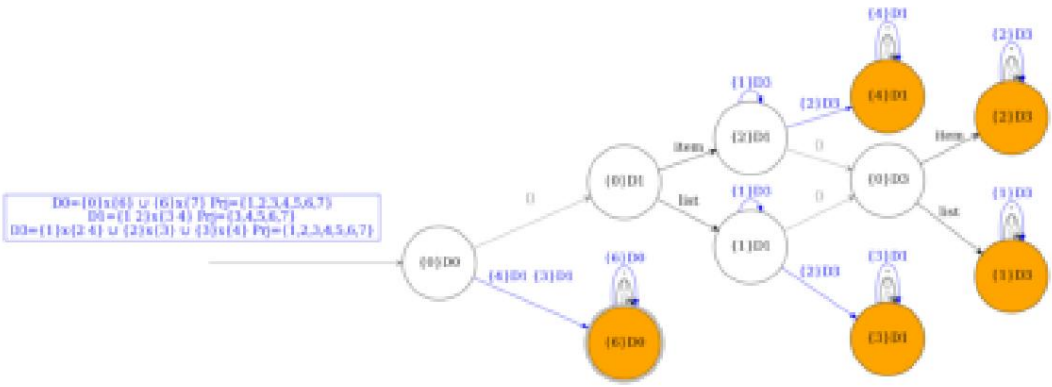


Рисунок 14. Конгруэнтная проекция dSHA A cgr(doc) для встречного примера полноты проекции без изменений, т.е. dSHA A для фильтра [child::item] на рисунке 12 с конечными состояниями схемы ФС = {0, 5, 6}. Обратите внимание, что состояние {1}D3 , которое достигается по префиксу list · item , и состояние {2}D3 , которое достигается по префиксу item · item , нерелевантны для субхеджа и, следовательно, являются состояниями проекции субхеджа.

Очевидный вопрос заключается в следующем: какое отношение эквивалентности мы выберем? Предположим, мы хотим проверить, соответствует ли хедж, удовлетворяющий регулярной схеме S , регулярному шаблону L. В ограниченном случае слов без ограничений схемы идея может заключаться в использовании сравнения сравнения Майхилла-Нероуда $\overline{L}$ и сопоставлены с штатами. Однако в общем случае ситуация усложняется, учитывая, что в общем случае diff $\overline{L}$ S может не быть

Machine Translated by Google

отношение эквивалентности. Значит, это может и не быть сравнением, как уже показано в примере 4. Кроме того, обработка вложенности живых изгородей потребует обновления рассматриваемого отношения при движении вниз по живой изгороди.

7.1.1. Отношения различий в состояниях

Алгоритм конгруэнтного проецирования будет поддерживать разностное отношение к состояниям dSHA , который вначале будет вызван разностным отношением на хеджах, diff и обновляется при каждом движении вниз по хеджированию.

Определение 11. Пусть (Σ , Q, , _, _) — dSHA. Разностное отношение для — это симметричное отношение на состояниях D Q × Q такое, что для всех h HΣ

(q q ' относительно p p ' относительно (q ', п ') € D) (q, p) € D.

Множество всех разностных отношений для обозначается D .

Подмножество Q Q назовем совместимым с разностным отношением D D , если Q 2 D = . Это означает, что никакие два состояния Q не могут быть разными относительно D.

Определение 12. Пусть (Σ , Q, , _, _) — SHA , a D — разностное отношение для . Подмножество состояний Q Q является нерелевантным для D относительно , если acc (Q) совместимо с D. Множество всех подмножеств состояний, нерелевантных для D относительно . Таким образом, выглядит следующим образом:

dPrj (D) = {Q Q | акк (Q) 2 D = }.

Состояние q Q нерелевантно для субхеджа для D , если одиночный элемент {q} нерелевантн для субхеджа для D. Множество всех нерелевантных для субхеджа состояний Q обозначается Prj (D).

Мы рассматриваем нерелевантность субхеджа для подмножеств состояний, поскольку алгоритм конгруэнтного проецирования должен устранять недетерминизм, который он вносит, с помощью своего рода детерминизации SHA . Детерминизация необходима для правильного распознавания префиксов, не относящихся к субхеджированию: все отдельные состояния в подмножестве могут быть нерелевантными для субхеджа, а все подмножество - нет.

7.1.2. Отношения наименьшей разницы

Для любого бинарного отношения R Q × Q пусть ldr (R) — отношение наименьшей разности состояний который содержит P.

Лемма 8. (p1, p2) ldr (R) (q1, q2) R. h HΣ . p1 q1 относительно p2 q2 относительно .

Доказательство. Множество {(p1, p2) Q2 | (q1, q2) R. h HΣ . p1 q2 относительно p2 очевидно, является q1 относительно разностным отношением, которое содержит R и, следовательно, содержит наименьшее такое разностное отношение ldr (R). И наоборот, каждая пара в приведенном выше множестве должна содержаться в любом разностном отношении, содержащем R, и, следовательно, в ldr (R). □

Лемма 9. Для любого R Q × Q разностное отношение ldr (R) представляет собой значение предиката D в наименее фиксированной точке программы регистрации наземных данных, созданное по следующим правилам вывода:

p, q Q / D(p, q) : R(p, q).

p1 a p2 относительно q1 a q2 / относительно D(p1, q1) : D(p2, q2).

p1@p p2 относительно q1@q q2 относительно / D(p1, q1) : D(p2, q2).

p, q Q / D(q, p) : D(p, q).

Machine Translated by Google

Доказательство. Первое правило вывода гарантирует, что R D. Три последующих правила вывода характеризуют разностные отношения D D . Второе и третье правила гласят, что различия в D распространяются назад по хеджам h. Это делается рекурсивно путем обработки буквенных изгорρдей h = a по второму правилу, а изгороди из деревьев h = h выражают симметрию разностных отношений D. по третьему правилу. Четвертые правила Таким образом, наименьшая фиксированная точка программы регистрации данных, сгенерированной вышеуказанными правилами, содержит R и является наименьшее отношение, удовлетворяющее всем аксиомам разностных отношений, то есть ldr (R). □

Предложение 4. Для любого R Q × Q наименьшее разностное отношение ldr (R) можно вычислить за время $O(|A|^2)$

Доказательство. Размер наземной программы регистрации данных, вычисляющей ldr (R) из леммы 9 , не более чем квадратичен по размеру A, поэтому ее наименьшая фиксированная точка может быть вычислена за время $O(|A|^2)$. □

7.2. Алгоритм

Теперь мы определяем алгоритм проекции конгруэнтности с помощью компилятора из dSHA и набора конечных состояний схемы в SHA , который при запуске на хедже может обнаруживать все нерелевантные префиксы субхеджа.

7.2.1. Входные и выходные

данные . В качестве входных данных алгоритму конгруэнтного проецирования предоставляется dSHA A = (Σ , Q, , I, F) для регулярного шаблона и набор FS с F FS Q. dSHA определяет регулярный язык L = L(A) , а регулярная схема S распознается dSHA A[F/FS] = (Σ , Q, , Таким образом, один и тот же базовый автомат определяет регулярный язык и регулярную схему, выбирая разные наборы конечных состояний.

Пример 10. В примере dSHA на рисунке 4 для фильтра XPath [self::list/child::item] с документом схемы мы имеем F = {4} и FS = {4, 5}. В автомате для примера счетчика [child::item] для проекции без изменений на рисунке 12 мы имеем F = {6} и FS = {5, 6}.

Заметим, что если L и S заданы независимыми SHA, то мы можем получить общий dSHA A и набор FS , как указано выше, вычислив произведение dSHA для L и S и дополнив его. Как отмечено в [16], может оказаться более эффективным определить SHA для S на первом этапе, построить произведение SHA для L с dSHA для S на втором этапе и определить это произведение на третьем этапе.

Конгруэнтная проекция A wrt. FS будет поддерживать в своей текущей конфигурации подмножество состояний Q Q и разностное отношение к состояниям в D D . Таким образом, конгруэнтная проекция dSHA всегда может определить, является ли текущий префикс нерелевантным для L по отношению к субхеджированию. схему S , проверяя, является ли текущий набор состояний Q субхедже не имеет значения для текущего разностного соотношения D.

7.2.2. Начальное разностное соотношение

Исходное разностное соотношение на состояниях D init ^{A,ФС} D индуцируется из разности отношение к префиксам diff $^{л}_{C}$ следующее:

$$D = {}^{A,ФС}_{\{(q\ инициализация}\ ,\ д'')\ |\ (v\ ,\ v'')\ diff\ {}^{л}_{S}.\ q0\ I.\ q0\ q^{hdg(v')}\ относительно\ q0^{hdg(v'')}\ q\ «_{относительно}\ \}.$$

Следующая лемма показывает, как D может быть $^{A,ФС}_{определен}$ из A и FS без ссылки на init.

языки L = L(A) и S = L(A[F/FS]).

Лемма 10. D init ^{A,FS} = ldr (F × (FS \ F)) acc (I) 2 .

Machine Translated by Google

Доказательство. Пусть для одного направления (p 'A,FS , p ") D init . Тогда существуют вложенные слова (v ', v ") diff л C и начальное состояние q0 I, такое, что q0 hdg(v p —) ' относительно и q0 , ") hdg(v « относительно . C ´v , в " вложенные слова, доступность хеджирования ' p ") acc (I) 2 следует p . Кроме того, (в ', (pv ") diff л S требует существования хеджа h HΣ , такого, что hdg(v hdg (v ") · h ') · h L и S \ L. Следовательно, существуют состояния q ' F и q , p ") « FS \ F, такой что p ' ч q ' и п « час q ». Используя лемму 8, отсюда следует, что (p , p ") ' ldr (F × (FS \ F)).

пусть (p . Используя лемму 8, , p ") ldr (F ldr (FS (F × (FS \ F)) показывает, что существует пара состояний p FS \ F, и , p ") acc (I) 2 имущество (п ' F, q « h HΣ , такой, что p ' ч q ' относительно и p « ч q « относительно . Из (п. ' , следует что существуют вложенные слова v ', в " NΣ и начальное состояние q0 I такое, что q0 hdg(v ----) ´п относительно и q0 ') ---- hdg(v p « относительно . Следовательно, (v ', v ") diff л C , так что (q ', q ") D A,ФС инициализировать □

Как следствие леммы 10 и предложения 4 исходное разностное отношение D A,FS) из A и FS. 2 может быть вычислено за время O(|A|

Предложение 5 (правомерность исходного разностного соотношения). Пусть A = (Σ , Q, , I, F) — полный dSHA, F FS Q, L = L(A) и S = L(A[F/FS]). Для любого хеджа h HΣ и состояния q относительно для некоторого q0 I вложенное слово nw(h) субхеджированию для час нерелевантно субхеджированию для q Q, так что q0 L и S тогда и только тогда, когда q нерелевантно DA ,ФС относительно .

Доказательство. Мы показываем, что nw(h) релевантно для L и S тогда и только тогда, когда q является субхеджем. актуально для D относительно. A,FS . инициализировать

« » Пусть nw(h) — субхедж, релевантный для L и S. Тогда существуют хеджирования h . час « HΣ и вложенное слово w NΣ такое, что

$$nw(ч) \cdot nw(ч\ ') \cdot w \quad L \quad nw(h) \cdot nw(h\ ») \cdot w \quad S \setminus L.$$

Поскольку A детерминирован и q0 q относительно для единственного q0 I, отсюда следует, что существуют состояния p Q, q ', пF« и q ' « FS \ F, такой, что

$$q \ ^{час}\ ' \ p \ ' \ \frac{hdg(w)}{---} \ q \ ' \text{ относительно } q \ ^{час}\ p \ « \ \frac{hdg(w)}{---} \ q \ « \text{ относительно .}$$

Из q ' F и q « FS \ F, то (q hdg(w) q ', q ") D A,FS A,FS инициализация . Поскольку D — это разница отношение, п ' hdg(w) q ' относительно и p « « относительно это означает, что (p ', p ") D A,ФС инициализировать слишком. Следовательно, (п 'A,FS , p ") acc ({q}) 2 D init , т. е. q, имеет отношение к D A, FS относительно. .

« » Пусть q — субхедж, релевантный для D относительно. A,FS . Тогда существует пара (p A,FS D ', p ") акк ({q}) 2 инициализировать . Итак, есть живые изгороди h ' и ч », такой, что

$$h\ ´ \ pq \ ' \ « h \text{ относительно } q \ p \ « \text{ относительно .}$$

Поскольку D A,ФС инициализирует = ldr (F × (FS \ F)) по лемме 10 существует вложенное слово w и a) принадлежит F × (FS \ F) и пара (q ', q ") Q2 , так что либо (q ', q ") или (q «, q '

$$п \ ' \ \frac{hdg(w)}{---} \ q \ ' \text{ относительно } p \ « \ \frac{hdg(w)}{---} \ q \ « \text{ относительно .}$$

Следовательно, nw(h) · nw(h) ') · w в L и nw(h) · nw(h ») · w S \ L или наоборот. Этот показывает, что nw(h) применимо для L и S.

□

Machine Translated by Google

7.2.3. Обновление отношения разницы

Для любого разностного отношения D D и подмножества состояний Q Q определим бинарное отношение down Q(D) Q × Q такое, что для всех состояний p1, p2 Q:

(p1, p2) down Q(D) тогда и только тогда, когда q1, q2 Q. (q1@ p1, q2@ p2) D.

Для любого подмножества состояний Q Q и разностного отношения D D пусть DQ D — наименьшее разностное отношение, содержащее down Q(D):

DQ = ldr (вниз Q(D)).

Лемма 11. Разностное отношение DQ можно вычислить за время O(|A| ^{2}) из Q, и D.

Доказательство. Бинарное отношение down Q(D) можно вычислить за время O(| ^{2}) из Q, D и . |. Отношение наименьшей разности ldr (down Q(D)) можно вычислить по наземным данным за время) O(|A| ^{2} от down Q(D) , используя Предложение 4. □

Пример 11. Пересмотрите dSHA для фильтра XPath [self::list/child::item] на рисунке 4 с набором конечных состояний схемы FS = {4, 5}. Поскольку F = {4}, имеем FS \ F = {5}. A ,FS , которое является симметричным равно D0 = D {(1, 4), (4, 5)}. Состояниями, не замыканием исходного разностного отношения ({0} × {1, 4, 5}) , имеющими отношения к подхеджированию, являются Prj (D0) = {1, 2, 3, 4, 5}, поскольку только состояние 0 может получить доступ к двум состояниям в разностном отношении D0, конечному состоянию в F = {4} и нефинальному состоянию. что является финалом схемы в FS \ F = {5} некоторых хеджей. Разностное ^) является инициализация ФС отношение down {0} (D — симметричное замыкание {1, 2} × {3}, равное разностному отношению D1 = D0{0} = ldr (down {0} (D0)) Следовательно , проекционными состояниями являются Prj (D1) = {2, 3, 4, 5}, поскольку из состояний 0 и 1 еще можно достичь как 1 , так и 3 , пока (1, 3) D1. Разностное соотношение D2 =. down{1} (D1) — симметричное замыкание {(1, 2), (2, 3)}. Следовательно, состояния проекции — это Prj (D2) = {1, 2, 3, 4, 5}. . Из состояния 1 можно достичь только состояний 1 и 3, которые не отличаются для D2.

Состояния проекции исходного разностного отношения содержат все состояния, безопасные для выбор или безопасное отклонение относительно схемы:

Лемма 12. Все состояния в Safe (F) и Safe (FS \ F) нерелевантны для DA,FS. инициализировать

Мы опускаем доказательство, так как результат в дальнейшем не используется.

Учитывая dSHA A = (Σ , Q, , I, F) и множество F FS Q, мы теперь построим конгруэнтную проекцию A cgr(S) как следующую dSHA :

A cgr(S) = (Σ , Qcgr , кгр, Я cgr(S) , F cgr(S)),

где набор состояний, начальных состояний и конечных состояний следующий:

Qcgr 2 Q × D

Я cgr(S) = {(Я, Д)} A,ФС

F cgr(S) знак равно (Q, D) A,ФС Q/ ε dPrj (D A,FS) Q F/ = Q dPrj (D A,ФС) acc (Q) F/ =

Правила перехода в cgr представлены правилами вывода на рисунке 15. Чтобы проиллюстрировать конструкцию и понять, почему необходима детерминизация, мы вновь рассмотрим пример 3, где ограничения схемы имеют последствия, которые на первый взгляд могут быть нелогичными.

Machine Translated by Google

$$\frac{a \in \Sigma \quad Q \in dPrj \quad (D)}{(Q, D)\ a \quad (Q, D)\ в \quad cgr} \qquad \frac{Q \quad dPrj \quad (D)}{(Q, D) \quad (Q, D)\ в \quad cgr}$$

$$\frac{Q@P \quad Q'\ в \quad ^{дет} \quad Q \quad dPrj \quad (D)}{(Q, D)@(Q, D) \quad (Q, D)\ в \quad cgr}$$

$$\frac{Q\ a \quad Q'\ в \quad ^{дет} \quad Q/ \quad dPrj \quad (D)}{(Q, D)\ a \quad (Q', D)\ в \quad cgr} \qquad \frac{Q/ \quad dPrj \quad (D)}{(K, Д) \quad (\ , DQ)\ в \quad cgr}$$

$$\frac{Q@acc \quad (P) \quad Q'\ в \quad ^{дет} \quad Q/ \quad dPrj \quad (D)\ P \quad dPrj \quad (DQ)}{(Q, D)@(P, DQ) \quad (Q', D)\ в \quad cgr}$$

$$\frac{Q@P \quad Q'\ в \quad ^{дет} \quad Q/ \quad \varepsilon\ dPrj \quad (D)\ P/ \quad dPrj \quad (DQ)}{(Q, D)@(P, DQ) \quad (Q', D)\ в \quad cgr}$$

Рис. 15. Правила переходов cgr конгруэнтной проекции A cgr(S) .

Пример 12 (Необходима детерминация при конгруэнтном прогнозировании). Повторно рассмотрим пример 3 с сигнатурой Σ = {a}, образцом L = { a } и схемой S = L { · a}. Этот язык может быть определен dSHA A на рисунке 16. Схема S может быть определена тем же автоматом, но с конечными состояниями схемы FS = {3, 5} и F = {3}. dSHA A cgr(S) , полученный с помощью конгруэнтной проекции , показан на рисунке 17. Уникальное хедж а языка L = L(A) принимается в состоянии ({3, 4}, D0), где D0= { (3, 5), (5, 3)}. Единственное хеджирование · a в S \ L отвергается: после прочтения дерева автомат переходит в состояние ({3, 4}, D0) , где он переходит в выступающий сток ({5}, D0) , когда читая последующую букву a. Заметим, что любое хедж в Σ принимается в состоянии ({3, 4}, D0) A cgr(S), даже если только один хедж а принадлежит L. Это правильно, учитывая , что хедж в (ε + a · a · a) в любом случае не принадлежат схеме S. Мы замечаем, что ({3, 4}, D0) не может быть состоянием проекции, поскольку бег по хеджированию · a должен продолжаться до стока ({5}, D0); следовательно, состояние ({3, 4}, D0) актуально для субхеджа.

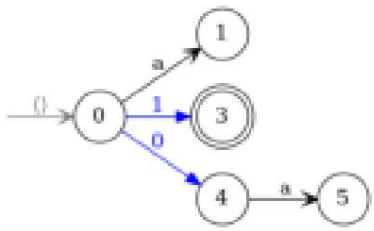


Рис. 16. dSHA A для L(A) = { a } и конечные состояния схемы FS = {3, 5} для схемы S = L(A) { · a}.

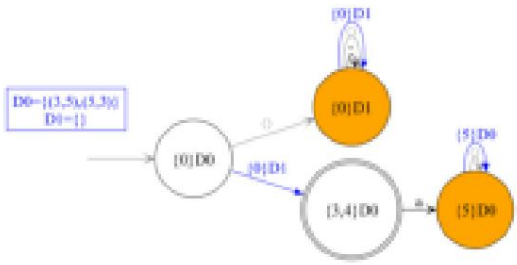


Рисунок 17. Конгруэнтная проекция A cgr(S) для dSHA A и конечных состояний схемы FS на рисунке 16.

Machine Translated by Google

Обратите внимание, что оба отдельных состояния 3 и 4 не имеют значения для D0, в то время как подмножество оба состояния {3, 4} релевантны субхеджированию для D0, поскольку (3, 5) acc ({3, 4}) 2 с D0. Это показывает, что детерминированная конструкция действительно необходима для принятия решения о нерелевантности субхеджирования для таких случаев, как в этом примере. Также обратите внимание, что состояние 0 не имеет отношения к субхеджированию зависит от h. Напротив, принятие хеджей формы ' для D1= . Это нормально, поскольку by A cgr(S) действительно не h · h зависит от h, так что подмножество состояний {3, 4} не может быть нерелевантным для D0.

Наконец, давайте обсудим, как выводятся правила перехода dSHA A cgr(S) . Наиболее интересное правило перехода — ({0}, D0)@({0}, D1) ({3, 4}, D0) в cgr. Это производится по следующему правилу вывода, где Q = P = {0}, acc (P) = {0, 1, 3, 4, 5}, Q ≠ {3, 4}, D = D0 и DQ. = Д1:

$$\frac{Q@acc(P) \quad Q \xrightarrow{'}_{det} в \quad Q/dPrj(D)\,P \quad dPrj(DQ)}{(Q, D)@(P, DQ) \quad (Q \xrightarrow{'}, D) в \quad cgr}$$

Это правило гласит, что если P является субхеджем, нерелевантным для DQ, то все состояния, доступные из P, должны быть опробованы, поскольку до всех них можно добраться с помощью какого-то другого субхеджа, который был спроецирован. В результате из-за субхедж-проекции невозможно узнать, в какое состояние Q следует перейти. Таким образом, чтобы оставаться детерминированными, мы переходим во все возможные '. В примере эти состояния, т.е. в подмножество Q входят все состояния, которые могут быть достигнуты при чтении некоторого хеджа в шаблоне { h | h HΣ }, учитывая , что подрези h Σ не проверяются с помощью проекции подрезей.

Пример 13 (фильтр XPath [self::list/child::item] с документом схемы). Конгруэнтная проекция примера dSHA на рисунке 4 представлена на рисунке 18. Этот dSHA аналогичен проекции «безопасно без изменений» на рисунке 7, за исключением того, что бесполезное состояние 2 удалено, и теперь четыре состояния проекции зацикливаются сами по себе. вместо перехода в состояние общего цикла Π. Также следует отметить, что там используются только одноэлементные наборы состояний, поэтому детерминизация не требуется. Как мы увидим, это типично для экспериментов с обычными запросами XPath, где большие подмножества встречаются редко.

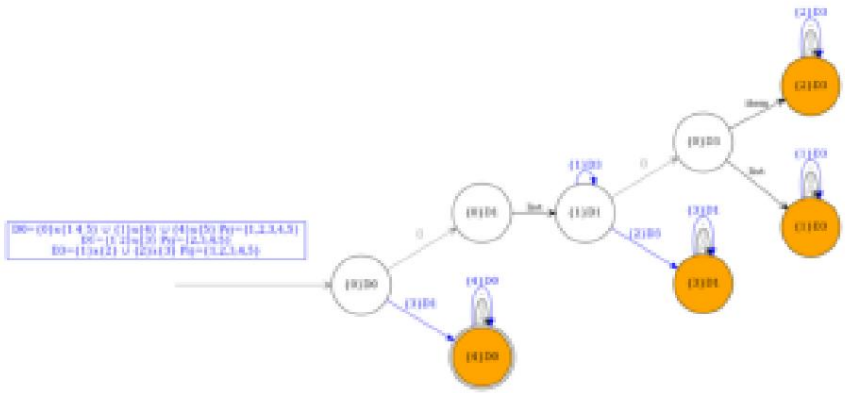


Рисунок 18. Конгруэнтная проекция A cgr(doc) dSHA A на рисунке 4 для фильтра XPath [self::list/child::item] со схемой doc. Нерелевантные состояния субхеджа окрашены в оранжевый цвет. Подчеркивание обозначает любую метку, список или элемент.

Пример 14 (Контрпример для полноты прогноза без изменений). Еще раз рассмотрим пример, противоречащий полноте проекции «безопасно без изменений», то есть dSHA на рисунке 12 для запроса XPath [child::item] с конечными состояниями схемы FS = {0, 5, 6}. Его конгруэнтная проекция показана на рисунке 14. Отметим, что префикс item. item приводит к состоянию {2}D3, которое является состоянием проекции субхеджа, поскольку 2 не имеет отношения к субхеджированию для D3. В частности, обратите внимание, что acc ({2}) = {2, 4} , поэтому никакие два состояния, доступные из 2 , не различаются в D3. Это означает, что состояние 2 все еще можно изменить на 4 , но тогда оно не станет другим по отношению к D3. Это решает проблему неполноты прогноза без изменений в этом примере.

Machine Translated by Google

Первое свойство состояний (Q, D) , заданное конгруэнтной проекцией, заключается в том, что Q совместимо с D. Это означает, что никакие два состояния в Q не отличаются по отношению к D. Интуитивно понятно, что каждое состояние Q так же хорошо, как и другое, за исключением выхода за пределы схемы. Итак, если (Q, D) присвоено префиксу, а суффикс ведет из какого-то состояния в Q в F, то он не может вести в FS \ F из какого-то другого состояния в Q.

Лемма 13. Если частичный запуск A cgr(S) присваивает префиксу состояние (Q, D) , то Q совместим с D.

Мы опускаем доказательство, так как этот поучительный результат не будет непосредственно использоваться в дальнейшем. Тем не менее, совместимость будет играть важную роль в доказательстве надежности.

7.3. разумность

Затем мы адаптируем результат и доказательство надежности от проекции «безопасно-без изменений» к проекции конгруэнтности.

Теорема 2 (правомерность конгруэнтного проектирования). Для любого dSHA A = (Σ , Q, , I, F) и множества F FS Q конгруэнтная проекция A относительно FS сохраняет язык A внутри схемы S = L(A[F/FS]), т.е.

L(A cgr(S)) S = L(A)

.

Заметим, что S является схемой для A, поскольку F FS. Более того, A является полной схемы для S, поскольку A детерминирован и S = L(A[F/FS]).

Доказательство. Доказательство включения L(A) L(A cgr(S)) S будет основано на следующем две претензии:

Утверждение 2.1a. Для всех h € HΣ , D € D и Q € dPrj (D), (Q, D) (Q, D) относительно cgr .

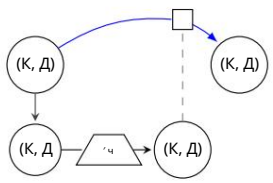
Доказательство. Утверждение 2.1a докажем индукцией по структуре h. Предположим, что Q dPrj (D).

Случай h = h ' . Гипотеза индукции, примененная к h ' дает (Q, D) (Q, D) ' относительно cgr .

Таким образом, мы можем применить следующие два правила вывода:

$$\frac{Q \quad dPrj \quad (D)}{(Q, D) \quad (Q, D) \text{ в } cgr} \qquad \frac{Q \quad dPrj \quad (D)}{(Q, D))@(Q, D) \quad (Q, D) \text{ в } cgr}$$

чтобы замкнуть следующую диаграмму относительно cgr:



Это доказывает (Q, D) (Q, D) относительно cgr, как того требует утверждение.

Случай h = a. Поскольку q dPrj (D), мы можем применить правило вывода:

$$\frac{a \in \Sigma \quad Q \in dPrj \quad (D)}{(Q, D)\ a \quad (Q, D) \text{ в } cgr}$$

Это доказывает данный случай иска.

Machine Translated by Google

Случай h = ε. Тривиально имеем (Q, D) ε (Q, D) относительно cgr.
Случай h = h · час «. По предположению индукции, примененному к h и h », получаем следующее:

(Q, D) ′ч (Q, D) и (Q, D) cgr « ч (Q, D) относительно cgr. Следовательно, (Q, D) ′ч « ·ч (Q, D) относительно .

На этом доказательство утверждения 2.1a заканчивается. Следующее утверждение является ключом к доказательству обоснованности. □

Для любого разностного отношения D определим бинарное отношение depa (D) 2 Q × 2 Q такое что для любых двух подмножеств состояний Q , вопрос « Q,

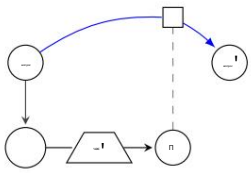
(B , Q ") депа (D) вопрос / dPrj (D) Q Q « (1
вопрос dPrj (D) (Q акк (Q ") Q « a) dPrj (D)) (2 a)

Кроме того, заметим, что QQ ") « dPrj (D) Q acc (Q ") влечет Q dPrj (D), и, следовательно, (B , depa (D).

Утверждение 2.2a. Пусть h HΣ – хедж, Q, Q Q подмножеств состояний и D D разность
ляция. Если вопрос час относительно det, тогда существует Q Q, такой, что (Q, D) (Q час « , D) относительно cgr и (Q , QQ ") depa (D).

Доказательство. Если Q dPrj (D), то (Q, D) час (Q, D) относительно cgr по п. 2.1a. Пусть Q « = B. Мы имеют Q (2a) acc (Q ") и Q тогда « dPrj (D), и, таким образом, Q dPrj (D), так что (1a) и из (QQ ") depa (Д). Итак, достаточно доказать утверждение в предположении, что Q/ ε dPrj (D). Доказательство проводится индукцией по структуре h.

Случай h = h ч . Предположение Q Q относительно det показывает, что существует подмножество состояний P Q, замыкая следующую диаграмму:



В частности, из Q@P Q можно сделать в дет. Пусть D′ = DQ. Поскольку Q/ ε dPrj (D), мы следующий вывод:

$$\frac{Q/\ \mathrm{dPrj}\ (D)}{(K, Д)\quad (\ , D') \text{ в } \mathrm{cgr}}$$

Мы должны различать два случая в зависимости от того, принадлежит ли dPrj (D′) или нет.

Подслучай П / ε dPrj (D′). Условие индукции применимо к h Q, такому что (P, P) depa (D′) и дает существование

(, Д) ′ч (П , Д) относительно cgr.

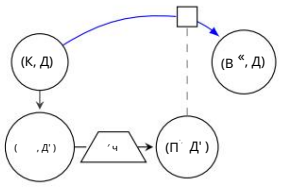
Подподслучай P/ ε dPrj (D′). Поскольку (P, P/) depa (D′), то P P в dPrj (D′). Пусть Q. « det. Мы Вопрос « Q таков, что Q@P .

Следовательно, P может затем применить следующее правило вывода:

$$\frac{K@П \quad \text{Вопрос} \ll \text{det в} \quad Q/\ \mathrm{dPrj}\ (D)\ P \quad / \varepsilon\ \mathrm{dPrj}\ (D')}{(Q, D)@(P \quad , D') \quad (Q \ll , D) \text{ в } \mathrm{cgr}}$$

Machine Translated by Google

Следовательно, мы можем замкнуть диаграмму следующим образом:

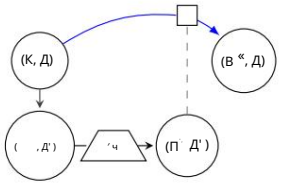


Это показывает, что (Q, D) (Q «, D) относительно cgr. Поскольку P ', K@P K ', и Q@P Вопрос « P в det, отсюда следует, что Q Q », и, следовательно, (Q , Q ") депа (D). Здесь показана претензия в данном случае.

Подподслучай P dPrj (D′). Поскольку (P, P) depa (D′), то P dPrj (D′). acc (P ′) и правила Следовательно, мы можем применить следующий вывод P для некоторого Q « НЭ :

$$\frac{Q@acc\ (P')\quad Q\ «\ det\ в\quad Q/\quad dPrj\ (D)\ P'\quad dPrj\ (D')}{(Q, D)@(P\ D')\quad (Q\ «, D)\ в\quad cgr}$$

Следовательно, мы можем замкнуть диаграмму следующим образом:

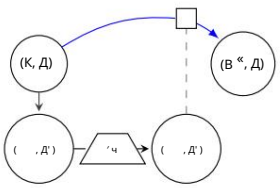


Это показывает, что (Q, D) (Q и «, D) относительно cgr. Поскольку P acc (P ′), отсюда следует QQ ") depa (D), Q@acc (P ′) Q из Q@P « что Q в определим , Q ". Таким образом, (B , поэтому утверждение справедливо и в этом случае.

Подслучай Prj (D′). Утверждение 2.1a показывает, что (, D′) ' (, D′) относительно cgr. « Пусть таково, что Q@acc ( ) Q′′ в det. Мы можем применить следующий вывод: Правило вопроса:

$$\frac{Q@acc\ (\quad)\quad Q\ «\ det\ в\quad Q/\ \varepsilon\ dPrj\ (D)\quad Prj\ (D')}{(Q, D)@(\quad, D')\quad (Q\quad «, D)\ в\quad cgr}$$

Таким образом, мы можем замкнуть приведенную ниже диаграмму относительно cgr следующим образом:



Это показывает, что (Q, D) (Q «, D) относительно cgr. Поскольку P acc ( ), отсюда следует , что Q Q ", и, таким образом, Q (QQ ") depa (D). и Q@acc ( ) Q из Q@P « в ,

Случай h = a. Поскольку Q/ dPrj (D), мы можем применить правило вывода:

$$\frac{Q\ a\quad Q'\ det\ в\quad Q/\quad dPrj\ (D)}{(Q, D)\ a\quad (Q\ , D)\ в\quad cgr}$$

Следовательно, (Q, D) ', D) относительно cgr. С Q « = K ', утверждение следует, поскольку (Q , Q ") (Q depa (D).

Случай h = ε. В этом случае имеем Q = Q. « и (Q, D) ε (Q, D) относительно cgr, поэтому утверждение

Machine Translated by Google

Случай h = h1 · h2. Поскольку Q K h1 относительно det, существует Q1 Q такой, что Q Q1 и относительно h2 Q1 det. По предположению индукции, примененному к h1, существует Q 1 K, Q такой, что (Q, D) (Q 1 , D) относительно cgr и (Q1, Q1) депа (D).

Подслучай Q1 dPrj (D). Поскольку (Q1, Q 1) depa (D), то Q1 acc (Q ′ 1) и вопрос1 dPrj (D). Кроме того, п. 2.1a показывает, что (Q 1 , Д) (Q 1 , Д), и, следовательно, (Q, D) 1 , D) относительно cgr. Поскольку Q acc (Q1) и Q1 acc (Q ′ 1), отсюда следует, что Q (Q acc (Q ′ 1). Следовательно, (Q Вопрос 1 ,) depa (D) и (Q, D) (B 1 , Д).

Подслучай Q1 / ε dPrj (D). В этом случае мы можем применить гипотезу индукции к Q1 Q относительно det, показывая, что существует Q « Q, такой что (Q 1 , Д) h2 (Q « , D) относительно cgr и , Q ") депа (D). Следовательно, (Q, D) (Q « , D) относительно cgr и (Q , (QQ ") depa (D).

На этом доказательство утверждения 2.2a заканчивается. □

Доказательство включения L(A) L(A cgr(S)) S. Поскольку S является схемой для A, имеем L(A) S, поэтому достаточно доказать, что L(A) L(A cgr(S)). Пусть h L(A). Тогда существуют q0 I и q относительно . Пусть Q = {q}. Поскольку A детерминирован, отсюда следует, что I q F, подмножества состояний Q Такой, что q0 A,FS Q относительно det. Используя (2a) из Лемма 2.2a, отсюда следует существование (K, K A,FS) depa (D) и (I, D) (Q ,) I cgr(S) . Таким образом , чтобы доказать, что h L(A cgr(S)), достаточно показать, что (Q , A,ФС) F cgr(S) . Мы различает два случая:

Случай Q / ε dPrj (D A, ФС). Условие (1a) из (Q, Q) depa (D), показывает, что Q так что q Q . Поскольку q F, то Q F/ = . Таким образом, (Q , A,ФС Q) F cgr(S), так что h L(Acgr(S)).

Случай Q dPrj (D A, ФС). Условие (2a) (Q, Q A,FS D init) depa (D) дает Q acc (Q ′). Следовательно, q acc (Q ′) F так, что (Q ,) F cgr(S) , и, следовательно, h L(A cgr(S)).

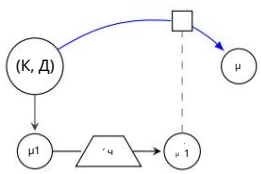
На этом доказательство первого включения заканчивается.

Далее мы хотим показать обратное включение L(A cgr(S)) S L(A). В конечном итоге это будет следует из следующих двух утверждений.

Утверждение 2.1б. Для любого хеджа h HΣ , разностного отношения D D , проекционного состояния Q dPrj (D) и состояния μ Qcgr: если (Q, D) μ относительно cgr, то μ = (Q, D) .

Доказательство. Индукцией по структуре h HΣ предположим, что (Q, D) μ относительно cgr:

Случай h = h . Должны существовать состояния μ1, μ 1 Qcgr , замыкая следующую диаграмму:



Поскольку Q dPrj (D), для вывода (Q, D) μ1 относительно cgr необходимо применить следующее правило :

$$\frac{Q \quad dPrj \quad (D)}{(Q, D) \quad (Q, D) \text{ в } cgr}$$

Machine Translated by Google

Следовательно, μ1 = (Q, D). Предположение индукции применимо и к (Q, D) относительно cgr μ 1 = (Q, D) также. Таким образом, μ должно было быть получено путем применения правила: показывает, что μ

$$\frac{Q \quad dPrj \quad (D)}{(Q, D)@(Q, D) \quad (Q, D) \text{ в } \quad cgr}$$

Следовательно, μ = (Q, D), что и требовалось.

Случай h = a. Необходимо применять следующее правило:

$$\frac{a \in \Sigma \quad Q \in dPrj \quad (D)}{(Q, D)\ a \quad (Q, D) \text{ в } \quad cgr}$$

Следовательно, μ = (Q, D).

Случай h = ε. Очевидный.

h = h1 · h2. Должен существовать такой μ1, что (Q, D) μ относительно cgr. Используя μ1 предположение индукции, примененное к h1, h1 h2 Случай

имеем μ1 = (Q, D). Таким образом, мы можем применить гипотезу индукции к h2 и получить μ2 = (Q, D).

На этом доказательство утверждения 2.1б заканчивается. □

Далее нам понадобится обращение к утверждению 2.2a. Чтобы связать серии A cgr обратно с сериями A, мы определяем для любого разностного отношения D другое бинарное отношение depb (D) 2 Q × 2 Q, такое, что для любых двух подмножеств состояний Q Qвопрос «

$$(B \ , \ Q \text{ "}) \quad depb\ (D) \qquad \begin{matrix} (Q \times Q \text{ »}) \quad D = \quad / \quad dPrj \quad (D) \qquad (0\ ^{б}) \\ Q \quad Q \quad (1 \quad dPrj \quad (D) \quad Q \quad acc \quad (Q \text{ ′ }) \ (2\ ^{б}) \\ \text{«} \qquad\qquad\qquad\

Machine Translated by Google

Поскольку Q/ dPrj (D), для вывода (Q, D) μ1 относительно cgr, где D′ = DQ, было применено следующее правило вывода :

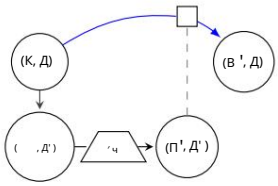
$$\frac{Q/\ dPrj\ (D)}{(K, Д)\quad (\ , D') \text{ в } cgr}$$

Следовательно, μ1 = (, D′). Если / ε dPrj (D), то предположение индукции, примененное к (, D′) ч μ относительно cgr показывает, что существует (P , P ″) depb (D′), такой, что 1 μ , , D′) и = (P ч П« относительно дет. В противном случае можно сделать тот же вывод, что и мы как говорилось в начале.

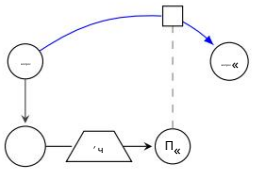
Подслучай P dPrj (D′). Правило перехода (Q, D)@μ1 μ должно быть выведено следующим образом для Q Вопрос:

$$\frac{Q@acc\ (P')\ Q\ \overset{\text{det в}}{\ }\ Q/\ dPrj\ (D)\ P\quad dPrj\ (D)}{(Q, D)@(P\ , D')\ (Q\ , D) \text{ в } cgr}$$

Это показывает, что μ = (Q , Д). Итак, мы имеем следующую схему:



Пусть « — уникальное подмножество состояний, такое что Q@P« Вопрос « относительно дет. Мы можем тогда Q закроет следующую диаграмму:

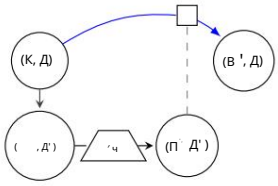


, т.е. , P ″) depb (D′), то (P From (PD = P ") D′ = , и, следовательно, (QQ ") × Q ") × условие (0b) из (Q tion (2b) из (PP ″) , depb (D). Поскольку P dPrj (D′), condi- acc (P ′). depb (D′) дает P и Q @acc (P ′) Q (2b) « Более того, Q@P в det, что дает Q , « Вопрос « из (QQ ″) depb (D) . так что условия (1b) и QQ ") depb (D). Подводя итог, мы Отсюда следует (Q h). покажем, что μ = (Q по , D), Q Q « относительно det и (Q , Q ") depb (D), что и требуется утверждению.

Подслучай P / ε dPrj (D′). Таким образом, правило перехода (Q, D)@μ1 μ должно быть выведено следующим образом для Q Вопрос:

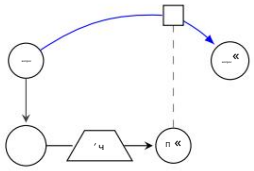
$$\frac{K@П\ \text{Вопрос}\ \overset{\text{det в}}{\ }\ Q/\ dPrj\ (D)\ P\quad / \varepsilon\ dPrj\ (D)}{(Q, D)@(P\ , D')\ (Q\ , D) \text{ в } cgr}$$

Это показывает, что μ = (Q , D), и мы можем замкнуть следующую диаграмму:



Machine Translated by Google

Условие (0b) (P ', P ") depb (D′) дает (P × P ") 'D′ = . Пусть Q относительно« быть уникального подмножества состояний, « Вопрос« дет. Поскольку D′ down Q(D) такое ' × P ") D′ = , то это следует из Q@P Q Q ") 'D = . ' и Q@P Q « Вопрос« относительно что Q@P и ' То есть условие (0b) для (QP ") depb (D') равно P ', ") depb (D). Поскольку П' (P det , что (Q/ dPrj (D′)), условие (1b) (P « П '. От Q@P « Q « и K@П ' Вопрос', отсюда следует, что Q « '. Следовательно, условия (1b) и (2b) из (Q ', Q ") depb (D) , так что (Q 'QQ ") depb (D) . Более того,



.

Это показывает, что Q h ' Q« относительно . Два других требования иска, μ = (Q ', D) и (Q ', Q ") depb (D), были показаны ранее.

Случай h = a. Поскольку Q/ dPrj (D), необходимо использовать следующее правило вывода:

$$\frac{Q\ a\quad Q' \text{ в } Q/\ \varepsilon\ dPrj\ (D)}{(Q, D)\ a\quad (Q', D)\ \text{в cgr}}$$

Итак, μ = (Q ', D), Q a Q ' относительно дет. Более того, поскольку Q совместим с D D – разностное соотношение, Q ' совместим и с D. Следовательно, (Q ' × Q ' и) D = , показывает условие (0b) из (Q) depb (D). Тривиально, Q условия (1b) ', Q ') depb (D). Следовательно, (Q ' acc (Q ′) , и (2b) из (Q. Случай h = ε. Очевидно. ' ' вопрос ', вопрос поэтому) depb (D).

Случай h = h1 · h2. Поскольку (Q, D) μ час относительно. cgr, существует такой μ1 Qcgr, что h1 h2 (Q, D) μ1 и μ1 μ относительно cgr. Мы можем применить гипотезу индукции к h1.

Следовательно, существуют подмножества состояний ' h1 Q, такой что μ1 = (Q1, D), Q Q ' относительно Q1, Q и (Q1, Q ' 1) depb (D). Мы различаем два случая:

h2 Подслучай Q1 dPrj (D). Поскольку μ1 = (Q1, D) μ относительно cgr, из утверждения 2.1б что μ = (Q1, D) , так что (Q1, D) ч2 следует (Q1, D) относительно. кгр. Условие (2b) (K1, Кв ' 1) depb (D) и Q1 dPrj (D) влекут, что Q ' 1 acc (Q1). Пусть Q2 будет acc (Q ′ — единственное подмножество состояний ' ч2 Q2 относительно. дет. Тогда Q2 1) такое что Q , что Q2 acc (Q1), и, следовательно, условие (2b) для (Q1, Q2) depb (D) выполнено. Условие (1b) (Q1, Q2) depb (D) тривиально, поскольку Q1 dPrj (D). Так как Q1 dPrj (D) и Q1 × Q2 acc (Q) 2 , то Q1 × Q2 D = , поэтому условие (0b) для (Q1, Q2) depb (D) также выполнено. Следовательно, Q (Q1, Q2) depb (D). час Q2 относительно det и

Подслучай Q1/ ε dPrj (D). Поскольку Q1 h1 ' и Q1 совместим с разницей , также совместим с отношение D, то Q ' 1 D. Таким образом, мы можем применить гипотезу индукции h2 к (Q1, D) μ, показав существование подмножеств состояний h2) depb (D), таких что μ = (Q2, (K2, Кв ' 2 D) и Q1 Q ' 2 относительно . Итак, у нас есть Q час Q ' 2 относительно и (Q2, Q2 ') depb (D), что и требовалось.

На этом доказательство утверждения 2.2б заканчивается. □

Доказательство включения L(A cgr(S)) S L(A). Пусть h L(A cgr(S)) S. Тогда существует финальное подмножество состояний Q F cgr(S) , такой, что (I, D) A,FS час (Q, D) относительно A,FS cgr. Поскольку я

Machine Translated by Google

— одноэлементный или пустой, он совместим с D init. Утверждение 2.26 показывает, что существует a) depb (D A,FS подмножество состояний Q Q, такой, что I ч Q относительно det и (Q, Q A, ФС). Состояние (0b) из (Q, Q) depb (D A,FS) показывает, что (Q, Q A,FS) D = . Поскольку h S, отсюда следует, что ФС/ = . Детерминизм A показывает, что Q = {q '}. FS, это одноэлемент. Итак, существует государство д что Q такой,

Случай Q/ ε dPrj (D A,ФС). Условие (1б) показывает, что Q Q, так что q Q. Поскольку Q F cgr , имеем Q F/ = , . Поскольку Q совместим с D Q и Q F/ = A,FS из init , q следует, что q F и, следовательно, h / L(A)FS \ F. Так как q FS, отсюда следует q

Случай Q dPrj (D A,ФС acc (Q)). Условие (2b) показывает, что Q′ acc (Q), так что q Поскольку Q F cgr, имеем acc (Q) F/ = . Пусть q acc (Q) F произвольно. Поскольку Q dPrj (D A,FS) и (q/ , q) acc (Q) 2 , отсюда следует, что (q, q A,FS)/ ε D инициализировать . Кроме того, q F , так что qh FS \ F. В сочетании с q FS, отсюда следует q FS , так что L(A).

На этом заканчивается доказательство обратного включения и, следовательно, доказательства L(A) = L(A cgr(S)) S. □

7.4. Полнота Далее

мы покажем полноту конгруэнтного проектирования для подхедж-проекции согласно определению 10. Пусть A = (Σ , Q, , I, F) — полный dSHA и F FS Q.
Автомат A определяет регулярный шаблон L = L(A) с конечными состояниями схемы в FS — регулярную схему S = L(A[F/FS]). Сначала мы покажем, что все нерелевантные к субхежу состояния разностных отношений являются субхедж-проекционными состояниями A cgr(S) согласно определению 7.

Лемма 14. Если Q dPrj (D), то (Q, D) — состояние субхедж-проекции cgr со свидетелем (Q, D).

Доказательство. Мы предполагаем, что q Prj (D) , и должны показать, что (Q, D) является состоянием подперегородочной проекции cgr. Мы должны показать, что все правила перехода, начинающиеся с (Q, D), разрешены для состояния субхедж-проекции (Q, D) с (Q, D). Они генерируются по следующим правилам, все из которых являются циклическими:

a € Σ Q € dPrj (D)	Q dPrj (D)	Q dPrj (D)
(Q, D) a (Q, D) в cgr	(Q, D)@(Q, D) (Q, D) в cgr	(Q, D) (Q, D) в cgr

□

(D). Итак, если частичная серия A на префиксе u присваивает некоторое состояние (P, D) такое, что P dPrj cgr(S) , то (P, D) является состоянием субхедж-проекции по лемме 14, и, таким образом , , подизгородь, не имеющая отношения к предложению 2.

Лемма 15. Пусть A = A[I/], L = L(A) и C = L(A [П/ФС]). Если (, D) hdg(u) (Q, D) относительно Acgr(S) для некоторого вложенного слова u NΣ и либо • Q/ dPrj (D) и q Q, либо • Q dPrj (D) и q acc (Q),

hdg(u , то существует вложенное слово u′ class (u) и q0 , такое что q0 q относительно .
C

Доказательство. Индукцией по длине u предположим, что (, D) hdg(u) --- (Q, D) относительно A cgr(S) .
В базовом случае u = ε. Следовательно, Q = .

Случай Q/ dPrj (D) и q Q. Тогда = {q}. Уникальный пробег A на u = ε начинается и заканчивается в начальном состоянии дерева q .

Machine Translated by Google

Случай Q dPrj (D) и q acc (Q). Поскольку Q dPrj (D), лемма 14 показывает, что (Q, D) ввиду ′(ε) = NΣ .
является состоянием проекции субхеджа, так что ε является субхеджем, нерелевантным для L. ' и C '
Предложение 2. Следовательно, C Более того, поскольку q acc (Q), существует
класс L хеджирует h и q0 I, такие что q0 час q относительно ' = нв(ч). Бег A по тебе ' заканчивается
u classL в q ' и u· Пусть C ′(y).

На этапе индукции различаем возможные формы вложенного слова u. Итак, существуют u˜, v NΣ и a Σ
такие, что имеет место любой из следующих случаев:) cgr(S)

Случай u = u˜ · · v · . Пусть (Q˜ , D) — состояние, в котором серия (A ' на u˜ заканчивается.

Подслучай Q˜ / dPrj (D). Пусть D′ = DQ˜ . Тогда существует P Q такой, что (, D′) v (P, D′)
относительно A cgr(S)) cgr(S) на v переходит в состояние (P, D′). Итак, пробег (A '

Подподслучай P dPrj (D′). По построению A cgr(S) , тогда мы имеем Q = Q˜
@ acc (P). Поскольку q Q, существуют q˜ Q˜ и p acc (P), такие что ′(v), такие ·
q˜@p q в . По предположению индукции существует v , что ' классL C '
серия A ' ′ на v заканчивается на стр. Следовательно, бег A ' = u˜ · · v '
заканчивается на q, и, кроме того, u ' на u classL (u). C

Подподслучай P/ ε dPrj (D′). По построению A cgr(S) , тогда мы имеем Q =
Q˜ @ P. Поскольку q Q, существуют q˜ Q˜ и p P такие, что q˜@p q в .
По предположению индукции существует v ′(v), такой что серия · ' классL C ' заканчивается в q, и, кроме
' ′ на v заканчивается на стр. Следовательно, бег A ' = u˜ · · v ' того, A , u
' на u classL (u). C

Подслучай Q˜ dPrj (D). Тогда Q = Q˜ и q acc (Q˜). Используя индукцию hdg(u˜ (u˜),
гипотеза, примененная к u˜, существует u˜ ' классL C такую, что q. classL
Поскольку u˜ не имеет значения для L и S ', у нас есть ты~ ' (u). C

Случай u = u˜ · a. Этого легче достичь, чем в предыдущих случаях, и поэтому он опускается.
□

Лемма 16. Частичный запуск A cgr(S) присваивает состояние (Q, D) префиксу u prefs(NΣ). Если Q/ dPrj (D)
и q Q, то существует префикс u classL (u), такой, что ' частичная C серия A присваивает u′ состояние q .

Доказательство. Индукцией по длине u. В базовом случае u = ε. Частичный запуск init . Предположим, что Q/ ε
A cgr(S) на u присваивает состояние (Q, D), где Q = I и D = D A ΦC dPrj (D)
и q Q. Тогда I = {q}. Единственный частичный пробег A на u = ε заканчивается в начальном состоянии = A[I/
возможные формы u. Итак, существуют u˜ '], L) и S [F/FS]. Мы q L(A ' На этапе индукции (пусть A различает
prefs(NΣ), v NΣ и a Σ , такие, что выполняется любое из следующих условий:

Случай u = u˜ · · v. Пусть (Q˜ , D˜) — состояние, в котором частичный пробег A cgr(S) на u˜ заканчивается.

Из Q/ dPrj (D) следует, что Q˜ / dPrj (D˜). Кроме того, (, D) v (Q, D) относительно
A cgr(S)) cgr(S) Итак, частичный пробег (A на v ' переходит в состояние (Q, D). По предположению индукции
существует v ' classL C ′(v), такой, что частичная серия A = u˜ · · v ' на
′ v заканчивается на q. Следовательно, частичный пробег A заканчивается в q, и, кроме того,
ты на u classL (u). C

Случай u = u˜ · · v · . Пусть (Q˜ , D˜) — состояние, в котором частичный пробег A cgr(S) на u˜ заканчивается.

Из Q/ dPrj (D) следует, что Q˜ / dPrj (D˜). Более того, (, D) v (P, D) относительно A cgr(S)
для некоторого P Q.

Подслучай P dPrj (D). Тогда Q = Q˜ @ acc (P). Итак, существуют q˜ Q и p acc (P), такие что q˜@p
q в . По лемме 15 существует v ′(v), ' классL C

Machine Translated by Google

такой, что hdg(v') p относительно . Используя предположение индукции, примененное к u˜, существует начальное состояние q0 I и префикс u˜ classL (u˜), таким, что частичное · ' · пробег A на u˜ переходит к состоянию q˜. Следовательно, частичный пробег A на = ты˜ v · переходит к q и u u classL (u).

Подслучай P/ ε dPrj (D). Тогда Q = Q˜ @ P. Итак, существуют q˜ Q и p P, такие classL ′(v), такие, что что q˜@p q в . Используя лемму 15, существует v

hdg(v') p относительно . Используя предположение индукции, примененное к u˜, существует начальное состояние q0 I и префикс u˜ на classL (u˜), такой что частичный пробег A · · v · u˜ переходит к состоянию q˜. Следовательно, частичный пробег A на = ты˜ продолжается к q и тебе u classL (u).

Случай u = u˜ · a. Этого легче достичь, чем в предыдущем случае, и поэтому он опускается.

□

Следующая лемма устанавливает ключевой инвариант конгруэнтной проекции, который в конечном итоге доказывает свою полноту для подхедж-проекции. Для этого определим для любого F FS F бинарное отношение как симметричное замыкание F × (FS \ F), т.е. для всех (q , q ") Q2 :

д′/ ФС q « (д F q FS \ F « FS \ F)) (д q F

Лемма 17 (ключевой инвариант). Пусть A = (Σ , Q, , I, F) a dSHA, F FS Q, L = L(A) и S = L(A[F/FS]). Если A cgr(S) имеет частичный пробег по префиксу u prefs(NΣ) для утверждения (P, D) 2 Q × D , то для всех (p , p ") P 2 , (p ,r ") D и (v , v ") N 2 , такой, что

п hdg(v ') ′---- r) относительно p « hdg(v ") относительно

существуют (ты , u ") classL (u), w suffs(NΣ), q′/ F FS q " и q0 I такие, что

q hdg(u ·w) ·v q0 относительно q0 hdg(u ·w) q « ·в « относительно

Доказательство. Индукция по количеству висячих открывающих скобок u.

В базовом случае u не имеет висячих открывающих скобок, поэтому u NΣ является A,FS init . Итак, A cgr(S) В этом случае D = DA ,FS to (P, D), где P имеет частичную серию во вложенном слове u NΣ вложенного слова. Пусть (p Q. , p ") P 2 , (p ,r ") D A,ФС инициализировать и (v , v ") N 2 такой, что Σ ,

п hdg(v ′) относительно p « hdg(v ") ′ ---- r относительно

Отсюда следует, что P/ ε dPrj (D A,ФС). Поэтому мы можем применить лемму 16. Она показывает, что существуют вложенные слова (u , u ") (classL (u))2 и q0 I, такие, что

hdg(u ') p q0 относительно q0 hdg(u ") p « относительно

Поскольку Д A,ФС ldr (F × (FS \ F)) по лемме 10 существуют вложенное слово w NΣ и ′/ F утверждает q FS q », такой, что

p ′ hdg(w) --- q относительно r « hdg(w) --- q « относительно

Тогда у нас есть следующее:

hdg(u ' ′ ·v · ш) q q0 относительно q0 hdg(u « ·в » · ш) q « относительно

Machine Translated by Google

Для шага индукции пусть u = u1 · · u2 для некоторого префикса u1 prefs(NΣ) и вложенного слова u2 NΣ так, что u1 имеет на одну открытую висячую скобку меньше, чем u. Пусть A cgr(S) имеет частичную серию слова u NΣ , вложенного в (P, D). Пусть (P1, D1) — состояние, которое этот прогон присваивает префиксу u1. Тогда D = (D1)P1 . Пусть (п , р ") P , (p ,r ") D и (v , v ") N 2 такой, что Σ ,

п hdg(v r) относительно p « hdg(v ") p относительно

Поскольку D = (D1)P1 , существуют (р и р 1 , п 1 «) (P1) и штаты (r 1 ,p 1 «) D1, такой что р 1@q p 1 1@q « p 1 . » В частности, P1 / dPrj (D1), так что мы можем применить лемму 16. Это показывает, что существуют (u « Тебе 2 , 2) классL C (u2), такой что p hdg(v hdg(v и р ». Позволять v 1 = · ты 2 · v · и v относительно « ты 2 = ты 2 · Тогда п 1 r) 1 п 1) . Используя гипотезу индукции, примененную к u1, на которой A cgr(S) имеет частичный переход к состоянию (P1, D1), такой, что (p 1 , p 1 «) (P1) 2 , п 1 hdg(v r 1) 1 , p 1 hdg(v p 1 ,) и (r 1 ,r 1 «) D существуют (ты « тебе 1 , 1) классL C (u1), w1 suffs(NΣ), q / F FS q " и q0 I, такие, что

hdg(u 1 w1) q0 относительно hdg(u 1 w1) « относительно

Позвольте тебе знак равно ты 1 · · ты « ты 2 , = ты 1 · · ты « , и w = · w1. Вышеизложенное дает следующее:

hdg(u · w) q0 относительно hdg(u · w) « относительно

Кроме того, (вы , u ") (classL (u)) 2 , так что это нужно было показать. □

Предложение 6. Пусть A = (Σ , Q, , I, F) dSHA , F FS Q, L = L(A) и S = L(A[F/FS]).

Если A cgr(S) имеет частичный пробег по префиксу u prefs(NΣ) до некоторого состояния (P, D) 2 Q × D так, что P / dPrj (D), то classL (u) является подходящим актуально для L относительно S.

Доказательство. Предположим, что A cgr(S) имеет частичный пробег по префиксу u prefs(NΣ) до некоторого состояния / = пусть D множество всех v ") P , N 2 dPrj (D). Поскольку P / dPrj (D), существует acc (P) 2 2 Q D Итак, существуют (п , р ") P , (p ,r ") D и (v Σ , такой, что

п hdg(v) относительно p « hdg(v ") r относительно

Используя лемму 17, существуют (u u ") classL / F (u), w suffs(NΣ), q FS q ", и q0 I, такие, что

hdg(u · v · ш) q q0 относительно hdg(u · w · ш) « относительно

Следовательно, класс L (u) актуально для L относительно S. □

Теорема 3 (Полнота конгруэнтного проектирования). Для любого полного dSHA (Σ , Q, , I, F) и множества F FS Q конгруэнтная проекция A cgr(S) корректна и полна для проекции субхеджа для регулярного шаблона L(A) относительно регулярной схемы S = L(A[FS/F]).

Доказательство. Правильность конгруэнтного проектирования показана в теореме 2. Для доказательства полноты пусть u prefs(NΣ) — вложенный словесный префикс, который является сильно нерелевантным для L относительно S. По определению сильной иррелевантности класс classL (u) является не имеет значения для L и S. Пусть (P, D) — уникальное состояние, назначенное частичной серией A cgr(S) префиксу u. Поскольку (u) не имеет значения для L и S, предложение 6 показывает, что P dPrj (D). Используя лемму 14, classL S (P, D) тогда является состоянием субхедж-проекции A cgr(S)

. □

Machine Translated by Google

7.5. Сложность в памяти Далее

мы обсудим сложность проверки принадлежности с полным проецированием субхеджа, основанным на оценке входного хеджа в памяти с помощью конгруэнтной проекции входного автомата.

Лемма 18. Число состояний |Qcgr| находится в O(2 n 2+n), где n — количество состояний A.

Доказательство. При детерминированной конструкции состояния A cgr(S) являются парами в 2Q × D . Итак, максимальное число состояний конгруэнтной проекции равно |Qcgr(S) | = 2 |Q| |Д |
2н 2 2 н = 2 п 2+п . □

Пусть diff-rel(Qcgr(S)) — набор разностных отношений, используемых в состояниях A cgr(S) .

Следствие 1. Пусть подпись Σ фиксирована. Для любого полного dSHA A со схемой S НΣ принадлежность любого хеджа h S может быть проверена в памяти за время O(|1|) на каждый непроецируемый узел h после времени предварительной обработки O(|A cgr(S) | + n 3d), где d — мощность diff -rel(Qcgr(S)) , а n — количество состояний A.

Поскольку |Qcgr(S) | находится в O(2 n 2+n) по лемме 18, время предварительной обработки O(|A cgr(S) | + n 3d|) тоже принадлежит O(2 2n 2+2n).

Доказательство. Поскольку A является полным, теорема 3 показывает, что A cgr(S) является полным для проекции субхеджа для схемы S. Вычислитель A cgr(S) в памяти может быть запущен на любом входном хедже h S за время O(|1|) на каждый непроецируемый узел h после времени предварительной обработки O(|A cgr(S) | + n 3d), где d — мощность diff-rel(A cgr(S)). Поскольку подпись Σ фиксирована и A cgr(S) детерминирован , |A cgr(S) | находится в O(|Qcgr(S)|), доминирующая часть |A cgr(S) | находится в O(|Qcgr| 2). □

Применительно к обычному ответу на запрос XPath, как в наших экспериментах в разделе 11, мы имеем n 80 и d 25, поэтому кубическая часть O(n 3d) не является ограничивающей. Размеры SHA меньше 2600, считая как состояния, так и правила . Таким образом, верхняя граница O(|A cgr(S) | + n 3d) предполагает, что время предварительной обработки может быть осуществимо на практике, даже если в худшем случае оно экспоненциально.

Здесь следует быть осторожными, поскольку размеры SHA , указанные в разделе 11, не относятся к к чистой конгруэнтной проекции A cgr(S) , а к самой ранней конгруэнтной проекции A , которую мы кгр(S) e(C) введем только в разделе 8. Несоздание правил перехода в состояния, безопасные для отклонения, может привести к радикальному уменьшению размера. Альтернативно, существенное уменьшение размера A cgr(S) можно получить, удалив все бесполезные состояния и правила перехода.
Но это не уменьшит время предварительного вычисления, а наоборот: увеличит время предварительной обработки, поскольку апостериорная очистка автоматов также требует времени, которое для SHA может быть нелинейным.

Вместо статического предварительного вычисления A cgr(S) на основе входных данных A и FS , мы можем вычислить только ту часть A cgr(S), которая необходима для оценки входного хеджа h «на лету». Заметим, что количество правил перехода необходимой части ограничено O(|h|), но может быть намного меньше из-за проекции. Тогда время предварительной обработки уменьшается до O(|A|). Поскольку в худшем случае полный автомат A cgr(S) может быть экспоненциальным по n , это может сократить общее время обработки.

С положительной стороны, оценка «на лету» сокращает общее время вычислений до O(|h| + n 3d(h)), где d(h) — количество разностных отношений в части A cgr(S) необходимо для вычисления h, поэтому d(h) min(|h|, d). С другой стороны, шаги d(h) больше не будут осуществляться за постоянное, а кубическое время за O(n). Следует также отметить, что общее время оперативной оценки прогноза без изменений составляет O(|A| + |h| + n s(h)), где s(h) — число подмножеств безопасных состояний A в состояниях Qcgr, т. е. s(h) |h|. Это связано с тем, что каждое подмножество безопасных состояний может быть вычислено за линейное время за O(n). Как утверждалось выше, общее время конгруэнтной проекции, полученной путем оперативной оценки, составляет O(|A| + |h| + n).

3 3
д(ч)). Это связано с тем, что вычисление разностного отношения занимает O(n

Machine Translated by Google

с использованием наземных программ Datalog кубического размера. Дополнительный кубический коэффициент ^{3} в n d(h) вместо линейного коэффициента в n s(h) — это цена, которую приходится платить за достижение полной проекции субхеджа с оперативной оценкой. Обратите внимание, что эта цена в значительной степени приемлема в случаях , когда n 80, d(h) 15 и s(h) 15.

Пример 15 (Экспоненциальное разрушение). Чтобы увидеть, как может произойти экспоненциальный худший случай, мы рассмотрим семейство регулярных языков, для которого минимальный DFA слева направо экспоненциально больше, чем минимальный DFA справа налево . Классическими примерами языков с этим свойством являются $L_n = \Sigma^* a \Sigma^n$, где n N и Σ = {a, b}. Интуитивно, слово из Σ принадлежит Ln тогда и только тогда, когда его n + 1-я буква с конца равна «a». Минимальный DFA слева направо для Ln имеет 2 n+1 состояний, поскольку ему необходимо запомнить окно из n + 1 букв. Напротив, его минимальный DFA справа налево имеет только n + 1 состояние; в этом направлении достаточно запомнить расстояние от конца по модулю n + 1.

Далее мы рассмотрим схему S, определенную $\mu z.\ (a + b) \cdot z$. Он содержит все изгороди, имеющие во всех своих поддеревьях метку из Σ = {a, b} в качестве первой буквы, и никаких дальнейших вхождений букв Σ где-либо еще. Затем мы рассматриваем семейство хедж-языков H_n HΣ со схемой S, такое, что последовательность меток некоторого пути от корня к листу hn принадлежит Ln. Обратите внимание , что Hn может распознаваться восходящим способом с помощью dSHA An с состояниями O(n + 1) , который имитирует минимальный детерминированный DFA Ln на всех путях входного хеджа. Для оценщика с проекцией субхеджирования ситуация иная. При движении сверху вниз ему необходимо запомнить последовательность s, а 2^{n+1} меток n + 1-последних предков, возможно, заполненных b таких последовательностей.
Если для некоторого листа его последовательность начинается с буквы «a», то следующие подизгороди со следующими листьями можно спроецировать. Как следствие, не может быть никакого dSHA , распознающего Hn , который проецирует все нерелевантные субхеджи с числом состояний менее 2^{n+1} . В частности, размер A должен $\geq 2^{cgr(S)}_{n}$ экспоненциально зависеть от размера An.

8. Самое раннее членство с помощью проекции субхеджирования.

Далее мы улучшаем наши компиляторы для введения состояний проекции субхеджирования, чтобы они могли также обнаруживать определенное членство или нечленство самым ранним образом. Для этого мы покажем, как объединить предыдущий самый ранний тестер членства для dSHA из [21] с любым алгоритмом проецирования поддерева, чтобы он применим как к проецированию без изменений, так и к проецированию конгруэнтности. Комбинируя оба аспекта ортогонально, мы получаем самый ранний dSHA с проекцией субхеджа, которую можно оценивать либо в памяти, либо в потоковом режиме.

8.1. Самое раннее членство

Мы начинаем с семантической характеристики самого раннего членства, т. е. префиксов вложенных слов хеджирования, суффиксы которых не имеют значения для членства, при условии обработки сверху вниз и слева направо. Обратите внимание, что та же самая характеристика различных вариантов представления была дана ранее в контексте потоковой обработки [21,22].

Определение 13. Пусть S HΣ — хедж-схема и L S — хедж-язык, удовлетворяющий этой схеме. Префикс вложенного слова v определен для принадлежности к L со схемой S для всех суффиксов w вложенных слов таких, что v · w nw(S) . Верно, что v · w nw(L). Оно называется достоверным при непринадлежности к L со схемой S для всех суффиксов w вложенных слов таких, что v · w nw(S) , верно , что v · w/ nw(L).

Теперь нас интересуют SHA , которые способны обнаруживать определенное членство и отсутствие членство синтаксически по самому раннему возможному префиксу или узлу выполнения.

Определение 14. dSHA A = (Σ , Q, , I, F) со схемой S HΣ называется самым ранним для схемы S , если существует состояние sel Q такое, что для всех хеджей h S единственное запуск R команды compl(A) на h удовлетворяет следующему:

Machine Translated by Google

- он переходит в состояние sel ровно для всех тех префиксов nw(h) , которые уверены в принадлежности к L(A)
- относительно S; он отправляется в приемник для всех префиксов nw(h) , которые уверены в нечленстве в L(A) с S.

Для dSHA A, полной схемы для схемы S = L(A[F/FS]), мы можем построить dSHA Ae(S) , который является самым ранним для S, как показано в Приложении A. Идея состоит в том, чтобы адаптировать самый ранний тестер членства для детерминированных SHA из [21] , который учитывает ограничения схемы . Кроме того, нам нужно скомпилировать входные dSHA в dSHA , а не в dNWA, как здесь используется. Это изменение станет очевидным, как только станет ясна взаимосвязь между обеими моделями автоматов.

Учитывая, что построение Ae(S) не является частью сути настоящей статьи, мы лишь проиллюстрируем его результат на нашем рабочем примере: для dSHA A со схемой S = doc на рисунке 4 мы получаем SHA Ae (S) на рисунке 19. Обратите внимание, что в элементе префикса list он переходит в состояние sel, показывая, что он уверен в членстве. В префиксе item он блокируется , показывая, что он не является членом. Однако в префиксе list list он не переходит в состояние проекции субхеджа, хотя этот префикс не имеет отношения к субхеджированию.

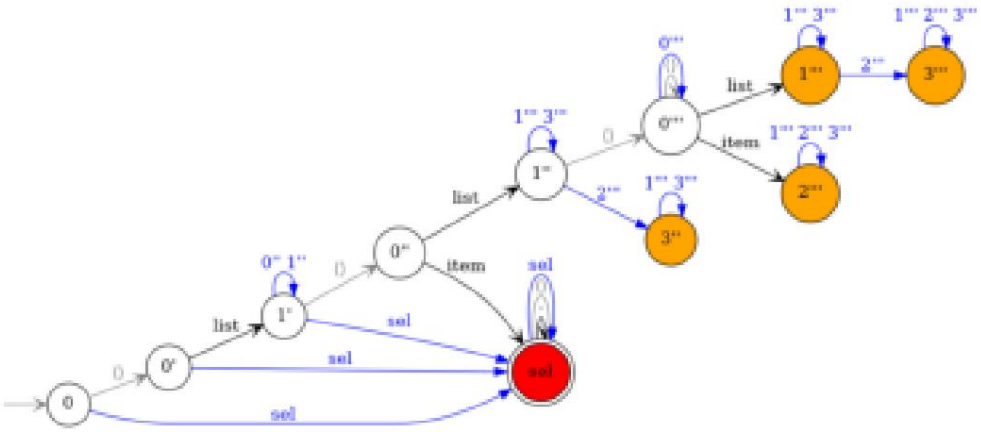


Рисунок 19. Самый ранний dSHA Ae(S) для dSHA A на рисунке 4 со схемой S = doc для фильтра XPath [self::list/child::item].

8.2. Добавление проекции субхеджа

Теперь мы расширяем любое раннее членство в dSHA с помощью прогноза субхеджирования. Идея состоит в том, чтобы объединить проекцию субхеджа с самым ранним членством. Итак, предположим, что нам даны схема S и два автомата A π и Ae со схемой S , которые распознают один и тот же язык с точностью до схемы, поэтому L(A π) L(S) = L(Ae) L (C). Кроме того, предположим, что Ae имеет состояние выбора sel, указывающее на определенное членство в самом раннем возможном префиксе.

Чтобы получить самое раннее членство с проекцией подхеджирования, мы объединяем два SHA в один SHA F e), по сути, запуская оба автомата параллельно, но под общим контролем. Набор состояний A $A^{\pi}_{e} = (\Sigma, Q^{\pi}_{e}, я^{\pi}_{e}, \Phi^{\pi}_{e})$ следующие:

$$Q^{\pi}_{e} = (Q\pi \times (Qe \setminus \{sel\})) \quad \{sel\}$$
$$я^{\pi}_{e} = (I\pi \times (Ie \setminus \{sel\})) \quad \{sel \mid sel \quad Ie\}$$
$$\Phi^{\pi}_{e} = (F\pi \times (Fe \setminus \{sel\})) \quad \{sel\}$$

Правила перехода в A^{π}_{e} приведены на рисунке 20. Любая серия A^{π}_{e} синхронизируется параллельно , и пробегает A^{π} и Ae следующим образом: всякий раз, когда Ae переходит в sel, тогда A^{π}_{e} всякий раз, когда он переходит в A^{π} состояние проекции суб-хеджа, он заставляет Ae перепрыгнуть через последующий суб- хедж.

Machine Translated by Google

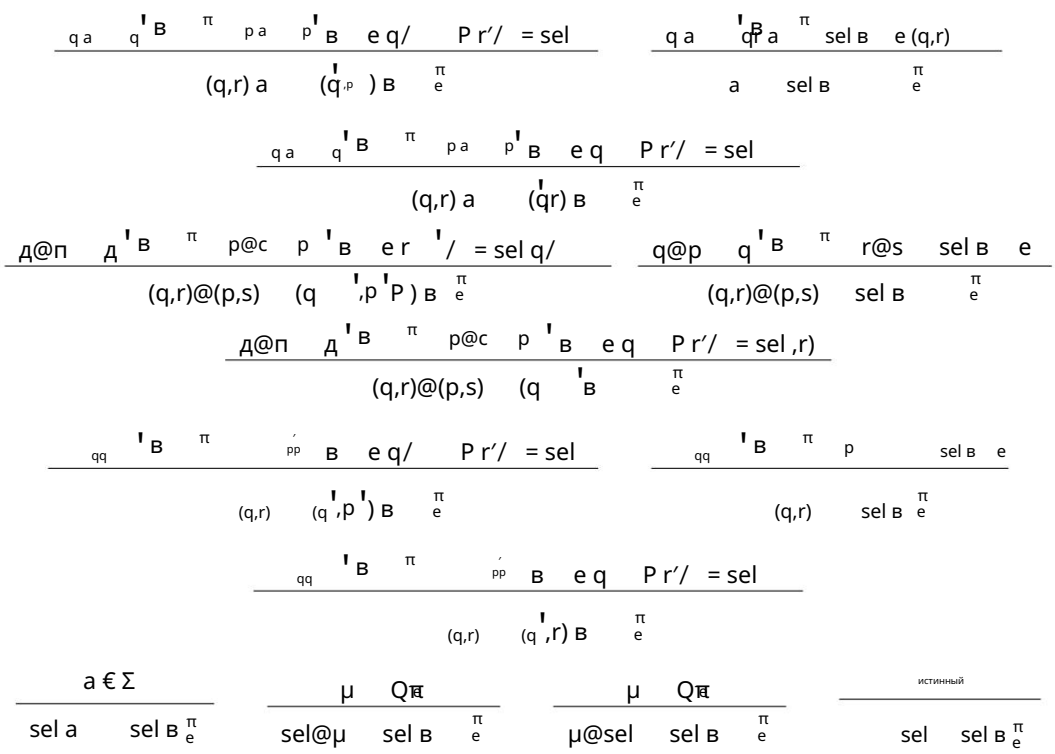


Рис. 20. Правила перехода Ae выводятся из состояний SHA s A с состояниями проекции P и с состоянием выбора sel.

Для иллюстрации мы пересматриваем dSHA с рисунка 4. Самый ранний dSHA с показан на проекция субхеджа A e(S) рисунке 21. Здесь мы объединяем безопасный вариант без изменений. проекция A из рисунка 11 с самым ранним тестером членства Ae(S) из рисунка 19. e(S) Их сочетание A представляет шаблон [self::list/child::item] с документом схемы в краткой форме. После префикса list item он обнаруживает определенное членство: для префикса list list он определяет, что субхедж не имеет значения, а после префикса item он обнаруживает определенное отсутствие членства.

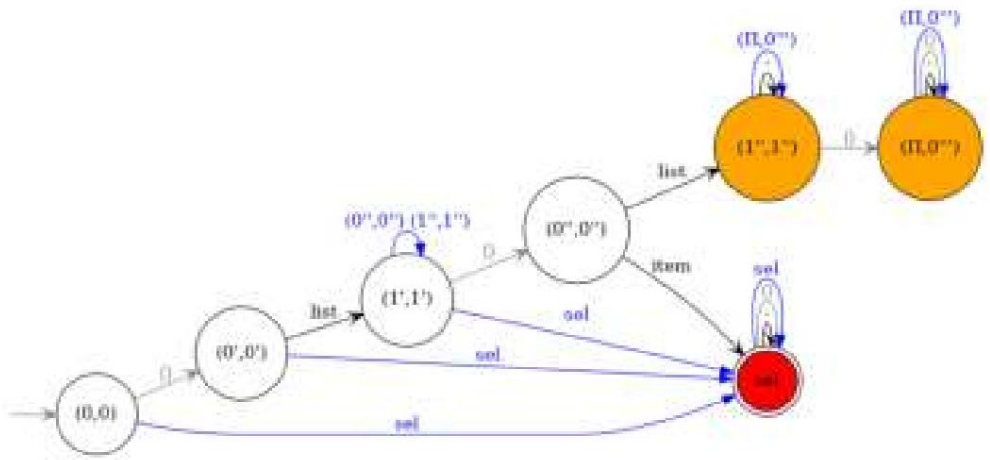


Рисунок 21. Самый ранний dSHA с проекцией подхеджирования без изменений для dSHA A в e(S) A Рисунок 4 со схемой S = doc для фильтра XPath [self::list/child::item].

Machine Translated by Google

8.3. Надежность и полнота

Далее мы покажем, что надежность и полнота алгоритма проецирования сохраняются в сочетании с каким-либо самым ранним тестером членства при допущении детерминизма.

предложения 7. Пусть A — π и Ae — dSHA s с одной и той же схемой S и одним и тем же языком с точностью до схема, т. е. L(A π) L(S) = L(Ae) L(S). Комбинация dSHA также является языком до схемы π_e тогда у него то же самое S. Более того, если Ae является самым ранним для S, то A переходит в некоторое π_e является самым ранним для S и идет состояние проекции субхеджа всякий раз, когда это делает Aπ .

подперегородкой . Доказательство. Если q является состоянием проекции субхеджатогда (q,r) является параллельно в состоянии π_e . Вычислитель для автомата A π_e бежит A A π , a r - состоянием Ae , π и Ae проекции входного хеджа A , при этом пропускаются субхеджи, начинающиеся в состояниях проекции субхеджа A π. К ним относятся субхеджи, начинающиеся в состоянии проекции при оценке A π_e на ч. Применение π Предложения 2 до A π, такие субхеджи не имеют значения для L(A π) по отношению к S, поскольку A предполагается детерминированным. Поскольку L(A π) S = L(Ae) S, пропуск такого подхеджа не влияет на принятие Ae хеджей внутри схемы S. Следовательно, оценщик A π_e также является самым ранним по отношению к S. □

Приведенное выше предложение показывает, что π полно для проекции подизгородья, то A π_e если A также является полным для проекции субхеджа, но при этом является самым ранним.

Теорема 4. Для любого полного dSHA A со схемой S dSHA A cgr(S)e(S) является самым ранним и полным для проекции субхеджа для схемы S.

Доказательство. Конгруэнтная проекция A cgr(S) корректна по предложению 2 и полна для подхедж-проекции относительно S по теореме 3. Автомат Ae(S) , обсуждаемый в разделе 8.1 , является самым ранним для S. Итак, их комбинация A cgr(S)e(S) дает самый ранний автомат с полной проекцией субхеджа относительно S с использованием предложения 7. □

8.4. Сложность в памяти Далее

мы обсудим сложность самой ранней проверки членства с полным проецированием субхеджа с помощью оценщика в памяти для A cgr(S)e(S) .

Лемма 19. Число состояний в Qcgr(S)e(S) находится в O(2 n 2+2n+log(n)), где n = |Q|.

Доказательство. Используя лемму 18, число состояний A cgr(S) находится в O(2 n 2+n), где n = |Q|. Число состояний Ae(S) находится в O(n2 n), а значит, в O(2 n+log(n)). Таким образом , число состояний Qcgr (S)e(S) находится в O(2 n 2+n 2 n+log(n)), что находится в O(2 n 2+2n+log(n)). □

Следствие 2. Пусть подпись Σ фиксирована. Для любого полного dSHA A со схемой S HΣ самое раннее членство во входном хедже h S может быть проверено в памяти за время O(| 1 |) на каждый непроецируемый узел h после времени предварительной обработки O(|A cgr(S)e(S) | + n 3d + n s), где d — количество разностных отношений, а s — количество подмножества безопасных состояний в Qcgr(S)e(S) . Время предварительной состояниобработкиO(также равно 2n 2+4n+2log(n)), где n — количество

Доказательство. Поскольку A является полным, теорема 4 показывает, что A cgr(S)e(S) является самым ранним, надежным и полным для проекции субхеджа для схемы S. Вычислителем A cgr(S)e(S) в памяти может быть запустить на любом входном хедже h S за время O(|1|) на каждый непроецируемый узел h после времени предварительной обработки O(|A cgr(S)e(S) | + n 3d + n s. Σ фиксируется как размер |A ограничен O(|Qcgr(S)e(S) | 2n ^{2} cgr(S)e(S) | Лемма 19 показывает, что |Qcgr(S)e(S) | находится в O(2 n 2+2n+log(n)) итак |A cgr(S)e(S) | находится за O(2 $^{2+4n+2log(n)}$). □

Machine Translated by Google

Как и раньше, вместо предварительного вычисления A cgr(S)e(S) на основе входных dSHA A и FS, мы можем вычислить только ту часть, которая необходима для оценки входного хеджа h на лету. Если необходимая часть A cgr (S)e(S) для вычисления h мала, общее время и пространство значительно уменьшаются.

9. Алгоритмы потоковой

передачи Мы показываем, что dSHA можно оценивать по вложенному слову хеджа потоковым способом , так что они дают те же результаты и поведение проекции, что и при их оценке на хеджах в памяти в нисходящем и нисходящем порядке. образом слева направо.

Все аспекты, связанные с проекцией субхеджирования, остаются неизменными. Самый ранний ответ на запрос в потоковом режиме означает, что суффиксы вложенных слов игнорируются, если они нерелевантны, а не только вложенные слова, как в случае проекции субхеджирования. Что меняется между обоими режимами оценки, так это потребление памяти. Для оценки в памяти должен храниться весь граф входного хеджа, тогда как в потоковом режиме при любом событии сохраняются только стек и состояние. Для проверки принадлежности к обычному языку пространство состояний конечно, а пространство стека становится бесконечным, поскольку глубина стека зависит от глубины входного хеджа.

Свойство наличия эквивалентных потоковых и хранящихся в памяти вычислителей уже было замечено в [26] для автоматов с выталкивающим лесом Неймана и Зейдля [25] и для автоматов с вложенными словами [11,15]. Здесь мы покажем, как перенести этот результат в dSHA s. Это позволяет показать корректность наших алгоритмов оценивания на основе свойств построенных автоматов. Это также позволяет получить свойства корректности потоковых алгоритмов, рассуждая о оценщике SHA в памяти .

9.1. Потоковые вычислители для автоматов нисходящего

хеджирования. Мы определяем потоковый вычислитель SHA A = (Σ , Q, , I, F) с помощью машины с видимым толканием вниз . Набор конфигураций этой машины — K = Q × Q , содержащий пары состояний и стопки состояний. Машина с видимым нажатием обеспечивает для любого слова v Σ ^ отношение потокового ул. K × K, такой, что для всех состояний q, q' Q и складывает перехода v Σ ^ σ' Q й слов v, v

$$\frac{\text{истинный}}{(q,\sigma)\ \varepsilon\ \text{ул}\ (q,\sigma)\ \text{относительно}} \qquad \frac{(q,\sigma)\ v\ \text{ул.}\ (q',\sigma) \qquad (q',\sigma)\ {}'v\ \text{ул}\ (q\ll,\sigma)\ \text{относительно}}{(q,\sigma)\ \text{ул.}\ {}'v{\cdot}v\ (q\ll,\sigma)\ \text{относительно}}$$

$$\frac{q\ a\ q'\ в}{q\ a\ д'\ \text{относительно}}\ \text{ул.} \qquad \frac{q\ q'\ в}{(q,\sigma)\ (q',\sigma\cdot q)\ \text{относительно}}\ \text{ул.} \qquad \frac{д@п\ д'\ в}{(p,\sigma\cdot q)\ (q',\sigma)\ \text{относительно}}\ \text{ул.}$$

Отметим, что ту же машину с видимой передачей можно получить путем компиляции SHA в NWA, чей потоковый оценщик также определяется машиной с видимой передачей (см. раздел 12.3 о соответствующей работе).

Понятие частичных запусков оценщика в памяти тесно связано с потоковыми запусками. Чтобы убедиться в этом, сначала отметим, что каждая частичная серия v относительно естественным образом определяет стек следующим образом. Поскольку v является префиксом вложенного слова, существуют уникальные вложенные слова v0, . . . , vn nw(HΣ Q), такие, что

$$v = v0 \cdot \ \cdot v1 \cdot \ \cdot \ldots \cdot \ \cdot вн.$$

Эти вложенные слова также должны быть частичными сериями , поэтому существуют состояния p0, . . . pn, q0, . . . , qn Q, такой, что vi переходит от pi к qi для всех 0 i n. Мы определяем стек σ . . . элемента v как · qn 1 и говорим, что v переходит от p0 к qn и имеет стек σ. Обратите внимание, что стек σ = q0 · пуст, если v — вложенное слово, поскольку в этом случае n = 0, так что v = v0 и σ = ε. Например, частичная серия v0 = 0 · · 0 · list · 1 · · 0 · item · 2 имеет стек σ0 = 0 · 1, а частичная серия v1 = v0 · · 3 · · 4 имеет стек σ1 = ε.

Machine Translated by Google

Лемма 20. Рассмотрим SHA A = (Σ , Q, , I, F), состояние q, q — префикс Q и стек σ Q . Пусть v будет с некоторого вложенного слова в nw(HΣ). Тогда существует частичный пробег r относительно от q до q стека σ такой, что projΣ (r) = v тогда и только тогда, когда (q,ε) v(q , σ) относительно .

Доказательство. Это осуществляется индукцией по длине префикса v. □

Из этой леммы следует, что любой переход A в памяти на изгороди h из q в q может можно отождествить с некоторым потоковым переходом A на nw(h) из (q,ε) в (q ,ε).

h Предложение 8. q q относительно (q,ε) (q ,ε) относительно .

Дрказательство. Если q относительно , существует серия R h c , которая начинается с q и д заканчивается . Рассмотрим частичный пробег v = nw(R) на h. Поскольку v — вложенное слово, его стек пуст. По лемме 20 имеем (q,ε) (q ,ε) относительно и projΣ (v) = nw(h). Для обратного импликация, предположим, что (q,ε) (q ,ε) относительно . Пусть u = nw(h). По лемме 20 там со существует частичный пробег v функции от q до q стеком ε. Тогда существует серия R группы на h такая, что h , что v = nw(R). Следовательно, q q относительно . □

Для любого детерминированного dSHA A, хеджа h и состояния q потоковый переход на nw(h) относительно , начиная с (q,ε), может быть вычислен за время O(1) на букву после предварительного вычисления за время O (|A|). Таким образом, общее время вычислений составляет O(|A| + |h|). Потоковая память, необходимая для хранения конфигурации, имеет размер O(глубина(h) + |A|), поскольку размер видимого стека ограничен глубиной входного хеджа.

9.2. Добавление проекции субхеджа

Затем мы расширяем отношение потокового перехода с помощью проекции субхеджа по аналогии с тем, что мы сделали для отношения перехода в памяти. Определим отношение перехода с помощью выступ изгороди h K × K относительно , такой, что для всех слов v, v shp Σ ˆ , буквы a Σ , состояния p, q, q , q « Q и складывает σ, σ ' « σ Q :

$$\frac{q \quad Q_{\text{шп}} \quad v \quad nw(H\Sigma)}{\text{ул.}}$$

(q, σ) v (q, σ) относительно shp

$$\frac{q/ \ \varepsilon \ Q_{\text{шп}} \quad q \ a \quad q}{\text{ул.}} \qquad \frac{q/ \ \varepsilon \ Q_{\text{шп}}}{\varepsilon \ \text{ул.}}$$

(q, σ) a (q шп , σ) относительно (q, σ) (q, σ) относительно shp

$$\frac{q/ \ \varepsilon \ Q_{\text{шп}} \quad (q, \sigma) \ v \quad (q_{\text{шп}}, \sigma) \text{ относительно } (q', \sigma)}{(q, \sigma) \quad (q_{\text{шп}}, \sigma) \text{ относительно}}$$

$$\frac{q/ \ \varepsilon \ Q_{\text{шп}} \quad qq}{\text{ул.}} \qquad \frac{p/ \ \varepsilon \ Q_{\text{шп}} \quad д@п \quad д}{\text{ул.}}$$

(q, σ) (q , σ · q) относительно (p, σ · q) (q , σ) относительно

Проекционное переходное отношение остается в конфигурации с состоянием проекции субхеджа до тех пор, пока не будет достигнут конец текущего субхеджа. Это правильно, поскольку состояние проекции субхеджа в любом случае изменить нельзя.

Machine Translated by Google

Предложение 9. Для любого SHA A = (Σ , Q, , I, F) состояния q, q не ' Q, вложенное слово v , на котором имеют блокирующего частичного пробега, начиная с q:

ул. ул.
(q,ε) v (q ',ε) относительно (q,ε) v (q shp',ε) относительно .

Доказательство. Пусть h НΣ — хедж такой, что v = nw(h). Доказательство проводится индукцией по размеру v . Для импликации слева направо мы предполагаем, что (q,ε) v Поскольку v = nw(h), предложение 8 доказывает q ул. (q ',ε) относительно .
ул
ч д ' относительно .

Случай q Q . По лемме 6 из q shp следует . ул
ч q ' относительно , что q = q '. Затем мы можем подать заявку следующее правило, поскольку v — вложенное слово:

$$\frac{q \quad Q_{\text{шп}} \qquad v = nw(H\Sigma)}{(q,\varepsilon)\ v \quad \overset{\text{ул.}}{(q,\varepsilon)\ \text{относительно}}\ shp}$$

Следовательно, (q,ε) v ул. (q,ε) относительно
. шп

Случай q/ ε Q шп. Выделим все возможные формы хеджирования h.

Подслучай h = h ' · час . Пусть v ' и v Должно быть « = nw(ч « = nw(h »), поэтому v = v ' » · в = нв(ч) . применено следующее правило:

$$\frac{(q,\varepsilon)\ {}'_{v}\ \overset{\text{ул.}}{(p,\sigma)\ \text{относительно}}\ (p,\sigma)\ {}_{в}{}''\ \overset{\text{ул.}}{(q\ ',\varepsilon)\ \text{относительно}}}{(q,\varepsilon)\ \overset{\text{ул.}}{{}_{---}{}'_{''\,v\cdot v}\ (q\ ',\varepsilon)\ \text{относительно}}}$$

Поскольку v ' является вложенным словом, отсюда следует, что σ = ε. По гипотезе индукции esis применяется к вложенным словам v ' и v », получаем (q,ε) ′v ул. (p,ε) относительно shp и (p,ε) в " ул. (q ',ε)
shp
относительно . Следовательно, мы можем применить следующее правило:

$$\frac{q/\ \varepsilon\ Q_{\text{шп}} \quad (q,\varepsilon)\ {}'_{v}\ shp\ \overset{\text{ул.}}{(p,\varepsilon)\ \text{относительно}}\ (p,\varepsilon)\ {}_{в}{}''\ \overset{\text{ул.}}{(q\ ',\varepsilon)\ \text{относительно}}\ _{\text{шп}}}{(q,\varepsilon)\ \overset{\text{ул.}}{{}_{---}{}'_{''\,v\cdot v}\ (q\ ',\varepsilon)\ \text{относительно}}\ _{\text{шп}}}$$

Это показывает (q,ε) v ул. (q ',ε) по мере необходимости.

Подслучаи h = h 'shp или h = a или h = ε. Прямолинейно.

Для импликации справа налево предположим, что (q,ε) v ул. (q ',ε) относительно .
шп

Случай q Q шп. Тогда должно было быть применено следующее правило при q = q ':

$$\frac{q \quad Q_{\text{шп}} \qquad v = nw(H\Sigma)}{(q,\varepsilon)\ v \quad \overset{\text{ул.}}{(q,\varepsilon)\ \text{относительно}}\ shp}$$

Поскольку мы предполагаем, что v не имеет блокирующих частичных серий с , существует частичная серия на v с , которая начинается с q. Поскольку v — вложенное слово, любая частичная

Machine Translated by Google

run on v — это вложенное слово некоторого прогона, поэтому на изгороди h существует прогон , начинающийся с q. Это показывает существование некоторого состояния p такого, что q p относительно . Поскольку q Q shp, Лемма 6 показывает, что q = p, и, следовательно, q q относительно . Тогда предложение 8 доказывает что (q,ε) (q,ε) относительно , равного (q,ε) v (q ,ε) относительно , что и требовалось.

Случай q/ ε Q . Выделим все возможные формы изгороди h. шп.

Подслучай h = h · « v Должно = nw(v . Пусть « = nw(h »), поэтому v = v) и » · в = нв(ч). быть применено следующее правило:

$$\frac{q/\varepsilon Q_{\text{шп}} \quad (q,\varepsilon)\,{}'v\,(p,\sigma) \text{ относительно } (p,\sigma)\ \text{shp}^{\text{в}}_{\text{шп}}\ (q',\varepsilon) \text{ относительно}}{(q,\varepsilon)\ {}'v \cdot \text{в}\ \text{shp}^{\text{str}}\ (q',\varepsilon) \text{ относительно}}$$

Поскольку v является вложенным словом, отсюда следует, что σ = ε тоже. По гипотезе индукции esis применяется к вложенным словам v и v », получаем (q,ε) (p,ε) относительно и (p,ε) (q ,ε) относительно . Следовательно, мы можем применить следующее правило:

$$\frac{(q,\varepsilon)\ {}'v\ (p,\varepsilon) \text{ относительно } (p,\varepsilon)^{\text{в}} \quad (q',\varepsilon) \text{ относительно}}{(q,\varepsilon)\ {}'v \cdot v\ (q',\varepsilon) \text{ относительно}}$$

Это показывает (q,ε) v (q ,ε), как требуется.

Подслучаи h = h или h = a или h = ε. Простой.

На этом доказательства обоих направлений заканчиваются. □

Таким образом , оценщик потоковой передачи с проекцией субхеджирования для детерминированного SHA s A на хеджах h может быть получен путем вычисления отношения потокового перехода с проекцией субхеджирования для A из nw(h), начиная с начальной конфигурации. Это требует времени не более O(1) на букву nw(h), т. е. постоянное время на одно событие потока.

9.3. Потоковая сложность самого раннего членства с помощью Subhedge Projection

Используя потоковые оценщики для SHA , мы можем проверить самое раннее членство с проекцией субхеджа в потоковом режиме, запустив SHA A cgr(S)e(S) .

Следствие 3. Для любых полных dSHA (Σ , Q, , I, F), F FS Q, языка L = L(A) и схемы S = L(A[F/FS]), самое раннее членство с Полная проекция субхеджа для L относительно S может быть проверена в потоковом режиме для любого хеджа h S за время O(|1 |) на каждую непроецируемую букву nw(h) после времени предварительной обработки O(|A cgr(S)e(S) | + n 3d + n s), где d — номер (S). Требуемое пространство находится в разностных отношениях, а число безопасных подмножеств в состояниях Q e(S) КГР

O(глубина(h) + |A cgr(S)e(S) |).

Доказательство. По теореме 4 dSHA A cgr(S)e(S) является самым ранним, надежным и полным для субхеджевой проекции. Таким образом, согласно предложениям 8 и 9 мы можем получить потоковый тестер членства с проекцией субхеджа для хеджа h S , запустив потоковый вычислитель с проекцией субхеджирования A cgr(S)e(S) на вложенном слове nw(h). Поскольку этот SHA является детерминированным, оценка потоковой передачи может выполняться за время O(|1 |) на каждую букву nw(h) после времени предварительной обработки O(|A cgr(S)e(S) | + n 3d + н с).

□

Что касается прогноза «безопасно без изменений», экспоненциального времени предварительной обработки можно снова избежать, создав часть SHA A cgr(S)e(S), необходимую для оценки хеджирования на лету. Размер необходимой части находится в O(|h |). Следовательно, требования к пространству также могут

Machine Translated by Google

сократиться до O(|h|), что может быть слишком много для больших потоков nw(h). Однако необходимая часть A cgr(S)e(S) может быть намного меньше |h|. Кроме того, необходимо поддерживать стек размером глубины (h), но эта зависимость от h может быть приемлемой даже для больших потоков nw(h).

10. Монадические регулярные запросы и XPath

Далее мы рассмотрим проблему того, как отвечать на монадические регулярные запросы о хеджах самым ранним способом и с помощью проекции субхеджирования. Каждый монадический запрос определяет набор позиций для каждого входного хеджа. Например, XPath - запросы self::list[child::item] можно применить к хеджированию list. list . item . list. item , где он выбирает позицию первое самое верхнее дерево, помеченное списком , поскольку оно содержит дочернее дерево, помеченное элементом. Мы определяем позиции хеджей целыми числами. Например, выбранный элемент списка в хедже выше идентифицируется целым числом 2. Следовательно, набор ответов для приведенного выше запроса в хедже выше равен {2}. Обратите внимание, что монадические запросы не возвращают никакой другой информации о выбранных позициях, в отличие от семантики поддерева XPath в официальном стандарте W3C.

Ответить на этот запрос сложнее, чем проверить, соответствует ли фильтр XPath [self::list/child::item] корню первого поддерева хеджа, поскольку необходимо также построить набор ответов. Что касается потребления памяти, действующие кандидаты на ответ должны быть буферизованы для ответа на монадический запрос. Таким образом, потребление памяти потоковыми оценщиками монадических запросов может расти линейно в зависимости от размера потока и числа активных кандидатов.

10.1. Монадические регулярные запросы

Монадические запросы на хеджах в HΣ можно отождествить с языками аннотаций, где x/ ε Σ с хеджирования в произвольно фиксированной переменной выбора. Монадический запрос HΣ {x} на хеджах в HΣ затем может быть идентифицирован с языком всех аннотированных хеджей в HΣ {x} , в котором одна позиция получает x-аннотацию и эта позиция выбирается запросом.

Следовательно, регулярные монадические запросы к хеджам в HΣ могут определяться с помощью вложенных регулярных выражений в nRegExpΣ {x} или с помощью dSHA с той же сигнатурой. Например, обычный XPath -запрос self::list[child::item] можно определить с помощью следующего вложенного регулярного выражения:

список · x · · · элемент · · · · ,

где подобен подстановочному знаку, допускающему любое хеджирование в HΣ без x, т. е. для некоторой рекурсивной переменной z V:

= μz.(+aεΣ a + z) ·

Обратите внимание, что вложенное регулярное выражение для фильтра XPath [self::list/child::item] в nRegExpΣ можно получить из выражения запроса XPath self::list[child::item], удалив x. Кроме того, для любого монадического запроса к хеджам мы должны ограничить распознавание аннотированных хеджей языком запроса так, чтобы ровно одна позиция в каждом хедже была помечена знаком x. Это можно сделать путем пересечения со следующим вложенным регулярным выражением onex:

мкз. (TxT + T. z .T)

Предыдущий документ схемы для сопоставления с образцом теперь необходимо изменить на S = docx onex, где docx nRegExpΣ {x} подобен doc , но с сигнатурой Σ {x}.

Для обычного запроса XPath self::list[child::item] мы получаем dSHA, показанный на рисунке 22. Правила перехода со специальной буквой x указывают, где позиции могут быть привязаны к переменной выбора: x может быть привязан только в состоянии 2, то есть в поддеревьях, начинающихся с элемента списка . Тот же автомат можно использовать для определения схемы S = docx onex , используя конечные состояния схемы FS = {14, 20} вместо конечных состояний F = {20}. Действительно, мы построили автомат путем пересечения автомата, полученного из выражения XPath, с автоматами, вычисленными из регулярных выражений docx и one

Machine Translated by Google

схема. Обратите внимание, что 20 — это конечное состояние запроса (и схемы), а 14 — состояние приемника запроса (но не схемы).

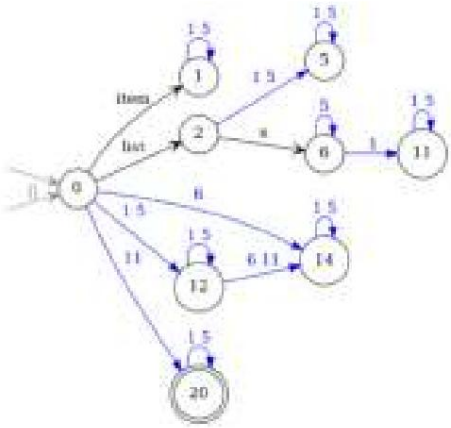


Рисунок 22. dSHA для запроса XPath self::list[child::item]. Позиция выбора обозначается x.

10.2. Самый ранний ответ на запрос с проекцией подхеджа.

Наивный алгоритм оценки запроса для хедж-автомата с сигнатурой Σ {x} получает входной хедж h НΣ , угадывает все возможные позиции того, куда вставить x в h, и проверяет для всех из них, является ли полученное аннотированное хеджирование принимается A. Менее наивным способом вставляется x только в ту позицию, где текущее состояние имеет исходящий переход x . Все это можно улучшить, если рассматриваемый хедж-автомат поддерживает состояния выбора, отклонения или проецирования. Во всех случаях алгоритм должен буферизовать всех кандидатов на привязку, которые еще не находятся в состоянии выбора или отклонения. В худшем случае ему придется буферизовать привязки ко всем позициям. Следовательно, пространство состояний вычислителя монадических запросов больше не является конечным.

Состояния выбора и отклонения можно идентифицировать путем адаптации самого раннего алгоритма ответа на запрос из [21]. Это можно сделать так, что можно получить SHA Ae(S) с сигнатурой Σ {x} в зависимости от схемы. Следует отметить, что в [21] используются автоматы dNWA, а не SHA , и схемы там игнорируются. Поэтому необходимые адаптации обсуждаются в Приложении A. Как получить эффективный алгоритм ответа на запросы в потоковом режиме из Ae(S) является основной темой [21]. Это базовый алгоритм, который мы расширим проекцией субхеджа для наших экспериментов.

Чтобы добавить проекцию конгруэнтности к самому раннему алгоритму ответа на запрос, нам нужно запустить автомат A cgr (S). Самая ранняя проекция конгруэнтности для запроса XPath e(S) self::list[child::item] показана на рисунке 23. Он связывает x в состоянии 2D1 после чтения элемента списка верхнего уровня , а затем проверяет его существование элемента дочернего элемента , переходящего в состояние выбора 1D3, как только он будет найден. Все остальные дочерние элементы элемента списка верхнего уровня проецируются в состоянии 2D3. Для добавления самого раннего запроса, отвечающего на проекцию без вместо A cgr нам нужно запустить π изменений, автомат A e(S) . Мы не приводим подробностей, но заявляем, что для данного запроса e(S) проекционная мощность обоих автоматов одинакова.

Machine Translated by Google

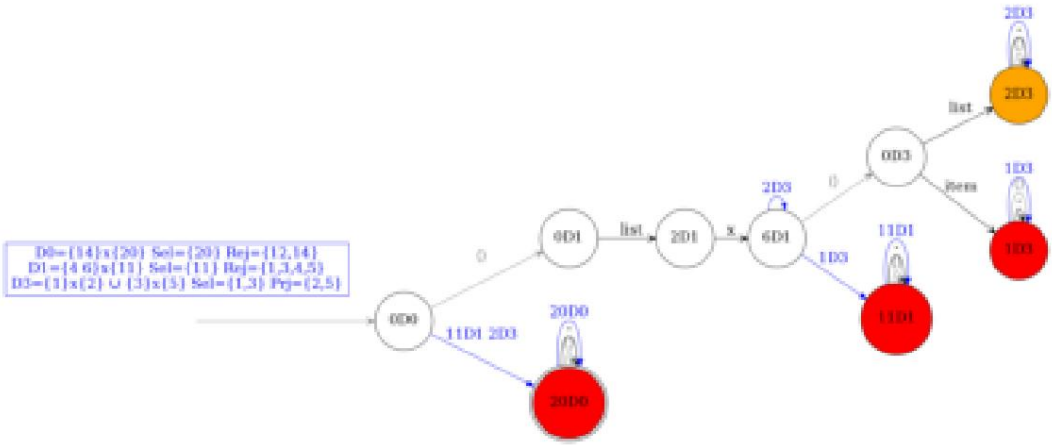


Рисунок 23. Самая ранняя конгруэнтная проекция A cgr(doc onex)e(doc onex) для dSHA A на рисунке 22 с FS = {14, 20}. Он определяет запрос XPath self::list[child::item].

К счастью, теоремы о надежности и полноте для самых ранних проверок членства с проекцией подхеджирования влияют только на конструкции этих SHA , но не на то, как их следует оценивать. Однако есть еще одна проблема: префиксы вложенных слов не могут считаться нерелевантными для последующего субхеджа, если там может быть связана переменная выбора x , даже если принятие не зависит от того, где она будет привязана. Это связано с тем, что позиция привязки должна быть возвращена любым алгоритмом ответа на запрос.

Определение 15. Префикс v назовем связующим, нерелевантным для L и S , если не существует хеджа h, содержащего x и суффикс w, такого, что v · w nw(S) и v · nw(h) · w nw(L).

Наш оценщик проецирования субхеджирования может проецировать только нерелевантные префиксы субхеджирования, которые также не имеют значения для привязки. Учитывая dSHA A и набор конечных состояний схемы FS, можно решить, является ли префикс нерелевантным для субхеджа, если текущее состояние q принимает хеджи, содержащие x , но может получить доступ к некоторому другому состоянию, которое небезопасно для отклонения и разрешает x-привязки. . В основном журнале данных мы можем различать нерелевантные состояния q, налагая следующие правила для всех q,q′ Q:

привязка_irrelev(q):-state(q), а не привязка_relev(q). привязка_relev(q) :- не привязка_x(q), acc(q,q'), привязка_x(q'), а не rej(q').

Обратите внимание, что в нашем примере запроса XPath self ::list[child::item] состояние 2D3 dSHA A cgr(doc onex)e(doc onex) на рисунке 23 не имеет значения привязки, поскольку x должен быть связан на элементе списка верхнего уровня в состоянии 2D1, поэтому его нельзя привязать к какому-либо элементу списка ниже.

11. Эксперименты с потоковой оценкой XPath Мы

экспериментально сравниваем три потоковых оценщика для dSHA, для самого раннего монадического ответа на запрос, без проекции подхеджирования, с проекцией без изменений и с проекцией конгруэнтности.

11.1. Тестирование dSHA для запросов XPath

Мы начнем с dSHA , определяющего монадические запросы для обычных запросов XPath A1-A8 из теста XPathMark [7] , которые приведены в таблице 1. Эти запросы XPath демонстрируют большинство функций обычного фрагмента XPath. В Таблицу 2 мы добавили еще 14 запросов пути XPath, которые также оказались полезными для тестирования.

Machine Translated by Google

Таблица 1. Обычные XPath-запросы A1-A8 из коллекции XPathMark.

Идентификатор запроса	XPath-выражение
A1	/сайт/закрытые_аукционы/закрытый_аукцион/аннотация/описание/текст/ключевое слово
A2	//закрытый_аукцион//ключевое слово
A3	/сайт/закрытые_аукционы/закрытый_аукцион//ключевое слово
A4	/сайт/закрытые_аукционы/закрытый_аукцион[аннотация/описание/текст/ключевое слово]/дата
A5	/сайт/закрытые_аукционы/закрытый_аукцион [потомок::ключевое слово]/дата
A6	/сайт/люди/человек[профиль/пол и профиль/возраст]/имя
A7	/сайт/люди/человек[телефон или домашняя страница]/имя
A8	/site/people/person[адрес и (телефон или домашняя страница) и (кредитная карта или профиль)]/имя

Таблица 2. Дополнительные регулярные запросы XPath для документов XPathMark.

A0	/сайт
A1_0a	/сайт/*
A1_0b	/сайт/@*
A1_0c	/сайт//@*
A1_1a	//участник торгов/personref[starts-with(@person,'person0')]
A1_1d	//участник торгов/personref[@person='person0']
A1_2	//человек
A1_3	/сайт/регионы/африка//@*
A1_4	/сайт/регионы/африка/*
A1_5	/сайт/регионы/*
A1_6	//закрытый_аукцион/аннотация//ключевое слово
A2_1	//закрытый_аукцион[потомок::ключевое слово]
A4_0	/сайт/закрытые_аукционы/закрытый_аукцион[аннотация]/дата
A4_1	/сайт[open_auctions]/closed_auctions

Детерминированные SHA для A1-A8 были предоставлены ранее в [21]. Для другого XPath запросы мы скомпилировали их в dSHA с помощью вложенного регулярного выражения.

Чтобы создать входные dSHA для наших оценщиков, мы пересекаем эти автоматы. с dSHA для документа схемы $_{x}$ onex, где doc′ $_{x}$ это схема, которую мы используем для хеджирования кодировки реальных XML- документов с аннотациями x . Тем самым мы смогли идентифицировать в окончательном варианте схемы указано FS. Мы свернули и доработали результат. В таблице 3 указаны размеры входных dSHA для наших оценщиков, полученных с помощью описанной выше процедуры. Для каждого dSHA A, размер задается двумя числами size(n), первое для общего размера, второе для количество штатов n.

Для входного dSHA A каждого запроса мы статически вычислили все dSHA s. A cgr(S)e(S) при использовании необходимых частей алгоритма детерминизации из [16]. размер полученных dSHA и количество d разностных отношений указаны в таблице 3. слишком. Самый большой размер — 2091(504) для входного dSHA для A8. Наибольшее количество различий отношения d = 24 также получается для этого запроса. Так что действительно размеры этих автоматов намного

Machine Translated by Google

меньше, чем можно было бы ожидать от конструкции, которая в худшем случае экспоненциальна. случай. Однако единственный запрос, для которого мы не смогли получить соответствующий детерминированный запрос самая ранняя конгруэнтная проекция A cgr(S)e(S) равна A5.

Время расчета самой ранней проекции конгруэнтности заняло от 2,3 до 26 с.

Таблица 3. Общий размер и количество n состояний входа dSHA A для запроса со схемой-final состояния FS, общий размер и количество состояний SHA A cgr(S)e(S), полученных путем самого раннего сравнения проекция, а число d – ее разностные отношения.

Запрос dSHA A Размер(n) Размер(#State) Отн. г	dSHA ID	#Diff- Query dSHA A ID	dSHA	#Diff-Size(n) Size(#State) Отн. д	
A1 482(68)	1296(324)	16	A1_0c 230(43)	238(62)	6
A2 224(42)	316(82)	7	A1_1a 305(54)	384(101)	8
A3 320(53)	662(156)	10	A1_1д 305(54)	382(101)	8
A4 629(74)	1651(404)	18	A1_2 194(39)	152(42)	5
A5 438(63)	1226(269)	13	A1_3 318(53)	672(159)	10
A6 675(76)	2090(500)	22	A1_4 312(52)	479(132)	10
A7 394(59)	728(184)	12	A1_5 266(47)	293(84)	8
A8 648(79)	2091(504)	24	A1_6 265(47)	588(142)	9
A0 203(40)	158(44)	5	A2_1 232(42)	295(78)	7
A1_0a 224(42)	145(44)	6	A4_0 392(59)	719(184)	12
A1_0б 203(39)	68(23)	5	A4_1 292(49)	287(78)	8

Отметим, что мы еще не вычислили полные dSHA статически для безопасной проекции без изменений A snc, чистой конгруэнтной проекции A cgr(S) и самого раннего ответа на запрос. Ae(C) . Мы считаем, что эти автоматы могут стать больше, чем A cgr(S)e(S) , который мы построили за один раз.

11.2. Инструмент оценки потоковой передачи: AStream

Мы используем и разрабатываем инструмент AStream для ответа на монадические запросы dSHA. в потоках XML с ограничениями схемы. Версия 1.01 AStream была представлена в [21] поддерживает самый ранний ответ на запрос без проекции. Учитывая dSHA A, определяющий монадический запрос, набор конечных состояний схемы FS и поток XML w, он создает на лету необходима часть SHA Ae(S) при анализе w и оценке A в хедж-кодировании w.

Для версии настоящей статьи для конференции FCT [24] мы расширили AStream с помощью безопасный прогноз без изменений. Это приводит к версии AStream 2.01. Он строит самый ранний прогноз безопасности и отсутствия изменений $dSHA_{e(C)}^{CHC}$ на лету при оценке определенного монадического запроса по A на хедже, кодирующем XML- поток w. Проекцию субхеджирования можно включить или off, чтобы сравнить обе версии без дальнейших различий в реализации.

Для данной версии журнала мы добавили самую раннюю проекцию конгруэнтности и интегрировали ее в AStream, что привело к созданию AStream 3.0.

AStream 3.0 отличается от двух предыдущих версий тем, что dSHA A cgr e(C) построен статически и целиком, независимо от входного хеджирования. Затем мы используем общий самый ранний потоковый оценщик для SHA , который отклоняет в состояниях отклонения, выбирает при выборе штатах и проектах во всех штатах проекции субхеджа. Этот оценщик также можно запустить с $A_{e(C)}^{CHC}$ или Ae(S) , если они не становятся слишком большими.

Следует отметить, что A cgr e(C) оказалось очень маленьким для всего нашего теста, с другими dSHA A $_{e(C)}^{CHC}$ и Ae(S) рискуют стать больше. Как уже говорилось ранее, мы не построить эти dSHA для наших контрольных запросов. Вот почему мы продолжали бежать

Machine Translated by Google

самая ранняя потоковая передача с проекцией без изменений с помощью AStream 2.01, тогда как мы использовали AStream 3.0 для самой ранней потоковой передачи с проекцией конгруэнтности.

Все версии AStream используют Java abc-Datalog для вычисления наименее фиксированных точек программ Datalog восходящим способом. Программы регистрации данных необходимы для всех логических рассуждений: для вычисления подмножеств состояний, которые безопасны для отклонения, выбора или проецирования субхеджирования на всех уровнях. Кроме того, отношения разностей вычисляются на основе журнала данных. Обратите внимание , что самая ранняя оценка запроса «на лету» требует запуска журнала данных во время оценки запроса, тогда как при статическом подходе для полного построения dSHA журнал данных необходим только во время предварительной обработки во время построения автомата.

11.3. Меры оценки

Мы хотим измерить время запуска трех оценщиков потоковых запросов для всех dSHA, полученные из выражений XPath в коллекции в таблицах 1 и 2.

Одним из преимуществ теста XPathMark является то, что он поставляется с генератором XML -документов, к которым можно применять запросы и масштабировать их по размеру. Мы создали XML- документ размером 1,1 ГБ, который достаточно велик, чтобы сделать потоковую передачу обязательной. Наши эксперименты показывают, что эффективность оценки запроса растет линейно с размером непроецируемой части XML- документа. Это справедливо для каждого из трех наших оценщиков. Таким образом, измерение времени оценщика на XML -документах других размеров не даст никаких новых результатов, за исключением того, что потребление памяти также остается умеренным.

Выигрыш во времени проецирования для запроса Q — это процент времени, сэкономленный проекцией во время оценки запроса при игнорировании чистого времени синтаксического анализа документа tparse секунд:

$$\text{time-gain}Q = 100 \quad (1 \quad (\text{tnoproject}Q \quad \text{tparse})/(\text{tproject}Q \quad \text{tparse})).$$

Мы можем измерить выигрыш во времени для самой ранней проекции без изменений с помощью AStream 2.01 и для самой ранней конгруэнтной проекции с помощью AStream 3.0, поскольку проекцию можно отключить в обеих из них. Недостатком выигрыша во времени является то, что он зависит от деталей реализации.

Прирост событий является лучшим показателем мощности проекции. Анализатор отправляет поток , необходимых событий в SHA , игнорируя части входного документа, которые могут быть спроецированы. Для этого автомат всегда должен информировать его о том, какие события можно спроецировать в текущем состоянии. Прирост событий проекции event-gainQ тогда представляет собой процент событий, которые признаны нерелевантными и, следовательно, больше не создаются оценщиком прогнозирования для запроса Q. Можно было бы ожидать, что выигрыш во времени всегда меньше, чем выигрыш по событиям.

$$\text{time-gain}Q \quad \text{event-gain}Q$$

Действительно, это подтвердят наши эксперименты. Мы считаем, что эти расхождения между этими двумя показателями указывают на то, сколько еще остается возможностей для оптимизации внедрения оценщика. Таким образом, мы можем заметить, что некоторое пространство для дальнейшей оптимизации AStream 3.0 все еще остается.

11.4. Эксперименты

Мы использовали Java- интерфейс XMLStreamReader версии 1.0 из пакета javax.xml.stream для анализа и потоковой передачи XML- файлов в Scala. Для анализа XML- документа размером 1,1 ГБ потребовалось tparse = 15 с, тогда как его запрос без проецирования занял у нас в среднем tnoprojectavg = 752 с, при этом время варьировалось от 600 до 900 с. Таким образом, среднее время обработки 1 МБ составило 0,72 с. Узнав это, мы спрогнозировали ожидаемое время оценки исходя из размера непроецируемой части XML - документа. Это происходит следующим образом:

$$\text{event-gain}Q \quad 1100 \quad 0{,}72 \text{ с} + \text{tparse с}.$$

Без проецирования синтаксический анализатор генерирует 1 012 018 728 событий для XML- документа размером 1,1 ГБ. Это базовый уровень для вычисления event-gainQ для всех запросов Q.

Machine Translated by Google

Мы запускали AStream со Scala v2.13.3 на машине с операционной системой Ubuntu.
20.04.06 ЛТС (64-бит). Это машина Dell Precision-7750, оснащенная
Процессор Intel® Core™ i7-10875H с частотой 2,30 ГГц × 16 и 32 ГБ ОЗУ.

11.4.1. Самая ранняя конгруэнтная проекция

Мы представляем показатели нашего оценщика с самой ранней проекцией конгруэнтности в
Таблица 4. Прирост событий высокий и варьируется от 75,7% для A1_0c до 100% для A0, A1_0a, A1_0b, A1_4, A1_5 и A4_1. Прирост событий для запросов без оси-потомка выше
98,2%. Для этих запросов все поддеревья, которые не находятся на основном пути запроса, могут быть
спроецировано прочь. В частности, необходимо проверять только поддеревья до фиксированной глубины. Эти
это запросы A1, A4, A6-A8 и A4_0 и запросы с приростом событий 100%, перечисленные выше.
Выигрыш времени для всех этих запросов немного меньше, чем выигрыш по событиям, но все же выше.
98,2%.

Таблица 4. Выигрыш во времени и прирост событий по самой ранней проекции сравнения с AStream3.01 на 1,1 ГБ
Поток XPathMark.

Запрос	#Ans-	Время	Время	Событие	Запрос	#Ans-	Время	Время	Событие
		Мы в выигрыше	в выигрыше	Прирост			Вер		Прирост
A1	38 267	29,3	98,1%	98,9%	A1_0c	3 600 816	226,6	72,9%	75,7%
A2	117 389	171,8	77,9%	81,1%	A1_1a		2	187,8	78,1% 80,3%
A3	117 389	33,4	97,4%	97,8%	A1_1d		2	217,3	74,8% 80,3%
A4	24 959	33,2	97,8%	98,9%	A1_2	1 062 965	238,2	68,2%	76,0%
A5	50 186				A1_3	25074		16,2	99,1% 99,8%
A6	30 307	45,9	96,9%	98,2%	A1_4	5170		17,5	99,7% 100,0%
A7	179 889	30,5	98,0%	98,7%	A1_5	6		14,3	100,0% 100,0%
A8	67 688	37,8	97,3%	98,7%	A1_6	117 389		188,7	75,4% 81,1%
A0	1	21,3	99,1%	100,0%	A2_1	50 186		199,7	76,5% 81,1%
A1_0a	6	15,2	99,9%	100,0%	A4_0	91 650		21,8	99,0% 99,3%
A1_0b	0	0,0	100,0%	100,0%	A4_1	1		12,0	100,0% 100,0%

Далее мы рассмотрим второй тип запросов, имеющих ось-потомок, а именно A2,
A1_0c, A1_1a, A1_1d, A1_2, A1_6 и A2_1. Интуитивно понятно, что в
в этих случаях, поскольку проекция субхеджа с нисходящей осью не может напрямую исключить
целое поддерево под данным элементом; все дочерние элементы должны быть проверены.
Например, запрос A1_0c, равный /site//@*, должен выбрать все атрибуты всех элементов.
под сайтом корневого элемента , требуя проверки каждого элемента XML в документе,
на наличие атрибута, кроме корневого элемента сайта.

Меньший прирост событий, указанный для вышеупомянутых запросов, соответствует приведенному выше
ожидание, показывая проценты в диапазоне от 75,7% до 81,1%. Кроме того, эти запросы демонстрируют
больший разрыв между выигрышем по времени и выигрышем по событиям, который колеблется от 2,2% до 5,7% и будет
подлежат дальнейшей оптимизации. Стоит отметить, что некоторые из этих запросов объединяют
несколько функций одновременно, например запросы A1_1a и A1_1d, имеющие ось-потомок,
фильтр с атрибутами и сравнение строк.

Единственным исключением из этой тенденции для запросов, имеющих ось потомков, являются A3 и
A1_3, с соответствующим приростом событий 97,8% и 99,8%. Причиной такого высокого выигрыша является
что эти запросы начинаются с корня, с путем дочерней оси глубины 3, прежде чем запрашивать
ось потомков. Этот первоначальный путь позволяет исключить и проецировать большинство элементов.
под сайтом , которые не соответствуют указанному пути со всеми своими поддеревьями.

Machine Translated by Google

11.4.2. Самый ранний прогноз «безопасного без изменений»

Мы сравниваем самый ранний прогноз «безопасного отсутствия изменений» с самым ранним конгруэнтным прогнозом. в таблице 5. Для этого ограничимся запросами A1–A8 из XPathMark в
Таблица 1.

Запросы XPath A1 , A4, A6, A7 и A8 не имеют дочерних осей. Время
Прирост для прогноза «безопасно без изменений» по этим запросам составляет 92,3–98,9%. Для самого раннего Согласно прогнозу конгруэнтности, выигрыш во времени был лучше для A1, A4 и A6, по крайней мере, на 96,6%. Для A7 и A7 выигрыш во времени при прогнозировании безопасного отсутствия изменений близок к выигрышу в случае события конгруэнтной проекции, но выигрыш во времени при конгруэнтной проекции на 1,5% ниже. Итак, мы надеюсь, что оптимизированная версия проекции конгруэнтности сможет превзойти вариант «безопасно без изменений» проекция во всех случаях.

Запросы XPath с осью потомков: A2, A3 и A5. Для A2 и A3 в
В частности, выигрыш во времени при прогнозировании «безопасного отсутствия изменений» очень низок, даже не 10,9%, в то время как конгруэнтный прогноз дает не менее 77,9%. Мы считаем, что это ошибка в нашем текущем реализация прогноза «безопасного отсутствия изменений», который может запрещать прогнозирование атрибуты и текстовые узлы для этих запросов (но не для остальных). Конгруэнтная проекция на A2 выигрывает 77,9% времени и 81,1% событий. Это намного лучше, но все еще далеко от мощность проекции для запросов без дочерних осей. Для A3 выигрыш в проекции конгруэнтности 97,4% времени. Это намного лучше, чем для A2, поскольку ось потомков в A3 находится в конце, тогда как для A2 он находится в начале пути. Но даже для A2 конгруэнтная проекция очень полезно.

Таблица 5. Выигрыш во времени для самых ранних прогнозов без изменений и самого раннего конгруэнтного прогноза, а также выигрыш для событий самой ранней проекции совпадения.

Время запроса (с)	Время (с)		Выигрыш по времени		Выигрыш по событию
идентификатор	СНС	Конгресс	СНС	Конгресс	Конгресс
A1	72,8	29,3	92,3%	98,1%	98,9%
A2	664,6	171,8	8,5%	77,9%	81,1%
A3	666,1	33,4	10,9%	97,4%	97,8%
A4	78,5	33,2	92,4%	97,8%	98,9%
A5	77,7		92,3%		
A6	65,2	45,9	95,1%	96,9%	98,2%
A7	24,6	30,5	98,8%	98,0%	98,7%
A8	24,8	37,5	98,9%	97,3%	98,7%

11.4.3. Сравнение с внешними инструментами

QuiXPath оказался наиболее эффективным внешним инструментом с большим охватом для регулярных Оценка XPath для потоков XML [4]. Поэтому сравниваем эффективность сравнения проекцию на QuiXPath в Таблице 6 и, следовательно, за счет транзитивности ко многим альтернативам инструменты. Мы отмечаем, что QuiXPath выполняет ранний ответ на запрос путем компиляции, возможно, недетерминированные NWA с состояниями выбора, которые не всегда являются самыми ранними. Кроме при этом его эффективность основывается как на проекции поддерева, так и на проекции потомков.

Machine Translated by Google

Таблица 6. Сравнение времени без разбора в секундах для запросов XPathMark с QuiXPath и самая ранняя конгруэнтная проекция.

Фракт без разбора запроса.	Фракт без разбора запроса.					
идентификатор	Время (с) Время (с)		идентификатор	Время (с)	Время (с)	
	Конференция QuiXPath. Прж.			Конференция QuiXPath. Прж.		
A1	11,0	14.3	1,3 A2	11.4	156,8	13,8
A4	11,6	18,2	1,6 A3	11,5	18,4	1,6
A6	10,7	30,9	2,9 A5	12,0		
A7	8,6	15,5	1,8			
A8	8,8	22,5	2.6			

В экспериментах с QuiXPath в [4] используется время, свободное от синтаксического анализа. Поэтому мы делаем то же самое для проекции конгруэнтности, уменьшив измеренное общее время на 15 с. время разбора.

По сравнению со временем без синтаксического анализа для запросов XPath без дочерней оси, самый ранний конгруэнтный прогноз демонстрирует значительные улучшения. В среднем, наша проекция медленнее QuiXPath лишь в 1,3–2,9 раза. Это показывает, что наш реализация уже близка к тому, чтобы конкурировать с существующими инструментами потоковой передачи XML , будучи единственным, кто гарантирует самый ранний ответ на запрос.

Для запроса A2 с осью-потомком проекция конгруэнтности происходит в 13,8 раза медленнее. чем QuiXPath. Причина в том, что QuiXPath поддерживает проекцию потомков, тогда как конгруэнтная проекция касается только субхеджа. Для запроса A3 с дочерней осью в конце: конгруэнтная проекция отстает всего в 1,6 раза, поэтому проекция потомка кажется меньше актуально здесь.

12. Родственные модели автоматов.

Было предложено большое количество альтернативных представлений автоматов для деревьев, лесов и живых изгородей. предложено в литературе. Мы сравниваем их с SHA и SHA в следующем подразделе. и обсудить последствия.

12.1. SHA против древесных, лесных и хедж-автоматов

Стандартные хедж-автоматы. Стандартные хедж-автоматы [9,10,32,33] работают на помеченных хеджирование осуществляется снизу вверх, аналогично SHA. И у них такая же выразительность, но горизонтальные языки определяются по-другому, что приводит к проблематичному представлению о детерминизм [34]. По этой причине однозначная минимизация невозможна для детерминированного стандарта. хедж-автоматы. Тем не менее, они имеют ту же выразительность, что и SHA, когда ограничены маркированные живые изгороди.

Стандартные древовидные автоматы и минимизация SHA. Синтаксически любой SHA A является стандартным. древовидный автомат по следующей ранжированной сигнатуре:

- унарный символ для любой буквы a Σ ;
- двоичный символ @;
- константа .

Семантически идеального общего соответствия не существует. Только для подклассов идеальный соответствие может быть составлено посредством двоичного кодирования. Это было показано в [8] для подкласса dSHA с I = . В [35] это можно было показать и для подкласса многомодульных дША. Это приводит к уникальным результатам минимизации для обоих подклассов детерминированных SHA.

Лесные автоматы Боянчика. Такой же выразительностью обладают и стандартные хедж-автоматы. как лесные автоматы Боянчика (см. раздел 3.3 в [36]). Они основаны на идее моноиды перехода, а не идея правил перехода, таких как SHA или конечные

Machine Translated by Google

государственные автоматы или лесные автоматы Неймана и Зейдля. Как следствие, существует экспоненциальная разница в краткости между SHA и лесными автоматами Боянчика [35].

12.2. SHA и SHA

Мы начинаем объяснять, почему SHA имеют такую же выразительность, как и SHA. По сути, это та же самая причина, по которой NWA имеют ту же выразительность, что и SHA (см., например, [8]). В более простом направлении мы уже утверждали, что любой SHA A = (Σ , Q, , I, F) может быть преобразован в

мы можем преобразовать любой dSHA A в эквивалентный, исходящую ликвидацию SHA . И наоборот, недетерминизм следующим образом:

$$Q^{\text{элим}}_{\text{я}} = Я \times Q \qquad \frac{q \xrightarrow{q} {}' в}{(q, q') \xrightarrow{\text{elim}} в}$$

$$Ф^{\text{исключаю}} = Ф \times Q$$

$$\frac{q \xrightarrow{a} {}' в \quad r \quad Q}{q\,(r, q) \xrightarrow{a} (r, q') \xrightarrow{\text{elim}} в} \qquad \frac{д@п \quad д \xrightarrow{} {}' в \quad r \quad Q}{(r, q)@(q, p) \quad (r, q) \xrightarrow{} {}' \xrightarrow{\text{elim}} в}$$

Предложение 10. L(A) = L(elim (A)).

Конструкция аналогична конвертации NWA в SHA [8] или хеджированию автоматы [26]. Доказательства корректности этих компиляторов стандартны.

Уникальная минимизация. Уникальная минимизация не работает для dSHA , как обычно для детерминированных многоходовых автоматов. Это происходит даже для детерминированных двусторонних конечных автоматов на словах. Напротив, подкласс dSHA с ограничением, что начальное состояние и начальное состояние дерева совпадают, обладает уникальной минимизацией [8].

Лесные автоматы Ноймана и Зейдля. Известно также, что стандартные хедж-автоматы обладают той же выразительностью, что и автоматы с выталкивающим лесом Неймана и Зейдля [25]. Последнее можно идентифицировать с помощью SHA для помеченных хеджей. Для сопоставления лесных автоматов с понижением уровня со стандартными хедж-автоматами необходимо нисходящее исключение, как и для сопоставления SHA с SHA.

12.3. SHA против NWA

Потоковый вычислитель для SHA с помощью машины с видимым выталкиванием также может быть получен путем компиляции SHA в автомат с вложенными словами (NWA) [11] , чей потоковый вычислитель задается машиной с видимым выталкиванием [12], ранее известной как входной управляемые автоматы [13–15].

Определение 16. Вложенный словесный автомат (NWA) — это набор (Σ , Q, Γ , , I, F), где Σ , Γ , Q — множества, I, F Q и = ((a)aεΣ , Q × Q,) содержит следующие соотношения : Q × (Γ × Q) и , Q × Γ × Q. NWA является детерминированным или, что то же самое, dNWA , если I содержит не более одного элемента, и все приведенные выше отношения являются частичными функциями.

Элементы Γ называются символами стека. Правила перехода в снова имеют три формы: внутренние правила q a q ' γ в , как и для SHA, правила открытия q q ' в , если (д) = (д ', γ), и правила закрытия q γ q ' в , если (q, γ) = q '.

Потоковые оценщики для NWA. Потоковый оценщик для NWA можно рассматривать как понижающую машину для оценки вложенных слов хеджирования в потоковом режиме. Конфигурация автомата с выталкиванием представляет собой пару из K = Q × Γ , содержащуюсостояние и стек.

Machine Translated by Google

Для любого слова v Σ ^ ′ мы определяем отношение потокового перехода ул. K × K, такой, что для всех q, q ' v Q и σ Γ, применимо следующее:

$$\frac{\text{истинный}}{(q,\sigma) \xrightarrow[(q,\sigma)\text{ относительно}]{\varepsilon\ \text{ул}} (q,\sigma)} \qquad \frac{(q,\sigma) \xrightarrow{\text{ул.}} (q',\sigma) \quad (q',\sigma) \xrightarrow{v\ \text{ул.}} (q'',\sigma)\ \text{относительно}}{(q,\sigma) \xrightarrow[(q,\sigma)\text{ относительно}]{v\text{-}v\ \text{ул}} (q'',\sigma)\ \text{относительно}}$$

$$\frac{q\ a\ q' \in B}{q\ a \xrightarrow[q'\ \text{относительно}]{\text{ул}}} \qquad \frac{qqy\ ' \in B}{(q,\sigma) \xrightarrow[(q',\sigma)\ \text{относительно}]{\text{ул.}}} \qquad \frac{y \quad qq\ ' \in B}{(q,\sigma\cdot\gamma) \xrightarrow[(q',\sigma)\ \text{относительно}]{\text{ул.}}}$$

От SHA до NWA. Для любого SHA A = (Σ , Q, , I, F) мы можем определить нва = правила нва , я нва , F nwa) при сохранении детерминизма, такой, что Γ = Q, а перехода NWA A (Σ , Q, Γ ,) для всех a Σ и q, p Q :

$$\frac{q\ a \quad ' \in B}{qq\ a \quad q' \in B}\ \text{нва} \qquad \frac{q \quad ' \in B}{q \quad q\ q \quad q' \in B}\ \text{нва} \qquad \frac{q@p \quad q \quad ' \in B}{p \quad q \quad q' \in B}\ \text{нва}$$

Например, dNWA A нва dSHA A на рисунке 9 , полученная из dSHA для [self ::list/child::item] из введения показан на рисунке 24. dNWA A dSHA A на нва рисунке 21 показан на рисунке 25.

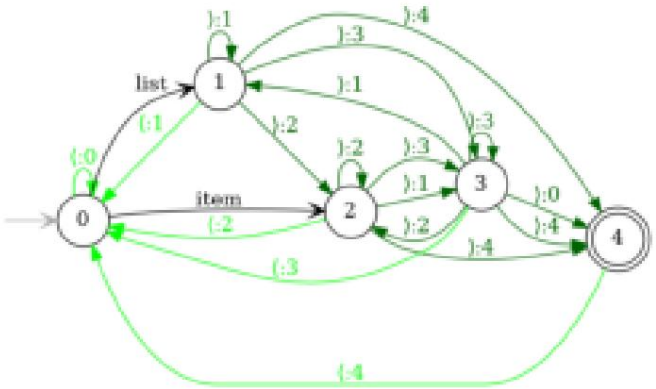
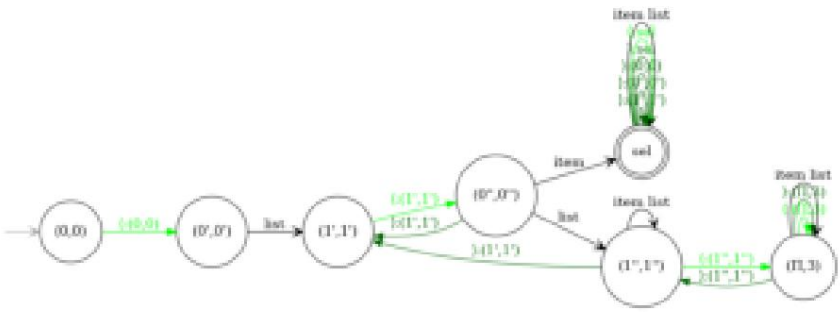


Рисунок 24. dNWA и dSHA нва для dSHA A на рисунке 9 , полученного применением оператора down к на рисунке 4 для фильтра [self::list/child::item] .



рисунке 25. dNWA (A snc) nwa e(doc) для самого раннего dSHA с прогнозом «безопасно без изменений» на Рисунок 21.

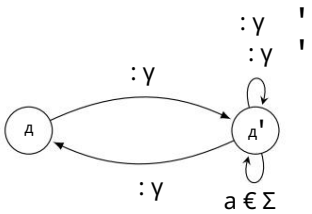
Machine Translated by Google

Лемма 21. L(A nwa) = L(A).

Запуски SHA A и NWA A нва можно идентифицировать.

Прогноз для NWA. Проектные оценщики CBA были предложены в контексте проектирования CBA [4]. Они основаны на следующем понятии состояний нерелевантных поддеревьев для NWA.

Определение 17 (вариант определения 3 из [4]). Мы называем состояние q NWA состоянием нерелевантных поддеревьев , если существуют два разных символа стека γ, γ, такие, что переходы ', показанные справа, существуют, но нет дальнейших открывающих переходов с γ, никаких дальнейших переходов с γ и никакого дальнейшего закрывающего перехода. в q выскакивает γ. В этом случае мы пишем q i-treeΣ \ .



Лемма 22. Любое состояние проекции субхеджа полного SHA A является состоянием нерелевантных поддеревьев Anwa . .

Затем в [4] было показано, как отобразить NWA с подмножеством Q состояний нерелевантных поддеревьев shp в проецируемое NWA, чья потоковая семантика дает потоковый вычислитель с проекцией субхеджа. Представление проекции субхеджирования для SHA без прохождения через NWA дает тот же результат более прямым способом.

Следует также отметить, что проецирование NWA поддерживает проекцию потомков, помимо проекции субхеджа. Что касается SHA, это оставлено на будущее.

13. Выводы и будущая работа

Мы ввели понятие нерелевантных субхеджей для членства в хедж-языке при ограничениях схемы. Мы разработали алгоритмы с проекцией субхеджирования, которые перескакивают нерелевантные субхеджи для проверки принадлежности к обычным языкам хеджирования, определенным dSHA . Все наши алгоритмы могут работать как в режиме памяти, так и в потоковом режиме. Трудность заключалась в том, как передать необходимую информацию о конечном состоянии для проекции поддерева сверху вниз, учитывая, что SHA работают снизу вверх. Мы решили ее, используя компиляторы от SHA к нисходящим SHA.

Этот первый компилятор распространяет информацию о безопасности относительно неизменяющихся состояний. Второй компилятор, который, как мы доказываем, является полным для проекции субхеджа, распространяет разностные отношения. Затем мы объединили проекцию подхеджирования с самым ранним тестированием членства и объединили его с самым ранним монадическим ответом на запрос. Мы интегрировали как «безопасность без изменений», так и полную конгруэнтность в нашем инструменте AStream и экспериментально подтвердили их полезность: мы показали, что они действительно могут ускорить работу инструмента AStream при ответе на обычные запросы XPath к потокам XML . Более того, проекция конгруэнтности показала гораздо более высокие показатели выигрыша от проекции событий, чем проекция без изменений, в диапазоне от 75,7% до 100% для класса запросов, не имеющих оси потомков, что со временем стало конкурентоспособным с лучшими существующими инструментами потоковой передачи для обычного XPath. запросы без рекурсивной оси.

Остающийся открытым теоретический и практический вопрос заключается в том, как получить проекцию потомков для SHA, что необходимо для ускорения оценки запросов XPath с осью потомков . Что касается реализации и экспериментов, было бы неплохо также изучить версию наших алгоритмов подхеджирования в памяти. Наконец, было бы неплохо оптимизировать

Machine Translated by Google

потоковую передачу инструментов ответа на самые ранние запросы до тех пор, пока они не превзойдут лучшие инструменты потоковой передачи во всех случаях, в то же время оказываясь лучше для запросов, где другие инструменты не являются самыми ранними.

Вклад авторов: написание — первоначальный вариант рукописи, AAS и JN. Все авторы прочитали опубликованную версию рукописи и согласились с ней.

Финансирование: Данное исследование не получило внешнего финансирования.

Заявление о доступности данных: набор данных доступен по запросу авторов.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Приложение А. Самое раннее членство Мы

адаптировали самый ранний тестер членства для dSHA из [21] так, чтобы он компилировался в SHA вместо NWA и учитывал схемы.

Приложение А.1. Безопасность схемы — проблема включения. Идея

тестирования определенного членства и определенного нечленства состоит в том, чтобы учитывать, что состояния могут безопасно достигать конечных состояний, за исключением случаев выхода из схемы. Пусть A = (Σ , Q, , I, F) — SHA со схемой S, такая что существует F FS Q c S = L(A[F/FS]). Заметим , что L(A) S HΣ , в частности. Набор состояний, в которых безопасно достичь подмножества P Q для всех хеджей, не выходящих за пределы схемы S, выглядит следующим образом:

$$\text{безопасный}_{C}(P) = \text{безопасный}(P \setminus (Q \setminus FS)).$$

Если A является детерминированным и полным по схеме для S, то безопасность состояния A по отношению к схеме S является проблемой включения.

Лемма A1. Пусть S = L(A[F/FS]). Если A детерминирован и схемно-полн для S, то для всех состояний q Q:

$$L(A[I/\{q\}, F/FS]) \quad L(A[I/\{q\}]) \quad q \quad \text{Safe}_{C}(F).$$

Доказательство. « ». Предположим, что q/ ε Safe (F). Определение безопасности на основе схемы дает acc ({q}) (FS \ F)/ = . Итак, существует такая изгородь h HΣ , что q FS \ F относительно . В частности, q L(A[I/{q}, F/FS]). По детерминизму не существует другого перехода на h из q, и, следовательно, q/ L(A[I/{q}]). Следовательно, L(A[I/{q}, F/FS])/

L(A[I/{q}]. “ ”. Предположим, что q Safe (F). Пусть h L(A [I/{q}, F/FS]). Тогда ' € ФС, h такой, что q q F. ' относительно . Следовательно, q' существует q acc ({q}) такой, что acc ({q}) F (Q \ FS q ' Таким образом, h L(A[I/{q}]). подразумевает □

Лемма A1 с FS = Q показывает лемму 5 из [21], которая утверждает, что для любого полного dSHA A что безопасность – это проблема универсальности:

$$L(A[I/\{q\}]) = H\Sigma \quad q \quad \text{Safe}(F).$$

Это связано с тем, что из схемной полноты A для S = HΣ следует полнота A, и , следовательно, L(A[I/{q}, F/Q]) = HΣ .

Приложение А.2. Алгоритм

Теперь мы представляем наш первый тестер членства для dSHA с ограничениями схемы. Учитывая SHA A, мы вычисляем SHA Ae(S) = (Σ , Qe , e ,I e(S) , Fe(S)) следующим образом. Пусть sel будет

Machine Translated by Google

свежий символ. Начнем с набора состояний, безопасных для выбора S0 = Safe (F) и, соответственно, безопасных для отклонения R0 = Safe (Q \ F). Тогда наборы состояний Ae(S) будут следующими:

$$Qe = (Q \times 2^{Q} \times 2^{Q}) \cup \{sel\} = \{(q, \bigcup_{Я \in e(C)} R0, S0) \mid q \in I, q/\varepsilon \in R0 \cap S0\} \cup \{sel \mid Я \cap S0/ = \varnothing\}$$

$$Fe(S) = \{(q, R0, S0) \mid q \in F\} \cup \{sel\}$$

Для иллюстрации рассмотрим dSHA на рисунке 4. Этот dSHA не является полным, но является полной схемой для документа схемы. Получаем s-down (0, {4}) = {3, 4}. Может показаться нелогичным, что только состояние 3, но и состояние 4 вниз остается безопасным для выбора. Это отражает тот факт, что ни одно надлежащее подхеджирование любого хеджа, удовлетворяющего намеченной схеме, никогда не может перейти в состояние 4. Применение самой ранней конструкции с проекцией «безопасно-без изменений» к dSHA на рисунке 4 дает dSHA на рисунке A1. Правила перехода в е приведены на рисунке A2.

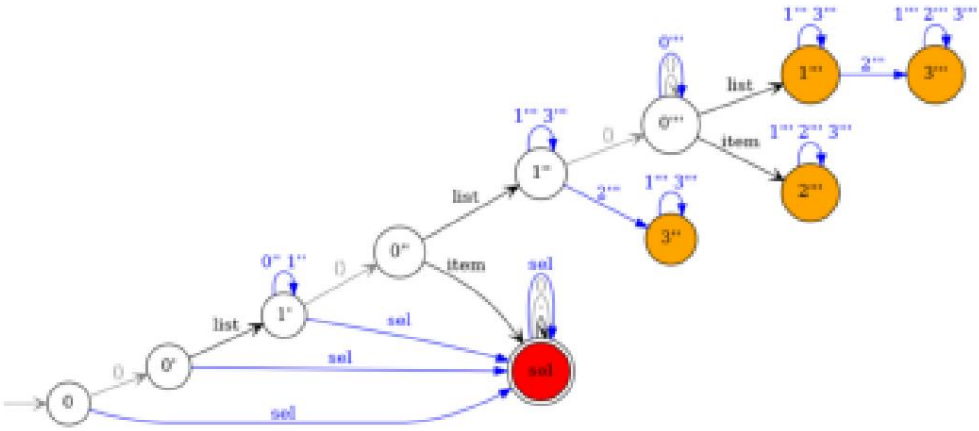


Рисунок A1. Самый ранний dSHA Ae для dSHA A на рисунке 19 с документом схемы для фильтра XPath [self::list/child::item]. Состояниям Ae соответствуют следующие кортежи: 0 = (0, {1, 2, 3, 5}, {4}), 0′ = (0, {2, 4, 5}, {3, 4}) , 1′ = (1, {2, 4, 5}, {3, 4}), 0″ = (0, , {2, «1 4, 5}), = (1, , {2, 4 , 5}), 0‴ = (0, ,), 1‴ = (1, ,), 2‴ = (1, ,) и 3‴ = (1, ,).

Machine Translated by Google

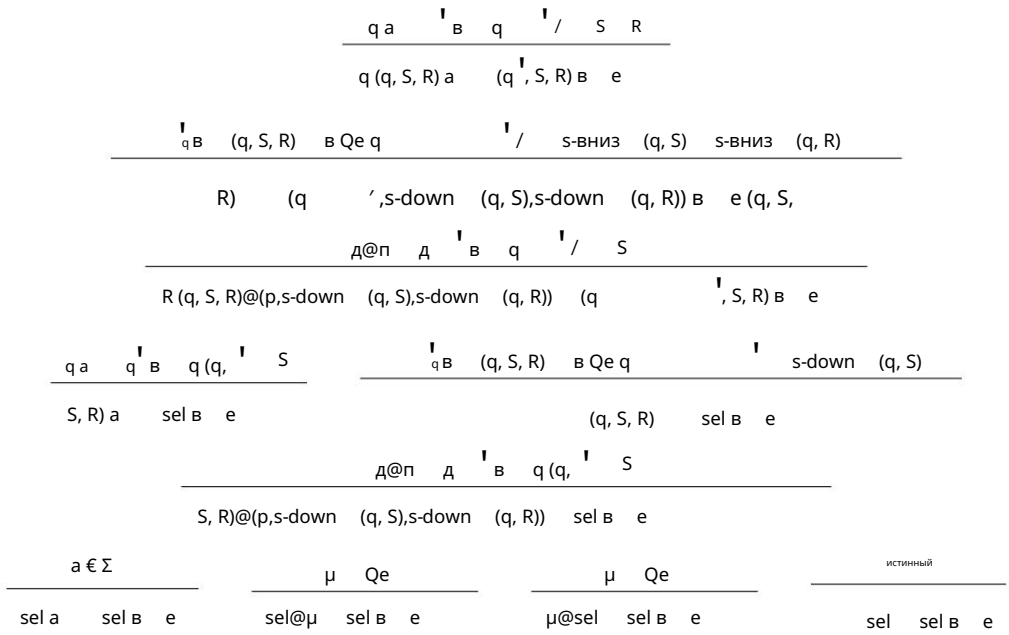


Рисунок A2. Самые ранние правила перехода e выводятся из правил перехода некоторого SHA.

Приложение A.3. Надежность и полнота

Затем мы предоставим результаты надежности и полноты нашего самого раннего тестера членства в памяти с проекцией.

Предложение A1 (Обоснованность). Для любого SHA A со схемой S = L(A[F/FS]), хеджем h S и частичным пробегом v Ae(S) на h, который заканчивается в состоянии sel, тогда h L(A).

Таким образом, если частичный прогон достигает состояния sel , то projΣ (v) наверняка принадлежит к L(A) относительно S. Оценка SHA s Ae(S) дает самый ранний тестер членства в памяти для dSHA A. A Единственная адаптация заключается в следующем: всякий раз, когда достигается sel , вычисление может полностью остановиться и принять входное хеджирование.

Теорема A1 (Полнота). Пусть A = (Σ , Q, , I, F) — dSHA , полная схемы для схемы S = L(A[F/FS]). Для любого хеджа h S и префикса v из nw(h): Если v достоверно принадлежит к L(A) c S, то

- существует частичная серия w для Ae(S) на h, такая, что v = projΣ (w) и w заканчивается состоянием sel.
- Если v достоверно не принадлежит к L(A) c S, то существует блокирующий частичный пробег w для Ae(S) на h, такой, что projΣ (w) является префиксом v.

Это означает, что достоверность членства и нечленства обнаруживается по самому раннему префиксу при запуске оценщика для Ae(S) .

Доказательство. Мы заметили, что этот алгоритм по сути такой же, как и самый ранний тестер членства из предложения 6 из [21], за исключением того, что он также проверяет безопасное отклонение и учитывает схемы. Тот факт, что здесь используются NWA вместо SHA , не является существенным. Таким образом, мы можем без проблем снять оттуда доказательство достоверности и полноты. □

Приложение B. Доказательства раздела 6 (Прогноз без изменений)

Доказательство. Нам нужно доказать, что отсутствие изменений в состояниях q P без изменений является правильным.

Если q no-change , это следует из полноты схемы , так что невозможно ни заблокировать какое-либо хеджирование из схемы, ни изменить состояние. В случае q P

Machine Translated by Google

Интуиция подсказывает, что состояние на уровне выше, скажем, r, не может ни измениться, поскольку P = s-no-change (r), и автомат не может заблокировать какое-либо изгородь от схемы из-за полноты схемы.

Сначала докажем включение L(A) L(A snc). Поскольку L(A) S по определению схем, отсюда следует L(A) L(A snc) S. Доказательство будет основано на следующих трех утверждениях A1.1a, A1.2a и A1.3a. Обратите внимание, что для этого направления не требуется полнота схемы .

h Пункт A1.1a. П Потносительно CHC для всех изгородей h HΣ .

Доказательство проводится непосредственно индукцией по структуре h. Он использует последние три правила перехода на CHC рисунке 10 , позволяющие всегда оставаться в Пдля любого последующего хеджирования.

Претензия A1.2a. Для всех h HΣ , q Q и P Q, таких что q P без изменений ,

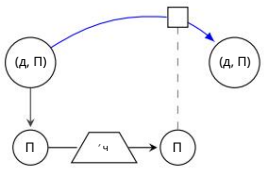
(q, P) час (q, P) относительно CHC.

Утверждение A1.2a докажем индукцией по структуре h.

Случай h = h ' ч . В этом случае мы можем использовать пункт A1.1a, чтобы показать П Потносительно snc и правила вывода

$$\frac{q \quad P \quad \text{без изменений}}{(д, П) \qquad П в \ CHC} \qquad \frac{q \quad P \quad \text{без изменений}}{(q, P)@П \quad (q, P) в \ CHC}$$

чтобы замкнуть следующую диаграмму относительно snc:



Это доказывает (q, P) час (q, P) относительно snc, как того требует утверждение.

Случай h = a. Поскольку q P без изменений , мы можем применить правило вывода:

$$\frac{a \quad \Sigma \quad q \quad P \quad \text{без изменений} \quad (q,}{P) a \quad (q, P) в \ CHC}$$

Это доказывает данный случай иска.

Случай h = ε. Тривиально имеем (q, P) ε (q, P) относительно CHC.

Случай h = h ' · час «. По предположению индукции, примененному к h ' и ч « имеем: (q, P) час ' (q, P) и (q, P) час « (q, P) относительно snc. Следовательно, (q, P) час ' час « (q, P) относительно CHC.

На этом заканчивается доказательство утверждения A1.2a. Следующее утверждение, доказательство которого с помощью индукционного шага несколько более утомительно, является ключом к доказательству корректности. Определим следующий предикат для всех q Q и P Q:

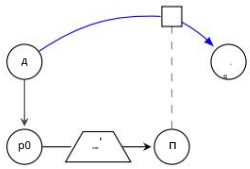
$$д \quad P \ q \ « \ \text{тогда и только тогда, q} \ = q \ « \quad д \ , q \ « \ € P)$$

Претензия A1.3a. Пусть h HΣ — хедж, q, q h такой, что acc (P) Q состояний и P Q — подмножество состояний, таких как / P no-change . Если q q относительно , то существует q ” такой, что (q, P) час (q « , P) относительно snc и q P q «.

Machine Translated by Google

Доказательство. Индукция по структуре h.

Случай h = h 'ч . Предположение q qp Q ' wrt показывает, что существуют состояния p0 и замыкает следующую диаграмму:

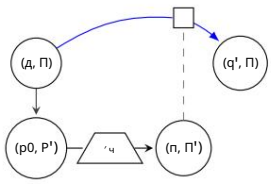


Пусть P' = s-no-change (q) и обратите внимание, что acc (P ′) P '. Поскольку q/ P без изменений , мы можно сделать следующий вывод:

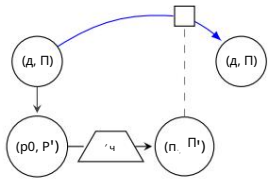
$$\frac{p0 \text{ в } q/\ P \text{ без изменений}}{(q, P)\ (p0, P') \text{ в}^{\text{СНС}}} \qquad \frac{д@п\ \ д\ '\text{ в } q/\ P \text{ без изменений}}{(q, P)@(p, P\ ')\ \ (д\ ', P) \text{ в}^{\text{СНС}}}$$

Подслучай p0/ P ' без изменений . Предположение индукции применимо к h такому, ' показывает, что существует p (p0) ') относительно snc и p P p '. Мы различаем последний предикат:

Подподслучай p- ' = п. Следовательно, (p0, P') '(p, P ') относительно snc, поэтому мы можем замкнуть диаграмма выглядит следующим образом:



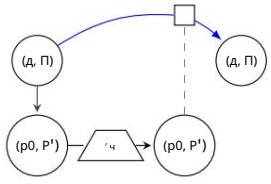
Это показывает, что (q, P) (q . ', P) относительно snc, и, таким образом, утверждение, ' P q '. Подподслучай p, pq в ' P 'Поскольку p ' P 'и П ' поскольку q = s-no-change (q), мы имеем q@p' . Следовательно, мы можем замкнуть диаграмму следующим образом:



Поскольку p ' и q@p q ' в имеем q = q ' по определению П ' = P , s-без изменений (q). Это показывает, что (q, P) (q, P) относительно snc. Поскольку q P q, претензия следующая.

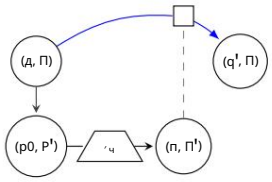
Подслучай p0 P ' без изменений . Тогда утверждение A1.2a показывает, что (p0, P') '(p0, P ') относительно СНС.

Подподслучай p0 P '. Поскольку p acc (p0) и acc (P ′) P = s-без ', отсюда следует, что P p P слишком. По определению ' изменений (q) и полнота , вытекает, что q@ p0 принадлежности p0 P теперь ' и p P ' = {q} = q@ p. Мы могут замыкать приведенную ниже диаграмму следующим образом:



Подподслучай p0 no-change . В этом случае p0 = p , так что q ' q@ p0. Следовательно,

Machine Translated by Google



Случай h = a. Поскольку q/ P no-change , мы можем применить правило вывода:

$$\frac{q\ a\ q'\ в\ q/\ P\ без\ изменений\ (q,\ P)\ a\ (q'\ ,\ P)\ в}{}\ CHC$$

Это показывает, что (q, P) (q' , P) подтверждает утверждение, поскольку q' P q' .

Случай h = ε. В этом случае имеем q = q ' и (q, P) ε (q, P) , поэтому утверждение справедливо.

h Случай h = h1 · h2. Поскольку q q ' h1 относительно , существует q1 Q такой, что q q1 относительно и h2 q1 q' относительно . Поскольку q/ P no-change , применим предположение индукции по h1. такой, что, Это означает, что существует q '1

$$h1\ (q,\ P)\ (q'_{1},\ P)\ относительно\ CHC\ и\ q1\ P\ q\ '_{1}.$$

Мы различаем два случая, когда q1 P q '1

Подслучай q1 = q '1 . Здесь мы также выделяем два подслучая: Подподслучай q1/ кроме P.

Индукционное предположение, примененное к h2, дает следующее:

инг:

$$q\ «.\ (q1,\ P)\ (q\ ч2\ «,\ P)\ относительно\ CHC\ д'\ P\ q\ «,$$

Следовательно

$$q\ «.\ (q,\ P)\ (q\ час\ «,\ P)\ относительно\ CHC\ д'\ P\ q\ «$$

.

Подподслучай q1 P. По утверждению A1.2a имеем (q1, P) (q1, P). Также имеем ч2 P. Следовательно, д' acc({q1}), и поскольку мы предполагаем acc(P) P, отсюда следует q ' (q, P) час (q1, P) и q ', q1 P, а это означает, что утверждение q ' P q1.

Подслучай q1, q '1 P. Поскольку q '1 P, из утверждения A1.2a следует, что (q'1 , П) h2 '1 , P) относительно CHC . acc (P) час (q 1 , P) относительно snc. Поскольку q ' (q acc ({q1}) и q1 P, то Таким образом, (q, P) д' P. Здесь, как и в предыдущем подслучае, мы использовали, что acc(P) П предполагается иском. Пусть q и q « = q '1 . Тогда (q, P) (q час «, P) относительно CHC ', q « P , доказывающий утверждение.

На этом доказательство утверждения A1.3a заканчивается. □

Доказательство включения L(A) L(A snc). Пусть h L(A). Тогда существуют q0 I и q F такие, что q0 q. Мы различаем два случая:

Случай q0 без изменений . По определению отсутствия изменений и поскольку q acc (q0), имеем q0 = q.

Утверждение A1.2a показывает, что (q0,) час (q0,) относительно snc и, следовательно, (q0,) час (q,), так что h L(A snc).

Случай q0/ ε без изменений . Утверждение A1.3a с P = показывает, что (q0,) (q,) час относительно snc и, следовательно, h L(A snc).

Machine Translated by Google

На этом доказательство первого включения заканчивается. Далее мы хотим показать обратное включение L(A snc) S L(A). В конечном итоге это будет следовать из следующих трех пунктов формулы A1.1b, A1.2b и A1.3b.

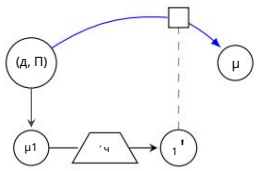
Пункт A1.1б. Для любого хеджа h и состояния μ Qsnc, если П μ относительно CHC тогда μ = П.

Доказательство простое, путем индукции по структуре h: единственные правила перехода snc с Пв левой части выводятся из последних трех правил на рисунке 10. Они требуют оставаться в П, какое бы хеджирование ни последовало.

Пункт A1.2б. Для любого хеджа h положим P Q, состояние q P без изменений и состояние μ Qsnc: если (q, P) μ относительно snc, то μ = (q, P).

Доказательство. Индукцией по структуре h. Предположим, что (q, P) μ относительно snc:

Случай h = h . Должны существовать состояния μ1, μ 1 Qsnc , замыкая следующую диаграмму:



Поскольку q P no-change , для вывода (q, P) μ1 относительно snc необходимо применить следующее правило :

$$\frac{q \quad P \quad \text{без изменений}}{(д, П) \qquad П в^{CHC}}$$

Следовательно, μ1 = П. Пункт A1.1b показывает, что μ 1 = П тоже. Итак, μ должно было быть получается путем применения правила:

$$\frac{q \quad P \quad \text{без изменений}}{(q, P)@П \quad (q, P) в^{CHC}}.$$

Итак, μ = (q, P), что и требовалось.

Случай h = a. Должно быть применено следующее правило:

$$\frac{q \quad P \quad \text{без изменений}}{(q, P) a \quad (q, P) в^{CHC}}$$

Следовательно, μ = (q, P).

Случай h = ε. Очевидный.

Случай h = h1 · h2. Должен существовать μ1 такой, что (q, P) μ относительно snc. По предположению индукции μ1 , примененному к h1, имеем μ1 = (q, P). Таким образом, мы можем применить гипотезу индукции к h2 и получить μ2 = (q, P). h1 h2

На этом доказательство утверждения A1.2b заканчивается. Далее нам понадобится обратный пункт A1.3a. □

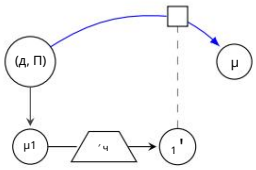
Пункт A1.3б. Пусть q Q, P Q, такой, что q/ ε P no-change и acc (P) P. Для любого h HΣ , такого, что (q, P) μ относительно любой блокирующей частичной CHC для некоторого μ Qsnc и такого, что не имеет запустить по h, начиная с q, существует q , q ”, такой, что

μ = (q , P), q q « относительно и q P q «.

Machine Translated by Google

Доказательство. Индукцией по структуре h НΣ выделим случаи всех возможных форм h.

Случай h = h μ По определению s(q, P) должны существовать μ1, μ , чтобы можно было замкнуть 1 Qsnc, такой следующую диаграмму:



Поскольку q/ P no-change , для вывода (q, P) μ1 было применено следующее правило, П где = s-no-change (q):

$$\frac{p \text{ в } q/P \text{ без изменений}}{(q,P) \quad (p,P\,) \text{ в } \text{СНС}}$$

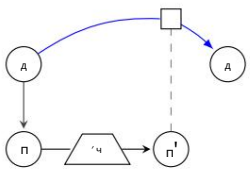
Следовательно, существует р такой, что μ1 = (p, P). Мы фиксируем такое состояние р произвольно.

Поскольку не имеет блокировок на h из q, существует р , q « Q, такой, что п р q, начиная с р. « относительно . Кроме того, не имеет блокирующего частичного пробега. на час

Подслучай р P · В этом случае мы можем применить утверждение A1.2b к) 1 h относительно в порядке μ показать, что μ, 1 = (p, P (p, P). Поскольку р acc ({p}) и р P = s-без изменений (q), чтобы отсюда следует, что q@ p = {q}. Следовательно, для вывода (q, P) μ было применено следующее правило :

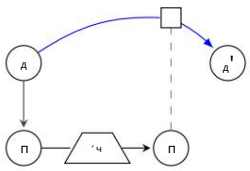
$$\frac{q@p \quad q \text{ в } q/P \text{ без изменений } (q,P)@(p,P\,) \quad (q,P) \text{ в } \text{СНС}}{},$$

Это показывает, что μ = (q, P). Пусть q = q так, что μ = (q , П). Поскольку р относительно , мы есть п acc ({p}) так, что q@p q относительно по определению s-без изменений ({q}). Следовательно, мы можем замкнуть следующую диаграмму:



= q, так « что q требуется = q ». Тогда верно, что q. Пусть q P q « h и q q относительно , так как согласно утверждению.

h Подслучай р no-change . В этом случае р р относительно следует, что р = п. Следовательно, мы может замкнуть следующую диаграмму:



Это показывает, что q q относительно .

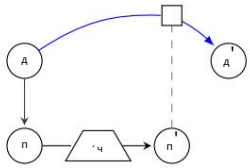
Machine Translated by Google

В подслучае p/ ε P ' без изменений . По предположению индукции, примененному к (p, P ' час) относительно μ 1 существует p ” такой, что

μ ' 1 = (п ', П ' ч), п ' п « относительно и р ' P' п «·

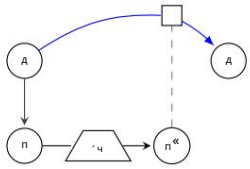
Поскольку не имеет блокирующих проходов на h , начинающихся с q, существуют q P′ p ”: « такой, что « Д« относительно . Есть два способа удовлетворить p ' q@p

Подподслучай p ' = п «. Тогда мы имеем следующее:



Это показывает q h ' относительно .

Подподслучай p ', п « q P. По определению ḥ отсюда следует, что q ' = q = q ». Следословательно,



h Это показывает q q относительно .

Случай h = a. Поскольку q/ P no-change , необходимо использовать следующее правило вывода:

$$\frac{q\ a\quad q'\ в\quad q/\quad P\quad без\ изменений\quad (q,}{P)\ a\quad (q\quad ',P)\ в}\ СНС$$

Итак, μ = (q ', P) и q a q ' относительно .

Случай h = ε. Очевидный.

Случай h = h1 · h2. Суждение (q, P) μ относительно час h2 СНС показывает, что существует μ1 такой, что h1 (q, P) μ1 μ относительно snc. Поскольку q/ P no-change , мы можем применить индукцию h1 гипотеза для h1. Это показывает, что существует q ' 1 и q «, такой, что μ1 = (q ' 1 , П), q д « относительно и q ' 1 P q «.

Подслучай q ' 1 = q «. Следовательно, (q1, P) x2 μ относительно СНС.

Подподслучай q ' 1 / P без изменений . В этом случае мы можем применить индукционную гипотезу. гипотеза (q μ = (q ' 1 , П) h2 μ относительно snc, что указывает на существование ', такой, что ', P) и состояние q ” такое, что q ' 1 h2 «q и q , так ' q P q ”. Следовательно, q« q-д « относительно и q ' P q « что утверждение справедливо.

Подслучай q ' 1 P без изменений . Пункт A1.2b применяется к (q 1 , P). ' 1 , P) ч2 μ относительно СНС показывает, что μ = (q '

Подслучай q ' 1 , q « P. Напомним, что (q «, П) h2 μ относительно « P. Пункт A1.2b показывает, что μ = (q «snc и q h1 1 , P) относительно snc. У нас также « относительно . Поскольку там есть q q , нет блокирующих частичных пробегов по h, начиная с q , существует состояние « такой q , в котором h2 q «q относительно . Поскольку q ' 1 P и P замкнуто по достижимости, имеем д « acc({q«}) acc(P) P. Из q h1 «q относительно получаемq час « относительно . С д «, д « P, то q «1 P q ”, и, таким образом, утверждение справедливо.

На этом доказательство утверждения A1.3b заканчивается.

Machine Translated by Google

□

Доказательство включения L(A snc) S L(A). Пусть h € L(A snc) S. Поскольку h € L(A snc), то существуют q0 € I и q € F такие, что (q0,) (q,) относительно snc. Используя лемму 7, имеем q относительно . q0 Выделим два случая:

Случай q0 без изменений . Утверждение A1.2b показывает, что q = q0. Поскольку A схемно-полна для S, h S и q0 I без изменений , лемма 7 показывает, что q0 q0 относительно . Поскольку q = q0, это дает h L(A).

Случай q0/ ε без изменений . Поскольку A является полным по схеме для S и h S, на h не существует блокировок , начинающихся с q0. Следовательно, мы можем применить утверждение A1.3b с P = к (q0,) (q,) относительно snc. Это показывает, что q0 q относительно и, следовательно, h L(A).

На этом заканчивается доказательство обратного включения и, следовательно, L(A) = L(A snc). □

Ссылки

1. Мариан А.; Симеон, Дж. Проектирование XML-документов. В трудах ВЛДБ; Морган Кауфманн: Сан-Франциско, Калифорния, США, 2003 г.; стр. 213–224.
2. Фриш, А. Распознавание обычного древовидного языка со статической информацией. В материалах исследования новых границ теоретической информатики, 18-й Всемирный компьютерный конгресс ИФИП, 3-я Международная конференция TCS по теоретической информатике, Тулуза, Франция, 22–27 августа 2004 г.; стр. 661–674.
3. Манет, С.; Нгуен, К. Оптимизация всего запроса XPath. Журнал ВЛПБ 2010, 3, 882–893.
4. Себастьян Т.; Нирен, Дж. Проекция для автоматов с вложенными словами ускоряет оценку XPath в потоках XML. В материалах Международной конференции по текущим тенденциям в теории и практике информатики (SOFSEM), Гаррахов, Чешская Республика, 23–28 января 2016 г.
5. Кей, М. Саксонский процессор XSLT и XQuery. 2004. Доступно онлайн: https://www.saxonica.com. (по состоянию на 29 июля 2024 г.).
6. Генечко, М.; Мурлак, Ф.; Пейперман, К. Поддержка потомков в JSONPath с SIMD-ускорением. В материалах 29-й Международной конференции ACM по архитектурной поддержке языков программирования и операционных систем, Ла-Хойя, Калифорния, США, 27 апреля – 1 мая 2024 г.; Том 1, стр. 15–29.
7. Аль Серхали, А.; Нирен, Дж. Эталонный набор детерминированных автоматов для запросов XPath. В материалах XML Prague 2022, Прага, Чехия, 9–11 июня 2022 г.
8. Нирен, Дж.; Сахо, М. Еще раз о детерминизации и минимизации автоматов для вложенных слов. Алгоритмы 2021, 14, 68. https://doi.org/10.3390/a14030068.
9. Комон, Х.; Доше, М.; Гиллерон, Р.; Лёдинг, К.; Жакмар, Ф.; Лужьес, Д.; Тайсон, С.; Томмаси, М. Техника работы с древовидными автоматами и приложения. Доступно онлайн: http://tata.gforge.inria.fr. (по состоянию на 29 июля 2024 г.).
10. Тэтчер Дж. В. Характеристика деревьев вывода контекстно-свободных грамматик посредством обобщения теории автоматов. Дж. Компьютер. Сист. наук. 1967, 1, 317–322. https://doi.org/10.1016/S0022-0000(67)80022-9.
11. Алур Р. Соединяя слова и деревья. В материалах 26-го симпозиума ACM SIGMOD-SIGACT-SIGART по принципам Системы баз данных, Пекин, Китай, 11–14 июня 2007 г.; ACM-Press: Нью-Йорк, Нью-Йорк, США, 2007, стр. 233–242.
12. Алур Р.; Мадхусудан, П. Языки с видимым опусканием вниз. В материалах 36-го ежегодного симпозиума ACM по теории вычислений, Чикаго, Иллинойс, США, 13–16 июня 2004 г.; Бабай Л., Ред.; ACM: Нью-Йорк, штат Нью-Йорк, США, 2004 г.; стр. 202–211. https: //doi.org/10.1145/1007352.1007390.
13. Мельхорн, К. Галечные горные хребты и применение метода распознавания DCFL. В трудах автоматов, языков и программирования, 7-й коллоквиум, Нордвайкерхаут, Нидерланды, 14–18 июля 1980 г.; разбирательства; де Баккер, Дж.В., ван Леувен, Дж., ред.; Шпрингер: Берлин/Гейдельберг, Германия, 1980 г.; Том 85, стр. 422–435. https://doi.org/10.1007/3-540-10003- 2_89.
14. фон Браунмюль, Б.; Вербек, Р. Языки, управляемые вводом, распознаются в пространстве log n. В разделах теории вычислений; Математические исследования Северной Голландии; Карпински М., ван Леувен Дж., ред.; Северная Голландия: 1985 г.; Том 102, стр. 1–19. https://doi.org/10.1016/S0304-0208(08)73072-X.
15. Охотин А.; Саломаа, К. Сложность автоматов с магазинным вводом. Новости SIGACT 2014, 45, 47–67. https://doi.org/10.1145/2636805.2636821.
16. Нирен, Дж.; Сахо, М.; Аль Серхали, А. Определение автоматов на основе схемы. В материалах 13-го Международного симпозиума по играм, автоматам, логике и формальной верификации, GandALF 2022, Мадрид, Испания, 21–23 сентября 2022 г.; Ганти П., Моника Д.Д., ред.; ЭПТКС: 2022; Том 370, стр. 49–65. https://doi.org/10.4204/EPTCS.370.4.
17. Лик А.; Шмитц, С. XPath Benchmark. Доступно онлайн: https://archive.softwareheritage.org/browse/directory/1ea68cf5bb3f9. f3f2fe8c7995f1802ebadf17fb5 (по состоянию на 29 июля 2024 г.).

Machine Translated by Google

18. Мозафари, Б.; Цзэн, К.; Заниоло, К. Высокопроизводительная обработка сложных событий в потоках XML. В материалах конференции SIGMOD; Кандан К.С., Чен Ю., Снодграсс Р.Т., Гравано Л., Фуксман А., Кандан К.С., Чен Ю., Снодграсс Р.Т., Гравано Л., Фуксман А., ред.; ACM: Нью-Йорк, штат Нью-Йорк, США, 2012 г.; стр. 253–264. https://doi.org/10.1145/2213836.2213866.

19. Грез А.; Риверос, К.; Угарте, М. Формальная основа для сложной обработки событий. В материалах 22-й Международной конференции по теории баз данных (ICDT 2019), Лиссабон, Португалия, 26–28 марта 2019 г.; стр. 5:1–5:18. https://doi.org/10.4230/LIPIcs.ИКДТ.2019.5.

20. Муньос, М.; Риверос, К. Потоковое перечисление вложенных документов. В материалах 25-й Международной конференции по теории баз данных, ICDT 2022, Эдинбург, Великобритания, 29 марта – 1 апреля 2022 г.; Олтяну, Д., Вортмайер, Н., ред.; Замок Дагштуль – Центр информатики Лейбница: 2022 г.; Том 220, стр. 19:1–19:18. https://doi.org/10.4230/LIPIcs.ICDT.2022.19.

21. Аль Серхали, А.; Нирен, Дж. Первые ответы на запросы для детерминированных пошаговых автоматов хеджирования. В материалах по внедрению и применению автоматов — 27-я Международная конференция, CIAA 2023, Фамагуста, Северный Кипр, 19–22 сентября 2023 г.; разбирательства; Надь, Б., Ред.; Шпрингер: Берлин/Гейдельберг, Германия, 2023 г.; Том 14151, стр. 53–65. https: //doi.org/10.1007/978-3-031-40247-0_3.

22. Гаувин, О.; Нирен, Дж.; Тайсон, С. Самые ранние ответы на запросы для детерминированных автоматов с вложенными словами. В материалах 17-го Международного симпозиума по основам теории компьютеров, Вроцлав, Польша, 2–4 сентября 2009 г.; Springer: Берлин/Гейдельберг, Германия, 2009 г.; Том 5699, стр. 121–132.

23. Дебарбье, Д.; Гаувин, О.; Нирен, Дж.; Себастьян, Т.; Зергауи, М. Ранние автоматы с вложенными словами для ответа на запросы XPath в потоках XML. Теория. Вычислить. наук. 2015, 578, 100–125. https://doi.org/10.1016/j.tcs.2015.01.017.

24. Аль Серхали, А.; Нирен, Дж. Проекция Субхеджа для автоматов пошагового хеджирования. В Proceedings of the Fundamentals of Computing Theory — 24-й Международный симпозиум, FCT 2023, Трир, Германия, 18–21 сентября 2023 г.; разбирательство; Фернау Х., Янсен К., ред.; Шпрингер: Берлин/Гейдельберг, Германия, 2023 г.; Том 14292, стр. 16–31. https://doi.org/10.1007/978-3-031-43587-4_2.

25. Нейман А.; Зайдль, Х. Поиск совпадений рисунков деревьев в лесах. В трудах Фонда программной технологии и теоретической информатики, Ченнаи, Индия, 17–19 декабря 1998 г.; Springer: Берлин/Гейдельберг, Германия, 1998 г.; Том 1530, стр. 134–145.

26. Гаувин, О.; Нирен, Дж.; Роос, Ю. Автоматы потокового дерева. Инф. Процесс. Летт. 2008, 109, 13–17. https://doi.org/10.1016/j.ipl.2008.08.002.

27. Франческе, М. Тест производительности XPathMark. Доступно онлайн: https://users.dimi.uniud.it/~massimo.franceschet/xpathmark/PTbench.html (по состоянию на 25 октября 2020 г.).

28. Хосоя, Х.; Пирс, Британская Колумбия XDuce: статически типизированный язык обработки XML. АКМ Транс. Интернет-технологии. 2003, 3, 117–148. https://doi.org/10.1145/767193.767195.

29. Гечег, Ф.; Стейнби, М. Древовидные автоматы; Академический Киадо: Будапешт, Венгрия, 1984 г.

30. Золото, ЭМ Языковая идентификация в пределе. Инф. Контроль. 1967, 10, 447–474. https://doi.org/10.1016/S0019-9958(67)91165-5.

31. Карме, Дж.; Нирен, Дж.; Томмаси, М. Опрос неранжированных деревьев с помощью автоматов пошагового дерева. В материалах о методах и приложениях перезаписи , 15-я Международная конференция, RTA 2004, Аахен, Германия, 3–5 июня 2004 г.; разбирательство; ван Остром, В., Ред.; Springer: Берлин/Гейдельберг, Германия, 2004 г.; Том 3091, стр. 105–118. https://doi.org/10.1007/978-3-540-25 979-4_8.

32. Пара, С.; Кере, А. Определение и исследование двуязычных правил. Инф. Контроль 1968, 13, 565–593.

33. Мурата, М. Хедж-автоматы: формальная модель XML-схем. 2000. Доступно в Интернете: http://www.xml.gr.jp/relax/hedge_. хороший.html (по состоянию на 29 июля 2024 г.).

34. Мартенс, В.; Нирен, Дж. О минимизации XML-схем и древовидных автоматов для неранжированных деревьев. Дж. Компьютер. Сист. наук. 2007, 73, 550–583.

35. Нирен, Дж. Автоматы пошагового хеджирования экспоненциально более кратки, чем лесные автоматы. 2024 год, в стадии подготовки.

36. Боянчик, М.; Валукевич, И. Лесные алгебры. В трудах по логике и автоматам: история и перспективы [в честь Вольфганга Томаса]; Флум Дж., Гредель Э., Вилке Т., ред.; Издательство Амстердамского университета: Амстердам, Нидерланды, 2008 г.; Том 2, стр. 107–132.

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Иерархическая структура оптимизации для проектирования макетов Сеть газосборных трубопроводов «Звездное дерево» в дискретных пространствах

Ю Линь [1], Яньхуа Цю [1], Хао Чен [2], Цзюнь Чжоу 3,*, Цзяи Хэ [3], Пэнхуа Ду 3 и Дафан Лю [4]

[1] Научно-исследовательский институт технологий сбора и транспорта, PetroChina Southwest Oil & Gasfield Company, Чэнду 610041, Китай; linyu223@petrochina.com.cn (YL); qiuyan0320@petrochina.com.cn (YQ)
[2] Sichuan Shale Gas Exploration and Development Co., Ltd., Чэнду 610051, Китай; chenhao05@petrochina.com.cn
3 Школа нефтяной инженерии, Юго-Западный нефтяной университет, Чэнду 610500, Китай; hjywyyx@163.com (Дж.Х.); scdupenghua@163.com (ПД)
[4] Департамент разведки и разработки сланцевого газа, CNPC Chuanqing Drilling Engineering Co., Ltd., Чэнду 610066, Китай; liudf_cqyyq@cnpc.com.cn
* Переписка: zhoujunswpu@163.com

Аннотация: Сеть газосборных трубопроводов представляет собой важнейшую инфраструктуру для сбора и транспортировки природного газа от места добычи до объекта переработки. В данной статье представлена модель оптимизации проектирования газосборной сети газопроводов типа «звезда» в дискретном пространстве, направленная на определение оптимальной конфигурации этой инфраструктуры. Целью является сокращение инвестиций, необходимых для построения сети. Ключевые переменные решения включают расположение станций, расположение завода, соединения между скважинами и станциями, а также взаимосвязи между станциями. Сформулированы несколько ограничений равенства и неравенства, в первую очередь касающиеся принадлежности скважин и станций, радиуса передачи и мощности станций. Проектирование трубопроводной сети типа «звезда-дерево» представляет собой сложную, недетерминированную полиномиальную (NP) задачу жесткой комбинаторной оптимизации. Для решения этой проблемы предлагается структура иерархической оптимизации в сочетании с улучшенным генетическим алгоритмом (IGA). Эффективность генетического алгоритма подтверждается путем тестирования и сравнения с другими традиционными алгоритмами. Впоследствии модель оптимизации и методология решения применяются к проектированию схемы трубопроводной сети. Результаты показывают, что оптимизированная конфигурация сети снижает инвестиционные затраты на 16% по сравнению с исходной конструкцией. Кроме того, при сравнении оптимальной компоновки в топологии звезда-звезда видно, что инвестиции, необходимые для топологии звезда-звезда, на 4% выше, чем необходимые для топологии звезда-дерево.

Ключевые слова: топологическая структура; дизайн макета; иерархическая оптимизация; генетический алгоритм

check for updates

Цитирование: Лин, Ю.; Цю, Ю.; Чен, Х.; Чжоу, Дж.; Он, Дж.; Ду, П.; Лю, Д. Структура иерархической оптимизации для проектирования компоновки газосборной сети газосборных трубопроводов типа «звезда – дерево» в дискретных пространствах. Алгоритмы 2024, 17, 340. https://doi.org/10.3390/a17080340

Академические редакторы: Франк Вернер, Мин-Фэн Гэ и Чэнь Лю

Поступила: 3 июля 2024 г.
Пересмотрено: 27 июля 2024 г.
Принято: 3 августа 2024 г.
Опубликовано: 5 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

1.1. Мотивация

В настоящее время мировое сообщество переходит в эпоху чистой и эффективной энергетики, когда природный газ становится преобладающим источником энергии. Согласно «Глобальному энергетическому прогнозу», мировое потребление природного газа, согласно прогнозам, будет непрерывно увеличиваться до 2050 года, при этом среднегодовые темпы роста превысят 1,4% (Ричард, 2020). Значительный рост спроса на природный газ требует активизации усилий по строительству сетей наземного газосбора. Инвестиции, необходимые для газосборных трубопроводных сетей, значительны и часто составляют более 30% общих затрат на разработку газового месторождения [1]. Следовательно, рациональная оптимизация конструкции этих трубопроводных сетей может принести значительные экономические выгоды.

Система обычно включает в себя скважины, газосборные станции, центральные очистные сооружения и трубопроводы (рис. 1). Первоначально добываемый из скважин природный газ транспортируется на

Machine Translated by Google

газосборную станцию по производственным трубопроводам. После сбора на газосборной станции газ затем транспортируется по сборным трубопроводам на центральную очистную станцию для очистки, обеспечивая его соответствие стандартам, необходимым для внешнего распределения.
По топологии сети газосборные трубопроводные сети можно разделить на три основных типа: звездообразная, древовидная и циклическая топологии [2]. В звездообразной топологии несколько скважин напрямую подключены к одной станции, как показано на рисунке 2a. В древовидной топологии скважины связаны со станцией через соседние скважины, как показано на рисунке 2b. Циклическая топология включает взаимосвязанные скважины, образующие петлю, как показано на рисунке 2c.

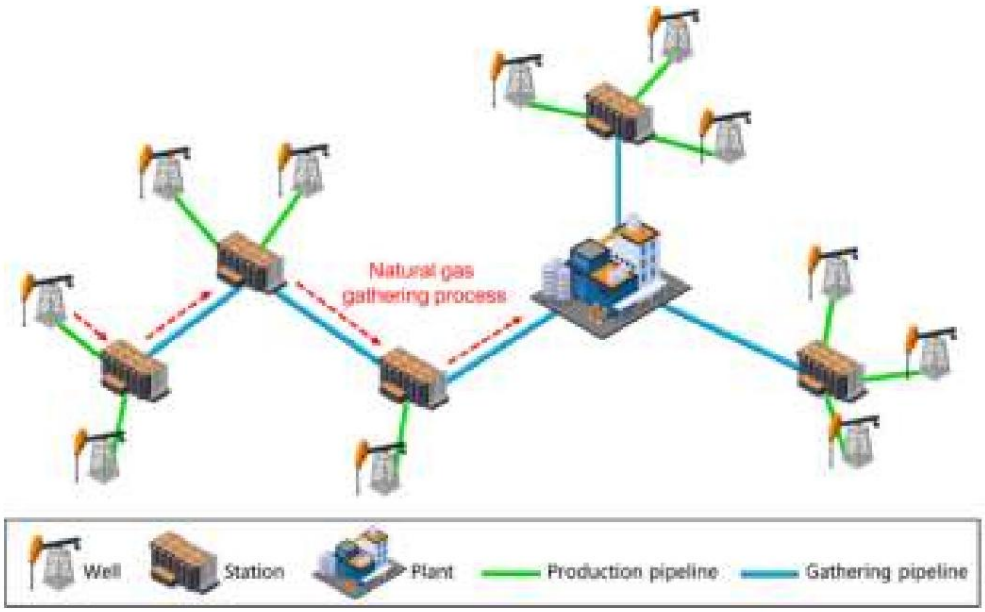


Рисунок 1. Типичная газосборная трубопроводная сеть.

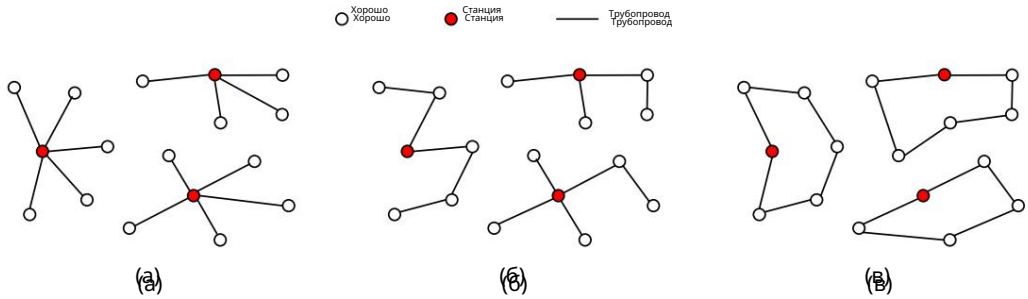


Рис. 2. Структура трубопроводной сети: (а) топология звезды, (б) древовидная топология, (в) циклическая топология.

На практике в крупных газосборных трубопроводных сетях часто наблюдаются различные топологические комбинации. В настоящее время при фактическом строительстве газовых месторождений на основе множества факторов, включая расположение газовых скважин, стоимость строительства, удобство управления, эффективность эксплуатации и т. д., используются сетевые структуры топологии звезда-звезда и сетевые структуры топологии звезда-дерево. широко используется при разработке газовых месторождений. Сетевая структура с топологией звезда-звезда, состоящая из нескольких газовых скважин, напрямую связанных с центральным пунктом сбора газа, делает компоновку простой и легкой в управлении и контроле. Его преимущества заключаются в том, что можно уменьшить общую длину трубопровода, снизить первоначальную стоимость строительства , а также удобно централизованно обрабатывать и транспортировать природный газ, добываемый на газовом месторождении, что подходит для газового месторождения с небольшой газоносная площадь, большое количество устьев и низкая дебит одной скважины. Сетевая структура с топологией звезда-дерево сочетает в себе характеристики звездной и дендритной структур. В этой структуре каждая газовая скважина и ее газосборные станции соединены звездообразно, а газосборные станции соединены магистральными газосборными линиями, образуя де

Machine Translated by Google

out уравновешивает стоимость и эффективность линии сбора на большие расстояния, обеспечивая при этом некоторую гибкость в настройке и расширении в зависимости от стадии разработки месторождения. На рисунке 3 показана типичная сеть газосборных трубопроводов типа «звезда-дерево». Такая структура сети особенно подходит для сценариев, когда скважины внутри локальных блоков сконцентрированы, но эти блоки широко рассредоточены со значительными расстояниями между ними.

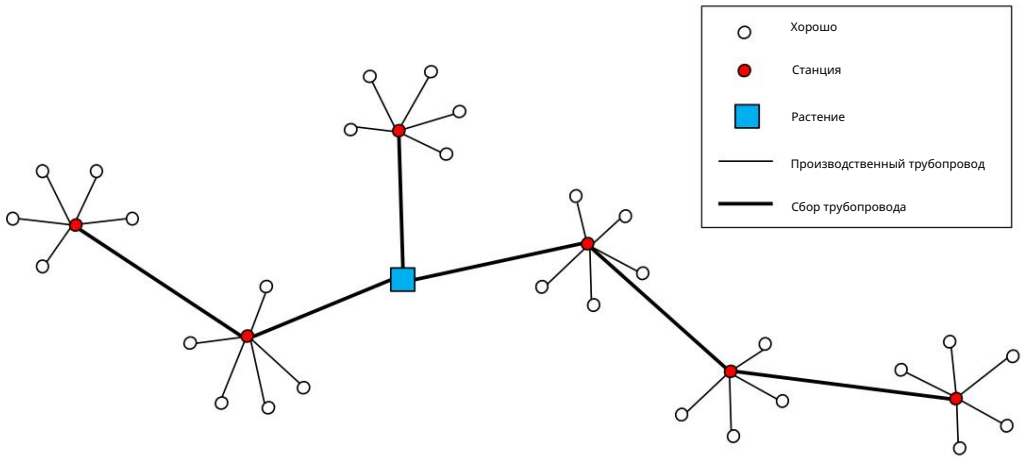


Рисунок 3. Трубопроводная сеть типа «звезда-дерево».

В целом проблему оптимизации можно разделить на оптимизацию конструкции и оптимизацию эксплуатации [3]. Оптимизация проектирования включает в себя выбор наиболее подходящей конфигурации компоновки, размещения станций, размеров труб и масштабов оборудования для новой сети трубопроводов для достижения оптимальных инвестиционных затрат на строительство. С другой стороны, эксплуатационная оптимизация направлена на совершенствование эксплуатационных стратегий созданной трубопроводной сети, что включает в себя оптимизацию потоков трубопроводов и параметров давления в узлах для достижения технических или экономических целей.

Задача оптимизации компоновки газосборных трубопроводных сетей включает в себя несколько подпроцессов, таких как группировка скважин, определение местоположения станций и заводов, а также установление связи между объектами [4]. На рисунке 4 показан процесс оптимизации макета. При этом определение местоположения станции (или завода) делится на два сценария. Первый сценарий заключается в том, что станцию можно построить в любом месте газового месторождения. Второй сценарий заключается в том, что станцию необходимо строить совместно со скважиной, чтобы сэкономить на инвестиционных затратах на строительство. Другими словами, местоположение станции может быть выбрано только из существующего местоположения скважины, которое определяется как выбор местоположения станции в дискретном пространстве. Поскольку этот сценарий более широко используется в инженерной практике, данная статья будет посвящена именно этой проблеме.

1.2. Литературный обзор

Судя по литературным исследованиям, видно, что академическое сообщество провело обширные исследования по оптимизации проектирования газопроводных сетей. Эти исследования в первую очередь сосредоточены на моделировании и решении задач оптимизации, связанных с расположением сети . В зависимости от топологии трубопроводной сети — например, звездообразной, древовидной и циклической — возникают различные задачи оптимизации. Санайе и Махмудимер [5] разработали модель оптимизации для сетей газопроводов с древовидной топологией, которая учитывает такие параметры, как расположение сети, диаметр трубы и давление в узлах. Родригес и др. [6] исследовали проектирование многоуровневых звездообразных сетей, уделяя внимание оптимальным конфигурациям и размещению объектов. Фу [7] исследовал многокритериальную оптимизацию кольцевых нефтесборных трубопроводных сетей. Ван и др. [8] сформулировали многопериодную модель MILP (смешанное целочисленное линейное программирование) для оптимизации сетей с древовидной топологией с учетом рельефа местности и препятствий. Чжоу и др. [9] создали MINLP (Mixed Integer Nonlin-

Machine Translated by Google

Ear Programming) модель для оптимизации двухуровневых конвейерных сетей с топологией «звезда» для подземное хранилище природного газа.

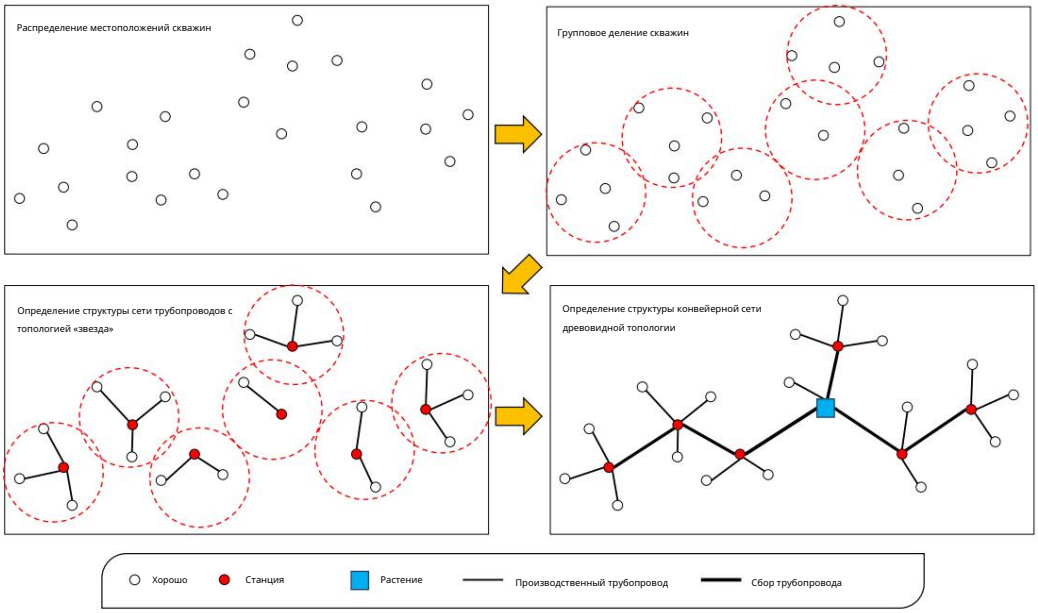


Рисунок 4. Диаграмма иерархической оптимизации.

Что касается методов решения оптимизации, в ранних исследованиях преимущественно использовались методы исследования операций, в частности эвристические алгоритмы, для решения проблемы сбора газа. модели оптимизации трубопроводной сети. Из математической оптимизации и эксплуатации С точки зрения исследований, проектирование оптимальных сетей газосборных трубопроводов классифицируется как NP-трудная задача комбинаторной оптимизации, которая ставит серьезные задачи и находит применение в различных промышленных областях [10,11]. Балинский [12] сформулировал модель оптимизации размещения сайтов в дискретных пространствах с неограниченной пропускной способностью, которая была в основном решаются с помощью эвристических алгоритмов. Общие точные алгоритмы включают ветвь и метод границ, алгоритм секущей плоскости и алгоритм генерации столбцов [13,14]. Однако традиционные теории оптимизации часто не могут эффективно решать проблемы с модели сети трубопроводов природного газа из-за длительного времени вычислений и случайных несоответствие между теоретическими результатами и практическим применением.

В последние годы коммерческие решатели и эвристические алгоритмы постепенно стали наиболее распространенный способ оптимизации схемы трубопроводной сети. Рекламный ролик решатель может устранить утомительную работу по программированию алгоритмов, но метод решения ограничено, и сложно изменить встроенные параметры алгоритма. Эвристика алгоритм имеет большую рабочую нагрузку по программированию, но алгоритм имеет хорошую масштабируемость и можно даже объединить преимущества нескольких алгоритмов для получения результатов оптимизации. Устер и Дилавероглу [15] решили задачу оптимизации проектирования и расширения сети газопроводов с помощью онлайн-решателя Bonmin. Чжан и др. [16] разработали всеобъемлющий Модель оптимизации компоновки MILP, учитывающая особенности местности и препятствий. Есть также ученые, которые используют методы теории графов и методы машинного обучения для решать оптимизационные задачи [17,18]. Они использовали решатель Gurobi для глобального решения задачи. модель. Роза и др. [19] представили практичный и эффективный подход к оптимизации подводных работ. производственных сетей с использованием решателя GAMS. Лю и др. [20] выдвинули расширенную Алгоритм оптимизации роя частиц с расширенными возможностями глобального поиска для решения модель MINLP для оптимизации схемы трубопроводной сети. Хонг и др. [21] предложил Модель оптимизации компоновки трубопроводной сети, учитывающая гидравлические характеристики и для решения использовал алгоритм колонии муравьев.

Machine Translated by Google

1.3. Взносы

Наземные инвестиции в газовые месторождения составляют от 30% до 40% общей стоимости инвестиций в разработку, а доля инвестиций в строительство наземных сетей сбора и транспортировки трубопроводов часто составляет от 60% до 70% стоимости наземных инвестиций. Видно, что сеть трубопроводов по сбору и транспортировке природного газа занимает важное место в обустройстве наземного строительства газовых месторождений , а оптимизация ее проектирования может существенно повысить экономический

В настоящее время некоторых традиционных методов оптимизации недостаточно для исследования многоуровневых структур трубопроводных сетей и поиска эффективного решения. Поэтому в данной статье предлагается набор методов оптимизации для более сложной сети газосборных трубопроводов типа «звезда-дерево» в дискретных пространствах.

На основании обзора литературы учеными достигнуты значительные успехи в оптимизации проектирования газопроводных сетей. Однако их внимание в первую очередь было сосредоточено на оптимизации топологий одной сети, игнорируя сложность, связанную с множеством топологических комбинаций, особенно многоуровневыми конфигурациями звездного дерева, преобладающими на нефтяных и газовых месторождениях. Включение различных типов сетей значительно усложняет решение этих моделей. Мы подходим к этому вопросу следующим образом: (1)

Мы устанавливаем модель MINLP для оптимального проектирования сетей звездного дерева в дискретном режи

космос.

(2) Предлагается иерархическая структура оптимизации. На первом этапе основное внимание уделяется

оптимизации конструкции трубопроводной сети звездообразной топологии, соединяющей скважины со станциями.

Второй этап предполагает оптимизацию конструкции древовидной топологии трубопроводной сети, соединяющей станции с станциями.

(3) Предлагается улучшенный генетический алгоритм (IGA) и используется алгоритм Краскала.

для достижения оптимальных компоновочных решений древовидной сети.

В разделе 2 представлена математическая модель. В разделе 3 представлена иерархическая

рамки оптимизации. В разделе 4 представлен практический пример и рассмотрен результат оптимизации. Наконец, в разделе 5 представлено резюме.

2. Математическая модель

В схеме сети трубопроводов для сбора и транспортировки природного газа существует два основных типа моделей: одна — модель MIP, а другая — модель MINLP. В модели MIP заранее рассчитываются различные расстояния между точками. Для разработки газовых месторождений масштаб количества скважин очень велик, а расчет расстояния заранее очень велик; следовательно, в модели данной статьи длина трубопровода от скважин до станций Lij и длина трубопровода между станциями Ljj' являются ев

расстояния, которые представляют собой нелинейные уравнения соответственно $L_{ij} = \left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2)^{0,5}$

и $L_{jj'} = \left(x_j - x_{j'}\right)^2 + \left(y_j - y_{j'}\right)^2)^{0,5}$, кроме того, что может повысить эффективность решения. В

переменная решения о местоположении станции (δ_j), переменная решения о местонахождении завода (δ_k) и переменная решения о подключении трубопровода (δ_{ij},$\delta_{jj'}$) являются двоичными переменными. Таким образом, модель в этой статье является моделью MINLP. Модель обладает хорошей масштабируемостью, что позволит увеличить количество узлов в будущем.

2.1. Целевая функция

Предлагается модель MINLP. Минимальные инвестиции используются в качестве целевой функции модели, как показано в уравнении (1).

$$\text{Мин } F = FPP + FGP + FS + FC \quad (1)$$

где F — инвестиции в трубопроводную сеть, юани. FPP — инвестиции в производственный трубопровод, юани. ПБР — это сбор инвестиций в трубопровод, юани. FS — инвестиции в станцию, юани. FC — инвестиции в завод, юани.

Machine Translated by Google

(1) Инвестиции в производственный трубопровод.

Инвестиции в добывающие трубопроводы – это затраты на подключение всех скважин и станций. в сети, как показано в уравнении (2).

$$FPP = \sum_{i \in I} \sum_{j \in J} \delta ijcijLij \quad (2)$$

где δij — переменная решения о соединении между скважинами и станциями, 0 или 1. cij — цена трубопровода между скважинами и станциями, юань/м. Lij – длина трубопровода от скважин до станций, м. $Lij = \sqrt{xi \quad xj \quad ^2 + йи \quad йджей \quad ^2}^{0,5}$, (xi , yi) – координата скважины, а (xj , yj) – координата станции.

(2) Привлечение инвестиций в трубопровод.

Инвестиции в сбор трубопроводов представляют собой стоимость соединения трубопроводов между всеми станциями сети, как показано в уравнении (3).

$$ПБР = \sum_{j \in J} \sum_{j' \quad J} \delta jj' \, cjj' \, Ljj' \quad (3)$$

где δjj' — переменная решения о соединении между станциями, 0 или 1. cjj' — цена трубопровода между станциями, юань/м. Ljj′ — длина трубопровода между станциями, м. $Ljj' = \sqrt{xj \quad xj' \quad ^2 + yj \quad yj' \quad ^2}^{0,5}$, (xj , yj) и (xj ′ , yj ′) — станции или завод координата.

(3) Инвестиции в станцию.

Инвестиции в станции представляют собой сумму инвестиционных затрат всех станций в сети, как показано в уравнении (4).

$$FS = \sum_{j \in J} \delta j \, fS \quad (4)$$

где δj — переменная решения о местоположении станции, 0 или 1. fS — стоимость станции, юани.

(4) Инвестиции в завод.

Инвестиции в завод представляют собой сумму инвестиционных затрат всех заводов в сети трубопроводов, как показано в уравнении (5).

$$ФК = \sum_{k \in K} \delta k \, фП \quad (5)$$

где δk — переменная решения о местоположении завода, 0 или 1. fP — стоимость завода, юани.

2.2. Ограничения

(1) Ограничение уникальности соединений скважин.

Каждая скважина связана исключительно с одной станцией, как показано в уравнении (6).

$$\sum_{j \in J} \delta ij = 1 \quad i \quad I \quad (6)$$

(2) Ограничение количества заводов.

В сети трубопроводов существует один завод, как показано в уравнении (7).

$$\sum_{k \in K} \delta к = 1 \quad (7)$$

(3) Ограничение на принадлежность скважин и станций.

Ну, меня можно присвоить станции j, если станция j существует, как показано в уравнении (8).

$$\delta ij \quad \delta j \quad i \quad I, \quad j \quad J \quad (8)$$

Machine Translated by Google

(4) Ограничение радиуса передачи.

Давление газа может снизиться во время транспортировки из-за трения труб. Чтобы поддерживать падение давления в разумных пределах, расстояние между скважиной и станцией должно быть меньше определенного значения, известного как радиус передачи, как показано на рисунке 5.

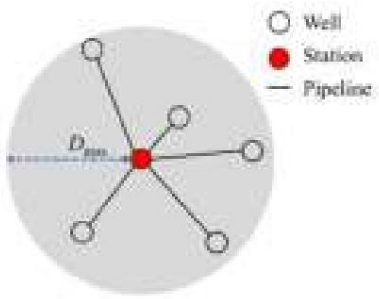


Рисунок 5. Схематическая диаграмма радиуса передачи.

Таким образом, ограничение радиуса передачи рассчитывается, как показано в уравнении (9).

$$\delta_{ij} L_{ij} \quad D_{max} \quad i, \quad j \quad J, \tag{9}$$

где D_{max} – максимально допустимая длина трубопровода от скважины до станции, м.

(5) Ограничение мощности станции.

Как правило, ограничение радиуса сбора не учитывается при решении проблемы ограничения пропускной способности станции, которое подпадает под проблему местоположения мощного оборудования (CFLP). В этой статье рассматриваются как ограничение радиуса, так и ограничение мощности. Когда ограничение радиуса сбора удовлетворено, а ограничение пропускной способности — нет, схема трубопроводной сети будет перепроектирована, как показано на рисунке 6.

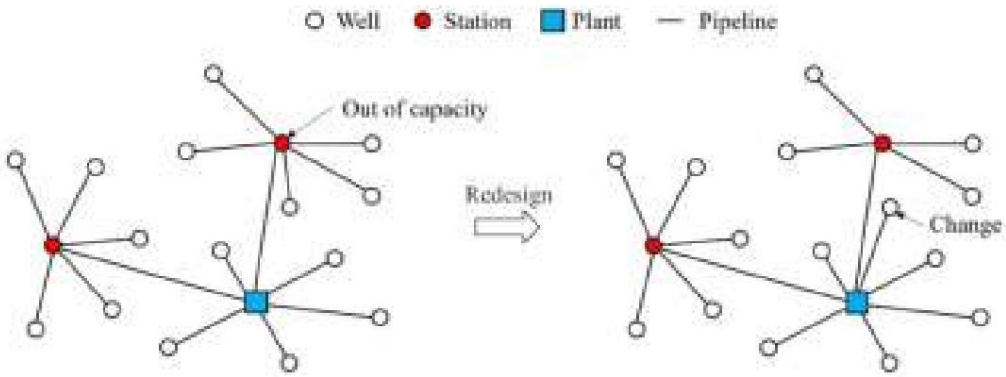


Рисунок 6. Реконструкция трубопроводной сети с нарушением ограничений пропускной способности.

Таким образом, мощность станции не должна превышать мощность, указанную в уравнении (10).

$$\delta_j q_j \quad q_{max} \quad j \quad J \tag{10}$$

где q_j – узловой расход газа станции j, м3/сут. q_{max} – производительность станции, м3/сут.

(6) Ограничение древовидной структуры.

Ограничение соединения дерева гарантирует, что сгенерированная топология будет самой короткой и без петель, как показано в уравнении (11).

$$\delta_{jj} = 2 \tag{11}$$

Machine Translated by Google

(7) Ограничение потока в узлах.

Ограничение потока в узле гарантирует, что общий объем газа остается постоянным, как показано в уравнении (12).

$$\text{ЦИ} \quad \delta kqk = 0 \qquad (12)$$

$$i\varepsilon I \quad k\varepsilon K$$

где qi – узловой расход газа скважины i, м3/сут. qk – узловой расход установки k, м3/сут.

2.3. Переменные решения

Переменные решения включают в себя переменную решения о местоположении станции (δj), переменную решения о местоположении завода (δk) и переменную решения о подключении трубопровода (δij,δjj'). Все эти переменные решения являются двоичными переменными, как показано в уравнении (13).

$$\delta j\ ,\ \delta k\ ,\ \delta ij,\ \delta jj' = \{0,\ 1\}\ i \quad I,\ j \quad J,\ j \quad ' \quad J,\ k \quad K \qquad (13)$$

3. Метод решения.

Оптимизация проектирования трубопроводной сети типа «звезда-дерево» признана NP-сложной, что затрудняет поиск комплексного решения [22]. Для решения этой проблемы в данном исследовании предлагается иерархическая структура оптимизации. Схема состоит из двух этапов: первый этап включает определение разделения групп скважин и местоположения станций в сети с топологией «звезда», а второй этап направлен на установление соединений станций и местоположений заводов в сети с древовидной топологией. Примечательно, что оптимизация сети с топологией «звезда» на первом этапе служит основополагающим и ключевым компонентом всей проблемы оптимизации. Ее сложность и масштаб превосходят оптимизацию сети второго этапа с древовидной топологией. Поэтому данная статья концентрируется на улучшении оптимизации конструкции и представляет IGA, адаптированный для решения этой проблемы. Кроме того, алгоритм Крускала, использующий подход минимального связующего дерева, используется для оптимизации проектирования конвейерной сети второго этапа с древовидной топологией.

3.1. Базовый генетический

алгоритм ГА — это эвристический метод поиска, основанный на принципах естественного отбора и генетики. Он направлен на оптимизацию распределения объектов путем моделирования эволюционного процесса биологических хромосом. Основной процесс решения ГА [23] разворачивается следующим образом:

(1) Кодирование: закодируйте информацию о распределении лунок и станций в последовательность хромосом.

(2) Генерация начальной популяции: фиксированный размер хромосом (особей) генерируется случайным образом для формирования начальной популяции P(t) (в данной статье генерируются 200 групп). Здесь t обозначает счетчик эволюционных генераций, а T представляет максимальное количество циклов эволюции, установленное для алгоритма.

(3) Функция приспособленности: каждый человек в популяции оценивается на основе функции приспособленности, обычно с использованием общей стоимости инвестиций в качестве критерия оценки.

(4) Отбор и наследование: исходная популяция отбирается и скрещивается путем моделирования процесса биологической эволюции, а популяция следующего поколения P(t + 1) получается на этапе 3.

(5) Оценка условия завершения: когда t T, тогда t + 1 заменяет t, и мы переходим к шагу 2; если t > T, мы рассматриваем индивидуума с оптимальной приспособленностью в процессе эволюции как оптимальное решение и прекращаем операцию [24].

Вышеописанный процесс также описывается блок-схемой, показанной на рисунке 7.

Machine Translated by Google

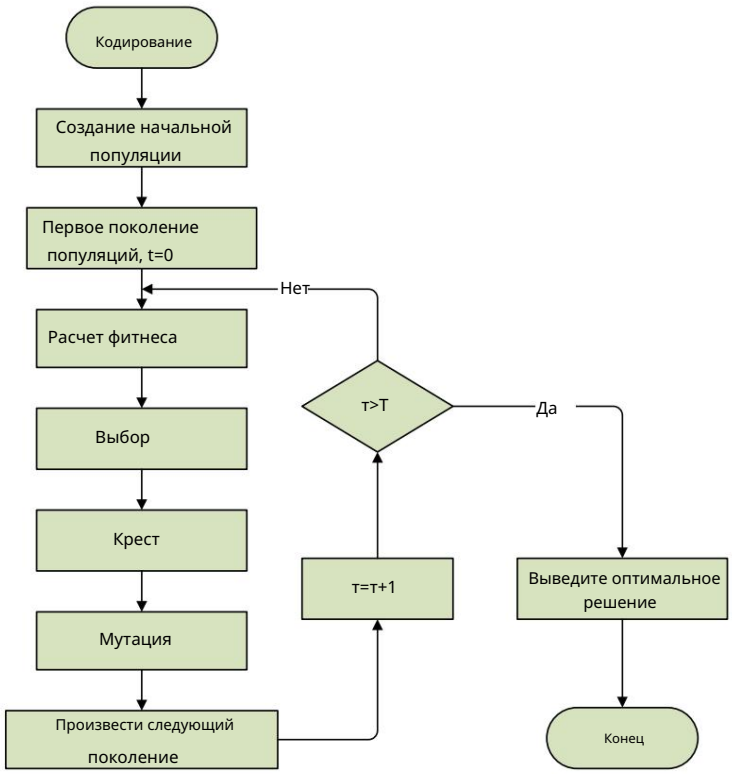


Рисунок 7. Блок-схема базового ГА.

3.2. Усовершенствование алгоритма

Учитывая дискретную природу пространства и специфическую структуру газопромысловых трубопроводных сетей, традиционный ГА и связанные с ним параметры могут не полностью соответствовать математической модели, предложенной в данной статье. Поэтому важно усовершенствовать базовый ГА, чтобы он лучше соответствовал этим характеристикам и эффективно оптимизировал проект трубопроводной сети. Улучшение включает в себя общее кодирование соответствия скважина-станция, формулировку функции пригодности сетевого решения, а также перепроектирование оператора скрещивания и оператора мутации.

(1) Дизайн кодирования.

Принимая во внимание характеристики ограничений, используя связь между скважинами и станциями в качестве основы кодирования, для повышения эффективности ГА используется кодирование действительных чисел. Последовательность, кодирующая хромосому, структурирована таким образом , что она напрямую влияет на целевую функцию, которая в данном случае представляет собой общий объем инвестиций. Длина каждой хромосомы соответствует общему количеству лунок, а внутри последовательности идентификатор каждой лунки систематически представлен в порядке возрастания. Этот метод кодирования гарантирует, что ГА оптимально учитывает проектные ограничения и эффективно нацелен на оптимизацию конфигурации сети газопроводов. Соответствующий номер станции каждой лунки стохастически генерируется в каждой позиции внутри хромосомы, чтобы обозначить связь между лункой и соответствующей станцией. В предложенной схеме хромосомного кодирования каждая цифра хромосомы = (C1, C2, ... Cm) представляет номер станции, соответствующей каждой лунке.

(2) Конструкция кроссоверного оператора.

Для ГА метод работы кроссовера и перекрестная вероятность являются чрезвычайно важными параметрами, которые напрямую влияют на качество оптимизации алгоритма. Используя данные CAP101 на основе OR-LIBRARY (50 скважин, 25 станций) [25], проверяются метод работы кроссовера и вероятность кроссовера на предмет их влияния на решение, а также определяются лучший режим кроссовера и вероятность кроссовера.

Каждый из методов кроссовера тестировался 50 раз, и их результаты усреднялись для сравнения. По мере увеличения количества итеративных генераций алгоритм постепенно совершенствовался.

Machine Translated by Google

оптимальному решению. Среди различных оцениваемых методов кроссовера, многоточечного пересечения продемонстрировал превосходную производительность, что привело к его выбору. Рисунок 8 иллюстрирует средние результаты теста.

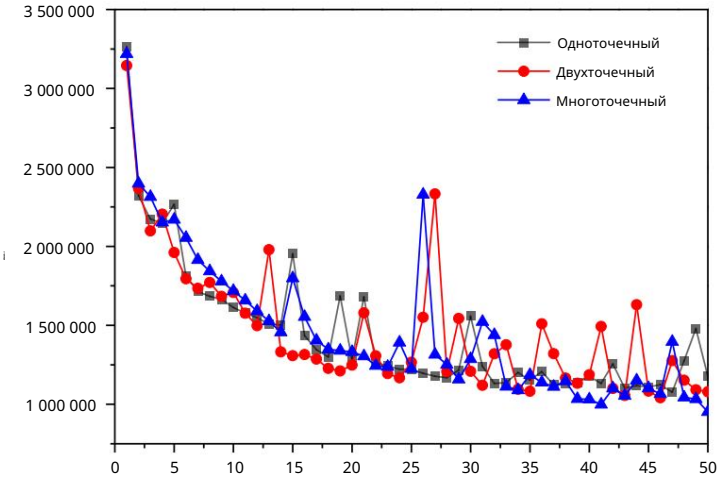


Рисунок 8. Кривая сходимости различных методов работы кроссовера.

Вероятность кроссовера варьируется от 0,5 до 0,9. Сравнение эффектов этих различных вероятностей пересечения результатов было проведено путем усреднения Результаты 50 тестов. Когда вероятность пересечения находится между 0,7 и 0,9, результаты похожи. Правильное увеличение вероятности пересечения может улучшить сходимость. решения, поэтому вероятность пересечения должна поддерживаться на высоком уровне при проектировании. На рисунке 9 показано влияние различных вероятностей пересечения на результаты многоточечного анализа. метод пересечения.

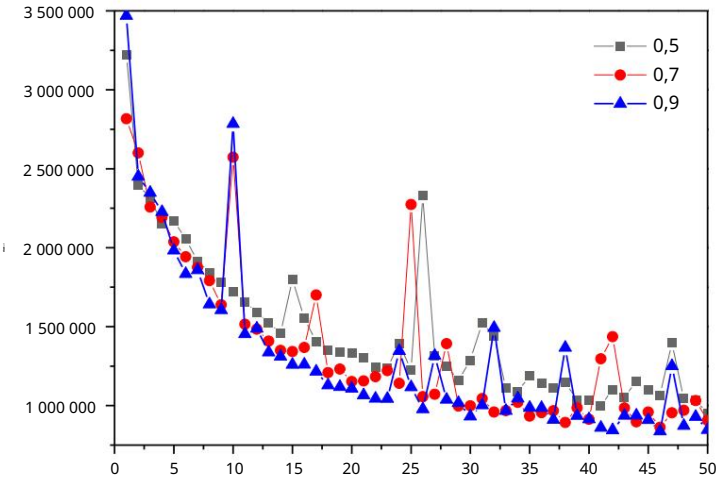


Рисунок 9. Кривая сходимости различных вероятностей пересечения.

(3) Конструкция оператора мутации.

Оператор мутации представляет собой меру, которая предотвращает попадание ГА в оптимально путем моделирования генетических и эволюционных значений аллели разрыва [26]. В процессе отладки оператора мутации тестируется вариационная функция, позволяющая ГА В процессе отладки оператора мутации для ГА тестируется функция вариации, чтобы снизить риск попадания в локальные оптимумы на более поздних этапах. Операторы мутации используют адаптивный подход, как указано в формуле (14).

$$= + \quad (14)$$

$$= +$$

Machine Translated by Google

снизить риск попадания в локальный оптимум на более поздних стадиях. Операторы мутации используют адаптивный подход, как указано в формуле (14).

$$\begin{cases} MP = \varepsilon \cdot \dfrac{Fitmax-Fit}{Fitmax-Fitmin} & Подходит \ \dfrac{Fitmax+Fitmin}{2} \\ MP = \varepsilon & Подходит < \dfrac{Fitmax+Fitmin}{2} \end{cases} \quad (14)$$

где MP – вероятность мутации; Fit – это показатель пригодности; Фитмакс и Фитмин – это максимальное и минимальное значение приспособленности каждого поколения; ε — коэффициент мутации.

В результате тестирования было обнаружено, что когда коэффициент мутации ниже 0,003, итеративная устойчивость и сходимость всего процесса решения лучше, а решение время более разумно; однако, если коэффициент мутации имеет значение более 0,1, определенное решение может быть далеко от оптимального решения, что влияет на стабильность итерации, скорости сходимости и других проблем. Следовательно, мутация коэффициент был выбран ниже 0,003.

Влияние вероятности мутации на инвестиции отражено в Таблице 1. эксперименты с CAP101 показывают, что когда вероятность мутации установлена на определенное значение ценность, это обеспечивает разнообразие населения без ущерба для конвергенции ГА. к глобальному оптимальному решению. Основная причина этого заключается в том, что когда действительное число используется кодирование, каждая позиция на каждой хромосоме содержит больше информации, чем при используются другие методы кодирования, и когда вероятность мутации высока, легко непосредственно уничтожить доминирующую особь.

Таблица 1. Влияние различных вероятностей мутаций на производительность ГА.

Размер проблемы	Общее количество вычислений	ε	Средний оптимальный результат
50 × 25	10	0,0001	861 682,74
		0,0002	830 511,15
		0,0003	818 015,08
		0,0004	796 668,16
		0,0005	806 771,78
		0,001	809 352,78
		0,002	855 745,99
		0,003	834 490,13

3.3. IGA-тестирование

Чтобы лучше решить оптимальную конструкцию трубопроводных сетей типа «звезда-дерево» в дискретных пространствах, в этой статье особое внимание уделяется усовершенствованиям схемы кодирования, оператора кроссовера и оператора мутации основного ГА. Шаги решения IGA показаны на рисунке 10.

Проверить эффективность модифицированного генетического алгоритма при решении конвейерной задачи. Проблема проектирования сети, была проведена серия тестов алгоритма. Все алгоритмы были реализованы в программном обеспечении MATLAB 2022. Численные эксперименты были выполнены на компьютере, оснащенном процессором Intel(R) Core (TM) i5-2450M с частотой 2,50 ГГц, 4 ГБ Оперативная память и операционная система Windows 7.

Для сравнения с улучшенным ГА были выбраны два классических алгоритма, а именно: Оптимизация роя частиц с коэффициентом сужения (CPSO) и поиск табу (TABU). Несколько задачи в OR-LIBRARY использовались для алгоритмического тестирования, а решение Результаты представлены в Таблице 2. ЛОКАЛЬНОЕ – это фактическое оптимальное решение. n означает хорошо число. m означает номер станции. Для меньшего количества скважин и станций, таких как в случае CAP101 преимущества использования IGA не очевидны. Фактически, IGA может потребовать больше времени вычислений по сравнению с другими алгоритмами. Однако, поскольку количество n (скважин) и m (станций) увеличивается, преимущества эффективности IGA в

Machine Translated by Google

решение проблем становится все более очевидным. имеет тенденцию сходиться к оптимальному решению быстрее, чем другие алгоритмы, по мере увеличения n и m .

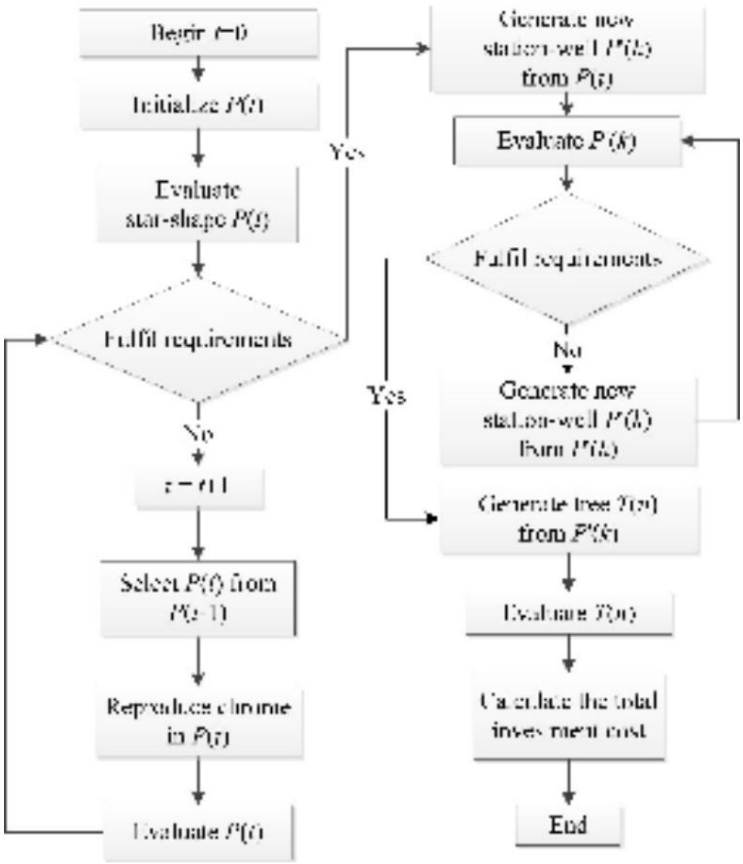


Рисунок 10. Блок-схема IGA.

Таблица 2. Сравнение с другими алгоритмами.

Проблема	Размер (n × м)	Алгоритм	Оптимальное решение (пр.)	Время (с)
CAP101	50 × 25	МЕСТНЫЙ	796 648,437	-
		ТАБУ	797 508,70	0,19
		КПСО	796 648,44	0,88
		ИГА	796 668,16	0,37
CAP103	50 × 25	МЕСТНЫЙ	893 782,11	-
		ТАБУ	894 008,10	0,24
		КПСО	893 782,11	0,99
		ИГА	895 192,66	0,17
CAP104	50 × 25	МЕСТНЫЙ	928 941,75	-
		ТАБУ	928 941,80	0,15
		КПСО	928 941,75	0,60
		ИГА	928 946,48	0,17

Machine Translated by Google

Таблица 2. Продолжение.

Проблема	Размер (n × м)	Алгоритм	Оптимальное решение (пр.)	Время (с)
CAP131	50 × 50	МЕСТНЫЙ	793 439,56	-
		ТАБУ	794 299,90	0,20
		КПСО	793 439,56	3,62
		ИГА	793 902,48	0,37
CAP133	50 × 50	МЕСТНЫЙ	893 076,71	-
		ТАБУ	893 782,10	0,30
		КПСО	893 076,71	3,78
		ИГА	893 994,01	0,22
КАПБ	1000 × 100	МЕСТНЫЙ	13 198 556,43	-
		ТАБУ	13 214 718,1	3,91
		ИГА	13 211 519,82	3.26

3.4. Алгоритм Краскала

При соединении каждой станции с использованием древовидной топологии основной задачей является решение Задача о минимальном связующем дереве. При проектировании минимального остовного дерева обычно используются алгоритмы Прима и алгоритм Краскала, ядро которых основан на жадном принципе. Сравнивая два алгоритма, выяснилось, что Алгоритм Краскала всегда превосходит алгоритм Прима по времени решения, поэтому мы выбрали использовать алгоритм Крускала в качестве алгоритма проектирования сети с древовидной топологией. Краскал Алгоритм представляет собой алгоритм связующего дерева, основанный на взвешенных ребрах. В процессе решения, последовательно выбирается ребро с наименьшим весом и добавляется в набор деревьев. Когда добавленное ребро генерирует циклическую топологию, взвешенное ребро удаляется до тех пор, пока все ребра не будут был выбран [27].

Процесс решения алгоритма Краскала показан на рисунке 11. Все узлы разделен на набор узлов соединения U и набор узлов разброса без соединения V. Вначале V = {v1, v2, v3, v4, v5} и U = {φ}. Во-первых, узлы, образующие самое короткое ребро, выбран и добавлен в набор узлов соединения, то есть U = {v4, v5} и V = {v1, v2, v3}. Затем в таким же образом ребра v1, v2, v1, v3, v4, v2 и v4, v3 выбираются в набор узлов соединения. Когда создается циклическая типология, создавшее ее ребро удаляется, и решение завершено. Наконец, V = {φ} и U = {v1, v2, v3, v4, v5}.

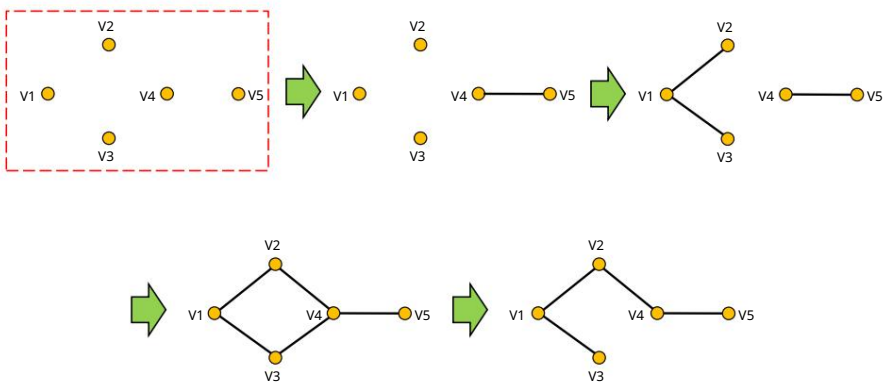


Рис. 11. Принципиальная схема процесса решения алгоритма Краскала.

4. Результаты и обсуждение.

Для системы трубопроводной сети сбора и транспортировки природного газа схема проектирование — это NP-задача, не способствующая расчетам. Чтобы улучшить эффективности решения сделаны следующие четыре предположения: (1) Для внешнего

Machine Translated by Google

условиях расчета в топологии, стандартные условия становятся стандартными для расчета (давление 0,1 МПа, температура 20 C). (2) Скорость потока в газопроводе нестабильна. Для решения проблемы удобства течение газа в трубопроводе упрощено как стабильное течение. (3) В трубопроводе температура подземного трубопровода обычно меняется в зависимости от длины трубопровода, а когда трубопровод слишком длинный, температура газа в конечном итоге становится близкой к температуре окружающей среды. Для облегчения расчетов, а также поскольку сеть собирающих трубопроводов обычно расположена в одном и том же районе, в данной работе температура трубопровода предполагается постоянной. (4) Для направления потока газа в сети предусмотрено, что каждый узел может транспортировать газ только из одного направления в другое.

Для анализа и поиска решения была выбрана реальная сеть газопроводов в Китае [28]. Первоначальная схема сети трубопроводов на газовом месторождении представлена на рисунке 12. Это газовое месторождение состоит из 93 скважин, каждая из которых производит от 2000 до 5000 м3/сутки газа, с устьевым давлением от 0,2 до 1,0 МПа и средней добычей. температура 20 C. Спецификация первичных добывающих трубопроводов между скважинами и станциями составила De160 × 9,5 мм, а вторичных сборных трубопроводов между станциями (и заводом) – De225 × 12,8 мм. Подробные затраты представлены в Таблице 3.

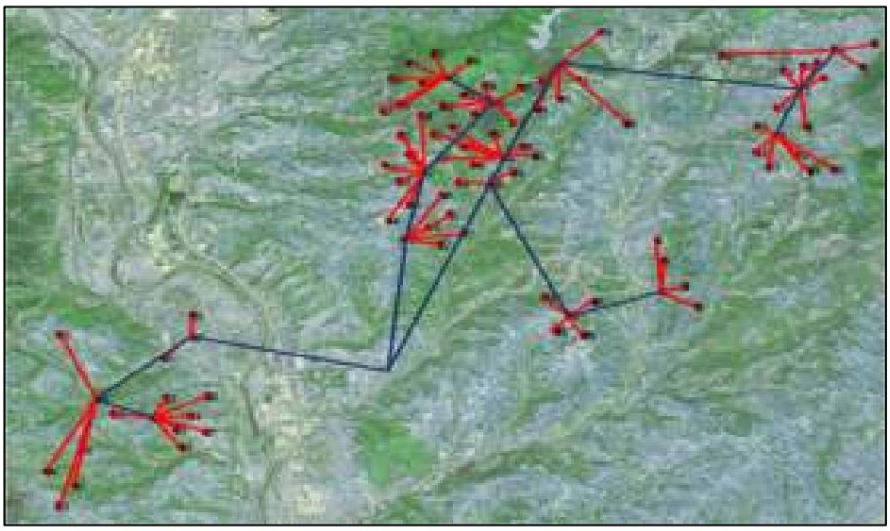


Рисунок 12. Эскиз исходной схемы трубопроводной сети.

Таблица 3. Прогнозируемые затраты на трубопроводную сеть.

Проект	Расходы
Станция	52,17 юаней × 104
Растение	1725,93 юаней × 104
Стоимость трубопровода (De160 × 9,5 мм)	юаней/км 15,636 × 104 юаней/
Стоимость трубопровода (De225 × 12,8 мм)	км 25,406 × 104

Максимальное ограничение радиуса составляло 1,5 км, а производительность одной станции — 20 000 м3/день. Поскольку данные получены из газового месторождения с низким давлением, ограничения по давлению в данном случае не учитывались. В исходном алгоритме (ГА) расчет ffi занимал более 2 часов за раз. После оптимизации структуры алгоритма (IGA) время расчета было сокращено до 15–30 мин, что повысило эффективность расчета более чем на 400%. IGA использовался для решения схемы расположения трубопроводной сети.

Используя сложившуюся структуру иерархической оптимизации, мы разработали оптимизированную схему компоновки. Первоначальная топология звезды на первом этапе показана на рис.

Machine Translated by Google

Рисунок 13. На рисунке 14 показаны результаты группировки скважин и определения местоположения станции . Окончательная оптимизированная схема показана на рисунке 15. Эта сеть состоит из четырнадцати газосборных станций и одной очистной станции, что на одну станцию меньше, чем в первоначальном проекте газового месторождения.



Рисунок 13. Связь между скважинами и станциями.

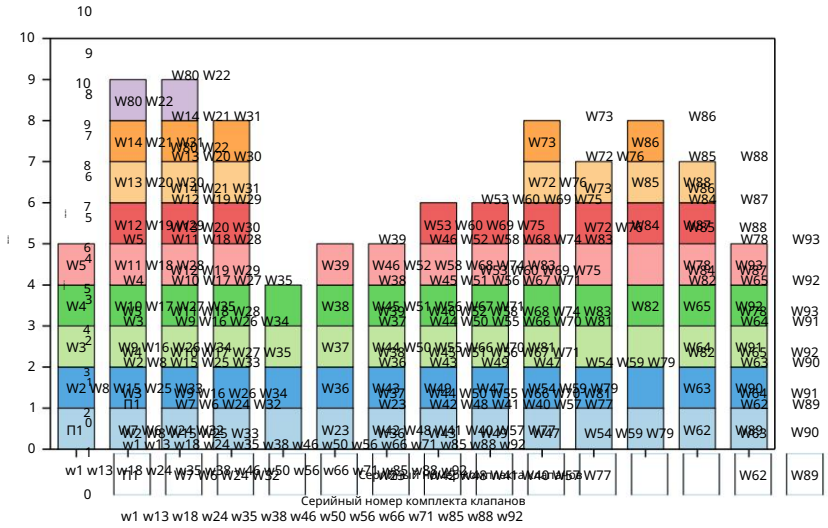


Рисунок 14. Результаты распределения скважин, расположения станции и завода.

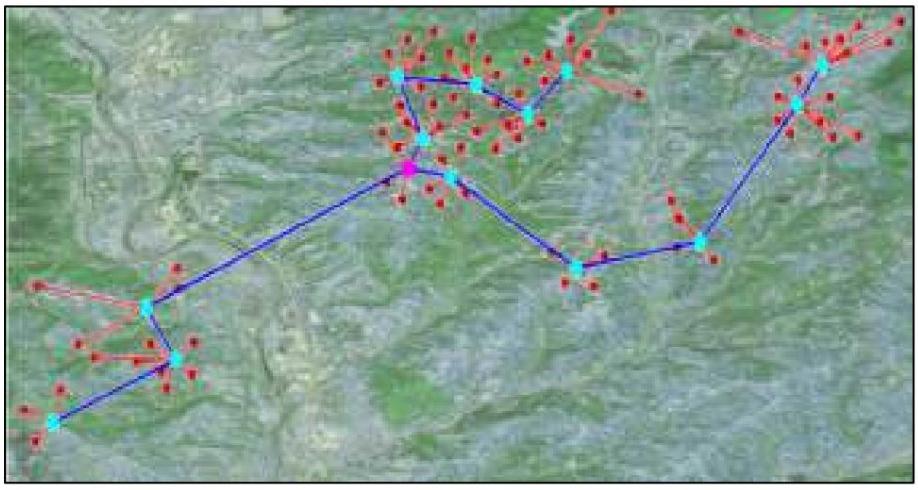


Рисунок 15. Результат проектирования схемы трубопроводной сети типа «звезда-дерево».

Machine Translated by Google

Используя программное обеспечение для моделирования для проверки осуществимости структуры компоновки трубопроводной сети и учитывая, что количество скважин в трубопроводной сети было слишком большим, количество скважин определялось как сумма объема газа и узла клапанного манифольда. для проверки соединения. Схема подключения представлена на рисунке 16.
Соответствующая калибровка перепада давления показана на рисунке 17. Падение давления находилось в пределах допустимого диапазона перепада давления, что соответствует требованиям транспортировки газа.

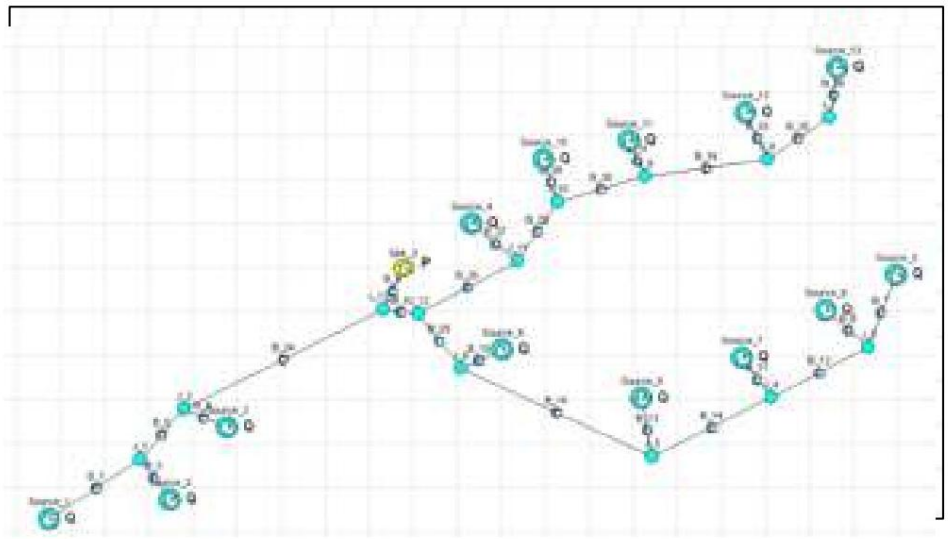


Рис. 16. Иллюстрация моделирования подключения трубопроводной сети.

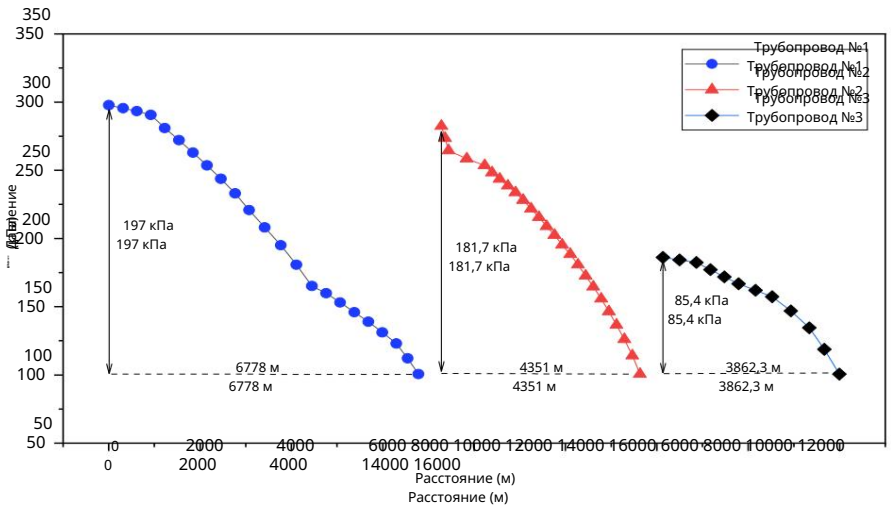
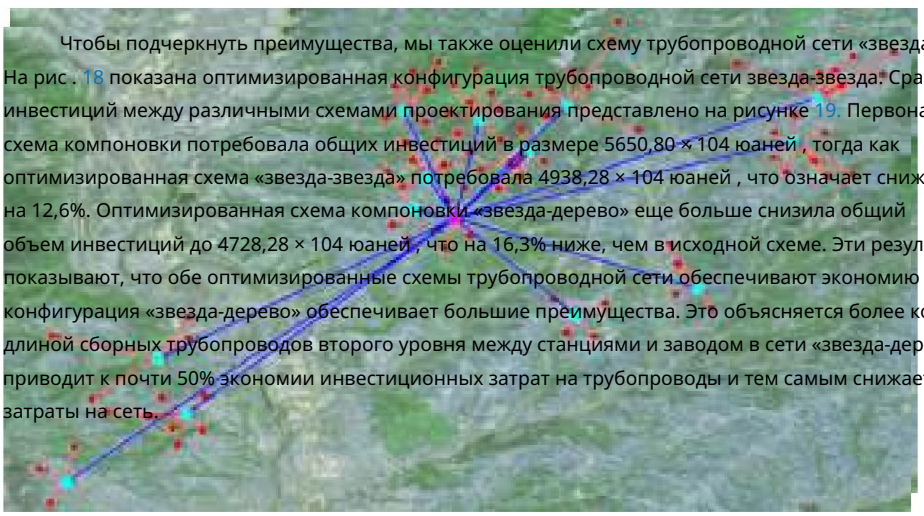


Рис. 17. Результаты калибровки перепада давления на схеме трубопроводной сети.



Чтобы подчеркнуть преимущества, мы также оценили схему трубопроводной сети «звезда-звезда». На рис . 18 показана оптимизированная конфигурация трубопроводной сети звезда-звезда. Сравнение инвестиций между различными схемами проектирования представлено на рисунке 19. Первоначальная схема компоновки потребовала общих инвестиций в размере 5650,80 × 104 юаней , тогда как оптимизированная схема «звезда-звезда» потребовала 4938,28 × 104 юаней , что означает снижение затрат на 12,6%. Оптимизированная схема компоновки «звезда-дерево» еще больше снизила общий , который объем инвестиций до 4728,28 × 104 юаней , что на 16,3% ниже, чем в исходной схеме. Эти результаты показывают, что обе оптимизированные схемы трубопроводной сети обеспечивают экономию средств, а конфигурация «звезда-дерево» обеспечивает большие преимущества. Это объясняется более короткой длиной сборных трубопроводов второго уровня между станциями и заводом в сети «звезда-дерево», что приводит к почти 50% экономии инвестиционных затрат на трубопроводы и тем самым снижает общие затраты на сеть.

Machine Translated by Google

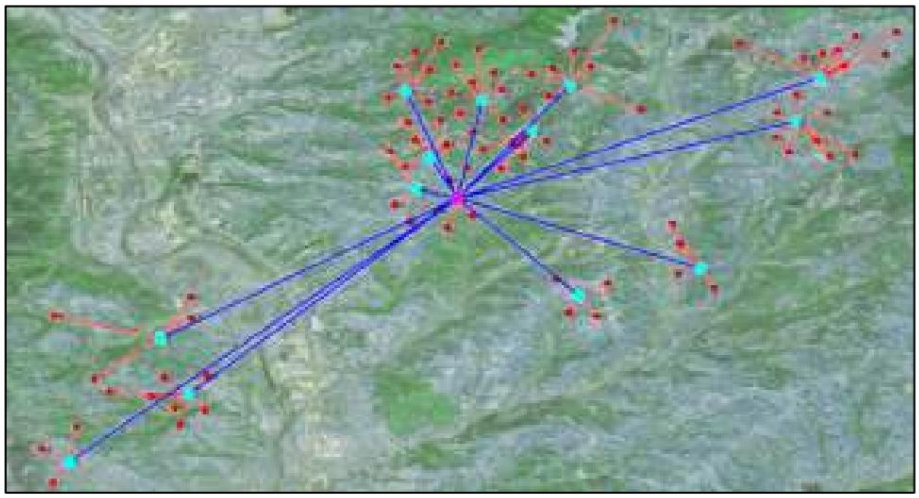


Рисунок 18. Результат проектирования схемы трубопроводной сети звезда-звезда.

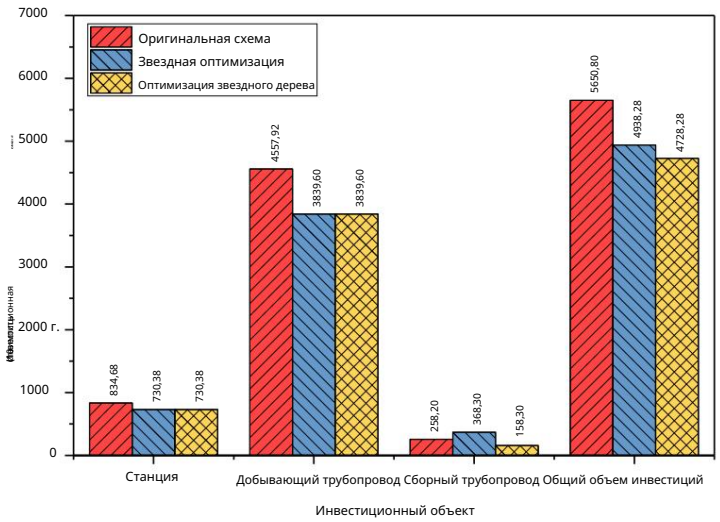


Рисунок 19. Сравнение инвестиционных затрат.

5. Выводы

Научные исследования по оптимизации схем газопроводных сетей преимущественно сосредоточены на одноуровневых, одноструктурных сетях, при этом ограниченное внимание уделяется многоуровневым и многотипным комбинированным сетям газопроводов. Чтобы устранить этот пробел, в данной статье создана модель оптимизации проектирования двухступенчатой сети газосборных трубопроводов типа «звезда-дерево» в дискретных пространствах. Признавая сложность проблемы и необходимость повышения вычислительной эффективности, наряду с IGA разрабатывается иерархическая структура оптимизации для эффективного решения этой проблемы оптимизации. ffi

Опираясь на эту модель, мы определили оптимизированный проект сети трубопроводов типа «звезда-дерево» для месторождения природного газа в Китае, что привело к оптимальным результатам проектирования. Результаты показывают, что оптимизированная схема компоновки сети обеспечивает сокращение инвестиционных затрат на 16% по сравнению с исходной схемой, что составляет экономию в размере 922,52×104 юаней . Кроме того , мы сравнили эту оптимизированную схему с топологией звезда-звезда. Было замечено , что использование схемы «звезда-звезда» увеличит инвестиционные затраты на 4% по сравнению со схемой расположения «звезда-дерево». Это сравнение подчеркивает превосходную применимость топологии звездообразного дерева для этой конкретной сетевой системы.

Хотя модель, созданная в данной статье, эффективно решает задачу оптимизации проектирования трубопроводной сети типа «звезда-дерево», она по-прежнему имеет следующие недостатки: Модель не учитывает гидравлические характеристики потока природного газа, поэтому она не может эффективно

Machine Translated by Google

судить, соответствуют ли изменения давления жидкости в трубопроводной сети технологическим требования. Кроме того, стратегия иерархической оптимизации, использованная в этом исследовании не гарантирует, что полученное решение является глобальным оптимальным решением. Поэтому будущее исследования должны всесторонне учитывать характеристики эксплуатации газа в трубопроводе. Модель оптимизации сетевого дизайна. Кроме того, принятие более систематического глобального Метод оптимизации будет иметь важное значение для эффективного решения этой проблемы.

Вклад автора: Проверка, YQ; формальный анализ, HC; расследование, Дж. Х.; программное обеспечение, ПД; данные курирование, DL; письмо — первоначальный вариант, YL и JZ; написание — обзор и редактирование, YL и JZ All авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Это исследование финансировалось за счет гранта [Национального фонда естественных наук Китая]. номер [51704253].

Заявление о доступности данных. Данные, подтверждающие выводы этого исследования, доступны на сайте соответствующим авторам по обоснованному запросу.

Благодарности: Авторы выражают благодарность за финансовую поддержку, оказанную Национальный фонд естественных наук Китая (51704253) для этого проекта.

Конфликт интересов: авторы Юй Линь и Яньхуа Цю работали в компании PetroChina. Юго-западная нефтегазопромысловая компания. Автор Хао Чен работал в компании Сычуань. Компания Shale Gas Exploration and Development Co., Ltd. Автор Дафань Лю работал в компании CNPC Chuanqing Drilling Engineering Co., Ltd. Остальные авторы заявляют, что исследование проводилось в отсутствие каких-либо коммерческих или финансовых отношений, которые могли бы быть истолкованы как потенциальный конфликт интересов.

Номенклатура

Наборы	
I	Хорошо задано, i I = {1, 2, . . . , Н.И.}
$J_ж$	Набор позиций кандидатов на станцию, j J = 1, 2, . . . , Нью-Джерси
K	Множество позиций кандидатов на объект, k K = {1, 2, . . . , НК}
Параметры	
$Ф$	Инвестиции в трубопроводную сеть, CNY
$c_{сидж}$	Цена трубопровода между скважинами и станциями, юаней/м
$c_{jj'}$	Цена трубопровода между станциями, юаней/м
$D_{макс}$	Максимально допустимая длина трубопровода от скважины до станции, м
$ф_П$	Стоимость завода, CNY
$ф_с$	Стоимость станции, CNY
$Ф_К$	Инвестиции в завод, CNY
$Ф_С$	Инвестиции в станцию, CNY
$Ф_{ПП}$	Инвестиции в производственный трубопровод, CNY
$П_{БР}$	Привлечение инвестиций в трубопровод, CNY
$L_{идж}$	Длина трубопровода от скважин до станций, м
$L_{jj'}$	Длина трубопровода между станциями, м
q_j	Узловой расход газа станции j, м3/сут
q_{max}	Производительность станции, м3/сут
q_i	Узловой расход газа скважины i, м3/сут
$q_к$	Узел расхода газа установки k, м3/сут
$δ_{ij}$	Переменная решения соединения между скважинами и станциями, 0 или 1
$δ_j$	Переменная решения о местоположении станции, 0 или 1
$δ_{jj'}$	Переменная решения о соединении между станциями, 0 или 1
$δ_k$	Переменная решения о местоположении завода, 0 или 1

Machine Translated by Google

Ссылки

1. Чжоу Дж.; Фу, Т.; Чен, Ю.; Сяо, Ю.; Пэн, Дж.; Лян, Г. Модель смешанного целочисленного нелинейного программирования для оптимального проектирования естественных Наземная двухтрубная сеть газохранилища. J. Energy Storage 2021, 44, 103379. [CrossRef]
2. Риос-Меркадо, Р.; Боррас-Санчес, К. Проблемы оптимизации систем транспортировки природного газа: современный обзор. Прил. Энергетика 2015, 147, 536–555. [Перекрестная ссылка]
3. Демисси А.; Чжу, В.; Белачев, Коннектикут. Модель многокритериальной оптимизации эксплуатации газопровода. Вычислить. хим. англ. 2017, 100, 94–103. [Перекрестная ссылка]
4. Хонг, Б.; Ли, Х.; Ди, Г.; Песня, С.; Ю, В.; Чен, С.; Ли, Ю.; Гонг, Дж. Интегрированная модель MILP для оптимального планирования многопериодных система береговых газосборных трубопроводов. Вычислить. Индийский англ. 2020, 146, 106479. [CrossRef]
5. Санайе, С.; Махмудимер, Дж. Оптимальный проект схемы сети транспортировки природного газа. хим. англ. Рез. Дес. 2013, 91, 2465–2476. [Перекрестная ссылка]
6. Родригес, Д.А.; Отейза, ПП; Бриньоль, Н.Б. Оптимизация моделирования отжига для сетей углеводородных трубопроводов. Индийский англ. хим. Рез. 2013, 52, 8579–8588. [Перекрестная ссылка]
7. Фу, Ю. Х. Многокритериальное проектирование оптимизации параметров однотрубной кольцевой сети нефтесборных водоводов смешения. Ключ англ. Матер. 2011, 467, 1285–1290. [Перекрестная ссылка]
8. Ван, Б.; Юань, М.; Чжан, Х.; Чжао, В.; Лян, Ю. Модель MILP для оптимального проектирования многопериодной сети транспортировки природного газа. хим. англ. Рез. Дес. 2018, 129, 122–131. [Перекрестная ссылка]
9. Чжоу Дж.; Чжоу, Х.; Лян, Г.; Пэн, Дж. Модель MINLP для сетевой компоновки подземных хранилищ природного газа. Дж. Интелл. Нечеткая система. 2020, 38, 4619–4642. [Перекрестная ссылка]
10. Даскин М. Сетевые и дискретные модели местоположения, алгоритмы и приложения. Дж. Опер. Рез. Соц. 1997, 48, 763–764. [Перекрестная ссылка]
11. Поста, М.; Ферланд, Дж.А.; Мишелон, П. Точный кооперативный метод решения проблемы размещения неиспользуемых объектов. Математика. Программа. Компьютер 2014, 6, 199–231. [Перекрестная ссылка]
12. Балинский М.Л. Целочисленное программирование: Методы, использование, вычисления. Менеджер. наук. 1965, 12, 253–313. [Перекрестная ссылка]
13. Клинцевич, Ю.Г.; Лусс, Х.; Розенберг Э. Оптимальные и эвристические алгоритмы размещения многопродуктовых непроизводительных объектов. Евро. Дж. Опер. Рез. 1986, 26, 251–258. [Перекрестная ссылка]
14. Эфроймсон, М.А.; Рэй, Т.Л. Алгоритм с ветвями определения местоположения предприятия. Опер. Рез. 1966, 14, 361–368. [Перекрестная ссылка]
15. Юстер, Х.; Дилавероглу С. Оптимизация проектирования и эксплуатации сетей транспортировки природного газа. Прил. Энергетика 2014, 133, 56–69. [Перекрестная ссылка]
16. Чжан Х.; Лян, Ю.; Чжан, В.; Ван, Б.; Ян, Х.; Ляо, К. Единая модель MILP для топологической структуры добывающей скважины. сбор трубопроводной сети. Дж. Пет. наук. англ. 2017, 152, 284–293. [Перекрестная ссылка]
17. Босиков И.И.; Клюев Р.В.; Силаев, ИВ; Стас Г.В. Комплексная оценка сложности и технологичности формализованной вентиляции. процессы в угольных шахтах. Поддерживать. Дев. Гора Террит. 2023, 15, 516–527. [Перекрестная ссылка]
18. Нгуен, Х.; Буй, XN; Чой, Ю.; Топал, Э. Применение искусственного интеллекта для прогнозирования устойчивости откосов в карьерах: тематическое исследование с использованием новой нейронной сети с радиальной базисной функцией, основанной на империалистическом конкурентном алгоритме. «Применение искусственного интеллекта в горном деле, геотехнике и геоинженерии»; Elsevier: Амстердам, Нидерланды, 2024 г.; стр. 97–111.
19. Роза, В.Р.; Кампоногара, Э.; Феррейра Фильо, VJM Оптимизация проектирования подводной инфраструктуры нефтяных месторождений с размещением манифольдов и расположение трубопровода. Вычислить. хим. англ. 2018, 108, 163–178. [Перекрестная ссылка]
20. Лю, Ю.; Чен, С.; Гуань, Б.; Сюй, П. Оптимизация компоновки крупномасштабной системы сбора нефти и газа на основе комбинированной стратегии оптимизации. Нейрокомпьютинг 2019, 332, 159–183. [Перекрестная ссылка]
21. Хонг, Б.; Ли, Х.; Ди, Г.; Ли, Ю.; Лю, Х.; Чен, С.; Гонг, Дж. Интегрированный метод MILP для сбора трубопроводных сетей с учетом гидравлические характеристики. хим. англ. Рез. Дес. 2019, 152, 320–335. [Перекрестная ссылка]
22. Монтанья, А.Ф.; Кафаро, округ Колумбия; Гроссманн, IE; Берч, Д.; Шао, Ю.; Ву, XH; Фурман, К. Проектирование трубопроводной сети для сбора нетрадиционной нефти и газа с использованием математической оптимизации. Оптим. англ. 2023, 24, 539–589. [Перекрестная ссылка]
23. Словик А.; Квасница, Х. Эволюционные алгоритмы и их приложения к инженерным задачам. Нейронный компьютер. Прил. 2020, 32, 12363–12379. [Перекрестная ссылка]
24. Тохьяма, Х.; Ида, К.; Мацуэда, Дж. Генетический алгоритм для решения проблемы размещения неиспользуемых объектов. Электр. Коммун. Япония. 2011, 94, 47–54. [Перекрестная ссылка]
25. Бизли, Дж. Э. ИЛИ-Библиотека: Распространение тестовых задач по электронной почте. Дж. Опер. Рез. Соц. 1990, 41, 1069–1072. [Перекрестная ссылка]
26. Аль-Султан, Канзас; Хусейн, МФ; Низами Дж.С. Генетический алгоритм решения задачи покрытия множества. Дж. Опер. Рез. Соц. 1996, 47, 702–709. [Перекрестная ссылка]
27. Петти, С.; Рамачандран, В. Оптимальный алгоритм минимального остовного дерева. Дж. ACM 2002, 49, 16–34. [Перекрестная ссылка]
28. Чжоу Дж.; Пэн, Дж.; Лян, Г.; Дэн, Т. Оптимизация компоновки сети газопроводов «дерево-дерево». Дж. Пет. наук. англ. 2019, 173, 666–680. [Перекрестная ссылка]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Быстрая минимальная энтропия ошибок для линейной регрессии

Цян Ли [1], Сяо Ляо [1], Вэй Цуй [1], Ин Ван [1], Хуэй Цао 2* и Циншу Гуань [2]

[1] State Grid Information & Telecommunication Group Co., Ltd., Пекин 102209, Китай; qiangli_sgcc@163.com (QL); xiaoliao_sgcc@163.com (XL); weicui_sgcc@163.com (туалет); yingwang_sgcc@163.com (YW)

[2] Школа электротехники, Сианьский университет Цзяотун, Сиань 710049, Китай; qsguan@stu.xjtu.edu.cn

* Переписка: huicao@mail.xjtu.edu.cn

Аннотация: Критерий минимальной энтропии ошибок (MEE) находит широкое применение в различных приложениях , особенно в контекстах, характеризующихся негауссовским шумом. Однако его вычислительные требования примечательны и в первую очередь связаны с операцией двойного суммирования, используемой при вычислении функции плотности вероятности (PDF) ошибки. Чтобы решить эту проблему, в нашем исследовании представлен новый подход, названный алгоритмом быстрой минимальной энтропии ошибок (FMEE), направленный на снижение сложности вычислений за счет использования полиномиальных разложений PDF ошибок. Первоначально PDF-аппроксимация случайной величины получается с помощью расширения Грама – Шарлье. Впоследствии мы приступаем к выяснению и оптимизации энтропии случайной величины. После этого энтропия ошибки, присущая модели линейной регрессии, очерчивается и выражается как функция вектора коэффициента регрессии. Наконец, используя алгоритм градиентного спуска, мы вычисляем вектор коэффициентов регрессии, соответствующий минимальной энтропии ошибки. Теоретическое исследование показывает, что временная сложность FMEE составляет O(n), [2]) сложность, связанная с что резко контрастирует с O(n MEE. Экспериментально наши результаты подчеркивают значительный прирост эффективности, обеспечиваемый FMEE, при этом потребление времени составляет менее 1 ‰ от наблюдаемого. Обнадеживает то, что этот скачок эффективности достигается без ущерба для точности, о чем свидетельствуют незначительные различия, наблюдаемые между точностью FMEE и MEE. Кроме того, комплексные регрессионные эксперименты на реальных наборах электрических данных на северо-западе Китая показывают, что наш FMEE превосходит базовый уровень. методы с явным преимуществом.

Ключевые слова: минимальная энтропия ошибок; линейная регрессия; расширение Грама – Шарлье; вычислительная сложность

check for updates

Цитирование: Ли, К.; Ляо, Х.; Кюи, В.; Ван, Ю.; Цао, Х.; Гуань, К. Быстрая минимальная энтропия ошибок для линейной регрессии. Алгоритмы 2024, 17, 341. https://doi.org/10.3390/a17080341.

Академический редактор: Фрэнк Вернер

Поступила: 23 июня 2024 г.

Пересмотрено: 1 августа 2024 г.

Принято: 3 августа 2024 г.

Опубликовано: 6 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария.

Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Энтропия служит ключевым показателем для количественной оценки присущей системе неопределенности , находя повсеместное применение во множестве областей, включая уменьшение размерности [1,2], оценку параметров [3,4], идентификацию [5,6], управление с обратной связью [7,8] и аномальное обнаружение [9–11]. Критерий минимальной энтропии ошибок (MEE) выделяется как примечательная основа, направленная на минимизацию энтропии, связанной с ошибками оценки, тем самым уменьшая неопределенность в модели оценки. Что особенно важно для решения задач оценки линейной регрессии, MEE отличается от традиционных методологий наименьших квадратов, поскольку не только учитывает дисперсию ошибок прогнозирования, но и включает в себя кумулянты более высокого порядка, что делает его пригодным для обработки негауссовых распределений шума [12,13] . Учитывая преобладание негауссовского шума в реальных сценариях, эффективность MEE была должным образом подтверждена в различных приложениях, включая адаптивную фильтрацию [14–16], распознавание лиц [17,18], идентификацию разреженных систем [19–21].], стохастические системы управления [22,23] и ви

Более того, свойства конвергенции MEE были тщательно изучены в предшествующей литературе [25–27]. Углубленный анализ углубился в динамику сходимости методологий MEE с фиксированной точкой [28], а исследования взаимодействия

Machine Translated by Google

MEE и минимальная среднеквадратическая ошибка (MMSE) [29], а также ее связь с максимальной коррентропией [30] были тщательно изучены. Заслуживающие внимания расширения традиционной структуры MEE включают предложение вариантов ядра MEE [31], разъяснения стратегий регуляризации для реализаций MEE [32] и разработку полуконтролируемых и распределенных адаптаций [33]. Более того, в области моделирования ошибок в переменных (EIV) анализ устойчивости методологий MEE был изложен [34], наряду с предложением новых методологий, таких как подход с минимальной общей энтропией ошибок [35].

Вычислительные требования, связанные с критерием MEE, вытекают из его временной характеристики . плексиметрия, которая масштабируется квадратично ^{2}Центральное место в вычислительной нагрузке MEE занимает как O(n) необходимость операции двойного суммирования во время расчета градиента, относящегося к функции плотности вероятности ошибки (PDF), особенно при использовании оконной обработки Парцена [36] и некоторых типов функций ядра. Кроме того, Выбор неподходящей функции ядра или ее параметров может отрицательно повлиять на точность полученного PDF-файла. Чтобы решить эту проблему, были предприняты усилия по оптимизации подхода MEE с градиентным спуском, о чем свидетельствует нормализация размера шага в. Кроме того, инновационные методологии, такие как использование оператора квантования, предложенного в [38], сыграли важную роль в уменьшении вычислительной сложности путем отображения ошибки на ряд. кодовые слова с действительными значениями , этот подход эффективно преобразует операцию двойного суммирования в сингулярную форму, следовательно, уменьшая вычислительные затраты алгоритма.

В этой статье представлен новый подход — алгоритм быстрой оценки минимальной энтропии ошибок (FMEE), специально разработанный для задач линейной регрессии, использующий технику полиномиального расширения, применяемую к функции плотности вероятности ошибок (PDF). В отличие от традиционной методологии минимальной энтропии ошибок (MEE), использующей окно Парцена для получения PDF ошибок, FMEE использует метод расширения Грама – Шарлье для аппроксимации PDF ошибок, что приводит к заметному снижению временной сложности за счет параметров O (n) . Алгоритм включает в себя несколько ключевых шагов: во- ^{2}) до O(n). Примечательно, что FMEE устраняет необходимость в функциях ядра или связанных с ними функциях. первых, получение аппроксимации PDF для случайной величины с помощью разложения Грама – Шарлье, за которым следует упрощение энтропии случайной величины, чему способствует ортогональность полиномов Эрмита. определяется модель линейной регрессии, при этом энтропия ошибок выражается как функция вектора коэффициентов регрессии. В конечном итоге используется метод градиентного спуска для определения вектора коэффициентов регрессии, соответствующего минимальной энтропии ошибок. Экспериментальная проверка подчеркивает эффективность FMEE. демонстрируя значительное сокращение затрат времени с расхождением менее 1 ‰ по сравнению с подходом MEE для идентичных задач . Важно отметить, что между уровнями точности методологий FMEE и MEE наблюдаются минимальные расхождения .

Остальная часть статьи организована следующим образом: в разделе 2 представлены соответствующие алгоритмы , FMEE подробно предложен в разделе 3, раздел 4 содержит эксперименты, а выводы приведены в разделе 5.

2. Сопутствующие алгоритмы

2.1. МЭЭ

MEE минимизирует информацию, содержащуюся в ошибке, и максимизирует информацию, полученную предполагаемой моделью. Обычно энтропия Реньи второго порядка [39] используется для измерения энтропии, содержащейся в ошибке модели.

$$H(E) = \log E_p(pE) = \log p \quad \frac{2}{Э}(e)\ де, \qquad (1)$$

Machine Translated by Google

где E — случайная величина ошибки MEE, а H(E) — энтропия E , а pE(e) передает PDF E. В этой статье Ep(·) жертвует ожиданием, чтобы избежать недоразумений. Затем PDF E можно оценить с помощью окна Парцена:

$$\hat{p}E(e) = \frac{1}{H} \sum_{i=1}^{H} X(e \quad ei), \tag{2}$$

где n — количество наблюдаемых данных, K — функция ядра, а h — полоса пропускания. Обычно выбирается ядро Гаусса, и Kh (z) = информация $\frac{1}{\sqrt{2\pi}h} \exp\left(\frac{z^2}{2h^2}\right)$. Затем эмпирический об ошибке Hˆ (f) может быть рассчитана как:

$$\hat{H}(f) = \log \frac{1}{2H} \sum_{i=1}^{H} \sum_{j=1}^{H} Kh(ei \quad ej). \tag{3}$$

Поскольку логарифмическая функция монотонно возрастает, ее можно удалить, не влияя на процесс минимизации. Затем преобразованную информацию об эмпирической ошибке можно получить по уравнению (4):

$$R(e) = - \frac{1}{2H} \sum_{i=1}^{H} \sum_{j=1}^{H} Kh(ei \quad ej). \tag{4}$$

Для линейной регрессии y = wTx + e цель модели — оценить w на основе ограниченных выборок. Поскольку ei = yi wTxi , соответствующая информация об эмпирической ошибке рассчитывается по уравнению (5):

$$R(w) = - \frac{1}{2H} \sum_{i=1}^{H} \sum_{j=1}^{H} Kh(yi - wT xi) - (yj - wT xj). \tag{5}$$

Тогда оценка MEE wˆ может быть получена путем минимизации R(w) с помощью градиентного спуска. Временная сложность MEE равна $O(n^2)$ из-за операции двойного суммирования, как показано на Уравнение (5). Поскольку затраты времени MEE квадратичны по отношению к количеству наблюдаемых данных, время работы MEE резко увеличится с увеличением количества наблюдаемых данных.

2.2. Полиномы Эрмита

Полиномы Эрмита [40] представляют собой классическую ортогональную полиномиальную последовательность. Они определяются производными стандартизованной гауссовой PDF φ(ξ) как уравнение (6):

$$\frac{i\varphi(\xi)}{} = (\quad 1)\, iHi(\xi)\varphi(\xi), \quad \xi i \tag{6}$$

где H(i) передает полином Эрмита i-го порядка.

Полиномы Эрмита образуют ортогональную систему, как в уравнении (7):

$$\varphi(\xi)Hi(\xi)Hj(\xi)d\xi = \begin{cases} я!, & \text{если } я = j, \\ 0, & \text{если } я/ = j \end{cases}, \tag{7}$$

В этой статье используются первые пять полиномов Эрмита, показанные в уравнении (8):

$$\begin{aligned} H0(x) &= 1, \\ H1(x) &= x, \\ H2(x) &= x^2 \quad 1, \\ H3(x) &= x^3 \quad 3x, \\ H4(x) &= x^4 - 6x^2 + 3. \end{aligned} \tag{8}$$

Machine Translated by Google

2.3. Расширение Грама – Шарлье

Разложение Грама-Шарлье дает аппроксимацию PDF случайной величины, и при этом разложении не происходит значительной потери точности, что продемонстрировано в [41]. В предположении, что случайная величина близка к нормальному распределению с тем же средним значением и дисперсией, что и она сама, разложение Грама – Шарлье PDF-распределения x можно выразить как:

$$p\hat{x}(\xi) = \exp\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)\frac{(\xi-\mu)^2}{2\sigma^2}\left(1+\frac{\kappa_3(x)}{3!\sigma^3}H_3\left(\frac{\xi-\mu}{\sigma}\right)+\frac{\kappa_4(x)}{4!\sigma^4}H_4\left(\frac{\xi-\mu}{\sigma}\right)\right), \quad (9)$$

где μ и σ — среднее и стандартное отклонение x. ξ — гауссова случайная величина, среднее значение и переменная которой такие же, как у x. κ3(x) и κ4(x) — кумулянты третьего и четвертого порядков x, которые также можно назвать асимметрией и эксцессом. H3 и H4 — полиномы Эрмита третьего и четвертого порядка.

3. Методология B

этом разделе выводится связь между дифференциальной энтропией и разложением Грама – Шарлье PDF ошибки для линейной регрессии. Более того, реализуется предположение о том, что PDF ошибки регрессии находится рядом с гауссовским распределением φ(ξ) , которое имеет то же среднее значение и дисперсию, что и ошибка.

Дифференциальная энтропия [42] случайной величины X показана в уравнении (10):

$$H(x) = \int p(x)\log p(x)dx. \quad (10)$$

Замените уравнение (9) на уравнение (10); результат можно получить с помощью уравнения (11):

$$\begin{aligned} H(x) &\approx \int \hat{p}x(\xi)\log \hat{p}x(\xi)d\xi \\ &= \int \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(\frac{(\xi-\mu)^2}{2\sigma^2}\right)\left(1+\frac{\kappa_3(x)}{3!\sigma^3}H_3\left(\frac{\xi-\mu}{\sigma}\right)+\frac{\kappa_4(x)}{4!\sigma^4}H_4\left(\frac{\xi-\mu}{\sigma}\right)\right) \\ &\quad \text{бревно}\left(\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(\frac{(\xi-\mu)^2}{2\sigma^2}\right)\left(1+\frac{\kappa_3(x)}{3!\sigma^3}H_3\left(\frac{\xi-\mu}{\sigma}\right)+\frac{\kappa_4(x)}{4!\sigma^4}H_4\left(\frac{\xi-\mu}{\sigma}\right)\right)\right)d\xi. \end{aligned} \quad (11)$$

Этот интеграл довольно сложно вычислить. Однако обратите внимание: если PDF x близка к нормальной плотности, как предполагалось, можно сделать вывод, что κ3(x) и κ4(x) очень малы. Тогда можно использовать приближение в уравнении (12) :

$$\log(1+\epsilon) \approx \epsilon - \frac{\epsilon^2}{2}, \quad (12)$$

где ϵ обозначает бесконечно малое.

Тогда уравнение (11) можно преобразовать в уравнение (13):

Machine Translated by Google

$$H(x) = \varphi(\xi) \quad \kappa 3(x)\ 1 \quad 3!\sigma^3 + H3 \quad \frac{\xi \quad \mu}{\sigma} + \frac{\kappa 4(x)}{4!\sigma^4} H4 \quad \frac{\xi \quad \mu}{\sigma} \quad \log \varphi(\xi) + H3\ 3!\sigma^3 \quad \frac{\kappa 3(x)}{\sigma} \quad \frac{\xi \quad \mu}{\sigma} + \frac{\kappa 4(x)}{4!\sigma^4} H4 \quad \frac{\xi \quad \mu}{\sigma} \quad \frac{1}{2} \quad \frac{\kappa 3(x)}{3!\sigma^3} H3 \quad \frac{\xi \quad \mu}{\sigma} + \frac{\kappa 4(x)}{4!\sigma^4} H4 \quad \frac{\xi \quad \mu}{\sigma} \quad ^2 \, d\xi. \tag{13}$$

Поскольку Hi образует ортогональную систему, уравнение (13) можно упростить до уравнения (14):

$$H(x) = \quad \log 2\pi e\sigma^2 \quad ^2 \quad \frac{^2 x_3(x)}{2 \quad 3!\sigma^6} \quad \frac{^2 x_4(x)}{2 \quad 4!\sigma^8}. \tag{14}$$

Детали процесса упрощения уравнения (14) проиллюстрированы следующим образом.

Эта часть предназначена для вывода уравнения (14). Во-первых, необходимо показать, что

$$2H3(\ \varphi(\xi)(\ )d\xi = 0 \quad \frac{\xi \quad \mu}{\sigma} \tag{15}$$

и

$$2H4(\ \varphi(\xi)(\ )d\xi = 0 \quad \frac{\xi \quad \mu}{\sigma} \tag{16}$$

Для уравнения (15) предположим, что t = (ξ μ)/σ,

$$2H3(\ \varphi(\xi)(\ )d\xi \quad \frac{\xi \quad \mu}{\sigma} = \frac{1}{\sqrt{2\pi}\sigma} \exp \quad \frac{(\xi \quad \mu)^2}{2\sigma^2} \quad (\ ) \quad \frac{\xi \quad \mu}{\sigma} \quad ^3 \quad 3(\ ) d\frac{\xi \quad \mu}{\sigma} = \frac{1}{\sqrt{2\pi}\tau^2} \exp(\quad)(t^2 \quad \frac{t^2}{} \quad ^3 \quad 3t)dt \tag{17}$$

Поскольку $\exp(-\frac{t^2}{2})$ — четная функция и $t^{5\,3} \quad ^{3\tau}$ нечетная функция, то

$$\frac{1}{2\tau\sqrt{2\pi}} \exp(- \quad \frac{\tau^2}{2})(\tau^3 \quad 3t)dt = 0 \tag{18}$$

и уравнение (15) выполняется.

Для уравнения (16) предположим, что t = (ξ μ)/σ,

$$2H4(\ \varphi(\xi)(\ )d\xi \quad \frac{\xi \quad \mu}{\sigma} = \frac{1}{\sqrt{2\pi}\sigma} \exp \quad \frac{(\xi \quad \mu)^2}{2\sigma^2} \quad \xi \quad \mu\ (\quad \frac{\xi - \mu}{\sigma} \quad ^6 - 6(\ ) \quad ^4 + 3(\ ) \quad ^2 \, d\xi = \frac{1}{\sqrt{2\pi}} \exp(\quad \frac{2\tau}{2})(\tau^6 \quad ^4 \quad ^2 \quad 6\tau + 3\tau \quad)d\tau \tag{19}$$

$$=3 \quad \frac{1}{\sqrt{2\pi}} \exp(\quad \frac{\tau^2}{2})t\ 2dt \quad 6 \quad \frac{1}{\sqrt{2\pi}} \exp(\quad \frac{\tau^2}{2})t\ 4dt + \frac{1}{\sqrt{2\pi}} \exp(\quad \frac{\tau^2}{2})t\ 6dt$$

Machine Translated by Google

Первый член в уравнении (19) можно рассчитать как уравнение (20):

$$\begin{aligned}
&\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{\tau^2}{2}\right)t\,2dt \\
&= \frac{1}{\sqrt{2\pi}}(\quad t)d\exp\left(-\frac{t^2}{2}\right) \\
&= \frac{1}{\sqrt{2\pi}}(\quad t)\exp\left(-\frac{\tau^2}{2}\right)+ \\
&\quad \frac{1}{\sqrt{2\pi}}\tau\exp\left(-\frac{2}{2}\right)dt \\
&= 1
\end{aligned} \tag{20}$$

Второй член в уравнении (19) можно рассчитать как уравнение (21):

$$\begin{aligned}
&\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{\tau^2}{2}\right)t\,4dt \\
&= \frac{1}{\sqrt{2\pi}}(\quad t^3)d\exp\left(-\frac{\tau^2}{2}\right) \\
&= \frac{1}{\sqrt{2\pi}}(\quad t^3\tau)\exp\left(-\frac{2}{2}\right)+ \\
&\quad 3\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{\tau^2}{2}\right)t\,2dt \\
&= 3
\end{aligned} \tag{21}$$

Последний член в уравнении (19) можно рассчитать как уравнение (22):

$$\begin{aligned}
&\frac{1}{\sqrt{2\pi}2}\exp\left(-\frac{\tau^2}{}\right)t\,6dt \\
&= \frac{1}{\sqrt{2\pi}}(\quad t^5)d\exp\left(-\frac{\tau^2}{2}\right) \\
&= \frac{1}{\sqrt{2\pi}}(\quad t^5)\exp\left(-\frac{\tau^2}{2}\right)+ \\
&\quad 5\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{\tau^2}{2}\right)t\,4dt \\
&= 15
\end{aligned} \tag{22}$$

Согласно уравнениям (20)–(22),

$$2H4\left(\varphi\left(\frac{\xi-\mu}{\sigma}\right)d\xi\right) = 15\left(\frac{\xi-\mu}{\sigma}\right)^6 - 3 + 3 = 0 \tag{23}$$

и уравнение (16) выполняется.

Machine Translated by Google

Тогда для уравнения (13)

$$H(x) = \int \varphi(\xi)\log \varphi(\xi)d\xi$$

$$\varphi(\xi)\left[\frac{к3(x)}{3!\sigma^3}\left(\frac{\xi-\mu}{\sigma}\right)^3 + \frac{к4(x)}{4!\sigma^4}\left(\frac{\xi-\mu}{\sigma}\right)^4 + H3\left(\frac{\xi-\mu}{\sigma}\right) + H4\left(\frac{\xi-\mu}{\sigma}\right)\right]d\xi +$$

$$-\int \frac{к3(x)\,к4(x)}{H3\left(\frac{\xi-\mu}{\sigma}\right) + H4\left(\frac{\xi-\mu}{\sigma}\right)}\left(\frac{\xi-\mu}{\sigma}\right)^3\left(\frac{\xi-\mu}{\sigma}\right)^4\frac{\varphi(\xi)}{\sigma^2}d\xi -$$

$$\int \varphi(\xi)\log \varphi(\xi)\left[\frac{к3(x)}{H3\left(\frac{\xi-\mu}{\sigma}\right)}\left(\frac{\xi-\mu}{\sigma}\right)^3\right.$$

$$\left.\frac{к4(x)}{4!\sigma^4}\left(\frac{\xi-\mu}{\sigma}\right)^4 H4\left(\frac{\xi-\mu}{\sigma}\right)\right]d\xi +$$

$$\int \frac{к3(x)}{H3\left(\frac{\xi-\mu}{\sigma}\right)}\left(\frac{\xi-\mu}{\sigma}\right)^3\frac{\varphi(\xi)}{\sigma^2}$$

$$\left[\frac{к4(x)}{4!\sigma^4}\left(\frac{\xi-\mu}{\sigma}\right)^4 H4\left(\frac{\xi-\mu}{\sigma}\right)\right]^2 d\xi +$$

$$\frac{1}{2}\int \frac{к3(x)\,к4(x)}{H3\left(\frac{\xi-\mu}{\sigma}\right) + H4\left(\frac{\xi-\mu}{\sigma}\right)}\left(\frac{\xi-\mu}{\sigma}\right)^3\left(\frac{\xi-\mu}{\sigma}\right)^4\frac{\varphi(\xi)}{\sigma^3}d\xi \quad (24)$$

Обратите внимание, что H0 = 1, тогда второй член уравнения (24) равен 0. Поскольку предполагается, что x близка к нормальной случайной величине, к3(x) и к4(x) очень малы. Тогда последний член уравнения (24) можно опустить, равный 0, поскольку он включает в себя мономы третьего порядка, которые бесконечно меньше, чем члены, содержащие только мономы второго порядка. Затем для уравнения (24) примите во внимание уравнения (15) и (16) и еще раз обратите внимание, что

$$H(x) = H(\xi) - \frac{1}{2}\left[\frac{к3(x)}{3!\sigma^3}\left(\frac{\xi-\mu}{\sigma}\right)^3 + \frac{к4(x)}{4!\sigma^4}\left(\frac{\xi-\mu}{\sigma}\right)^4 - \frac{к3(x)}{к4(x)}\right]^2 +$$

$$\int \left[\frac{к4(x)}{4!\sigma^4}\left(\frac{\xi-\mu}{\sigma}\right)^4 - \frac{к3(x)}{к4(x)}\right]^2 \left[H4\left(\frac{\xi-\mu}{\sigma}\right) + H3\left(\frac{\xi-\mu}{\sigma}\right)\right]\varphi(\xi)\log\left[\frac{к4(x)}{4!\sigma^4}\left(\frac{\xi-\mu}{\sigma}\right)^4 + \frac{к3(x)}{3!\sigma^3}\left(\frac{\xi-\mu}{\sigma}\right)^3 + H4\left(\frac{\xi-\mu}{\sigma}\right) + H3\left(\frac{\xi-\mu}{\sigma}\right)\right]d\xi +$$

$$\int \varphi(\xi)\log\left[\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(\xi-\mu)^2}{2\sigma^2}\right)\right]d\xi -$$

$$\int \varphi(\xi)\log\left[1 + \frac{к3(x)}{3!\sigma^3}\left(\frac{\xi-\mu}{\sigma}\right)^3 + \frac{к4(x)}{4!\sigma^4}\left(\frac{\xi-\mu}{\sigma}\right)^4 + H3\left(\frac{\xi-\mu}{\sigma}\right) + H4\left(\frac{\xi-\mu}{\sigma}\right)\right]d\xi$$

$$= H(\xi) - \frac{\chi_3^2(x)}{18\sigma^6} - \frac{\chi_4^2(x)}{32\sigma^8} +$$

$$\int \log\left[\frac{1}{\sqrt{2\pi\sigma^2}}\right]\varphi(\xi)\left[\frac{8\,к3(x)}{3!\sigma^3}\left(\frac{\xi-\mu}{\sigma}\right)^3 + \frac{к4(x)}{4!\sigma^4}\left(\frac{\xi-\mu}{\sigma}\right)^4 + H3\left(\frac{\xi-\mu}{\sigma}\right) + H4\left(\frac{\xi-\mu}{\sigma}\right)\right]d\xi +$$

$$\frac{1}{2\sigma^2}\int \varphi(\xi)(\xi-\mu)^2 d\xi - \int \frac{к3(x)\,к4(x)}{3!\sigma^3\,4!\sigma^4}\left[H4\left(\frac{\xi-\mu}{\sigma}\right) + H3\left(\frac{\xi-\mu}{\sigma}\right)\right]\varphi(\xi)d\xi$$

$$= \frac{1}{2}\log 2\pi e\sigma^2 - \frac{\chi_3^2(x)}{18\sigma^6} - \frac{\chi_4^2(x)}{32\sigma^8} \quad (25)$$

Итак, уравнение (14) справедливо.

Для линейной регрессии $e_i = y_i - w^T x_i$ стандартное отклонение ошибки e_w находится в уравнении (26):

$$\sigma^2(e_w) = E_p(E_w^2) - [E_p(E_w)]^2 = E_p(E_w^2) - (y - w^T\bar{x})^2, \quad (26)$$

Machine Translated by Google

где y — среднее значение y , а $\bar{x}$ — среднее значение x.

Чтобы упростить расчет к4(ew), пусть ew станет случайной величиной со средним значением 0:

$$ewi = yi \quad wT xi \quad y^{-} + wT x^{-}. \tag{27}$$

Обратите внимание, что преобразование в уравнении (27) не меняет стандартное отклонение а энтропия ew как $\sigma^2(e + c) = \sigma^2(e)$ и $H(e + c) = H(e)$, когда c — постоянное число. В остальной части статьи для ясности $\sigma^2(\cdot)$ заменяется на var(·).

Тогда для линейной регрессии энтропию ошибки можно выразить уравнением (28):

$$H(ев) = 1 \quad \frac{1}{2} \log 2\pi e + \log\sqrt{var(ew)} \quad \frac{2 к_3}{var3(ew)} \quad \frac{3!}{2} \quad \frac{2 к_4 (ew)}{4! var4(ew)}, \tag{28}$$

где

$$к3(ew) = Ep[(y \quad wT x \quad y^{-} + wT x^{-})^3], \tag{29}$$

$$к4(ew) = Ep[(y \quad wT x \quad y^{-} + wT x^{-})^4] \quad 3. \tag{30}$$

Чтобы минимизировать H(ew), производная H(ew) по w вычисляется в Уравнение (31):

$$\begin{aligned} \frac{H(ев)}{w} = {} & \frac{1}{2var(ew)\ln 2} \quad \frac{var(ew)}{w} \quad \frac{1}{2} \quad \frac{1}{3! var6(ew)} \\ & 2var3(ew) к3(ew) \quad \frac{к3(ew)}{w} \\ & 2 3к_3(ew) var2(ew) \quad \frac{var(ew)}{w} \\ & \frac{1}{2} \quad \frac{1}{4! var8(ew)} \quad 2var4(ew) к4(ew) \quad \frac{к4(ew)}{w} \\ & 2 4к_4(ew) var3(ew) \quad \frac{var(ew)}{w}, \end{aligned} \tag{31}$$

где

$$\frac{вар(ев)}{} = 2Ep[(y \quad wT x)x] + 2(y^{-} \quad wT x^{-})x^{-}, \quad w \tag{32}$$

$$\frac{к3(ев)}{} = 3Ep[(y \quad wT x \quad y^{-} + wT x^{-}) \quad w \quad ^2(\quad x + x^{-})], \tag{33}$$

$$\frac{к4(ев)}{} = 4Ep[(y \quad wT x \quad y^{-} + wT x^{-}) \quad w \quad ^3(\quad x + x^{-})]. \tag{34}$$

Тогда оптимальное w для минимума H(ew) можно получить по итерационной схеме в уравнении (35) с градиентным спуском:

$$w\hat{}\, k+1 = w\hat{}\, k - \alpha \quad \frac{H(ewk)}{w\hat{}\, k}, \tag{35}$$

где α — размер шага, определяемый условием Армихо [43].

Временная сложность вычисления $\frac{H(ewk)}{wk}$, wk , $\frac{var(ewk)}{wk}$, $\frac{к3(ewk)}{wk}$, $\frac{к4(ewk)}{wk}$ и wk , wk для FMEE на каждой итерации, var(ewk), к3(ewk), к4(ewk), Чтобы (26) и (29)–(34) из уравнений

var(ewk) из var(ewk), к3(ewk), $\frac{к4(ewk)}{}$, $\frac{}{}$ $\frac{}{}$ равно O(n) соответственно. Тогда) из-

wk , вычислительная сложность FMEE равна O(n). Поскольку временная сложность MEE равна $O(n)^2$ за для операции двойного суммирования, FMEE работает быстрее, чем MEE.

Machine Translated by Google

4. Эксперименты

В этом разделе проводится обширный набор экспериментов, включающий как численное моделирование, так и сценарии реального мира. Численное моделирование служит для подтвердить эффективность подхода FMEE и тщательно изучить временную эффективность обоих Методологии FMEE и MEE. Параллельно проводятся практические эксперименты, направленные при прогнозировании отключений электроэнергии в городе, расположенном на северо-западе Китая.

Что касается численного моделирования, мы рассматриваем сценарий, где x R10, с модель, определяемая как y = w ^{T} x + e, где w = [1, 1, 1, 1, 1 , 1, 1, 1, 1] ^{T} и x соответствует нормальному распределению N (0, I10). Два различных типа шума рассматриваются в наши эксперименты. Во-первых, используется гауссовский шум e N (0, 1) , а второй тип включает обобщенный гауссовский шум, характеризуемый функцией плотности вероятности f(e) exp(α|e|) 0,3. Это распределение с тяжелым хвостом, используемое здесь, соответствует предыдущему. работы, такие как ссылки [27,35]. Хотя FMEE выводится в предположении, что Шум, близкий к Гауссу, ожидается, что FMEE останется эффективным при наличии негауссовских шумов с тяжелыми хвостами. Здесь α обозначает константу, скорректированную для поддержания дисперсии из 1 для эл. Для MEE размер шага установлен равным 0,005π, а параметр масштаба для гауссова ядро — 10, что соответствует настройкам, принятым в [27]. В ходе экспериментов размеры выборки варьируются от 100 до 500, при этом каждая комбинация размера выборки и типа шума повторено 100 раз. Полученные экспериментальные результаты подробно описаны ниже.

В таблице 1 представлен сравнительный анализ затрат времени между MEE и FMEE при наличии гауссова шума. FMEE демонстрирует заметное ускорение вычислительная эффективность по сравнению с MEE. В частности, хотя время вычислений MEE демонстрирует квадратичный рост относительно размера выборки, вычислительные затраты FMEE масштабируется линейно с количеством выборок. Примечательно, что FMEE достигает поразительной скорости, обработка 500 объектов примерно за 0,1 с, тем самым позиционируя его как весьма перспективный кандидат для приложений, чувствительных ко времени.

Таблица 1. Время работы MEE и FMEE для гауссовского шума.

Размер выборки	Среднее время работы		Оптимальное время работы	
	МЭЭ	FMEE	МЭЭ	FMEE
100	103,89	0,0401	77,77	0,0156
200	384,31	0,0580	310.05	0,0312
300	801,52	0,0769	654,21	0,0468
400	1430,61	0,1000	1254,12	0,0468
500	2306,24	0,1104	2072,50	0,0624

В таблице 2 сравнивается среднеквадратическая ошибка между MEE и FMEE в контексте Гауссов шум. В целом, MEE демонстрирует превосходные характеристики по сравнению с FMEE с точки зрения МШЭ. Особо следует отметить существенное преимущество MEE перед FMEE, особенно очевидно, когда размер выборки равен 100. Однако по мере увеличения размера выборки расхождение в MSE между двумя алгоритмами уменьшается. К тому времени, когда размер выборки достигнет 500, разница сокращается до всего 2%. Это наблюдение предполагает, что, хотя FMEE может отставать немного отстает от MEE, особенно в крупномасштабных наборах данных, такое несоответствие остается приемлемым в практических приложениях, учитывая значительно сокращенное время вычислений FMEE по сравнению в МЭЭ.

Machine Translated by Google

Таблица 2. Среднеквадратическая ошибка MEE и FMEE для гауссовского шума.

Размер выборки	Средняя MSE		Оптимальная MSE	
	МЭЭ	FMEE	МЭЭ	FMEE
100	0,9108	1,4027	0,5697	0,6542
200	0,9311	1,0528	0,6295	0,7258
300	0,9573	1,0190	0,7600	0,7462
400	0,9696	0,9769	0,8024	0,7588
500	0,9604	0,9938	0,8024	0,8302

В таблице 3 анализируется количество итераций, необходимое для сходимости между MEE и FMEE. в контексте гауссовского шума. Примечательно, что FMEE достигает конвергенции со значительно меньше итераций по сравнению с MEE. Кроме того, замечено, что количество итераций MEE постепенно снижается по мере увеличения размера выборки. И наоборот, количество итераций FMEE остается относительно постоянным, независимо от изменений в размере выборки.

Таблица 3. Количество итераций MEE и FMEE для гауссовского шума.

Размер выборки	Среднее число итераций		Оптимальное число итераций	
	МЭЭ	FMEE	МЭЭ	FMEE
100	797.01	38.21	592	6
200	697,09	38,29	592	7
300	656,61	36,88	542	7
400	650.03	40,67	571	9
500	637.03	36.17	586	7

В таблице 4 представлена сравнительная оценка затрат времени между MEE и FMEE в контексте негауссовского шума. Во всех пяти экспериментальных группах FMEE последовательно демонстрирует значительное сокращение затрат времени. что составляет менее 1‰ временных затрат MEE. Более того, в соответствии с Согласно теоретическим предсказаниям, вычислительные затраты MEE демонстрируют квадратичный рост относительно в зависимости от размера выборки, тогда как временные издержки FMEE линейно зависят от размера выборки.

Таблица 4. Время работы MEE и FMEE для негауссовского шума.

Размер выборки	Среднее время работы		Оптимальное время работы	
	МЭЭ	FMEE	МЭЭ	FMEE
100	107,96	0,0413	85,24	0,0156
200	348,66	0,0635	270,26	0,0312
300	800,50	0,0819	727,45	0,0468
400	1431,34	0,1027	1274,87	0,0624
500	2179,33	0,1178	1952,27	0,0624

В Таблице 5 приведена сравнительная оценка среднеквадратической ошибки (MSE) между MEE и FMEE в контексте негауссова шума. Как правило, разница между Значения MSE FMEE и MEE относительно невелики. Примечательно, что FMEE превосходит MEE когда размер выборки составляет 100 или 400, в то время как MEE демонстрирует превосходную производительность для выборки размеры 200, 300 и 500. Кроме того, FMEE достигает оптимального MSE для всей выборки. размеров, за исключением n = 100. Кроме того, замечено, что значения СКО для негауссовых шум, как показано в Таблице 5, охватывает более широкий диапазон по сравнению с шумом Гаусса, что указывает на большую вариативность результатов как для MEE, так и для FMEE.

Machine Translated by Google

Таблица 5. Среднеквадратическая ошибка MEE и FMEE для негауссовского шума.

Размер выборки	Средняя MSE		Оптимальная MSE	
	МЭЭ	FMEE	МЭЭ	FMEE
100	1,0844	0,9604	0,1584	0,1630
200	0,8485	1,1554	0,2654	0,2607
300	0,9110	1,2314	0,3140	0,2659
400	1,1037	1,0550	0,3672	0,2891
500	0,9536	1,0491	0,4995	0,4668

В таблице 6 сравнивается количество итераций, необходимых для сходимости между MEE. и FMEE в контексте негауссовского шума. FMEE постоянно демонстрирует более высокую скорость конвергенция по сравнению с MEE. Кроме того, замечено, что количество итераций MEE уменьшается по мере увеличения размера выборки. Напротив, количество итераций FMEE остается относительно стабильны, несмотря на различия в размере выборки.

Таблица 6. Количество итераций MEE и FMEE для негауссовского шума.

Размер выборки	Среднее число итераций		Оптимальное число итераций	
	МЭЭ	FMEE	МЭЭ	FMEE
100	782,88	34.33	618	8
200	689,67	35,72	563	9
300	644,95	36,72	584	10
400	648,70	37,55	580	12
500	633,25	35.35	571	12

Судя по результатам, представленным в таблицах 1–6, очевидно, что FMEE последовательно дает результаты, сравнимые с MEE, хотя и со значительно меньшими затратами времени. и количество итераций. Примечательно, что FMEE отличается своей замечательной эффективностью: он работает примерно в 1000 раз быстрее, чем MEE, и обеспечивает сходимость примерно за 0,1 с для 500 экземпляров. Учитывая линейную временную сложность O(n), FMEE имеет значительные обещают развертывание в сценариях реального времени, характеризующихся негауссовским шумом.

Впоследствии мы приступаем к практическим экспериментам с использованием данных о характеристиках трансформатора для прогнозирования отказов распределительной сети. Трансформаторы делятся н две группы в зависимости от их географического положения на разных уровнях городской активности в конкретном городе на северо-западе Китая. Первоначально наше внимание направлено на подгруппа из 1265 трансформаторов, расположенных на меньшей, менее густонаселенной территории вышеупомянутый город. Мы предпринимаем случайное разделение этого набора данных на два сегмента: обучающий набор, содержащий 80% данных, всего 1012 экземпляров, и проверочный набор включая оставшиеся 20%, включающие 253 экземпляра. Если быть более конкретным, набор данных по трансформаторам включает в себя различные характеристические переменные, в том числе стандартизированные тяжелые продолжительность перегрузки, максимальный коэффициент активной нагрузки, средний коэффициент активной нагрузки, средний трехфазный несимметрия, стандартизированная продолжительность тяжелого трехфазного несимметрии и т. д.

Мы проводим сравнительный анализ, сопоставляя предлагаемый нами FMEE с тремя установленными базовыми методологиями: логистической регрессией, нейронной сетью и вектором поддержки. машина. Набор данных содержит 1265 переменных характеристик трансформатора, связанных с значительные уровни нагрузки, подвергающиеся 30 раундам случайного разделения. Впоследствии каждый случайно разделенный набор данных подвергается воздействию четырех различных алгоритмов. Прогностическая эффективность этих алгоритмов на верификационной выборке оценивается на основе F-меры и ошибки. показатели тарифа.

На рисунке 1 показаны оценки F-меры, полученные на основе 30 примеров прогнозирования. проводится по четырем различным алгоритмам. Примечательно, что каждая оценка включает последовательное разделение набора данных по четырем алгоритмам. Проявляется отчетливая тенденция: все алгоритмы дают значения F-меры, превышающие 0,8. Примечательно, что предлагаемый нами бывший FMEE

Machine Translated by Google

подавляет наиболее многообещающие результаты, имея средний показатель F 0,904, тем самым значительно превосходя три другие базовые методологии.

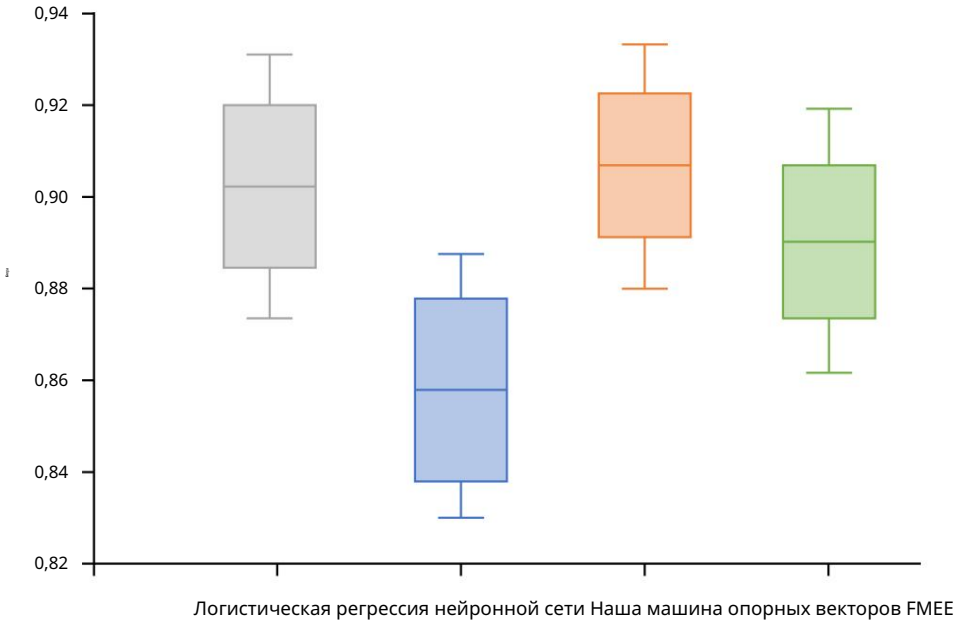


Рисунок 1. Блок-схема F-меры четырех алгоритмов прогнозирования сбоев.

Чтобы облегчить всестороннее сравнение четырех алгоритмов, на рисунке 2 показана частота ошибок и F-мера результатов прогнозирования отключений из-за сбоев в виде линейных и прямоугольных диаграмм. Примечательно, что FMEE становится лидером, демонстрируя существенное преимущество перед сложной машиной опорных векторов с точки зрения частоты ошибок. В частности, наш FMEE достигает средней частоты ошибок 9,1%, превосходя нейронную сеть, логистическую регрессию и машину опорных векторов на 0,3%, 4,2% и 4,3% соответственно. Эти результаты подчеркивают явное превосходство предложенной нами методологии FMEE.

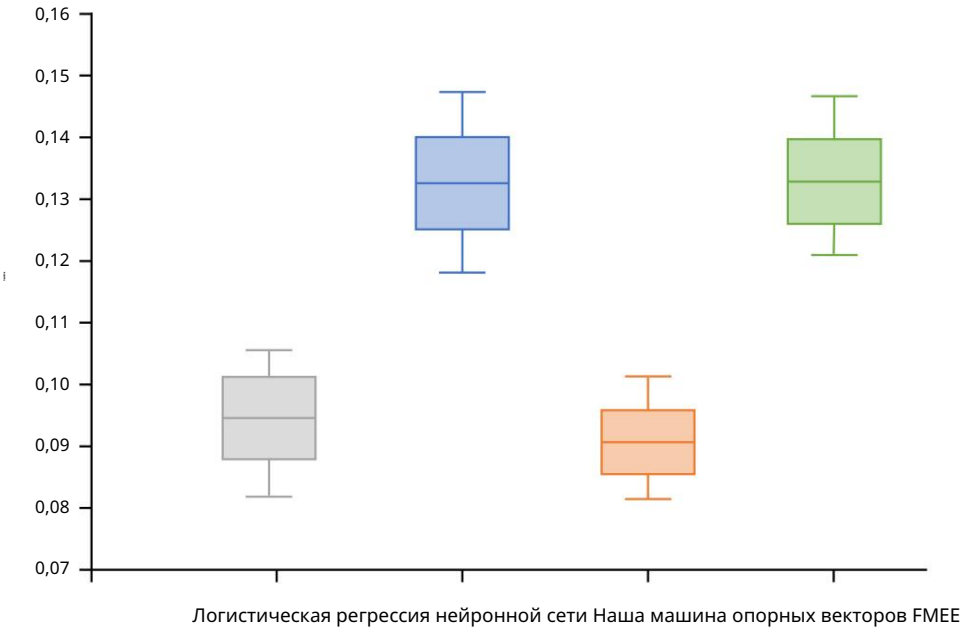


Рисунок 2. Боксовая диаграмма частоты ошибок среди четырех алгоритмов прогнозирования сбоев.

Machine Translated by Google

Кроме того, мы распространили наш эксперимент на более густонаселенный городской район в городе на северо-западе Китая. На рисунке 3 показаны оценки F-меры, полученные в результате 30 независимых тестов, проведенных по четырем сравнительным алгоритмам. В широком масштабе предлагаемый нами FMEE достигает F-меры 0,891, превосходя базовые методологии, такие как нейронные сети, логистическая регрессия и машины опорных векторов, с позицией 0,013, 0,023 и 0,028 соответственно.

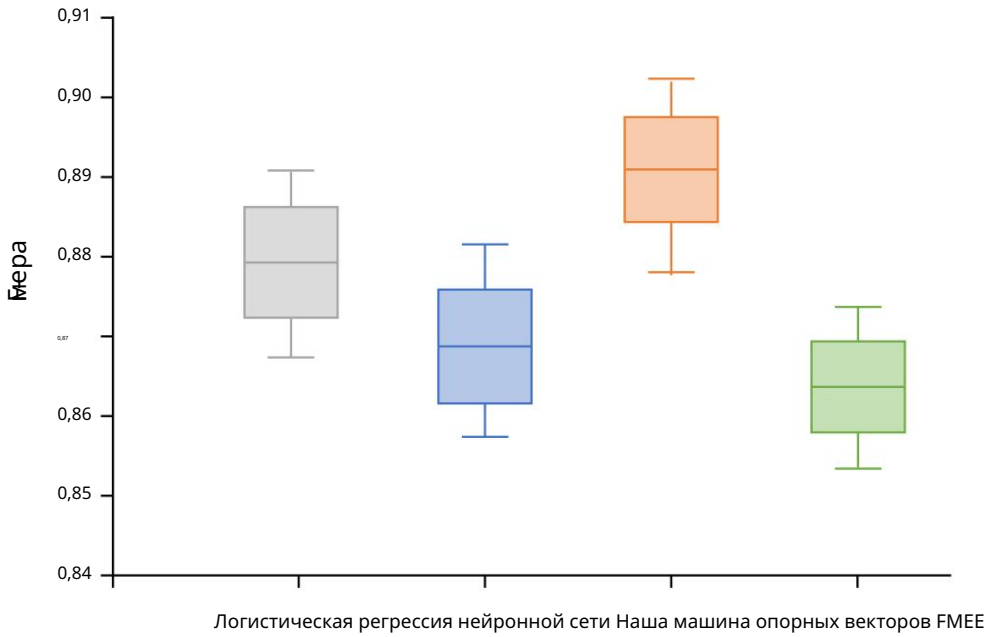


Рисунок 3. Блок-схема F-меры четырех алгоритмов прогнозирования сбоев.

Наконец, мы сообщаем о частоте ошибок F-меры, касающейся результатов прогнозирования отказов в работе, полученных на основе четырех сравнительных алгоритмов, представленных в виде линейных и прямоугольных диаграмм. Как показано на рисунке 4, предлагаемый нами FMEE достигает среднего уровня ошибок 11,2%, демонстрируя заметное превосходство над тремя базовыми методологиями. Эти результаты убедительно подчеркивают эффективность нашего алгоритма в решении проблем регрессии.

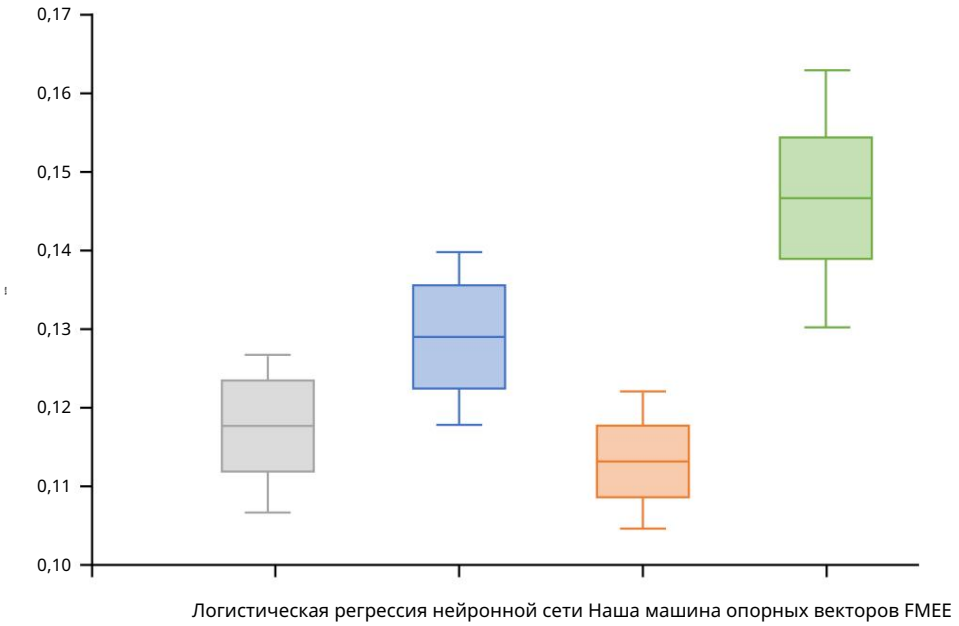


Рисунок 4. Блок-схема частоты ошибок четырех алгоритмов прогнозирования сбоев.

Machine Translated by Google

5. Выводы

В этом исследовании представлен быстрый алгоритм минимальной энтропии ошибок, адаптированный для линейной регрессии. В результате нашего расследования можно сделать несколько примечательных выводов. Во-первых, предлагаемый FMEE демонстрирует значительно более быструю сходимость и требует меньше итераций по сравнению с MEE. Во-вторых, хотя FMEE не превосходит традиционный MEE по производительности для гауссовского шума и дает сопоставимые результаты для негауссовского шума, измеренного по среднеквадратичной ошибке, его вычислительная эффективность поразительно выше, занимая менее 1‰ времени, требуемого MEE. В-третьих, несмотря на то, что FMEE был получен на основе предположения о близости шума к распределению Гаусса, он демонстрирует эффективность при обработке негауссовских распределений шума с тяжелыми хвостами. В-четвертых, благодаря исключительной эффективности использования времени FMEE перспективен для приложений реального времени в сценариях с негауссовским шумом. В дальнейшем дальнейший анализ условий сходимости FMEE и исследование методов аппроксимации шума для произвольных распределений открывают возможности для будущих исследований.

Вклад автора: концептуализация, QL и XL; методология, КЖ; программное обеспечение, туалет; валидация , QL, XL и WC; формальный анализ, YW; расследование, XL; ресурсы, QL; курирование данных, YW; письмо — подготовка первоначального проекта, QG; написание — рецензирование и редактирование, HC; визуализация, КГ; надзор, ХК; администрация проекта, WC; приобретение финансирования, QL и HC. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Работа поддерживается научно-техническими инновационными проектами Государственной сетевой информационно-телекоммуникационной группы «Исследование систем энергетических цифровых космических технологий и ключевых технологий» (грант №: SGIT0000XMJS2310456).

Заявление Институционального наблюдательного совета: Неприменимо.

Заявление об информированном согласии: Не применимо.

Заявление о доступности данных: данные, представленные в этом исследовании, доступны по запросу у соответствующего автора. Данные не являются общедоступными из-за особой значимости используемой в экспериментальном процессе электроэнергетической системы, которая предполагает конфиденциальность в некоторых областях исследований, поэтому полностью раскрывать их данные неудобно.

Конфликт интересов: Авторы (Цян Ли, Сяо Ляо, Вэй Цуй и Ин Ван) работали в компании State Grid Information & Telecommunication Group Co., Ltd. Остальные авторы заявляют, что исследование проводилось в отсутствие каких-либо коммерческих или финансовые отношения, которые могут быть истолкованы как потенциальный конфликт интересов.

Ссылки

1. Баггенстосс, П.М. Проект PDF с максимальной энтропией с использованием ограничений плотности объектов: приложения в обработке сигналов. IEEE Пер. Сигнальный процесс. 2015, 11, 28:15–28:25. [Перекрестная ссылка]
2. Ли, Ю.; Бай, Х.; Юань, Д.; Ю, Ц.; Сан, Х.; Го, Ю.; Чжан, Л.; Йе, Дж. Высокоэнтропийный двумерный оксид на основе меди как стабильный и активный фототермический катализатор. Нат. Коммун. 2023, 14, 3171. [CrossRef] [ПабМед]
3. Бисикало, О.; Харченко В.; Ковтун В.; Крак, И.; Павлов С. Параметризация стохастической модели для оценки переменных малых данных в энтропийном базисе Шеннона. Энтропия 2023, 25, 184. [CrossRef] [ПабМед]
4. Чен, К.; Чжан, Ф.; Су, Л.; Лин, Б.; Чен, С.; Чжан Ю. Оценка слияния параметров состояния интеллектуальных транспортных средств на основе ИММ-МССКФ. Прил. наук. 2024, 14, 4495. [CrossRef]
5. Карли, ФП; Чен, Т.; Люнг, Л. Ядра максимальной энтропии для идентификации системы. IEEE Транс. Автомат. Контроль 2016, 62, 1471–1477. [Перекрестная ссылка]
6. Фан, Х.; Чен, Л.; Хуанг, Д.; Тиан, Ю.; Чжан, Х.; Цзяо, М.; Чжоу, З. От отдельных металлов к высокоэнтропийным сплавам: как машинное обучение ускоряет разработку металлических электрокатализаторов. Адв. Функц. Матер. 2024, 1, 2401887. [CrossRef]
7. Хан, Т.Т.; Ге, СС; Ли, Т.Х. Нелинейные системы с постоянным временем пребывания и переключением: парадигма вариаций и конструкция датчиков. IEEE Пер. Автомат. Контроль 2010, 55, 321–337.
8. Чези, Г. Стабилизация и снижение энтропии с помощью SDP-проектирования контроллеров с обратной связью по выходу фиксированного порядка и параметров настройки. IEEE Транс. Автомат. Контроль 2017, 62, 1094–1108. [Перекрестная ссылка]
9. Вэй, П.; Li, HX Пространственно-временная энтропия для обнаружения аномалий и локализации литий-ионных аккумуляторных блоков. IEEE Транс. Индийский Электрон. 2023, 70, 12851–12859. [Перекрестная ссылка]
10. Ван З.; Ли, Г.; Яо, Л.; Кай, Ю.; Лин, Т.; Чжан, Дж.; Донг, Х. Интеллектуальная схема обнаружения неисправностей для ветряных турбин с постоянной скоростью, основанная на улучшенной многомасштабной нечеткой энтропии и адаптивной хаотической машине опорных векторов на основе оптимизации Aquila. ИСА Транс. 2023, 138, 582–602. [Перекрестная ссылка] [ПабМед]

Machine Translated by Google

11. Данешпажух, А.; Сами, А. Обнаружение выбросов на основе энтропии с использованием полуконтролируемого подхода с несколькими положительными примерами. Распознавание образов. Летт. 2014, 49, 77–84. [Перекрестная ссылка]
12. Цюй, Б.; Ван, З.; Шен, Б.; Донг, Х. Децентрализованная оценка динамического состояния многомашинных энергосистем с негауссовскими шумами: обнаружение и локализация выбросов. Автоматика 2023, 153, 111010. [CrossRef]
13. Фэн, З.; Ван, Г.; Пэн, Б.; Он, Дж.; Чжан, К. Новый надежный фильтр Калмана с энтропией распределения Вассерштейна с минимальной ошибкой неопределенность модели и негауссов шум. Сигнальный процесс. 2023, 203, 108806. [CrossRef]
14. Вс, М.; Дэвис, Мэн; Праудлер, ИК; Хопгуд, Дж. Р. Адаптивный ядро-фильтр Калмана. IEEE Транс. Сигнальный процесс. 2023, 71, 713–726. [Перекрестная ссылка]
15. Ву, Л.; Лин, Х.; Центр.; Тан, К.; Гао, З.; Лю, З.; Ли, С.З. За пределами гомофилии и предположения об однородности: частотно-адаптивные графовые нейронные сети на основе отношений. IEEE Транс. Нейронная сеть. Учиться. Сист. 2023, 35, 8497–8509. [Перекрестная ссылка] [ПабМед]
16. Го, Л.; Ван, Х. Фильтрация минимальной энтропии для многомерных стохастических систем с негауссовскими шумами. IEEE Транс. Автомат. Контроль 2006, 51, 695–700.
17. Ван Ю.; Тан, ГГ; Ли, Л. Разреженное представление на основе энтропии с минимальными ошибками для устойчивой кластеризации подпространств. IEEE Транс. Сигнальный процесс. 2015, 63, 4010–4021. [Перекрестная ссылка]
18. Ван, М.; Дэн, В. Глубокое распознавание лиц: опрос. Нейрокомпьютинг 2021, 429, 215–244. [Перекрестная ссылка]
19. Лин, М.; Ченг, К.; Пэн, З.; Донг, Х.; Цюй, Ю.; Мэн, Г. Идентификация нелинейной динамической системы с использованием методов разреженной регрессии и разделимых наименьших квадратов. Дж. Звук Виб. 2021, 505, 116141. [CrossRef]
20. Ву, З.; Пэн, С.; Ма, В.; Чен, Б.; Принсипи, Дж. К. Алгоритмы минимальной энтропии ошибок с ограничениями штрафа за разреженность. Энтропия 2015, 17, 3419–3437. [Перекрестная ссылка]
21. Хуан, С.; Тран, Т.Д. Восстановление разреженного сигнала путем минимизации обобщенных энтропийных функций. IEEE Транс. Сигнальный процесс. 2018, 67, 1322–1337. [Перекрестная ссылка]
22. Юэ, Х.; Ван, Х. Минимальный энтропийный контроль ошибок отслеживания с обратной связью для динамических стохастических систем. IEEE Транс. Автомат. Контроль 2003, 48, 118–122.
23. Ван Х. Минимальное энтропийное управление негауссовскими динамическими стохастическими системами. IEEE Транс. Автомат. Контроль 2002, 47, 398–403. [Перекрестная ссылка]
24. Митра Р.; Бхатия, В. Оценка канала на основе критерия минимальной энтропии ошибок для массивного MIMO в VLC. IEEE Транс. Вех. Технол. 2018, 68, 1014–1018. [Перекрестная ссылка]
25. Эрдогмус, Д.; Принсипи, Дж. К. Свойства конвергенции и эффективность данных критерия минимальной энтропии ошибок при обучении адалина. IEEE Транс. Сигнальный процесс. 2003, 51, 1966–1978.
26. Чен, Б.; Юань, З.; Чжэн, Н.; Принсипи, Дж. К. Алгоритм энтропии минимальной ошибки ядра. Нейрокомпьютинг 2013, 121, 160–169. [Перекрестная ссылка]
27. Ху, Т.; Ву, К.; Чжоу, DX. Сходимость градиентного спуска для принципа минимальной энтропии ошибок в линейной регрессии. IEEE Транс. Сигнальный процесс. 2016, 64, 6571–6579. [Перекрестная ссылка]
28. Чжан Ю.; Чен, Б.; Лю, Х.; Юань, З.; Принсипи, Дж. К. Сходимость алгоритма с минимальной энтропией ошибок с фиксированной точкой. Энтропия 2015, 17, 5549–5560. [Перекрестная ссылка]
29. Чен, Б.; Чжу, Ю.; Ху, Дж.; Чжан М. Об оптимальных оценках с критерием минимальной энтропии ошибок. Дж. Франкл. Инст. 2010, 347, 545–558.
30. Херави, Арканзас; Ходтани, Г.А. Новое теоретико-информационное соотношение между минимальной энтропией ошибок и максимальной коррентропией. Сигнальный процесс IEEE. Летт. 2018, 25, 921–925.
31. Ян, С.; Тан, Дж.; Чен, Б. Надежное непрерывное метаобучение на основе всплесков, улучшенное с помощью ограниченного критерия минимальной энтропии ошибок. Энтропия 2022, 24, 455. [CrossRef] [ПабМед]
32. Данг, Л.; Чен, Б.; Ван, С.; Ма, В.; Рен, П. Робастная оценка состояния энергосистемы с минимальной энтропией ошибок без запаха Фильтр Калмана. IEEE Транс. Инструмент. Измер. 2020, 69, 8797–8808. [Перекрестная ссылка]
33. Ван, Б.; Ху, Т. Принцип полуконтролируемой минимальной энтропии ошибок с использованием распределенного метода. Энтропия 2018, 20, 968. [CrossRef] [ПабМед]
34. Чен, Б.; Син, Л.; Сюй, Б.; Чжао, Х.; Принсипи, Дж. К. Анализ надежности оценки энтропии минимальной ошибки. IEEE Транс. Нейронная сеть. Учиться. Сист. 2016, 29, 731–737. [Перекрестная ссылка] [ПабМед]
35. Шен, П.; Ли, К. Метод минимальной общей энтропии ошибок для оценки параметров. IEEE Транс. Сигнальный процесс. 2015, 63, 4079–4090. [Перекрестная ссылка]
36. Котари, П.; Крайсс, С.; Алахи, А. Прогнозирование траектории движения человека в толпе: перспектива глубокого обучения. IEEE Транс. Интел. Трансп. Сист. 2021, 23, 7386–7400. [Перекрестная ссылка]
37. Ким, Н.; Квон, К. Нормализованный алгоритм минимальной энтропии ошибок с рекурсивной оценкой мощности. Энтропия 2016, 18, 239. [Перекрестная ссылка]
38. Чен, Б.; Ма, Р.; Ю, С.; Ду, С.; Цинь, Дж. Грейнджер Анализ причинно-следственной связи на основе квантованного критерия минимальной энтропии ошибок. IEEE Сигнальный процесс. Летт. 2019, 26, 347–351. [Перекрестная ссылка]
39. Никнам, М.; Сантос, ЛФ; Кори, Д.Г. Экспериментальное обнаружение корреляционной энтропии Реньи в модели центрального спина. Физ. Преподобный. Летт. 2021, 127, 080401. [CrossRef] [ПабМед]
40. Ким, Т.; Сан Ким, Д.; Джанг, LC; Ли, Х.; Ким, Х. Представления вырожденных полиномов Эрмита. Адв. Прил. Математика. Мех. 2022, 139, 102359. [CrossRef]

Machine Translated by Google

41. Лин, В.; Чжан, Дж. Э. Действительные области плотностей Грама – Шарлье с кумулянтами высокого порядка. Дж. Компьютер. Прил. Математика. 2022, 407, 113945. [Перекрестная ссылка]

42. Мацусита, Р.; Брандао, Х.; Нобре, И.; Да Силва, С. Дифференциальная оценка энтропии с помощью ядра Паретиана: тяжесть хвоста и сглаживание. Физ. Стат. Мех. Прил. 2024, 1, 129850. [CrossRef]

43. Каналес, К.; Галарсе, К.; Рубио, Ф.; Пинеда, Ф.; Ангита, Дж.; Баррос, Р.; Парраге, М.; Дайль, ЛК; Агирре, Дж.; Армихо, Ф.; и др. Тестирование теста: сравнительное исследование морской микробной коррозии в лабораторных и полевых условиях. АСУ Омега 2021, 6, 13496–13507. [Перекрестная ссылка] [ПабМед]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Оптимизация отбора генов для классификации рака в Высокомерные данные с использованием улучшенного африканского Алгоритм стервятников

Мона Г. Гафар 1,2,* , † Амр А. Абохани 3, , Ахмед Э. Эльхули 4† и Амр А. Абд Эль-Магид 5,†

1 Факультет компьютерной инженерии и информации, Инженерный колледж в Вади Альдавасир, Университет принца Саттама бин Абдель Азиза, Хардж 16278,
2 Саудовская Аравия. Факультет машинного обучения и информационного поиска, искусственный интеллект, Университет Кафрельшейха,
3 Кафрельшейх 33511, Египет. Факультет компьютеров и информации, Университет Кафрельшейха. ,
4 Кафрельшейх 33511, Египет; amrabohany8@gmail.com Кафедра биомедицинской инженерии, факультет электротехники, Университет Менофия,
5 Менофия 32951, Египет; eng.elkholy.ahmed@gmail.com Факультет информационных систем Университета Сохаг, Сохаг 82511, Египет; amr.atef@commerce.sohag.edu.eg * Адрес для переписки:
† m.gafar@psau.edu.sa или dr.mona.gamal85@gmail.com Эти авторы внесли равный вклад в эту работу.

Аннотация: В этом исследовании представлен новый метод, названный RBAVO-DE (Оптимизация бинарных африканских стервятников на основе дифференциальной эволюции), направленный на решение проблемы отбора генов (GS) в многомерных данных RNA-Seq, в частности, rnaseqv2 lluminaHiSeq rnaseqv2 un. Набор нормализованных данных по генам RSEM уровня 3 edu , который содержит более 20 000 генов. Секвенирование РНК (RNA-Seq) — это преобразующий подход, который позволяет проводить комплексную количественную оценку и характеристику экспрессии генов, превосходя возможности технологий микрочипов, предлагая более детальное представление данных об экспрессии генов RNA-Seq. Количественный анализ экспрессии генов может иметь решающее значение для выявления генов, которые отличают нормальные ткани от злокачественных. Однако управление этими многомерными плотными матричными данными представляет собой серьезные проблемы. Алгоритм RBAVO-DE предназначен для тщательного выбора наиболее информативных генов из набора данных, содержащего более 20 000 генов, и оценки их значимости для двадцати двух наборов данных о раке. Чтобы определить эффективность выбранных генов, в этом исследовании используются классификаторы машины опорных векторов (SVM) и k-ближайшего соседа (k-NN). По сравнению с двоичными версиями широко известных метаэвристических алгоритмов RBAVO-DE демонстрирует превосходную производительность. Согласно критерию суммы рангов Уилкоксона с уровнем значимости 5%, RBAVO-DE достигает точности классификации до 100% и уменьшает размер признаков до 98% в большинстве из двадцати двух исследованных наборов данных о раке. Это достижение подчеркивает потенциал RBAVO-DE в повышении точности отбора генов для исследований рака, тем самым способствуя более точной и эффективной идентификации ключевых генетически

Ключевые слова: секвенирование РНК; отбор генов; высокая размерность; метаэвристические алгоритмы; алгоритм африканских стервятников; дифференциальная эволюция; техника рельефа

check for updates

Цитирование: Гафар, М.Г.; Абохани, А.А.; Эльхули, А.Е.; Абд Эль-Магид, А.А. Оптимизация отбора генов для Классификация рака в Большие данные с использованием улучшенного алгоритма африканских стервятников. Алгоритмы 2024, 17, 342. https://doi.org/10.3390/a17080342 .

Академические редакторы: Мухаммед Аднан Хан и Фрэнк Вернер

Поступила: 27 апреля 2024 г.
Пересмотрено: 16 июля 2024 г.
Принято: 1 августа 2024 г.
Опубликовано: 6 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Дезоксирибонуклеиновая кислота (ДНК) составляет наш генетический код, содержащий рецепт нашего существования. Хотя каждая клетка имеет одинаковую ДНК, структура каждой ткани уникальна и служит определенной цели. Механизм транскрипции РНК определяет, какие гены в клетке активны, позволяя трансформировать РНК в белки, ответственные за структуру и функциональность клетки [1]. Мы анализируем данные экспрессии генов RNA-Seq (GED), чтобы определить генетические изменения и оценить биомаркеры заболеваний. Анализ дифференциальной экспрессии выявляет количественные различия в экспрессии генов, что позволяет нам классифицировать гены, экспрессия которых меняется в различных условиях. Этот метод п

Machine Translated by Google

болезней и найти способы борьбы с ними. Технологии профилирования экспрессии генов за последние годы значительно развились [2].

Двумя популярными методами являются микрочипы, в которых используется подход, основанный на гибридизации, и RNA-Seq, основанный на секвенировании следующего поколения. Эти технологии продемонстрировали свою ценность в продвижении нашего понимания экспрессии генов [3]. Известные исследования, например, Chen et al. [4] и Нуньес и др. [5] подчеркнули потенциал этих методов, а Wang et al. [6] предоставили ценную информацию о подходе RNA-Seq.

Два рассматриваемых метода служат цели количественной оценки экспрессии генов для классификации и статистического исследования. В этой статье были выбраны данные количественного определения, полученные с помощью подхода RNA-Seq на основе секвенирования следующего поколения (NGS), поскольку он обеспечивает большую точность определения уровней количественного определения РНК по сравнению с данными микрочипов [7]. RNA-Seq преодолел некоторые ограничения микроматричного анализа, такие как зависимость от предшествующих знаний о секвенировании, что ограничивало диапазон обнаружения [8,9]. В отличие от микрочипов, RNA-Seq не требует каких-либо предварительных знаний и значительно расширяет динамический диапазон обнаружения. Это усовершенствование повышает точность результатов и позволяет идентифицировать полный набор генов, обеспечивая точное понимание биомаркеров заболеваний. В результате RNA-Seq стал мощным инструментом для исследователей, предлагающим ценную информацию о молекулярных механизмах, лежащих в основе различных заболеваний [10].

«Проклятие размерности» [11] стало распространенной проблемой в наше время из-за обилия доступных данных. Это вызвало всплеск развития методов и алгоритмов выбора признаков (FS). Алгоритмы FS можно разделить на четыре различных метода: фильтр, оболочка, встроенный и гибридный [12,13]. Эти подходы направлены на выявление наиболее информативных признаков, которые позволяют различать разные классы.

В нашем случае мы озабочены выявлением генов, связанных с опухолями.

Фильтрационный подход — это широко используемый метод в FS, который включает оценку релевантности отдельных генов на основе статистических показателей. Этот метод известен своей высокой точностью выбора наиболее удовлетворительной группы генов. Тем не менее, подход с фильтрацией имеет некоторые ограничения, поскольку функционирование каждого гена индивидуально игнорирует взаимосвязи между генами, что может привести к проблеме локального оптимума [14]. Стоит отметить, что подход с фильтрацией можно разделить на два подтипа — одномерный и многомерный — причем последний учитывает корреляции между генами. Некоторые примеры подхода с фильтрацией включают Релиф [15], показатель Фишера [16], t-критерий [17] и прирост информации [18].

Метод-обертка исследует все возможные подмножества генов для тестирования и создания подмножества генов для определения их значения. Для определения результатов каждого подмножества используется специальный классификатор, а метод категоризации используется несколько раз для каждой оценки. По сравнению с подходом с фильтрацией подход с использованием оболочки обеспечивает превосходную производительность за счет использования метода категоризации, который направляет процедуру обучения. Однако этот метод требует значительных временных и вычислительных ресурсов, особенно при работе с крупномасштабными данными [19].

Метаэвристические методы (МНМ) — это эвристики высокого уровня, используемые в математической оптимизации и информатике для поиска удовлетворительных решений задач оптимизации, когда информация несовершенна или ресурсы ограничены [20]. МНМ выбирают подмножество решений, что делает их полезными для различных задач оптимизации из-за небольшого количества предположений о проблеме [21]. Однако они не гарантируют глобально оптимальных решений. Многие МНМ используют стохастическую оптимизацию, то есть решение основано на случайных величинах, генерируемых в ходе процесса [22].

Метаэвристические методы (МНМ) более практичны, чем традиционные итеративные методы и алгоритмы оптимизации, поскольку они могут исследовать более широкий диапазон возможных решений. Следовательно, МНМ стали предпочтительной стратегией решения задач оптимизации [23]. Многочисленные исследовательские работы показали, что среди различных методов-оболочек МНМ хорошо подходят для решения проблемы выбора функций (FS). Стохастические методы, включая МНМ, могут быстро дать оптимальные или почти оптимальные результаты. Они

Machine Translated by Google

предлагают такие преимущества, как гибкость, самоуправление без детальных математических свойств и способность оценивать множество результатов одновременно. Недавно для решения проблемы FS были разработаны различные MHM , обеспечивающие надежные, почти оптимальные решения при значительно сниженных вычислительных затратах [24].

Встроенные методы FS используют технику обучения для выбора подходящих генов, которые взаимодействуют с процессом классификации. Этот метод сочетает в себе технику FS как часть модели обучения. Алгоритм обучения обучается с использованием начального подмножества атрибутов для оценки меры оценки ранговых значений атрибутов [25]. Конечная цель — сократить время вычислений для переклассификации различных подмножеств за счет включения этапа FS в процесс обучения. Некоторые методы в этом подходе выполняют взвешивание признаков на основе моделей регуляризации с целевыми функциями, которые минимизируют ошибки аппроксимации, обеспечивая при этом небольшие или точно равные нулю коэффициенты признаков. Примерами встроенных методов являются алгоритм выбора подмножества признаков на основе правил первого порядка индуктивного обучения (FOIL) и SVM на основе рекурсивного исключения признаков (SVM-RFE) [26].

Гибридный метод — это хорошо продуманный метод, объединяющий методы фильтра и оболочки для использования сильных сторон каждого из них. Этот подход начинается с фильтра, который уменьшает размерность пространства признаков, генерируя несколько подмножеств средней сложности. Впоследствии в качестве метода обучения для выбора наиболее подходящего подмножества кандидатов используется оболочка [27]. Гибридный подход объединяет точность оберток и эффективность фильтров, что приводит к оптимальной методологии [28,29].

1.1. Мотивация и вклад

В этой статье представлен эффективный глобальный метаэвристический алгоритм оптимизации, который называется алгоритмом оптимизации африканских стервятников (AVO) [30], который имитирует образ жизни и привычки питания африканских стервятников. Предлагаемый алгоритм состоит из четырех основных этапов: разделение совокупности на три группы — лучшее решение, второе лучшее решение и оставшиеся решения; измерение уровня голода для формулирования математической модели эксплуатации, исследования и перемещения стервятников; исследование стервятников с использованием двух стратегий, позволяющих стервятникам преодолевать большие расстояния в течение длительных периодов времени в поисках пищи в случайных местах; и эксплуатация стервятников, включающая два подэтапа. На первом подэтапе используются две различные стратегии: осадный бой для сильных стервятников и ротационный полет, который образует спиральную активность между выдающимся стервятником и остальными. На втором подэтапе используются две стратегии : сбор стервятников вокруг источника пищи и враждебная осада, при которой стервятники становятся более агрессивными и пытаются украсть еду, оставленную здоровыми стервятниками.

В этом исследовании предлагается расширенный бинарный вариант метода AVO, а именно метод RBAVO-DE, который представляет собой продуктивный подход, демонстрирующий точную производительность при решении проблемы GS. Во-первых, существует высокая вероятность того, что рекомендуемый метод позволит избежать локальных оптимумов и обеспечит достаточную точность исследования, быструю сходимость и улучшенную стабильность. По сравнению с недавними MHM, предлагаемая методика RBAVO-DE обеспечивает повышенную эффективность за счет создания оптимальных или существенно оптимальных решений для многих анализируемых ситуаций. Алгоритм Relief используется для идентификации только связанных объектов и создания окончательного набора данных классификации. RBAVO-DE сочетает в себе алгоритм Relief с подходом DE для увеличения возможностей исследования и получения наилучших результатов в пространстве решений посредством повторений. Он также использует передаточную функцию для преобразования реальных значений позиции в двоичные значения. Подход RBAVO-DE имеет смысл в GS, поскольку он прост для понимания и реализации, позволяет решать различные задачи оптимизации, дает ценные результаты за приемлемое время , требует меньше вычислительной мощности и использует небольшое количество управляющих параметров. Основной

вклад этой статьи заключается в следующем: • Данные уровня 3, основанные на секвенировании следующего поколения (RNA-Seq), были предварительно обработаны. • Способность предлагаемого метаэвристического алгоритма AVO решать проблему GS не исследовалась, поскольку GED RNA-Seq никогда не использовался с алгоритмом AVO.

Machine Translated by Google

• Для создания двоичной версии, известной как алгоритм RBAVO-DE, алгоритм AVO изменяется и воссоздается. • Предлагаемый алгоритм RBAVO-DE сочетает в себе бинарный вариант AVO с подходом Relief и DE для улучшения возможностей исследования пространства поиска и повышения достигаемых оптимальных результатов. • Предложенный алгоритм RBAVO-DE впервые применяется к GED RNA-Seq. • Для оценки результатов используются несколько показателей эффективности, включая среднюю приспособленность, точность классификации, количество выбранных генов, точность, отзыв и показатель F1. • Проведено сравнение влияния представленного алгоритма RBAVO-DE, использующего два рекомендуемых классификатора ML (SVM и k-NN), и других алгоритмов, представленных в литературе.

• Для оценки предлагаемого метода RBAVO-DE используются двадцать два различных набора данных о раке , и показаны результаты. • Выбранные гены исследуются с использованием биомаркеров, связанных с раком.

1.2. Структура

Данная статья состоит из пяти основных разделов. В разделе 2 рассматриваются предыдущие исследования ФС с использованием RNA-Seq GED. Далее следует раздел 3, в котором подробно обсуждается алгоритм RBAVO-DE, расширенная версия AVO, включая его параметры для адресации GS. В разделе 4 представлены экспериментальные результаты в сравнении с несколькими недавними МНМ. Наконец, раздел 6 завершает данное исследование и предлагает пути дальнейшего исследования.

2. Сопутствующие работы

В этом разделе мы обсуждаем литературу, посвященную методам, которые исследователи используют для классификации GED RNA-Seq, которая обычно имеет высокую размерность. Для достижения оптимальной эффективности методов классификации крайне важно игнорировать нерелевантные и несвязанные гены; следовательно, выбор подходящих генов является жизненно важным этапом перед использованием методов машинного обучения и глубокого обучения (DL) [31] или любых других методов классификации. В связи с этим мы изучили некоторые соответствующие статьи в этой области, чтобы достичь цели категоризации RNA-Seq для идентификации рака.

Якуб и др. [32] представили передовую технику для GS, известную как алгоритм поиска синус-косинус-кукушка (SCCSA), гибридный метод, адаптированный для работы вместе с устоявшимися моделями машинного обучения, такими как SVM. Этот инновационный алгоритм GS был оценен с использованием эталонного набора данных по раку молочной железы, где его эффективность была тщательно проанализирована и сравнена с другими методологиями GS. Для уточнения выбора признаков изначально применялся метод минимальной избыточности и максимальной релевантности в качестве предварительного этапа фильтрации. Впоследствии гибридный подход SCCSA был применен для дальнейшего улучшения и точной настройки процесса GS. Заключительный этап включал использование классификатора SVM для классификации набора данных на основе выбранных генов. Учитывая решающую важность GS для декодирования сложных наборов биологических данных, SCCSA становится важнейшим инструментом для классификации наборов данных, связанных с раком.

Джоши и др. [33] представили инновационную стратегию оптимизации под названием PSO-CS, интегрированную с DL для классификации опухолей головного мозга. Этот метод повышает эффективность метода оптимизации роя частиц (PSO) за счет включения алгоритма поиска кукушки (CS), оптимизирующего процесс классификации. После этой оптимизации PSO-CS использует DL для классификации GED, связанного с опухолями головного мозга, идентифицируя различные классы, связанные с конкретными опухолями, наряду с методом оптимизации PSO-CS. Благодаря интеграции метода PSO- CS с DL он значительно превзошел другие модели DL и ML в отношении точности классификации, о чем свидетельствуют различные показатели производительности.

Махто и др. [34] представили новаторский подход к классификации рака с помощью интегрированного метода, основанного на CS-SMO (оптимизация CS и Spider Monkey) для GS. Первоначально фитнес-функция алгоритма оптимизации Spider Monkey (SMO) модифицируется с использованием алгоритма CS. Эта модификация использует сильные стороны обоих МНМ.

Machine Translated by Google

идентифицировать подмножество генов, способных предсказывать рак на ранней стадии. Для дальнейшего повышения точности алгоритма CS-SMO используется этап предварительной обработки MRMR, чтобы уменьшить сложность наборов данных об экспрессии генов рака. Впоследствии эти подмножества генов обрабатываются с использованием DL для классификации различных типов рака. Эффективность подхода CS-SMO в сочетании с DL оценивалась с использованием восьми эталонных наборов данных по экспрессии генов на микрочипах для рака, исследуя его эффективность по различным показателям. Метод CS-SMO, интегрированный с DL, продемонстрировал превосходную точность классификации во всех исследованных крупномасштабных наборах данных об экспрессии генов рака, превосходя существующие модели DL и ML.

Неггаз и др. [35] представили улучшенную версию оптимизации поиска пищи мантами , названную MRFO-SC, в которой для решения проблемы GS используются тригонометрические операторы, основанные на алгоритме синус-косинус (SC). Модель k-NN использовалась для отбора набора генов.
Кроме того, статистическую значимость MRFO-SC оценивали с использованием критерия суммы рангов Уилкоксона при уровне значимости 5%. Результаты были оценены и сравнены с некоторыми недавними MHM. Результаты сравнения и экспериментов подтвердили эффективную работу предлагаемого MRFO-SC на наборах эталонных данных высокой и низкой размерности, получив наибольшую точность классификации на 85% наборов эталонных данных GS.

Люу и др. [36] исследовали биомаркеры рака, сосредоточив внимание на значении генов в их влиянии на классификацию. Они использовали двухэтапный подход — предварительную обработку данных и использование сверточной нейронной сети — для классификации типа опухоли. На втором этапе они создали тепловые карты для каждой категории, чтобы идентифицировать гены, связанные с пикселями с самой высокой интенсивностью на тепловых картах. Затем они оценили пути выбранных генов. Во время предварительной обработки они удалили уровни экспрессии генов, которые не были изменены во время фазы GS, используя порог дисперсии 1,19, что уменьшило количество генов с 19 531 до 10 381. Окончательная полученная точность классификации составила 95,59, что является хорошим показателем, но его можно улучшить с помощью более эффективной методологии FS для дальнейшего уменьшения размерности данных.

Халифа и др. [37] построено на основе вышеупомянутой статьи [36]. В центре внимания их исследования были пять типов данных о раке: карцинома эндометрия тела матки (UCEC), плоскоклеточная карцинома легких (LUSC), аденокарцинома легких (LUAD), светлоклеточная карцинома почек и почек (KIRC) и инвазивная карцинома молочной железы (BRCA).). Базовый набор данных, использованный для исследования, включал 2086 записей и 972 атрибута. Каждая запись содержала подробную информацию об образце, а каждый атрибут включал значения RNA-Seq для конкретного гена, представленные как RPKM (чтений на килобаз на миллион) [38]. Исследователи применили смешанный подход, используя метод двоичного PSO и деревьев решений (BPSO-DT) для предварительной обработки данных. Из 971 атрибута 615 были выбраны как лучшие атрибуты PHK-SEquation. Предложенный метод достиг общей точности классификации тестирования 96,90%, о чем свидетельствуют предложенные результаты и использованные меры оценки.

Сяо и др. [39] оценили свою методологию, используя три набора данных RNA-Seq: аденокарцинома желудка (STAD), BRCA и LUAD. Их подход основывался на методах DL, в которых они использовали пять классификаторов, а затем использовали технику DL для ансамблирования каждого результата пяти классификаторов. Это привело к повышению точности классификации всех прогнозов: набор данных BRCA достиг точности 98,4%, STAD достиг точности 98,78%, а LUAD достиг точности 99,20%.

Лю и др. [40] использовали данные микрочипов и гибридный подход для решения этой проблемы. В отличие от ранее упомянутых работ, они изучали каждый класс рака отдельно. Для оценки эффективности они использовали четыре генных показателя, связанных с небольшими круглоклеточными опухолями, раком толстой кишки, раком легких и лейкемией. Предложенный ими метод использовал Relief в качестве метода предварительной обработки для исключения генов с более низкой корреляцией с конкретным классом рака, а затем PSO в качестве метода поиска. Наконец, они использовали модель SVM для оценки точности классификации выбранного подмножества генов и получения окончательного оптимального подмножества генов для каждого типа рака.

Судя по имеющимся исследованиям, кажется, что большинство исследований с использованием RNA-Seq GED все еще находятся на ранних стадиях, и исследователи пытаются реализовать и протестировать различные идеи.

Machine Translated by Google

в этой перспективной сфере. Хотя в литературе содержится множество экспериментов с использованием нескольких методов, таких как новейшие методы FS и DL, ни один метод не является идеальным из-за высокой размерности RNA-Seq GED. ФС RNA-Seq GED играет решающую роль в определении связи между геном и его категорией. Это важнейшая задача предварительной обработки для проверки генных биомаркеров рака и преодоления проклятия размер Следовательно, это исследование направлено на внедрение нового подхода-оболочки, алгоритма RBAVO-DE , и впервые применение его к данным RNA-Seq. Он также сравнивает эффективность предлагаемого алгоритма с эффективностью других методов FS.

3. Предлагаемый RBAVO-DE для GS.

В этой статье предлагается улучшенный вариант AVO, известный как RBAVO-DE, для обнаружения наименьших релевантных подмножеств генов в процессе классификации и игнорирования нерелевантных генов. RBAVO-DE использует двоичный алгоритм AVO на основе рельефа в сочетании с методом DE. Основной особенностью RBAVO-DE является его способность максимизировать точность при использовании минимально возможного количества функций. Предлагаемый RBAVO-DE состоит из двух основных этапов. Во-первых, существует этап предварительной обработки, на котором алгоритм Релифа используется для определения того, какие функции являются значимыми, путем присвоения каждому объекту веса, а затем удаления объектов, которые не имеют значения и имеют наименьшие веса. На втором этапе применяются бинарный алгоритм AVO и метод DE для выявления наиболее подходящих и уникальных функций. Алгоритм AVO склонен к ловушке локального оптимума при решении крупномасштабных задач. Алгоритм AVO включает в себя технику DE, позволяющую предотвратить

Предлагаемый алгоритм RBAVO-DE для реализации стратегии GS требует применения алгоритма Relief, инициализации, повышения позиции с использованием алгоритма AVO, двоичного преобразования, оценки пригодности и интеграции с DE. В следующих подразделах объясняются эти шаги.

3.1. Применение алгоритма рельефа для фильтрации

признаков. Этот шаг направлен на предварительную обработку совокупности с использованием алгоритма рельефа [41], который считается быстрым, простым и эффективным методом фильтрации для поиска связанных друг с другом признаков. Основная цель этого алгоритма — найти характеристики, которые различают значения выборки, и сгруппировать похожие выборки близко друг к другу. В результате метод зависит от взвешенного ранжирования признаков, где признак с более высоким весом указывает на лучшую эффективность классификации.

После случайного выбора выборки алгоритм Релифа исследует два разных типа ближайших выборок: близкие к совпадению выборки, которые связаны с выборками из одного и того же класса, и близкие к совпадению выборки, которые связаны с выборками из других классов. Значения почти попадания и почти попадания можно использовать для определения весов объектов. Чтобы оценить важность в процессе классификации, веса признаков расположены от наиболее значимых к наименее значимым. Наконец, выбираются признаки с наиболее значимыми весами . Для определения веса W для признака A можно использовать следующую формулу:

$$BA = \sum_{j=1}^{H} {}^{x\,j}_{A}\,HM(x\,j)_A^{\,2} - {}^{x\,j}_{A}\,NH(x\,j)_A^{\,2}. \quad (1)$$

где WA обозначает вес признака A, x j в одном и том $\tilde{A}\,j$ обозначает значение признака A для данных x , же или разных экземплярах, a N обозначает количество выборок. Ближайшими точками данных к классам x являются NH(x j) и NM(x j).

Алгоритм Relief сужает фокус только на необходимых функциях и сводит к минимуму область поиска, чтобы помочь алгоритму AVO быстрее находить лучшие функции.

3.2. Инициализация населения

Предлагаемый алгоритм BAVO начинается со случайной генерации совокупности из N позиций. Вектор размерностей D, эквивалентный количеству объектов в исходном наборе данных, описывает каждую позицию как возможное решение в ее ограниченной нижней и верхней точке.

Machine Translated by Google

границы. Этот случайно инициализированный шаг, ограниченный диапазоном [-1, 1] для каждой переменной вектора положения, использует случайно созданную позицию.

3.3. Повышение позиций с помощью алгоритма AVO

В этой статье представлен алгоритм AVO [30], метаэвристический алгоритм оптимизации, вдохновленный африканскими стервятниками и их жизненным и пищевым поведением. Основываясь на фундаментальных представлениях о стервятниках, алгоритм AVO настроен следующим образом: Первоначально алгоритм AVO предполагает, что размер популяции африканских стервятников состоит из N стервятников. Затем популяция африканских стервятников делится на три группы, чтобы отразить основную естественную функцию стервятников на основе рассчитанной функции приспособленности. В первую группу входят самые сильные стервятники, что является лучшим решением; во второй группе содержится стервятник, более слабый, чем в первой, что является вторым лучшим решением; и последняя группа включает оставшихся, более слабых стервятников, которые являются худшим решением.

На основании вышеизложенного предлагаемый алгоритм AVO состоит из четырех фундаментальных этапов для моделирования поведения различных типов стервятников. Следующие подразделы поясняют эти этапы.

3.3.1. Фаза разделения населения

Исходная популяция разбивается на группы путем оценки функции приспособленности решения. Лучшее решение выбирается как лучший стервятник в первой группе, а вторая группа содержит второе лучшее решение. Третья группа содержит остальные решения. Поскольку решения постоянно стремятся приблизиться к лучшим и вторым лучшим решениям, совокупность необходимо переоценивать для каждой итерации следующим образом:

$$P_g = \begin{cases} BestVultureg, & \text{если } prg = L1, \\ SecondBestVultureg\ 2, & \text{если } prg = L2, \end{cases} \quad (2)$$

$$prg = \frac{U_g}{\sum_{g=1}^{H} U_g}. \quad (3)$$

Лучший гриф в первой группе и второй лучший гриф во второй итерации группы обозначаются выражениями BestVultureg и SecondBestVultureg $_{2,\text{итерация}}$ соответственно. Вероятность выбора лучшего решения для каждой группы в точке g_i определяется с использованием подхода колеса рулетки, как показано в уравнении (3), prg . Два случайных параметра в диапазоне [0, 1] — это L1 и L2.

3.3.2. Этап измерения уровня голода

Этот этап используется для формулирования математической модели эксплуатации, исследования и перемещения стервятников. Когда они не голодны, стервятники обладают большей способностью искать пищу и могут летать дальше. Кроме того, голодные стервятники не могут летать на большие расстояния в поисках еды и могут стать агрессивными. Уровень голода грифа i (F_i) во время итерации i можно записать следующим образом, что используется для разработки математической модели эксплуатации, исследования и перемещения стервятников:

$$F_i = (2 \times rand + 1) \times z \times \left(1 - \frac{g_i}{G_{макс}}\right) + t, \quad (4)$$

$$t = час \times \sin w \left(\frac{\pi}{2} \times \frac{ги}{G_{макс}}\right) + потому\ что \left(\frac{\pi}{2} \times \frac{ги}{G_{макс}}\right) - 1. \quad (5)$$

Стервятники, по-видимому, сыты с момента исследования g_i показывает переход стервятников от переменной F до эксплуатации. Случайное число от 0 до 1 называется рандом . z обозначает случайное значение в интервале [1, 1]. Текущий номер итерации обозначается, а максимальный номер итерации ги , обозначается Gmax. Уравнение (5) вычисляет значение t , чтобы помочь более эффективно решать сложные задачи оптимизации и предотвратить достижение локального оптимума. Произвольное значение в интервале [2, 2] обозначается h. Вероятность проведения процесса разведки регулируется заранее установленными

Machine Translated by Google

постоянный параметр w; вероятность разведки растет по мере увеличения его стоимости. Разведка менее вероятна, поскольку ее ценность снижается.

Уравнение (4) утверждает, что по мере увеличения количества итераций F в F_i^t постепенно уменьшается. Как результате следующий шаг в предлагаемом алгоритме AVO можно определить следующим образом:

$$\begin{cases} \text{Фаза исследования (поиск еды в различных областях),} & \text{если } |F_i^t| \geq 1, \\ \text{Фаза эксплуатации (поиск еды в окрестностях),} & \text{если } |F_i^t| < 1. \end{cases} \quad (6)$$

3.3.3. Фаза исследования.

Стервятники характеризуются великолепной зрительной способностью находить подходящую пищу на этом этапе. На этом этапе используются две разные стратегии, позволяющие стервятникам преодолевать большие расстояния в течение длительного времени в поисках пищи в случайных местах. Эти местоположения выбираются с использованием случайного числа randP1 и заданного параметра P1, оба из которых имеют значения в диапазоне [0, 1]. На этапе разведки уровень голода |F больше или равен 1. Методы разведки F_i^t | являются описываются следующим образом:

$$x_i^{t+1} = \begin{cases} p_1^t - D_{gg} \times F_i, & \text{если randP1} < P1, \\ p_1^t - F_i \text{ ig } P1, + \text{rand} \times ((UB - LB) \times \text{rand} + LB), & \text{если randP1} > P1, \end{cases} \quad \text{если } |F_i^t| \geq 1, \quad (7)$$

$$D_i^t = |(2 \times \text{rand}) \times R_1^t - X_i^t|. \quad (8)$$

g+1 , где предстоящая обновленная позиция на следующей (g + 1) итерации обозначается X. Лучший $_i$. стервятник, выбранный для текущей итерации g, обозначается R g . Это определяется с помощью уравнения (2). Стервятники перемещаются хаотично, чтобы защитить пищу от других стервятников и обеспечить высокую степень случайности в их поисковом поведении. rand — это случайное значение между нулем и единицей. Верхняя граница переменных обозначается UB; их нижняя граница по LB; и текущая позиция в точке g $_{\text{итерация g}} — X_i$.

3.3.4. Этап эксплуатации

На этом этапе $|F_i^t|$ меньше 1. Эффективность предлагаемого алгоритма AVO оценивается в два подэтапа, которые включают этап эксплуатации. На каждом из этих подэтапов используются две разные стратегии. Для каждого внутреннего подэтапа соответствующая стратегия выбирается с использованием двух предопределенных параметров: P2 для первого подэтапа и P3 для второго, со значениями в диапазоне от 0 до 1. Эти два внутренних подэтапа объясняются следующим образом:

- Первая подфаза эксплуатации: на этой подфазе применяются две разные стратегии | меньше когда $|F_i^t|$ 1 и больше или равно 0,5. Выбор одной из этих двух стратегий осуществляется с использованием случайной величины randP2 в диапазоне [0, 1] и заданного параметра P2.

Начальная стратегия этой подфазы называется осадным боем, в котором участвуют достаточно мощные и несколько сытые стервятники. Более крепкие и здоровые стервятники стараются не делиться пищей с другими, потому что собираются вокруг одного источника пищи. Роясь рядом со здоровыми грифами и вступая в небольшие драки, более слабые грифы пытаются отобрать у них пищу. И наоборот, вторая стратегия называется ротационным полетом; это создает спиральное движение между превосходящим стервятником и остальными. Ниже показаны стратегии для первого подэтапа эксплуатации:

$$x_i^{t+1} = \begin{cases} Д_{ги} \times (Ф + \text{рэнд}) \quad d_t^t, & \text{если randP2} \geq P2, \\ P_1^t - (S_{\text{концерт 1}} + C_2^t), & \text{если randP2} < P2, \end{cases} \quad \text{если } 1 > |F_i^t| \geq 0{,}5, \quad (9)$$

$$Д_т^t = p_1^t - X_i^t, \quad (10)$$

$$C^t = P_1^t \times 1 \quad \frac{\text{ранд} \times X_i^t}{2\pi} \times \text{соз}(X_i^t), \quad (11)$$

Machine Translated by Google

$$C_2^г = P г \times \frac{\text{ранд} \times X^г_я}{2\pi} \times \text{грех}(X^г_я). \quad (12)$$

где следующее обновленное положение стервятника в следующем (g + 1) итерация обозначается по $X^{g+1}_я$ определяется с помощью уравнения (8), а rand представляет собой произвольное значение от 0 до 1. Расстояние d g между стервятником и одним из лучших два стервятника оцениваются с использованием уравнения (10); $R^т_g$ обозначает соответствующий лучший стервятник в расчете; ток g с $g^й$ итерация, которая вычисляется с использованием уравнения (2); $C^г_1$ и $C^г_2$ обозначает использованием уравнений (11) и (12) соответственно; и X в точке $X^г_я$ текущую позицию.

g итерация .

• Вторая подфаза эксплуатации: Эта подфаза выполняется, когда $|F_{ги}|$ составляет менее 0,5. На этом подфазе различные виды стервятников собираются вокруг запасов еды и участвуют во многих осадах и драках. На этом подэтапе используются две разные стратегии. Предопределенный параметр P3 и произвольное значение randP3 со значениями в диапазоне от 0 до 1 используются для принятия решения о том, какую из этих двух стратегий использовать.

Первоначальная стратегия этого подэтапа называется «собрать стервятников вокруг источника пищи». В этой стратегии разные виды стервятников ищут пищу и могут конкурировать вблизи одного источника. Вторая стратегия известна как враждебная осада. В этой стратегии стервятники становятся более агрессивными и пытаются отобрать остатки еды у здоровых стервятников, сбиваясь вокруг них разными способами. С другой стороны, здоровые стервятники деградируют и не могут противостоять другим стервятникам. Ниже показаны стратегии для второго подэтапа эксплуатации:

$$x^{г+1}_я = \begin{cases} \frac{г}{2}, & \text{если randP3} \quad P3, \\ p г - |d^г_т| \times Ф^г_я \times \text{Левид}, & \text{если randP3} > P3, \end{cases} \quad \text{если } |F^г_я| < 0,5, \quad (13)$$

$$г_1 = \text{Лучший гриф}_1 \frac{\text{БестГриф 1} \times X^г}{\text{ЛучшийГриф}_1 (X^г_и)^2} \times \text{Фи}^г_я, \quad (14)$$

$$г_2 = \text{Второй лучший гриф}_2 \frac{\text{ВторойЛучшийГриф 2} \times X^г_я}{\text{SecondBestVultureg 2-}(X^г_я)^2} \times \text{Фи}^г_я, \quad (15)$$

$$\text{Левид} = 0{,}01 \times \frac{\mu \times \sigma}{|\nu|^{\frac{1}{\beta}}}, \quad (16)$$

$$\text{сигма} = \frac{\Gamma(1+\beta) \times \sin(\frac{\pi\beta}{2})}{\Gamma(1+\beta 2) \times \beta \times 2^{(\frac{\beta-1}{2})}}^{\frac{1}{\beta}}. \quad (17)$$

где следующее обновленное положение стервятника при следующем (g + 1) $g+1$ A g и A g скопление $й$ итерация, отражающая стервятников обозначено $X^г_я$, оцениваются с использованием соответственно 1 2 уравнений (14) и (15). Чтобы обеспечить эффективность алгоритма AVO, Levyd представляет распределения сборов, полученную с использованием уравнения (16). d – размерность пространства; σ определяется уравнением (17), где β = 1,5 — постоянная величина; а μ и ν — случайные числа, равномерно распределенные в диапазоне [0, 1].

3.4. Преобразование в двоичную природу

В представленном алгоритме AVO позиции показаны как реальные значения. Как таковые, они не могут быть немедленно применимы к проблеме бинарных опционов GS. Следовательно, существует необходимость преобразовать эти реальные значения позиции в двоичные значения, чтобы соответствовать двоичной природе GS и сохранить структуру исходного алгоритма. В этом преобразовании 1 представляют реальные значения соответствующих выбранных генов в векторе бинаризации, тогда как 0 представляют реальные значения невыбранных генов, которые не имеют отношения к делу. На каждой итерации g можно использовать следующее математическое выражение для преобразования $X^г_я$ реальной позиции X в двоичную позицию ($X^г_{я\ \text{двоичная}}$)

Machine Translated by Google

$$(X^{г}_{я})_{бин} = \begin{cases} 1, \text{ если } X_g > \delta, \\ я \ 0 \text{ иначе.} \end{cases} \quad (18)$$

где произвольная пороговая точка в пределах [0, 1] обозначается δ. Согласно этому фундаментальному подходу двоичного преобразования, двоичная «1» (выбранный признак) заменяет свое реальное значение, если $(X^{г}_{я})_{bin}$ меньше δ, в противном случае она заменяется на двоичный «0» (признак, который не был выбран). .

3.5. Оценка значения функции фитнеса

Нахождение наименьшего количества выбранных признаков и оптимизация точности классификации доступных классификаторов (моделей k-NN и SVM) — две противоречивые цели , которые следует сбалансировать для достижения наилучшего решения и определения его качества. Поскольку точность классификаторов k-NN и SVM может быть снижена, если количество выбранных признаков меньше оптимального, функция приспособленности уравновешивает размер и точность выбранных признаков. Функция пригодности концентрируется на снижении частоты ошибок классификации, а не точности, следующим образом:

$$f\,it = w1 \times Errrate + w2 \times \left|\frac{|\,f\,съел\,|}{D|\ w1}\right., \quad [0, 1], w2 = 1 \quad w1. \quad (19)$$

где f eatpicked обозначает количество выбранных объектов, D обозначает общее количество объектов в наборе данных, а Errrate представляет собой частоту ошибок классификации по классификаторам k-NN и SVM. Важность точности классификации и количество выбранных признаков обозначаются весовыми параметрами w1 и w2 соответственно. Значения w1 (0,99) и w2 (0,01) определяются посредством обширных испытаний, проведенных в предыдущих исследованиях [21,42,43].

3.6. Использование метода DE DE

[44] характеризуется высокой эффективностью и простотой поиска подходящего решения сложных задач оптимизации. Это может быстро принести ценные результаты. Для DE необходимы три основных процесса — мутация, кроссинговер и отбор. Процесс дифференциальной мутации стремится создать модифицированный вектор ui для вектора решения каждой итерации . Математически это можно вычислить следующим образом:

$$^{ы}_{я} = \quad X_{p1} + WM(\quad X_{\ p2} \quad X_{p3}) \quad (20)$$

В диапазоне [1, размер популяции] три случайно выбранных асимметричных вектора обозначаются символами Xr1 , Xr2 и Xr3 . WM обозначает весовой коэффициент мутации в интервале [0, 1].

После операции мутации DE выполняет операцию скрещивания, чтобы увеличить разнообразие популяции. Вектор-потомок ui создается путем объединения значений целевого вектора Xi и измененного вектора ui . Наиболее часто используемым и основным фактором поиска кроссовера является двоичное кроссовер, которое имеет следующее математическое выражение:

$$ui,d = \begin{cases} ui, d, & \text{если rand} \quad CR \text{ или } d = jrand, \text{ в} \\ Си, г, & \text{противном случае.} \end{cases} \quad (21)$$

где равномерно распределенное случайное число jrand [1, 2, ..., DX] используется для обеспечения того, чтобы мутировавший вектор имел хотя бы одно измерение. Возможность пересечения каждого элемента рассчитывается с использованием коэффициента пересечения CR, который часто определяется как большая величина (CR = 0,9).

Затем выполняется операция выбора, как показано в уравнении (22). Здесь выполняется сравнение функции пригодности f(Xi) целевого вектора и соответствующего вектора-потомка f(ui) , и функция пригодности с наименьшим значением сохраняется для предстоящей итерации.

Machine Translated by Google

$$Си = \begin{cases} уи, & \text{если } f(ui) < f(Xi) \\ Си, & \text{в противном случае.} \end{cases} \quad (22)$$

3.7. Полный алгоритм RBAVO-DE

Для реализации стратегии GS шаги рекомендуемого алгоритма RBAVO-DE описаны в следующих подразделах. Псевдокод предлагаемого алгоритма RBAVO-DE представлен в алгоритме 1. Блок-схема предлагаемого алгоритма RBAVO-DE показана на рисунке 1, иллюстрирующая его основные этапы.

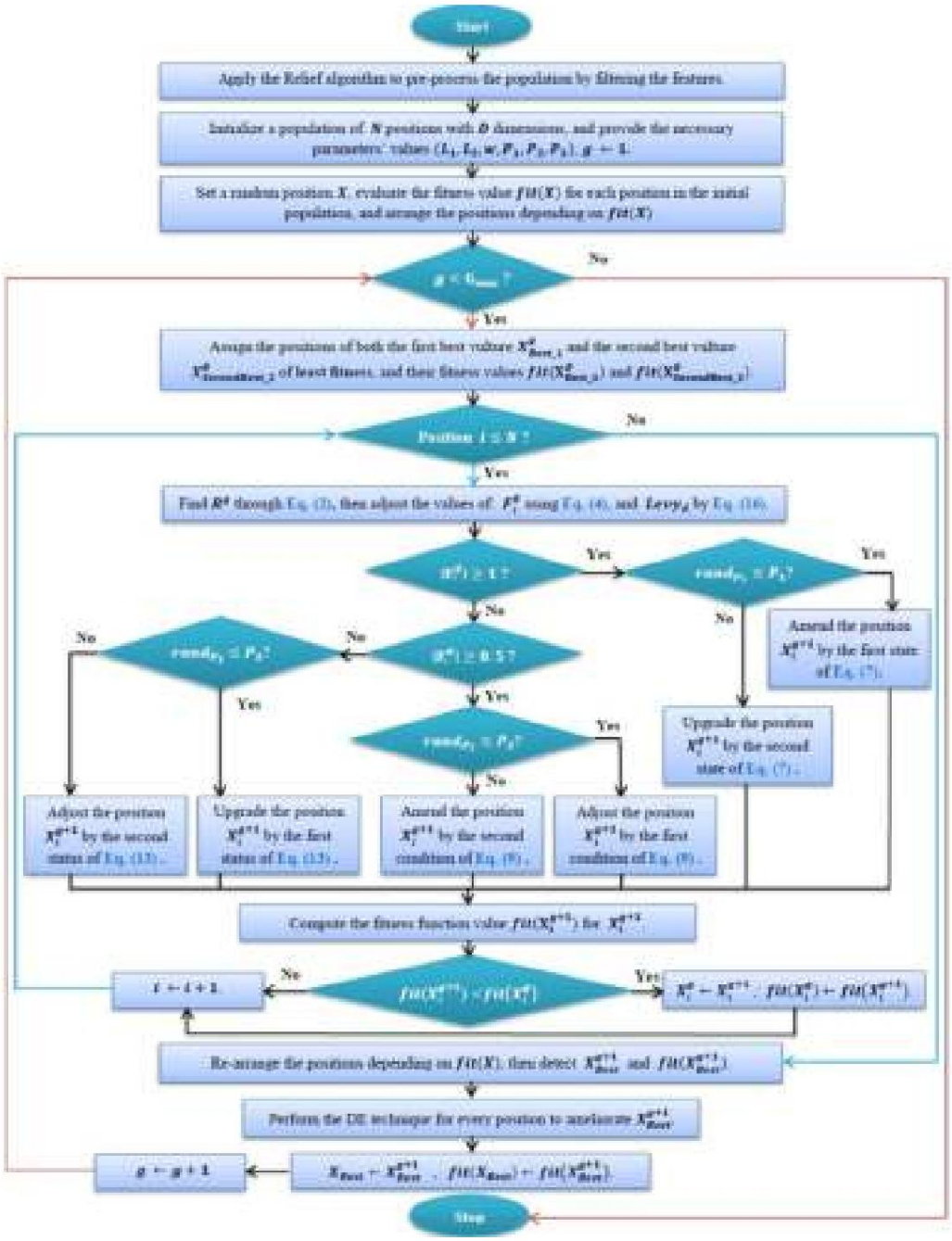


Рисунок 1. Блок-схема алгоритма RBAVO-DE.

Machine Translated by Google

Алгоритм 1. Предлагаемый алгоритм RBAVO-DE.

Вход:
N — общее количество должностей (численность населения)
Gmax — максимальное количество разрешенных итераций.
D — размерное пространство задачи.
LB – нижние границы переменных
UB – верхние границы переменных
CR — скорость пересечения
WM — весовой коэффициент мутации
Выход:
XBest — лучшая в мире позиция стервятника, найденная при поиске.
f it(XBest) — найденное глобальное лучшее значение фитнес-функции, которое следует уменьшить.
1: Старт
2. Примените подход Relief для выбора связанных объектов и их фильтрации, как показано в разделе 3.1;
3. Инициализируйте совокупность из N позиций и укажите значения необходимых параметров (L1, L2, w, P1, P2 и P3);
4: Установите случайную позицию X в исходной популяции;
5: Оценить значения пригодности f it(X) для каждой позиции в исходной популяции;
6: Расположите позиции в порядке возрастания в зависимости от их функции приспособленности f it(X);
7: г 1; 8: пока g Текущее количество итераций
< Gmax делать
9: Назначьте позиции первого лучшего стервятника X $_{\text{Лучший1}}^{г}$ и второй лучший стервятник X $_{\text{ВторойЛучший2}}^{г}$, а также их фитнес
значения f it(X $_{\text{Лучший1}}^{г}$) и f it(X $_{\text{ВторойЛучший2}}^{г}$), среди всех позиций населения;
для позиции стервятника i = 1: N do
Найдите лучшего грифа R g, используя уравнение (2);
Отрегулируйте уровень голода стервятника F. $_{я}^{г}$ используя уравнение (4);
Измените функцию распределения сборов Levyd, используя уравнение (16);
если |F $_{я}^{г}$ | 1, тогда
10:11:12:13:14:15: если randP1 P1 , то
Изменить положение стервятника X g+1 $_{я}$ на основе первого этапа фазы разведки с использованием уравнения (7);
16:17: иначе, если randP1 > P1 , то
Улучшите позицию стервятника X g+1. $_{я}$ на основе второго этапа фазы разведки с использованием уравнения (7);
конец если
иначе, если |F $_{я}^{г}$ | < 1 тогда
если |F $_{я}^{г}$ | 0,5, тогда
18:19:20:21:22: если randP2 P2 , то
23: Отрегулируйте положение стервятника X g+1 $_{я}$ на основе первого условия первого подэтапа эксплуатации с использованием
Уравнение (9);
24: иначе, если randP2 > P2 , то
25: с Изменить позицию стервятника X $_{я}^{г+1}$ на основе второго условия первого подэтапа эксплуатации
использованием уравнения (9);
26: конец если
иначе, если |F $_{я}^{г}$ | < 0,5 тогда
27:28: если randP3 P3 , то
29: Улучшите позицию стервятника X g+1. $_{я}$ на основе первого статуса второго подэтапа эксплуатации с использованием
Уравнение (13);
30: иначе, если randP3 > P3 , тогда
31: Отрегулируйте положение стервятника X g+1 $_{я}$ на основе второго статуса второго подэтапа эксплуатации с использованием
Уравнение (13);
конец если
конец если
конец если
32:33:34:35: если это(X $_{я}^{г+1}$) Вычислить значение фитнес-функции для X $_{я}^{г+1}$;
36: если f это(X $_{я}^{г+1}$) < f it(X $_{я}^{г}$) затем
37: X $_{я}^{г}$ X $_{я}^{г+1}$;
если это(X $_{я}^{г}$) f it(X $_{я}^{г+1}$);
конец если
конец для
38:39:40:41: Переставьте позиции в порядке возрастания в зависимости от их функции приспособленности f it(X);
Определить глобальную лучшую позицию X $_{\text{лучший}}^{г+1}$ и его лучшее глобальное значение пригодности f it(X $_{\text{Лучший}}^{г+1}$) (лучше всего) когда текущая итерация g + 1 закончилась;
42:43: Выполните технику DE для каждой позиции, чтобы улучшить X. $_{\text{Лучший}}^{г+1}$, как показано в разделе 3.6;
44: XBest X г+1 $_{\text{Лучший}}$;
45: f it(XBest) f it(X $_{\text{Лучший}}^{г+1}$);
46: г г + 1;
47: закончить пока
48: Конец

4. Экспериментальные результаты и обсуждение.

В этом разделе представлены эмпирические результаты предлагаемого RBAVO-DE и его аналоги: Бинарная искусственная пчелиная семья (BABC) [45], Алгоритм бинарного роя солей. (BSSA) [46], Бинарный PSO (BPSO) [47], Бинарный алгоритм летучей мыши (BBA) [48], Бинарный алгоритм Серого Волка Оптимизация (BGWO) [49], Алгоритм оптимизации двоичного кузнечика (BGOA) [50], Алгоритм оптимизации двоичного кита (BWOA) [51], двоичный ASO (BASO) [52], двоичная птица Роевой алгоритм (BBSA) [53], бинарный HGSO (BHGSO) [54] и бинарный алгоритм Харриса Хокса.

Machine Translated by Google

тимизация (BHHO) [55]. Оптимизаторы проходят оценку посредством обучения и тестирования.
контрольные показатели, причем убедительные результаты получены на основе средних значений показателей оценки .
Тесты, используемые для оценки эффективности предлагаемой модели:
подробно в разделе 4.1. Изложены параметры, используемые в операционной среде.
в разделе 4.2. Метрики, используемые для оценки, описаны в разделе 4.3. Анализ
Результаты эксперимента обсуждаются в разделе 4.4.

4.1. Описание набора данных

Для этого были применены глубокие экспериментальные подходы и различные алгоритмы-оболочки.
двадцать два набора данных описаний генов; данные включают нормализованную секвенацию РНК уровня 3.
данные об экспрессии генов для двадцати двух видов опухолей из Института Броуда. Эти данные
являются общедоступными и их можно найти в [56]. Мы придерживались описанной методологии
в [36] и наблюдали расхождения между данными, указанными на GitHub в статье.
и цифры, приведенные в документе, взяты с веб-сайта.
данные веб-сайта включали смесь опухолей и стандартных образцов, тогда как в статье рассматривались
данные единообразно, как образцы опухолей. В связи с этим мы провели детальное обследование.
данных. Изначально на сайте предлагались различные форматы одного и того же набора данных, который мы планировали.
анализировать. Изучив данные, мы столкнулись со следующими проблемами:

- Некоторые гены идентифицируются по идентификатору, но не имеют соответствующего символа.
- Некоторые гены отсутствуют в файле аннотаций.
- Образцы смешаны, включая как нормальные, так и опухолевые типы.

Следовательно, предварительная обработка была необходима для разделения и идентификации образцов,
различия между нормальными и опухолевыми образцами для использования в бинарной классификации и оптимизации.
процесс ГС. Мы решили вышеперечисленные проблемы следующим образом:

- Мы искали соответствующий символ гена для каждого идентификатора, найденного в файле аннотаций.
- После перекрестных ссылок с файлом аннотаций было исключено более ста генов .
- На основе образца отчета каждая строка листа Excel была организована в соответствии с
своего рода образец для бинарной классификации.

Более того, подход Relief, как подробно описано в разделе 3.1, использовался на этапе предварительной
обработки для расчета веса каждого гена в эталоне. Эти веса
впоследствии были упорядочены от высшего к низшему. Затем были выделены гены с меньшим весом.
удаленный. Подход Relief оказался полезен для исключения генов, которые не вносят вклада.
к классификации.

После этапа предварительной обработки данные были уточнены и готовы для процесса GS.
В отличие от подхода, использованного в [36], в котором рассматривалась мультиклассификация всех типов
рака вместе, мы решили проанализировать каждый тип рака индивидуально для большей специфичности.
В таблице 1 [57] представлен комплексный обзор всех двадцати двух типов опухолей и их
соответствующие количества образцов. Количество функций, используемых в эталонных наборах данных, равно
32 000 признаков (генов).

Таблица 1. Наборы данных, использованные в этой статье [57].

# Рак	Нормальный Рекорды	Опухоль Рекорды
1. Уротелиальная карцинома мочевого пузыря (BLCA)	19	408
2. Рак щитовидной железы (THCA)	59	501
3 Рак шейки матки и эндоцервикальный рак (CESC)	3	304
4 Холангиокарцинома (ХОЛ)	9	36
5 Аденокарцинома толстой кишки (COAD)	41	458
6. Рак пищевода (ESCA)	13	184
7. Мультиформная глиобластома (ГБМ)	5	153

Machine Translated by Google

Таблица 1. Продолжение.

# Рак	Нормальный Рекорды	Опухоль Рекорды
8 Тимома (ТИМ)	2	120
9. Плоскоклеточный рак головы и шеи (ХНСК)	44	520
10. Почечный хромофоб (KICH).	25	66
11 КИРЦ	72	533
12 Почечный папиллярный узел (КИРП)	32	290
13 Гепатоцеллюлярная карцинома печени (LIHC)	50	371
14 ЛУАД	59	515
15 ЛУСК	51	501
16 Аденокарцинома поджелудочной железы (ПААД)	4	178
17 УЦЭК	35	176
18 Феохромоцитома и Параганглиома (ПКПГ)	3	179
19 Аденокарцинома прямой кишки (ЧИТАЙТЕ)	10	94
20 Саркома (САРК)	2	259
21. Кожная меланома (SKCM)	1	103
22 СТАД	37	415

4.2. Настройка параметров

Предложенный алгоритм RBAVO-DE сравнивался с бинарными вариантами различных метаэвристические оптимизаторы, такие как BABC, BSSA, BPSO, BBA, BGWO, BWOA, BBSA, BGOA, БХХО, БАСО и БХГСО. Критические параметры моделей МО, используемых в этом исследования подробно описаны в таблице 2.

Таблица 2. Ключевые параметры используемых моделей ML.

Модель	Параметры
СВМ	Полиномиальное ядро = 2
к-НН	Метрика евклидового расстояния k = 5 [21,58,59]

В рамках предложенной структуры достоверность результатов была проверена с использованием 10-кратного перекрестный подход для обеспечения надежности результатов. Это произошло случайно разделение каждого набора данных на два отдельных подмножества: 80% для обучения и оставшиеся 20% для тестирования. Обучающая часть использовалась для обучения представленных классификаторов с использованием методы оптимизации, в то время как часть тестирования использовалась для оценки выбранных генов. эффективность. Параметры, наиболее часто применяемые во всех сравниваемых алгоритмах: обобщены в таблице 3. Для выполнения всех экспериментов в этой статье использовался Python вычислительная среда с двойным процессором Intel® Xeon® Gold 5115 2,4 ГГц и 128 ГБ памяти оперативной памяти в операционной системе Microsoft Windows Server 2019.

Таблица 3. Конфигурации параметров всех используемых оптимизаторов.

Оптимизатор	Параметры
Все оптимизаторы	Количество выполнений = 30
	Максимальное количество итераций Gmax = 100
	Численность населения N = 10

Machine Translated by Google

Таблица 3. Продолжение.

Оптимизатор	Параметры
Предлагаемый РБАВО-ДЕ	Размерность D = количество генов в используемых наборах данных. Верхние границы UB Нижние границы LB Параметр L1 = 0,7 Параметр L2 = 0,2 Параметр w = 2
БССА	Параметр P1 = 0,6 Параметр P2 = 0,6 Параметр P3 = 0,5 Число сборщиков SD = 0,1 × N Число производителей PD = 0,2 × N Порог безопасности ST = 0,8 Число
БЭБК	работающих пчел = 16 Число пчел-разведчиков = 3 Число пчел-наблюдателей = 4 Инерционный
БПСО	вес (ωmax = 0,9ωmin = 0,4) Коэффициенты ускорения (c2 = c1 = 1,2)
_____	Громкость A = 0,8 Частота излучения импульсов r = 0,95 Нижняя и верхняя частоты импульсов = 0, 10 a
БВОА	линейно уменьшается от 2 до 0 p = 0,5 b = 1,0 Энергия
БХХО	кролика E [1, 1] a линейно
БГВО	уменьшается от 2 до 0 Cmin = 0,00004 и
БГОА	Cmax = 1 Частота полета ff = 10
ББСА	Влияние на бдительность птиц Коэффициент слежения fl = 0,5 Вероятность поиска пищи p = 0,8 Коэффициенты ускорения (c1 = c2 = 1,5) Поведение (a1 = a2 = 1,0)
БАСО	Вес глубины α = 50 Вес
БХГСО	множителя β = 0,2 Количество кластеров = 2 α = β = 0,1 K = 1,0 и l3 = 1E 02 l1 = 5E 03, l2 = 1E + 02

4.3. Критерии оценки

Чтобы оценить эффективность предлагаемого RBAVO-DE по сравнению с другими методами, каждая стратегия была протестирована независимо тридцать раз по каждому критерию, чтобы обеспечить статистическую проверку результатов. Для этой цели использовались следующие установленные показатели эффективности по вопросам GS:

- Средняя точность классификации (MeanAC): эта мера представляет точность правильной классификации данных, рассчитанную путем независимого запуска алгоритма тридцать раз. Это определяется следующим образом:

$$\text{Среднее AC} = \sum_{k=1}^{30} \sum_{r=1}^{m} \text{совпадение}(PLr, ALr) \quad (23)$$

В этой формуле m обозначает количество выборок в наборе тестовых данных, а PLr и ALr обозначают предсказанные метки классификатора и фактические метки классов для выборки r соответственно. Функция match(PLr , ALr) служит механизмом сравнения: если PLr равен ALr , то match(PLr , ALr) присваивается значение 1; если они не совпадают, ему присваивается значение 0.

Machine Translated by Google

• Средняя приспособленность (MeanFi): эта мера вычисляет среднее значение meanFi, полученное путем независимого запуска предложенного подхода тридцать раз. Он измеряет баланс между снижением частоты ошибок классификации и минимизацией количества выбранных генов. Более низкое значение указывает на лучшее решение, оцененное на основе пригодности:

$$\text{Среднее Fi} = 30 \frac{1}{k=1} \quad ^{30} \quad ж^{к} \tag{24}$$

где f ^{K} показывает оптимальное значение приспособленности, достигнутое в k -ом выполнении .

• Среднее количество выбранных генов (MeanGe): этот показатель рассчитывает среднее количество выбранных генов (или соотношение GS) путем независимого запуска предложенного подхода тридцать раз и определяется следующим образом:

$$\text{Среднее Fe} = 30 \frac{1}{k=1} \quad ^{30} \quad \frac{|д^{к}|}{|Д|} \tag{25}$$

го ^{K} | обозначает общее количество генов, выбранных в оптимальном решении для k- решения. Здесь |d ecution и |D| представляет общее количество генов, присутствующих в исходном эталоне. • Стандартное отклонение (STDE). На основании приведенных выше результатов общие средние результаты, полученные в результате тридцати отдельных запусков каждого оптимизатора в каждом тесте, оценивались на стабильность следующим образом:

$$\text{СТАНДАРТ} = \sqrt{\frac{1}{29} \sum_{k=1}^{30} (Yk \quad \text{Среднее})^{2}} \tag{26}$$

где Y показывает меру, которая будет использоваться, Y k обозначает значение меры Y в k-м прогоне, а MeanY — среднее значение меры из тридцати независимых выполнений.

Данные в последующих таблицах представляют собой средние значения , полученные в результате тридцати независимых экспериментов с упором на точность классификации, количество выбранных генов, среднюю приспособленность, точность, отзыв и показатель F1. В последующих подразделах тщательно рассматриваются и обсуждаются эти экспериментальные результаты, жирным шрифтом выделены оптимальные результ

4.4. Результаты сравнения предложенного RBAVO-DE с различными метаэвристическими алгоритмами

Предложенный подход RBAVO-DE с моделями SVM и k-NN сравнивался с другими недавними метаэвристическими методами, выполняемыми в тех же условиях, чтобы показать его превосходство над аналогами. Предложенный алгоритм RBAVO-DE сравнивался с бинарными вариантами нескольких оптимизаторов, включая BABC, BSSA, BPSO, BBA, BGWO, BWOA, BBSA, BGOA, BHHO, BASO и BHGSO.

4.4.1. Результаты с использованием модели k-NN

В таблице 4 показаны результаты предложенного алгоритма RBAVO-DE и других методов оптимизации, использующих модель k-NN, на основе показателей точности, сравниваемых в аналогичных условиях. Результаты эксперимента показывают, что предложенный RBAVO-DE достиг наиболее многообещающих результатов в четырех тестах. Примечательно, что все конкурентные методы, включая предложенный RBAVO-DE, использующий модель k-NN, дали сопоставимые результаты по восемнадцати критериям.

Остальные метрики модели k-NN приведены в Приложении A.1. По средним значениям приспособленности предложенный алгоритм RBAVO-DE продемонстрировал более высокую эффективность по сравнению с аналогами, использующими модель k-NN в аналогичных условиях.

Алгоритм RBAVO-DE показал самые низкие результаты пригодности и самый конкурентоспособный STDE по всем тестам. Кроме того, все используемые тесты являются многомерными, что демонстрирует, что предлагаемый RBAVO-DE может эффективно работать на всех тестах независимо от размера. Алгоритм RBAVO-DE перспективен, поскольку эффективно балансирует использование и исследование пространства поиска, избегая попадания в ловушку локальных оптимумов во время итераций. В отличие от

Machine Translated by Google

многие другие алгоритмы, которые имеют тенденцию попадать в ловушку, продемонстрировали способность выбраться такие ловушки.

Что касается средних результатов для генов, выбранных с помощью предложенного алгоритма RBAVO-DE и его аналогов, использующих k-NN, предложенный алгоритм RBAVO-DE превзошел другие методы по всем критериям, касающимся количества выбранных генов. Более того, способность RBAVO-DE идентифицировать важные гены обусловлена его эффективным
исследование возможных областей с одновременным повышением точности.

С точки зрения средней точности предложенного алгоритма RBAVO-DE и его аналогов с k-NN предложенный RBAVO-DE превзошел альтернативные подходы для
три из двадцати двух наборов данных. Для девятнадцати наборов данных BWOA добилась аналогичных результатов: в то время как BABC, BPSO, BGWO, BGOA и BHHO получили результаты, аналогичные результатам
предложил РБАВО-ДЕ. BASO и BHGSO, которые дали те же результаты, что и
предложил RBAVO-DE по семнадцати наборам данных, заняв четвертое место. В конечном итоге BBA занял самое низкое место среди всех подходов, давая результаты, аналогичные предложенному RBAVO-DE для
пятнадцать наборов данных.

Что касается среднего отзыва предложенного RBAVO-DE и его аналогов, использующих k -NN, RBAVO-DE превзошел альтернативные подходы для трех из двадцати двух наборов данных. С другой стороны, BSSA, BABC, BPSO, BGWO, BWOA, BGOA и BBSA.
дал те же результаты, что и RBAVO-DE, для 19 наборов данных, в то время как BHHO добился аналогичных результатов для 18 наборов данных. BHGSO, который получил те же результаты, что и предложенный RBAVO-DE по семнадцати наборам данных занял четвертое место. В конечном итоге BASO и BBA заняли места самый низкий среди всех подходов, давая результаты, аналогичные RBAVO-DE для шестнадцати наборов данных.

Что касается среднего показателя F1 предлагаемого RBAVO-DE и его аналогов.
с использованием k-NN предложенный RBAVO-DE превзошел альтернативные подходы для три из двадцати двух наборов данных. Напротив, для восемнадцати наборов данных BSSA, BABC, BPSO, BGWO, BGOA и BBSA дали те же результаты, что и RBAVO-DE, тогда как BWOA и BBSA дали те же результаты, что и RBAVO-DE. BHHO предоставила сопоставимые результаты для семнадцати наборов данных. БАСО и БХГСО, которые получили те же результаты для 16 наборов данных, что и предложенный RBAVO-DE, занявший четвертое место. В конечном итоге BBA занял самое низкое место среди всех подходов, давая результаты, аналогичные предложил RBAVO-DE для четырнадцати наборов данных.

Таблица 4. Результаты предложенного алгоритма RBAVO-DE и его аналогов с использованием k-NN относительно результаты средней точности классификации.

Набор данных	Метрический	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	Среднее	0,998	0,993	0,998	0,993	0,989	0,995	0,994	0,993	0,991	0,994	0,994	0,989
	Среднее	0,984	0,006	0,005	0,006	0,003	0,006	0,006	0,006	0,005	0,006	0,006	0,002
CESC	0,000	CHOL	0,984	0,984	0,984	0,984	0,984	0,984	0,984	0,984	0,984	0,984	0,984
	Среднее	1,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
COAD	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	Среднее	0,974	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
0,000 GBM	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
HNSC	Среднее	0,993	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
STDE 0,004	Среднее	1,000	0,974	0,974	0,974	0,974	0,974	0,974	0,974	0,974	0,974	0,974	0,974
	Среднее	0,998	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
0,003	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	Среднее	1,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
0,000 LUAD	Среднее	0,991	0,991	0,991	0,991	0,992	0,991	0,991	0,991	0,991	0,991	0,991	0,991
	Среднее	1,000	0,002	0,002	0,000	0,000	0,002	0,000	0,000	0,000	0,002	0,000	0,000
PAAD	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
			0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
КИРЦ			0,993	0,994	0,992	0,992	0,994	0,992	0,994	0,993	0,993	0,993	0,992
			0,003	0,004	0,002	0,000	0,004	0,002	0,004	0,003	0,003	0,003	0,000
КИРП			1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
			0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
ЛИГК			1. 000	1. 000	1. 000	1. 000	1. 000	1. 000	1. 000	1. 000	1. 000	1. 000	1.000
			0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
			0,991	0,991	0,991	0,991	0,991	0,991	0,991	0,991	0,991	0,991	0,991
	0,000		0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
			1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	0,000		0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
			1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	0,000		0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000

Machine Translated by Google

Таблица 4. Продолжение.

Набор данных	Метрический	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
PCPG	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
ЧТЕНИЕ	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
SARC	Среднее	0,981	0,981	0,981	0,981	0,981	0,981	0,981	0,981	0,981	0,981	0,981	0,981
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
SKCM	Среднее	1,000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
СТАД	Среднее	0,999	0,996	0,999	0,996	0,989	0,996	0,995	0,997	0,992	0,995	0,996	0,989
	STDE	0,000	0,005	0,004	0,005	0,000	0,005	0,006	0,005	0,005	0,006	0,005	0,000
THCA	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
THYM	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
UCEC	Среднее	0,998	0,985	0,988	0,977	0,969	0,984	0,977	0,978	0,978	0,979	0,976	0,968
	STDE	0,008	0,012	0,013	0,006	0,011	0,012	0,006	0,008	0,009	0,009	0,005	0,011
Рейтинг W|T|L		4|18|0	0|18|4	0|18|4	0|18|4	0|18|4	Значения, выделенные		0|18|4	0|18|4	0|18|4	0|18|4	0|18|4	0|18|4	0|18|4

жирным шрифтом, обозначают лучшие результаты.

4.4.2. Результаты с использованием модели SVM

В таблице 5 показаны результаты предложенного RBAVO-DE и различных оптимизаций. методы, использующие модель SVM в отношении оцениваемых результатов точности классификации в одинаковых условиях эксплуатации. Результаты экспериментов показывают, что предложенный алгоритм RBAVO-DE превзошел другие подходы, получив наибольшее многообещающие значения для четырех наборов данных. Все конкурентные методы, включая предлагаемые RBAVO-DE, используя SVM, получил эквивалентные значения по 18 тестам.

Остальные метрики, использующие SVM, представлены в Приложении A.2. Что касается средние значения пригодности предлагаемого RBAVO-DE и его аналогов, использующих SVM модели в эквивалентных условиях эксплуатации предложенный RBAVO-DE оказался более эффективен, чем другие методы. Предлагаемый RBAVO-DE, использующий модель SVM. показали самые низкие показатели пригодности и самый конкурентоспособный STDE по всем критериям. Кроме того, все используемые тесты являются многомерными, демонстрируя, что предлагаемые RBAVO-DE может эффективно работать во всех тестах независимо от размера. РБАВО-ДЕ Алгоритм показывает многообещающие результаты, эффективно балансируя использование и исследование поиска. пространстве, избегая попадания в ловушку локальных оптимумов во время итераций. В отличие от многих других алгоритмов, которые имеют тенденцию попадать в ловушку, он продемонстрировал способность избегать таких ловушек.

Что касается средних результатов для генов, выбранных с помощью предложенного RBAVO-DE и его По сравнению с аналогами, использующими SVM, предложенный RBAVO-DE показал более многообещающие результаты. чем другие методы во всех тестах, использованных в этом исследовании. Также превосходство предлагаемого RBAVO-DE, использующего SVM, в этом контексте демонстрирует свои возможности эффективно исследовать ценные области поискового пространства, избегая при этом областей с нереализуемые решения.

Что касается средних значений точности предлагаемого РБАВО-ДЕ и его аналогов, использующих SVM, предложенный RBAVO-DE превзошел альтернативу подходы для трех из двадцати двух наборов данных. С другой стороны, для восемнадцати наборов данных BABC, BGWO, BWOA, BBSA и BASO дали сопоставимые результаты. Для семнадцати наборов данных BSSA, BPSO, BGOA и BHHO дали результаты, аналогичные результатам RBAVO-DE, тем самым занимает четвертое место. В конечном итоге BBA занял самое низкое место среди всех подходов, обеспечив аналогичные результаты. результаты совпадают с результатами предлагаемого RBAVO-DE для двенадцати наборов данных.

С точки зрения среднего отзыва предложенного метода RBAVO-DE и его аналогов с использованием SVM предложенный RBAVO-DE превзошел альтернативные подходы для трех из двадцать два набора данных. Напротив, BPSO и BHHO показали аналогичные результаты для восемнадцати наборы данных, в то время как BSSA, BABC, BGWO, BWOA, BGOA и BBSA дали одинаковые результаты как предлагаемый RBAVO-DE для девятнадцати наборов данных. BHGSO, который получил то же самое результаты как RBAVO-DE для семнадцати наборов данных, заняли четвертое место. В конце концов, BASO и BBA занял самое низкое место среди всех подходов, давая результаты, аналогичные результатам RBAVO-DE. для шестнадцати наборов данных.

Machine Translated by Google

Что касается среднего показателя F1 предложенного алгоритма RBAVO-DE и его аналогов, использующих SVM, предложенный RBAVO-DE превзошел альтернативные подходы. для пяти из двадцати двух наборов данных. С другой стороны, BWOA дала аналогичные результаты. с предлагаемыми RBAVO-DE для семнадцати наборов данных, тогда как BSSA, BABC, BPSO, BGWO, BGOA и BBSA дали сопоставимые результаты для шестнадцати наборов данных. БХХО, который получил те же результаты, что и предложенный RBAVO-DE для пятнадцати наборов данных, заняв четвертое место. В конечном итоге BBA занял самое низкое место среди всех подходов, давая результаты, аналогичные тем, RBAVO-DE для 12 наборов данных.

Таблица 5. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих SVM относительно средних значений точности классификации.

Набор данных	Метрический	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	Среднее	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
CESC	Среднее	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
CHOL	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
COAD	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
GBM	Среднее	0,986	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,013	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
HNSC	Среднее	0,975	0,977	0,974	0,974	0,977	0,974	0,975	0,975	0,975	0,974	0,974	
	STDE	0,004	0,008	0,000	0,000	0,008	0,000	0,005	0,005	0,005	0,000	0,000	
KICH	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
KIRC	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
LUAD	Среднее	0,993	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,003	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
PAAD	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
PCPG	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
ЧТЕНИЕ	Среднее	1,000	1,000	1,000	0,9992	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,0029	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
		0,991	0,991	0,991	0,991	0,991	0,991	0,991	0,991	0,991	0,992	0,991	
		0. 000	0. 000	0. 000	0,000	0. 000	0. 000	0. 000	0. 000	0. 000	0. 002	0,000	
ЛИГК		1.000	1.000	1.000	1,000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	
		0.000	0.000	0.000	0,000	0.000	0.000	0.000	0.000	0.000	0,000	0,000	
ЛЮСК		1.000	1.000	1.000	1,000	1.000	1.000	1.000	1.000	1.000	1,000	1.000	
		0.000	0.000	0.000	0,000	0.000	0.000	0.000	0.000	0.000	0,000	0,000	
		1.000	1.000	1.000	1,000	1.000	1.000	1.000	1.000	1.000	1,000	1.000	
		0.000	0.000	0.000	0,000	0.000	0.000	0.000	0.000	0.000	0,000	0,000	
		1.000	1.000	1.000	1,000	1.000	1.000	1.000	1.000	1.000	1,000	1.000	
		0,000	0.000	0.000	0.000	0,000	0.000	0.000	0.000	0.000	0,000	0,000	0,000
		1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	
		0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	
SARC	Среднее	0,994	0,986	0,989	0,986	0,981	0,989	0,987	0,986	0,984	0,986	0,984	0,981
	STDE	0,008	0,008	0,009	0,008	0,000	0,009	0,009	0,008	0,006	0,008	0,007	0,000
SKCM	Среднее	0,990	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0,003	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
THCA	Среднее	0,000	0,989	0,989	0,989	0,989	0,989	0,989	0,989	0,989	0,989	0,989	0,989
	STDE	0,000	0,000	0,000	0,002	0,000	0,000	0,000	0,002	0,000	0,000	0,000	
THYM	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
UCEC	Среднее	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1.000
	STDE	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
		1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	
		0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	
		1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	
		0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	
Рейтинг W|T|L		4|18|0	0|18|4	0|18|4	0|18|4	0|18|4	0|18|4	0|18|4	0|18|4	0|18|4	0|18|4	0|18|4	0|18|4

Цифры, выделенные жирным шрифтом, обозначают лучшие результаты.

4.5. Анализ конвергенции

В Приложении B [57] очевидно , что предложенный RBAVO-DE, использующий SVM и Модели k-NN достигли оптимальной сходимости во всех тестах. Поэтому, производительность сходимости предлагаемого RBAVO-DE с использованием SVM и k-NN модели демонстрирует свою способность быстро достигать оптимальных результатов, сохраняя при этом эффективное равновесие между поиском и эксплуатацией.

Machine Translated by Google

4.6. Критерий суммы рангов Уилкоксона

В этой статье значения пригодности, достигнутые предложенным RBAVO-DE и его аналогами, попарно сравниваются с использованием критерия суммы рангов Уилкоксона [60], целью которого является определить, существует ли статистически значимая разница между различными подходами. Этот тест является важным инструментом для оценки успешности предлагаемого алгоритма. Критерий Уилкоксона используется при проверке гипотез для сравнения совпадающих данных. Он включает в себя сортировку абсолютных различий в результатах парных процедур по j из N_j вопросов. Затем наименьшее значение находится путем суммирования положительного (R +) и отрицательного (R -) рейтингов. Нулевая гипотеза отклоняется, если результирующий уровень значимости составляет менее 5%, и не отвергается в противном случае.

Изучив результаты попарного сравнения RBAVO-DE и его аналогов с использованием критерия Уилкоксона, мы обнаружили, что все значения R + равны 253, все значения R равны 0 и все значения p равны $4,768 \times 10^{-7}$. которые меньше 0,05 (уровень значимости 5%). Следовательно, можно сделать вывод, что предлагаемый RBAVO-DE, использующий SVM и k-NN, во всех случаях работает лучше, чем любой другой алгоритм. Таким образом, все значения p менее 0,05 (уровень значимости 5%) убедительно свидетельствуют о том, что результаты предложенной стратегии статистически значимы, а не просто случайны.

4.7. Вычислительная сложность алгоритма RBAVO-DE и различных метаэвристических алгоритмов 4.7.1.

Вычислительная сложность времени выполнения алгоритма RBAVO-DE

Каждый из пяти основных этапов RBAVO-DE можно проанализировать отдельно, чтобы определить его вычислительную сложность. Эти шаги включают в себя функции фильтрации, инициализацию совокупности, повышение и изменение позиции, оценку функции приспособленности и использование DE. Затем $O_{execute_time}(RBAVO - DE)$ можно использовать для представления общей вычислительной сложности предлагаемого алгоритма RBAVO-DE. Это можно вычислить, используя следующие формулы обозначения big-O:

$$\begin{aligned} O_{execute_time}(RBAVO\ DE) = {} & O_{execute_time}(\text{функции фильтрации}) + \\ & O_{execute_time}(\text{инициализация совокупности}) \\ & + O_{execute_time}(\text{повышение и изменение позиции}) + \\ & O_{execute_time}(\text{оценка функции приспособленности}) \\ & + O_{execute_time}(\text{включение техники DE}). \end{aligned} \quad (27)$$

Поскольку N определяет размер популяции, Gmax — максимальное количество итераций. разрешено, а D — размерное пространство задачи. Следовательно,

$O_{execute_time}(\text{Функции фильтрации}) = O_{execute_time}(D).$
$O_{execute_time}(\text{инициализация заполнения}) = O_{execute_time}(N).$
$O_{execute_time}(\text{Усиление и изменение позиции}) = O_{execute_time}(Gmax \times N \times D).$
$O_{execute_time}(\text{Оценка функции пригодности}) = O_{execute_time}(Gmax \times N).$
$O_{execute_time}(\text{с использованием техники DE}) = O_{execute_time}(N \times D).$

Для этого

$$\begin{aligned} O_{execute_time}(RBAVO\ DE) = {} & O_{execute_time}(D) + \\ & O_{execute_time}(N) + O_{execute_time}(Gmax \times N \times D) + \\ & O_{execute_time}(Gmax \times N) + O_{execute_time}(N \times D) = \\ & O_{execute_time}(Gmax \times N \times D). \end{aligned}$$

Machine Translated by Google

4.7.2. Вычислительная сложность использования памяти алгоритма RBAVO-DE

Это включает в себя измерение объема памяти, необходимого алгоритму для решения проблемы по мере увеличения количества входных данных. Это часто выражается как дополнительная память, необходимая алгоритму помимо ввода. Это влечет за собой объединение следующих двух основных элементов: 1. Использование памяти

входными переменными: это объем памяти, необходимый алгоритму для хранения входных данных. Существует 13 входных переменных, связанных с предлагаемым алгоритмом RBAVO-DE, а именно: N, Gmax, D, LB, UB, CR, WM, L1, L2, w, P1, P2 и P3. Поскольку каждая переменная хранит числовые значения, каждая из них использует 4 байта памяти . Следовательно, общая сложность использования памяти этими 13 входными переменными составляет 52 байта (13 × 4 байта = 52 байта). Сложность использования памяти для входных значений постоянна.

2. Использование дополнительной памяти: показывает, сколько еще памяти требуется алгоритму помимо входных данных. Он включает в себя память, необходимую для структур данных, внутренних переменных и других компонентов алгоритма. Независимо от размера входных данных, алгоритм RBAVO-DE требует определенного объема дополнительной памяти. Следующее задействованы переменные:

• Сложность использования памяти, потребляемая Xinitial, составляет (4 × N × D) байт. Это связано с тем, что каждая позиция в векторе позиций Xinitial требует 4 байта памяти, а ее размер равен (N × D) и пропорционален исходной совокупности N позиций с размерностью D. Поскольку требуемый объем памяти возрастает линейно со значением (N × D), сложность использования памяти линейна.

• Переменные pr, g, Fi , t, R, Di , dt , S1, S2, A1, A2, sigma, f it(Xi), f it(XBest1), f it(XSecondBest2) и f it(XBest) требуют по 4 байта памяти каждый, поскольку они представляют только числовые значения. В результате общая сложность использования памяти этими 16 переменными составляет 64 байта (16 × 4 байта = 64 байта). Эта сложность использования памяти является постоянной.

• Векторы положения BestVulture1, SecondBestVulture2, Xi , Levyd , μ, σ, X бин , ui , Xr1 , Xr2 , Xr3 , ui и XBest требуют по 4 байта памяти каждый, а размер каждого вектора равен D, пропорционально размеру полученных позиций.

В результате векторы в этих 13 точках имеют общую сложность использования памяти (52 × D) байт (13 × 4 × D байт). Поскольку необходимый объем памяти растет линейно со значением D, сложность потребления памяти является линейной.

В результате общая сложность использования памяти для всех перечисленных выше дополнительных переменных составит (4 × N × D) + 64 + (52 × D) байт.

Наконец, следующую формулу можно использовать для определения вычислительной сложности общего использования памяти предлагаемым алгоритмом RBAVO-DE:

Сложность использования памяти (RBAVO-DE) = Использование памяти входных значений+

Использование дополнительной памяти = 52 + (4 × N × D) + 64 + (52 × D) байт.

Имейте в виду, что есть константные байты, которые не учитываются. В нотации big-O вычислительная сложность общего потребления памяти RBAVO-DE может быть представлена как Omemory_usage(RBAVO - DE). Затем это можно вычислить в нотации big-O после исключения всех констант следующим образом:

Omemory_usage(RBAVO DE) = Omemory_usage(использование входных значений памяти)

+ Omemory_usage(дополнительное использование

памяти) = Omemory_usage(1) + Omemory_usage(N × D) + Omemory_usage(1) + Omemory_usage(D) =

Omemory_usage(N × D) .

Machine Translated by Google

Может быть сложно провести тщательное сравнение использования памяти и сложности времени выполнения различных метаэвристических алгоритмов, поскольку эти сложности могут различаться в зависимости от конкретной реализации, размера задачи и других операторов. Кроме того , не все перечисленные алгоритмы имеют всестороннюю оценку времени выполнения и сложности использования памяти, и их свойства могут варьироваться в зависимости от решаемой задачи.

5. Преимущества и недостатки алгоритма RBAVO-DE.

В этой части представлено сбалансированное обсуждение преимуществ и недостатков RBAVO-Алгоритм ДЭ. Преимущества алгоритма RBAVO-DE можно перечислить следующим образом:

- Высокая точность классификации. Алгоритм RBAVO-DE продемонстрировал высокую точность классификации, в некоторых случаях достигающую 100%. Это примечательное достижение, особенно в отношении наборов данных о раке, поскольку точный выбор генов может напрямую влиять на диагностические и терапевтические результаты. •
- Эффективное уменьшение размера объекта: алгоритм продемонстрировал замечательную способность уменьшать размер объекта до 98% при сохранении или повышении точности классификации. Такое уменьшение размерности имеет решающее значение для эффективной обработки многомерных наборов
- данных. • Надежность в различных наборах данных. Эффективность алгоритма RBAVO-DE в двадцати двух наборах данных о раке указывает на его надежность и адаптируемость к различным характеристикам генетических данных. Эта универсальность полезна для более широкого применения в
- исследованиях в области геномики. • Превосходная производительность по сравнению с конкурентами: по сравнению с бинарными вариантами широко признанных метаэвристических алгоритмов RBAVO-DE выделяется своей выдающейся производительностью по точности и сокращению признаков, что подчеркивает его инновационный подход к отбору генов.

С другой стороны, недостатки предлагаемой методологии можно перечислить следующим образом: •

- Вычислительная сложность: высокая размерность наборов данных и итеративный характер метаэвристических алгоритмов позволяют предположить, что RBAVO-DE может иметь значительные вычислительные затраты. Этот аспект может ограничить его применимость в средах с ограниченными вычислительными ресурсами. • Проблемы обобщения: несмотря
- на успех алгоритма в различных наборах данных о раке, возможность обобщения алгоритма на другие типы биологических данных или заболеваний еще предстоит тщательно изучить. Крайне важно протестировать RBAVO-DE в более широком контексте, чтобы подтвердить его применимость за пределами исследованных наборов данных.
- Чувствительность параметров. Как и многие метаэвристические алгоритмы, RBAVO-DE может быть чувствителен к настройкам параметров, влияя на производительность и эффективность. Детальные исследования влияния различных конфигураций параметров и стратегий их оптимизации могут повысить полезность алгоритма.

Подводя итог, можно сказать, что алгоритм RBAVO-DE представляет собой значительный прогресс в отборе генов для классификации рака, обладающий заметными преимуществами в точности и эффективности. Однако устранение его потенциальных недостатков посредством дальнейших исследований может расширить его применимость и улучшить его эффективность, что сделает его еще более ценным инструментом в геномике и персонализированной медицине.

6. Выводы и будущие направления

Подход RBAVO-DE, предложенный в этой статье, был первым, который был реализован для решения проблем GS в данных экспрессии генов RNA-Seq и определения потенциальных биомаркеров для различных классов опухолей для улучшения найденного лучшего решения. Результаты были многообещающими:

эффективность и возможности предложенного алгоритма RBAVO-DE были значительно улучшены. SVM и k-NN, две широко используемые классификационные модели, использовались для оценки эффективности каждого набора выбранных генов. Возможности предложенного алгоритма RBAVO-DE сравнивались с двоичными вариантами одиннадцати широко используемых алгоритмов.

Machine Translated by Google

метаэвристические методы для оценки его на разных классах опухолей в разных случаях. Оценка проводилась с использованием комбинации оценочных показателей, таких как средняя приспособленность, точность классификации, количество выбранных генов, точность, полнота и показатель F1. Предложенный алгоритм RBAVO-DE с использованием классификаторов SVM и k-NN достиг более многообещающих результатов, чем другие оптимизаторы при решении проблем GS. Несмотря на многообещающие результаты, это исследование открывает несколько возможностей для будущих исследований для дальнейшего развития этой области:

- Алгоритмические усовершенствования: повышение эффективности алгоритма и снижение сложности вычислений.
- Применимость при перекрестных заболеваниях: тестирование предлагаемого алгоритма на генетических данных из различные заболевания, помимо рака.
- Сравнительный анализ с моделями глубокого обучения: оценка RBAVO-DE по сравнению с передовые модели глубокого обучения для анализа генетических данных.
- Клиническая проверка в реальных условиях: сотрудничество с клиническими экспертами для подтверждения практической полезности выбранных генов при лечении рака.
- Масштабируемость и параллелизация: повышение масштабируемости алгоритма за счет параллельных вычислений для эффективной обработки больших наборов генетических данных.
- Междисциплинарные приложения: изучение потенциала алгоритма в других областях, связанных с многомерными данными, таких как финансовое и экологическое моделирование.

В заключение, хотя алгоритм RBAVO-DE представляет собой значительный шаг вперед в области отбора генов для классификации рака, пути, намеченные для будущих исследований, подчеркивают потенциал для дальнейшего развития и более широкого применения этой работы. Продолжающееся междисциплинарное сотрудничество и инновации будут иметь решающее значение для раскрытия полного потенциала методологий отбора генов для улучшения результатов здравоохранения и углубления нашего понимания сложных заболеваний.

Вклад автора: MGG: Ресурсы, проверка и написание — обзор и редактирование. AAA: ресурсы, формальный анализ, обработка данных, проверка, написание — исходный черновик и написание — проверка и редактирование. AEE: исследование, визуализация, обработка данных, проверка, написание — исходный черновик и написание — проверка и редактирование. AAAE-M.: Концептуализация, методология, программное обеспечение, формальный анализ, исследование, обработка данных, проверка, написание — исходный проект и написание — обзор и редактирование. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Университет принца Саттама бин Абдель Азиза профинансировал эту исследовательскую работу в рамках проекта PSAU/2023/01/25612.

Заявление о доступности данных. Для повышения применимости и воспроизводимости этого исследования разработанное программное обеспечение и соответствующий код Python этого исследования общедоступны и доступны в [61].

Благодарности: Авторы выражают благодарность Университету принца Саттама бин Абдель Азиза за финансовую поддержку этого исследовательского проекта.

Конфликты интересов: авторы заявляют, что у них нет известных конкурирующих финансовых интересов или личных отношений, которые могли бы повлиять на работу, представленную в этой статье.

Сокращения

В исследовании используются следующие сокращения:

ABO	Оптимизация африканских стервятников
ХНСК	Плоскоклеточный рак головы и шеи
DE	Дифференциальная эволюция
КИЧ	Почки хромофоб
RBAVO-DE	Оптимизация рельефа бинарных африканских грифов на основе дифференциальной эволюции
КИРП	Почки Почечные Папиллярные
GS	Генный отбор
ЛИГК	Гепатоцеллюлярная карцинома печени
RNA-Seq	Секвенирование РНК

Machine Translated by Google

ПААД	Аденокарцинома поджелудочной железы
ДНК	Дезоксирибонуклеиновая кислота
ПКПГ	Феохромоцитома и параганглиома
СВМ	Машина опорных векторов
ЧИТАТЬ	Аденокарцинома прямой кишки
к-НН	k-ближайший сосед
САРК	Саркома
ФС	Выбор функции
СККМ	Кожная меланома кожи
МХМ	Метаэвристические методы
Среднее Fi	Средняя пригодность
МЛ	Машинное обучение
среднее значение	Средний размер выбранных генов
ДЛ	Глубокое обучение
СТДЕ	Стандартное отклонение
СА	Имитация отжига
РФ	Случайный лес
ФОЛЬГА	Индуктивный обучающийся первого порядка
ГСО	Гравитационный оптимизатор
SCCSA	Алгоритм поиска синус-косинус-кукушка
СМО	Оптимизация обезьяны-паука
БХГСО	Оптимизация растворимости двоичного газа по Генри
БПСО	Оптимизация роя бинарных частиц
БЭБК	Бинарная искусственная пчелиная семья
БССА	Двоичный алгоритм Salp Swarm
_____	Алгоритм двоичной летучей мыши
БГВО	Бинарная оптимизация Серого Волка
БГОА	Алгоритм оптимизации двоичного кузнечика
БВОА	Алгоритм оптимизации двоичного кита
БАСО	Оптимизация поиска двоичных атомов
ББСА	Бинарный алгоритм роя птиц
БХХО	Оптимизация двоичных файлов Харриса Хоукса
БББО	Бинарная оптимизация «бурого медведя»
БАО	Бинарная оптимизация Aquila
БМОА	Алгоритм оптимизации двоичного суриката
ЛУАД	Аденокарцинома легких
ЛЮСК	Плоскоклеточный рак легких
БРЦА	Инвазивный рак молочной железы
КИРЦ	Светлоклеточный рак почки почки
УЦЭК	Карцинома эндометрия тела матки
РПКМ	Чтений на килобазу на миллион сопоставленных чтений
СТАД	Аденокарцинома желудка
СДАЭ	Автоэнкодер с накопленным шумоподавлением
CGA	Атлас генома рака
БЛКА	Уротелиальная карцинома мочевого пузыря
ТНСА	Рак щитовидной железы
CESC	Рак шейки матки и эндоцервикальный рак
ЧОЛ	Холангиокарцинома
COAD	Аденокарцинома толстой кишки
ЭСКА	Рак пищевода
ГБМ	Мультиформная глиобластома
ТИМ	Тимома

Machine Translated by Google

Приложение А. Результаты сравнения предлагаемого RBAVO-DE и его аналогов

Приложение А.1. Результаты сравнения на основе модели k-NN

Таблица А1. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих k-NN. относительно средних значений фитнеса.

Набор данных	Метрический	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	Среднее	0,00190	STDE	0,00430	0,01110	0,00710	0,01120	0,01470	0,00940	0,01070	0,01140	0,01310	0,01030	0,01080	0,01590
	Среднее	0,01610	STDE		0,00530	0,00440	0,00540	0,00240	0,00550	0,00540	0,00540	0,00440	0,00550	0,00540	0,00180
CESC		0,00000	CHOL	Среднее	0,01990	0,02020	0,02050	0,01980	0,02000	0,02020	0,02020	0,01970	0,02020	0,02030	0,02060
		0,00000	STDE		0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
					0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
				0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
COAD	Среднее	0,00000	STDE	Среднее	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	0,02560		0,00000		0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
ЭСКА			STDE		0,02930	0,02960	0,02990	0,02930	0,02940	0,02970	0,02960	0,02910	0,02960	0,02980	0,03000
				0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00020
GBM	Среднее	0,00000	STDE		0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
				0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
HNSC	Среднее	0,00710	STDE	0,00350	0,01250	0,01290	0,01330	0,01280	0,01240	0,01310	0,01320	0,01270	0,01290	0,01320	0,01360
	Среднее	0,00000	STDE		0,00160	0,00140	0,00010	0,00030	0,00210	0,00010	0,00010	0,00020	0,00150	0,00010	0,00010
КИЧ		0,00000	Среднее	0,00250	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	STDE	0,00380	Среднее		0,00020	0. 00010	0. 00010	0,00020	0,00020	0,00020	0. 00010	0. 00010	0. 00010	0. 00010	0,00010
КИРЦ		0,00000	STDE	0,00000	0,01110	0,01040	0,01220	0,01200	0,01020	0,01200	0,01070	0,01040	0,01120	0,01170	0,01290
	Среднее	0,00000	STDE		0,00270	0,00340	0,00190	0,00020	0,00340	0,00190	0,00320	0,00300	0,00280	0,00220	0,00010
КИРП		0,00000	Среднее	LUAD	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	0,00870	STDE	0,00000		0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
ЛИГК		Среднее	0	0,00000	STDE	0. 00390	0,00430	0,00450	0. 00390	0. 00400	0. 00420	0,00430	0,00380	0,00430	0,00420	0,00460
	0,00000	PAAD	Среднее		0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
0,00000	STDE	0,00000	PCPG	Среднее	0,01260	0,01290	0,01310	0,01240	0,01270	0,01290	0,01290	0,01240	0,01290	0,01300	0,01340
	0,00000	STDE	0,00000		0,00020	0,00010	0,00010	0,00020	0,00020	0,00010	0,00010	0,00020	0,00010	0,00010	0,00010
ЛЮСК		READ	Среднее	0,00000	0,00400	0,00430	0,00450	0,00380	0,00410	0,00430	0,00430	0,00390	0,00430	0,00430	0,00480
	STDE	0,00000	SARC		0,00020	0,00010	0,00010	0,00020	0,00010	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
Среднее	0,01890	STDE	0,00000	SKCM	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	Среднее	0,00000	STDE		0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
0,00000	Среднее	0,00070	STDE	0,00280	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
					0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
					0,00390	0,0043 0	0,0045 0	0,00390	0,00400	0,00430	0,0043 0	0,0037 0	0,0042 0	0,0042 0	0,00470
					0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
					0,02260	0,02290	0,02320	0,02260	0,02270	0,02300	0,02290	0,02240	0,02290	0,02300	0,02330
					0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
					0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
					0,0002 0	0,00010	0,00010	0,0002 0	0,0002 0	0,0002 0	0,00010	0,00010	0,00010	0,00010	0,00010
СТАД					0,00830	0,00620	0,00930	0,01470	0,00810	0,00950	0,00800	0,01170	0,00940	0,00840	0,01570
					0,00500	0,00360	0,00510	0,00020	0,00500	0,00510	0,00480	0,00480	0,00520	0,00490	0,00010
THCA	Среднее	0,00000	STDE		0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00460
				0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
THYM	Среднее	0,00000	STDE		0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
				0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
UCEC	Среднее	0,00000	STDE		0,01920	0,01710	0,02780	0,03540	0,02010	0,02770	0,02690	0,02560	0,02520	0,02860	0,03670
				0,00250	0,01200	0,01220	0,00610	0,01000	0,01190	0,00620	0,00720	0,00830	0,00910	0,00450	0,01050
Рейтинг W|T|L		22|0|0			0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22

Значения, выделенные жирным шрифтом, обозначают лучшие результаты.

Таблица А2. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих k-NN. относительно средних значений выбранных генов.

Набор данных	Метрический	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	Среднее	142,03	STDE	017,18	210,17	237,70	235,00	196,03	218,17	228,50	226,30	195,23	228,27	232,03	237,13
	Среднее	122,50	STDE		020,50	019,07	017,12	023,83	018,27	019,43	014,81	013,17	014,60	018,90	014.14
CESC		005,45	CHOL	Среднее	195,13	213,03	225,30	194,10	199,50	213,63	212,67	185,60	213,33	215,90	233,43
	122,13	STDE	006,18		011,17	005,31	003,11	009,66	009,30	008,35	005,66	006,49	005,88	006,02	005.64
Среднее	COAD	120,30	STDE	005,82	195,13	213,03	225,30	194,10	199,50	213,63	212,67	185,60	211,00	211,57	232,63
	Среднее	120,17	STDE		011,17	005,31	003,11	009,66	009,30	008,35	005,66	006,49	006,44	005,08	004.36
006,81	GBM	Среднее	120,73	STDE	195,13	213,03	225,30	194,10	199,50	213,63	212,67	185,60	211,00	211,57	232,63
					011,17	005,31	003,11	009,66	009,30	008,35	005,66	006,49	006,44	005,08	004.36
ЭСКА					195,13	213,03	225,30	194,10	199,50	213,63	212,67	185,60	211,57	218,87	230.07
					011,17	005,31	003,11	009,66	009,30	008,35	005,66	006,49	006,94	004,68	007.64
					195,13	212,77	225,30	194,10	199,73	213,17	212,80	185,43	211,00	211,83	232,63
				006.54	011,17	005.40	003,11	009,66	009,05	008,51	005,72	006,40	006,44	004,99	004.39
HNSC	Среднее	133,43	STDE	017,63	202,87	222,67	227,43	203,00	211,37	217,97	220,87	195,93	219,83	223,63	241,27
	Среднее	121,43	STDE		009,26	011,96	003.91	013,24	007,65	007,41	005,32	008,26	007,71	005,68	006.21
КИЧ		006,58			195,13	213,03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211,57	232,63
					011,17	005,31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005,08	004.36

Machine Translated by Google

Таблица А2. Продолжение

Набор данных Метрический RBAVO-DE BSSA BABC BPSO BBA BGWO BWOA BGOA BHHO BBSA BASO BHGSO												
КИРЦ	В среднем 137,57 STDE 017,10 В среднем	200,83 009,63	220,17 011,74	227,70 008,27	192,17 008,73	209,87 014,08	216,13 010,11	222,30 015,18	193,27 017,36	218,73 013,82	218,60 012,78	234.10 004.21
КИРП	122,33 STDE В среднем 123,40 009.08	195,13 011,17	213,03 005,31	225,30 003.11	194,10 009,66	199,50 009.30	213,63 008,35	212,67 005,66	185,60 006,49	211,00 006,44	211,57 005.08	232,63 004.36
ЛИГК	STDE 006,59 LUAD B среднем 127,03 STDE	197,00 007,78	214,03 005,22	225,67 003,36	193,33 009,17	200,63 010,02	212,33 009,35	213,63 004,89	189,30 006,89	213,10 005,69	212.03 004.81	232,43 004.97
008,62 В среднем 125,73 STDE 006,35 PAAD В среднем 121,10		197,47 008,98	216,13 004,74	222,93 006,16	190,83 008,61	205,33 010,57	212,57 006,15	215,67 004,43	191,80 008,50	215,27 006,13	217.83 004.71	237,90 004.94
ЛЮСК	STDE 007,71 PCPG B среднем 122,17 STDE	198,87 008,77	215,63 003,54	223,00 004,89	189,13 010,69	203,60 007,20	213,23 009,70	216,37 003,44	194,67 006,40	215,53 005,54	215.23 006.16	241,23 004.66
006,42 В среднем READ 120,13 STDE 007,72 SARC Среднее		195,13 011,17	213,03 005,31	225,30 003,11	194,10 009,66	199,50 009,30	213,63 008,35	212,67 005,66	185,60 006,49	211,00 006,44	211.57 005.08	232,63 004.36
122,47 STDE 005,79 SKCM Среднее 123,03 STDE 008,83		195,13 011,17	213,03 005,31	225,30 003,11	194,10 009,66	199,50 009,30	213,63 008,35	212,67 005,66	185,60 006,49	211,00 006,44	211.57 005.08	232,63 004.36
Среднее 142,60 STDE 013,31 THCA Среднее 117,97 STDE		195,13 011,17	213,03 005,31	225,30 003,11	194,10 009,66	199,50 009,30	213,63 008,35	212,67 005,66	185,60 006,49	211,00 006,44	211.57 005.08	232,63 004.36
006,67 Среднее THYM 120,37 STDE 004,89 Среднее UCEC		195,13 011 .17	213,03 005 .31	225,30 003,11	194,10 009,66	199,50 009,30	213,63 008,35	212,67 005,66	185,60 006 .49	212,90 006 .30	216.43 004.98	231,80 003.40
143,93 STDE 013,18		195,13 011,17	213,03 005,31	225,30 003,11	194,10 009,66	199,50 009,30	213,63 008,35	212,67 005,66	185,60 006,49	211,00 006,44	211.57 005.08	232,63 004.36
СТАД		215,00 018,43	236,17 013,94	243,33 016,29	187 .50 010,27	221,17 019 .24	234 .87 022,63	232,77 014,25	201,73 014,00	231,13 014,04	237 .10 015,35	236,97 005.86
		195,43 011,20	213,40 005,29	225 .13 003,10	194,10 009,66	199,63 009,81	214,13 008,59	212,80 005,69	186,53 005,88	211,00 006,44	211,57 005,08	231,90 004.78
		195,13 011,17	213,03 005,31	225,30 003,11	194,10 009,66	199,50 009,30	213,63 008,35	212 .67 005,66	185,60 006,49	211,00 006,44	211,57 005,08	232,63 004.36
		216,33 015,41	237,87 015,82	236,00 010,11	242,37 034,13	220,07 011,26	230,80 011,27	229,17 011,66	206,77 010,84	228,40 010,32	235,00 006,14	266,67 030.55
Рейтинг W|T|L	22|0|0	0|0|22 0|0|22 Значения,		0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22

выделенные жирным шрифтом, обозначают лучшие результаты.

Таблица А3. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих k-NN. относительно средних значений точности.

Набор данных RBAVO-DE BSSA		БЭБК	БПСО	_____	БГВО БВОА БГОА БХХО ББСА					БАСО БХГСО	
BLCA 1,00000 CESC 0,98390	1,00000	1,00000	1,00000	0,99960	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
CHOL 1,00000 COAD	0,98390	0,98390	0,98390	0,98390	0,98390	0,98390	0,98390	0,98390	0,98390	0,98390	0,98390
1,00000 ESCA 0,97370 GBM	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 HNSC 0,99180	1,00000	1,00000	1,00000	0,99890	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
KICH 1,00000 KIRC 1,00000	0,97370	0,97370	0,97370	0,97370	0,97370	0,97370	0,97370	0,97370	0,97370	0,97370	0,97370
KIRP 1,00000 LIHC 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
LUAD 1. 00000 LUSC	0,96470	0,96540	0,96500	0,96350	0,96690	0,96660	0,96570	0,96540	0,96540	0,96100	0,96290
1,00000 PAAD 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
PCPG 1,00000 ЧТЕНИЕ	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 SARC 0,98110	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
SKCM 1,00000 STAD	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
0,99920 THCA 1,00000	0,99770	0,99740	0,99840	0,99870	0,99770	0,99800	0,99540	0,99710	0,99740	0,99670	0,99800
THYM 1,00000 UCEC	1,00000	1,00000	1,00000	0,99790	1,00000	0,99970	1,00000	1,00000	1,00000	0,97900	0,99930
1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1. 00000	1. 00000	1. 00000	1. 00000	1. 00000	1. 00000	1. 00000	1. 00000	1. 00000	1. 00000	1,00000
	0,98110	0,98110	0,98110	0,98110	0,98110	0,98110	0,98110	0,98110	0,98110	0,98110	0,98110
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	0,95180	0,95180	0,95180	0,94870	0,95180	0,95180	0,95180	0,95180	0,95180	0,99000	0,95090
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	0,99910	1,00000	1,00000	1,00000	1,00000	1,00000	0,93330	0,99910
Рейтинг (B|T|Л)	3|19|0	0|19|3	0|19|3	0|19|3	0|15|7	0|19|3	0|18|4	0|19|3	0|19|3	0|19|3	0|17|5	0|17|5

Значения, выделенные жирным шрифтом, обозначают лучшие результаты.

Таблица А4. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих k-NN. относительно средних значений отзыва.

Набор данных RBAVO-DE BSSA		БЭБК	БПСО	_____	БГВО БВОА БГОА БХХО ББСА					БАСО БХГСО	
BLCA 1,00000 CESC 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
CHOL 1,00000 COAD	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 ESCA 1,00000 GBM	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000

Machine Translated by Google

Таблица A4. Продолжение

Набор данных RBAVO-DE BSSA		БЭБК	БПСО	_____	БГВО БВОА БГОА БХХО ББСА					БАСО БХГСО	
HNSC 1,00000 KICH 1,00000	0,99520	0,99450	0,99480	0,99590	0,99280	0,99310	0,99420	0,99450	0,99450	0,99790	0,99730
KIRC 0,99710	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	0,99040	0,99040	0,99040	0,98970	0,99040	0,99040	0,99040	0,99040	0,99040	0,99010	0,99040
KIRP 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
LIHC 1,00000 LUAD 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
LUSC 1,00000 PAAD 1,00000	0,99240	0,99270	0,99170	0,99140	0,99240	0,99210	0,99470	0,99310	0,99270	0,99340	0,99210
PCPG 1,00000 ЧТЕНИЕ	1,00000	1,00000	1,00000	0,99580	1,00000	1,00000	1,00000	0,99930	1,00000	0,99310	0,99270
1,00000 SARC 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
SKCM 1,00000 STAD 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
THCA 1,00000 THYM	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 UCEC 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	0,99750	1,00000	1,00000	1,00000	1,00000	1,00000	0,99620	0,99750
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	0,99370	1,00000	1,00000	1,00000	1,00000	1,00000	0,99640	1,00000
Рейтинг (B|T|Л) 3|19|0	0|19|3	0|19|3	0|19|3	0|16|6	0|19|3	0|19|3	0|19|3	0|18|4	0|19|3	0|16|6	0|17|5

Значения, выделенные жирным шрифтом, обозначают лучшие результаты.

Таблица A5. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих k-NN.

относительно средних значений F1-балла.

Набор данных RBAVO-DE BSSA		БЭБК	БПСО	_____	БГВО БВОА БГОА БХХО ББСА					БАСО БХГСО	
BLCA 1,00000 CESC 0,99190	1,00000	1,00000	1,00000	0,99980	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
CHOL 1,00000 COAD	0,99190	0,99190	0,99190	0,99190	0,99190	0,99190	0,99190	0,99190	0,99190	0,99190	0,99190
1,00000 ESCA GBM 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
HNSC 0,99590 KICH KIRC	1,00000	1,00000	1,00000	0,99940	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
KIRP 0,98670	0,98670	0,98670	0,98670	0,98670	0,98670	0,98670	0,98670	0,98670	0,98670	0,98670	0,98670
1,00000 LIHC 1,00000 LUAD	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
0,99500 LUSC 1,00000	0,97970	0,97970	0,97970	0,97940	0,97970	0,97970	0,97970	0,97970	0,97970	0,97980	0,97970
PAAD 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,000 0,99860	0,99520	0,99520	0,99520	0,99480	0,99520	0,99520	0,99520	0,99520	0,99520	0,99500	0,99520
00 PCPG 1,00000 ЧТЕНИЕ	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 SARC 0,99050	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
SKCM 1,00000 STAD	0,99500	0,99500	0,99500	0,99500	0,99500	0,99500	0,99500	0,99500	0,99500	0,99500	0,99500
0,99960 THCA 1,00000	1,00000	1,00000	1,00000	0,99690	1,00000	0,99980	1,00000	0,99970	1,00000	0,99630	0,99600
THYM 1,00000 UCEC 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	0,99050	0,99050	0,99050	0,99050	0,99050	0,99050	0,99050	0,99050	0,99050	0,99050	0,99050
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	0,97530	0,97530	0,97530	0,97240	0,97530	0,97530	0,97530	0,97530	0,97530	0,97180	0,97360
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	0,99640	1,00000	1,00000	1,00000	1,00000	1,00000	0,99640	0,99960
Рейтинг (B|T|Л) 3|19|0	0|18|4	0|18|4	0|18|4	0|14|8	0|18|4	0|17|5	0|18|4	0|17|5	0|18|4	0|16|6	0|16|6

Значения, выделенные жирным шрифтом, обозначают лучшие результаты.

Приложение A.2. Результаты сравнения на основе модели SVM

Таблица A6. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих SVM.

относительно средних значений фитнеса.

Набор данных Метрический RBAVO-DE BSSA BABC BPSO BBA BGWO BWOA BGOA BHHO BBSA BASO BHGSO											
BLCA Среднее 0,00000 STDE 0,00000	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00460
Среднее 0,00000 STDE	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
CESC 0,00000 CHOL Среднее	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
0,00000 STDE 0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
COAD Среднее 0,00000 STDE Среднее	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
0,01370 STDE	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
ESCA	0,02840	0,02720	0,02990	0,02930	0,02690	0,02970	0,02880	0,02830	0,02880	0,02980	0,03000
0,01280	0,00460	0,00750	0,00010	0,00020	0,00750	0,00020	0,00450	0,00460	0,00440	0,00010	0,00020
GBM Среднее 0,00000 STDE	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00460
0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
HNSC Среднее 0,00000 STDE	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00460
0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010

Machine Translated by Google

Таблица A6. Продолжение

Набор данных Метрический RBAVO-DE BSSA BABC BPSO BBA BGWO BWOA BGOA BHHO BBSA BASO BHGSO												
КИЧ	Среднее 0,00000 STDE	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	0,00000 Среднее 0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
КИРЦ	STDE Среднее 0,00000	0,00390	0,00430	0,00450	0,00380	0,00400	0,00430	0,00430	0,00380	0,00430	0,00430	0,00460
	STDE 0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00020
КИРП	Среднее 0,00000 STDE	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
ЛИГК		0,00410	0,00440	0,00450	0,00500	0,00430	0,00440	0,00450	0,00400	0,00450	0,00450	0,00500
	0,00000	0,00010	0,00010	0,00010	0,00280	0,00010	0,00020	0,00010	0,00020	0,00010	0,00010	0,00020
LUAD Среднее 0,00840 STDE 0,00160		0,01250	0,01290	0,01310	0,01250	0,01260	0,01290	0,01290	0,01230	0,01290	0,01260	0,01330
	Среднее 0,00000 STDE	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00150	0,00010
ЛЮСК	0,00000 PAAD Среднее	0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	0,00000 STDE 0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
Среднее PCPG 0,00000 STDE 0,00000		0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	Среднее READ 0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
STDE 0,00000 Среднее SARC 0,00570		0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	STDE 0,008 60 SKCM	0,00020	0. 00010	0. 00010	0,00020	0,00020	0,00020	0. 00010	0,00010	0. 00010	0. 00010	0,00010
Среднее 0,00000 STDE 0,00000 Среднее		0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	0,01040 STDE 0,00280	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
THCA Среднее 0,00000 STDE 0,00000		0,01780	0,01570	0,01840	0,02260	0. 01480	0. 01760	0,01870	0,02000	0,01870	0,02000	0,02330
	THYM Среднее 0,00000	0. 00800	0,00880	0,00800	0. 00020	0,00900	0,00820	0,00770	0. 00610	0,00770	0,00670	0,00010
STDE 0,00000 UCEC Среднее 0,00000		0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
	СТАНДАРТНОЕ 0,00000	0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
СТАД		0,01490	0,01530	0,01550	0,01450	0,01500	0,01530	0,01530	0,01470	0,01520	0,01530	0,01560
		0,00020	0,00010	0,00010	0,00200	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
		0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
		0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
		0,00390	0,00430	0,00450	0,00390	0,00400	0,00430	0,00430	0,00370	0,00420	0,00420	0,00470
		0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00010	0,00010	0,00010	0,00010
		0,00390	0,00430	0,00450	0,00380	0,00400	0,00430	0,00430	0,00380	0,00420	0,00430	0,00470
		0,00020	0,00010	0,00010	0,00020	0,00020	0,00020	0,00010	0,00020	0,00010	0,00010	0,00010
Рейтинг W|T|L	22|0|0	0|0|22 0|0|22 0|0|22 Значения, выделенные			0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22

жирным шрифтом, обозначают лучшие результаты.

Таблица A7. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих SVM.

относительно средних значений выбранных генов.

Набор данных Метрический RBAVO-DE BSSA BABC BPSO BBA BGWO BWOA BGOA BHHO BBSA BASO BHGSO												
BLCA Среднее 121,37 STDE 007,49		194,70	213.03	225.30	194,10	199,50	213,40	212,67	186,27	211,00	211,83	232,37
	Среднее 119,83 STDE	011,44	005.18	003.11	009,66	009,30	008,44	005,66	005,77	006,44	005,25	004.42
CESC	007,02 CHOL Среднее	195,13	213.03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211,57	232,63
	125,57 STDE 007,14	011,17	005.31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005,08	004.36
Среднее COAD 122,63 STDE 007,73		195,13	213.03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211,57	232,63
	Среднее 125,50 STDE	011,17	005.31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005,08	004.36
011,18 GBM Среднее 119,73 STDE		195,13	213.03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211,57	232,63
	008,07 Среднее HNSC	011,17	005.31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005,08	004.36
ЭСКА	122,17 DE 008,29	195,13	215.40	225.30	194,10	201,77	213,63	213,13	185,70	212,40	218,87	230.07
	Среднее 121,73 STDE	011,17	009.58	003.11	009,66	012,33	008,35	006,44	006,47	008,98	004,68	007.64
008,49 Среднее 123,87 STDE 007,57		194,93	213.03	225.30	193,90	198,77	213,70	212,83	185,50	211,00	211,00	232,33
	Среднее 120,07 STDE	011,27	005.31	003.11	009,47	010,28	008,36	005,69	006,51	006,44	004,92	004.15
007,87 Среднее 122,63 STDE 008,01		196,40	213.90	225.13	194,03	202,10	213,83	213,50	186,93	211,47	210,87	232,17
	LUAD Среднее 121,17	008,49	005.22	003.19	009,56	010,76	009,34	005,22	005,74	005,66	004,79	005.02
КИЧ	STDE 007,56 Среднее	195,13	213.03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211,57	232,63
	122,67 STDE	011,17	005.31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005,08	004.36
КИРЦ		197,10	214.70	225.00	191,93	200,30	212,97	213,13	187,57	213,83	214,63	231,27
		009 .42	005 .09	004 .23	008 .70	011 .59	007 .64	004 .94	005 .31	005 .81	004 .20	008.02
КИРП		195,13	213,03	225,30	194,10	199,50	213,63	212,67	185,60	211,00	211,57	232,63
		011,17	005,31	003,11	009,66	009,30	008,35	005,66	006,49	006,44	005,08	004.36
ЛИГК		204,27	220,37	226,40	212,83	214,07	219,07	223,70	200,60	223,03	225,17	249,90
		006,99	006,29	006,80	017,60	007,06	007,88	004,13	007,66	006,48	004,81	007.97
		195,13	213,03	225,30	194,10	199,43	213,63	212,67	185,87	212,40	214,50	233.00
		011,17	005,31	003,11	009,66	009,37	008,35	005,66	006,40	006,10	005,98	004.34
ЛЮСК		195,13	213,03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211,57	232,63
	006.28	011,17	005,31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005.08	004.36
PAAD Среднее 119,23 STDE 006,90 PCPG		195,13	213,03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211.57	232,63
	Среднее 122,43 STDE	011,17	005,31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005.08	004.36
006,68 ЧТЕНИЕ Среднее 123,93 STDE		195,13	213,03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211.57	232,63
	007,45 SARC Среднее	011,17	005,31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005.08	004.36
135,27 STDE 018,57 SKCM Среднее		195,13	213,03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211.57	232,63
	122,47 STDE 007,66	011,17	005,31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005.08	004.36
		202,90	226,90	233.77	194,10	210,53	223,87	217,80	189,93	220,57	220.87	231,53
		020,86	020,25	014.44	009,66	017,74	017,39	013,90	013,41	016,40	011.76	003.70
		195,13	213,03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211.57	232,63
		0 11.17	0 05.31	003.11	009,66	009,30	008,35	005,66	0 06.49	0 06.44	005.08	004.36

Machine Translated by Google

Таблица A7. Продолжение

Набор данных Метрический RBAVO-DE BSSA BABC BPSO BBA BGWO BWOA BGOA BHHO BBSA BASO BHGSO												
СТАД	Среднее 123,53 STDE	194,60	213,40	225.03	194,30	198,80	214,10	212,70	185,70	211,97	215.07	231,77
	010,65 THCA Среднее	010,22	005,38	003.29	009,79	009,09	008,68	005,54	006,92	005,88	005.12	004.58
119,47 STDE		195,13	213,03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211.57	232,63
	005.58	011,17	005,31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005.08	004.36
Среднее значение THYM 121,80 STDE		195,13	213,03	225.30	194,10	199,50	213,63	212,67	185,60	211,00	211.57	232,63
	007,21 Среднее	011,17	005,31	003.11	009,66	009,30	008,35	005,66	006,49	006,44	005.08	004.36
значение UCEC 122,23 STDE 005,93		196,00	213,27	224.83	192,03	200,97	212,60	213,03	189,23	212,23	212.93	233,93
		009,06	006,31	004.27	007,85	008,24	007,74	005,67	008,78	004,98	004.97	003.97
Рейтинг W|T|L	22|0|0	0|0|22 0|0|22 0|0|22 Значения, выделенные		0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22	0|0|22

жирным шрифтом, обозначают лучшие результаты.

Таблица A8. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих SVM. относительно средних значений точности.

Набор данных RBAVO-DE BSSA		БЭБК	БПСО	_____	БГВО БВОА БГОА БХХО ББСА					БАСО БХГСО	
BLCA 1,00000 CESC 1,00000	1.00000	1.00000	1.00000	1,00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1,00000
CHOL 1,00000 COAD	1.00000	1.00000	1.00000	0,99890	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1,00000
1,00000 ESCA 1,00000 GBM	1.00000	1.00000	1.00000	1,00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1,00000
1,00000 HNSC 1,00000	1.00000	1.00000	1.00000	0,99890	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1,00000
KICH 1,00000 KIRC 1,00000	1.00000	1.00000	1.00000	0,99820	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	0,99210
KIRP 1,00000 1,00000	1.00000	1.00000	1.00000	1,00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1,00000
	0.94140	0.94170	0.93750	0,92840	0.94140	0.93990	0.94170	0.94110	0.94050	0.84550	0,93150
	1.00000	1.00000	1.00000	1,00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1,00000
	1.00000	1.00000	1.00000	0,99940	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1,00000
	1.00000	1.00000	1.00000	1,00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1,00000
LIHC	1.00000	1.00000	1.00000	1,00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1,00000
LUAD 1,00000 LUSC 1,00000	1.00000	1.00000	1.00000	0,99870	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	0,99930
PAAD 1,00000 PCPG	0.99860	0.99860	0.99860	0,99830	0.99660	0.99860	0.99900	0.99930	0.99860	0.81400	0,99970
1,00000 ЧТЕНИЕ 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
SARC 1,00000 SKCM	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 STAD 0,98830	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
THCA 1,00000 THYM	0,99940	1,00000	0,99940	0,99500	0,99940	1,00000	0,99940	0,99620	0,99940	1,00000	0,99120
1,00000 UCEC 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	0,94840	0,95110	0,94170	0,93610	0,94730	0,94690	0,94690	0,93910	0,94650	0,95200	0,93500
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	0,99910	0,99820	0,99820	0,99300	1,00000	0,99820	0,99820	0,99910	1,00000	1,00000	0,99390
Рейтинг (B|T|Л) 3|19|0	0|17|5	0|18|4	0|17|5	0|12|10	0|18|4	0|18|4	0|17|5	0|17|5	0|18|4	0|19|3	0|15|7

Значения, выделенные жирным шрифтом, обозначают лучшие результаты.

Таблица A9. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих SVM. относительно средних значений отзыва.

Набор данных RBAVO-DE BSSA		БЭБК	БПСО	_____	БГВО БВОА БГОА БХХО ББСА					БАСО БХГСО	
BLCA 1,00000 CESC 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
CHOL 1,00000 COAD	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 ESCA 1,00000 GBM	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 HNSC 1,00000	1,00000	1,00000	1,00000	0,99940	1,00000	1,00000	1,00000	1,00000	1,00000	0,99940	1,00000
KICH KIRC KIRP LIHC	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 LUAD 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
LUSC 1,00000 PAAD 1,000	1,00000	1,00000	0,99970	0,99930	1,00000	1,00000	1,00000	1,00000	1,00000	0,99930	0,99930
00 PCPG 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 1,00000	0,99040	0,99040	0,99040	0,99010	0,99040	0,99040	0,99040	0,99040	0,99040	0,98780	0,99040
ЧТЕНИЕ 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 SARC 1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
SKCM 1,00000 STAD	1,00000	1,00000	1,00000	0,99500	1,00000	1,00000	1,00000	0,99930	1,00000	0,99830	0,99770
1,00000 THCA 1,00000	0,99100	0,99100	0,99100	0,99130	0,98960	0,99100	0,99060	0,99030	0,99100	0,99100	0,98990
THYM 1,00000 UCEC	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	0,99550	0,99640	0,98380	0,98200	0,99460	0,99100	0,99010	0,98470	0,99370	0,97840	0,97930
Рейтинг (B|T|Л) 3|19|0	0|19|3	0|19|3	0|18|4	0|16|6	0|19|3	0|19|3	0|19|3	0|18|4	0|19|3	0|16|6	0|17|5

Значения, выделенные жирным шрифтом, обозначают лучшие результаты.

Machine Translated by Google

Таблица A10. Результаты предложенного алгоритма RBAVO-DE и его аналогов, использующих SVM.

относительно средних значений F1-балла.

Набор данных RBAVO-DE BSSA		БЭБК	БПСО	.....	БГВО БВОА БГОА БХХО ББСА					БАСО БХГСО	
BLCA 1,00000 CESC 1,00000	1.00000	1.00000	1,00000	1,00000	1.00000	1.00000	1,00000	1,00000	1,00000	1,00000	1,00000
CHOL 1,00000 COAD	1.00000	1.00000	1,00000	0,99950	1.00000	1.00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 ESCA 1,00000 GBM	1.00000	1.00000	1,00000	1,00000	1.00000	1.00000	1,00000	1,00000	1,00000	1,00000	1,00000
1,00000 HNSC 1,00000	1.00000	1.00000	1,00000	0,99910	1.00000	1.00000	1,00000	1,00000	1,00000	0,99970	1,00000
KICH 1,00000 KIRC 1,00000	1.00000	1.00000	1,00000	0,99910	1.00000	1.00000	1,00000	1,00000	1,00000	0,99820	0,99600
KIRP 1,00000 LIHC 1,00000	1.00000	1.00000	1,00000	1,00000	1.00000	1.00000	1,00000	1,00000	1,00000	1,00000	1,00000
LUAD 1. 00000 LUSC	0.96980	0.97000	0,96760	0,96250	0.96980	0.96900	0,96970	0,96970	0,96940	0,95730	0,96420
1,00000 PAAD 1,00000	1.00000	1.00000	1,00000	1,00000	1.00000	1.00000	1,00000	1,00000	1,00000	1,00000	1,00000
PCPG 1,00000 ЧТЕНИЕ	0.99520	0.99520	0,99520	0,99470	0.99520	0.99520	0,99520	0,99520	0,99520	0,99320	0,99520
1,00000 SARC 0,99900	1.00000	1.00000	1,00000	1,00000	1.00000	1.00000	1,00000	1,00000	1,00000	1,00000	1,00000
SKCM 1,00000 STAD	1.00000	1.00000	1,00000	1,00000	1.00000	1.00000	1,00000	1,00000	1,00000	1,00000	1,00000
0,99410 THCA 1,00000	1,00000	1,00000	1,00000	0,99690	1,00000	1,00000	1,00000	0,99970	1,00000	0,99880	0,99850
THYM 1,00000 UCEC	0,99480	0,99480	0,99480	0,99480	0,99480	0,99480	0,99480	0,99480	0,99480	0,99480	0,99480
1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	0,99970	0,99810	0,99970	0,99750	0,99970	1,00000	0,99970	0,99810	0,99970	0,99710	0,99560
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	0,97350	0,97490	0,96990	0,96700	0,97290	0,97270	0,97270	0,96860	0,97250	0,96560	0,96640
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
	0,99730	0,99730	0,99090	0,98730	0,99730	0,99450	0,99410	0,99180	0,99680	0,98680	0,98640
Рейтинг (В|Т|Л) 5|17|0	0|16|6	0|16|6	0|16|6	0|12|10	0|16|6	0|17|5	0|16|6	0|15|7	0|16|6	0|13|9	0|14|8

Значения, выделенные жирным шрифтом, обозначают лучшие результаты.

Приложение Б. Графики сходимости предлагаемых RBAVO-DE и Конкурентные методы

Приложение Б.1. Графики сходимости, использующие модель SVM

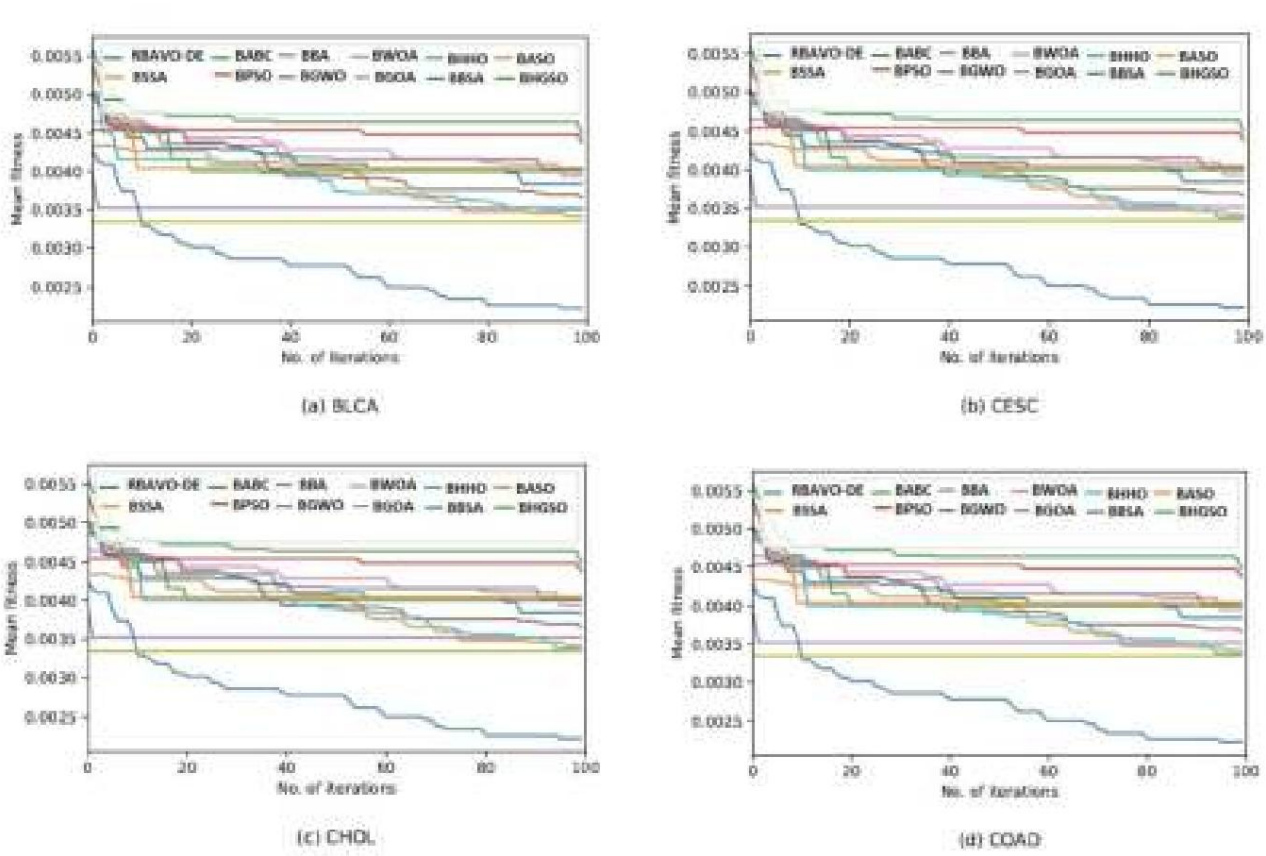
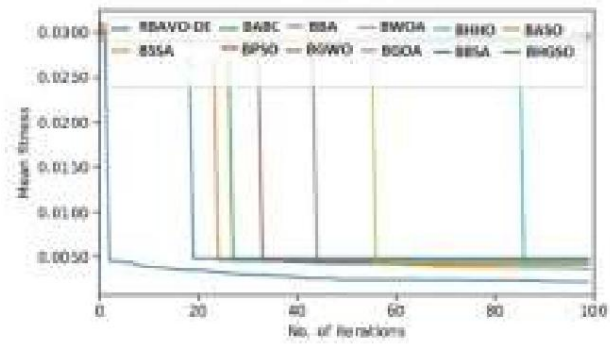
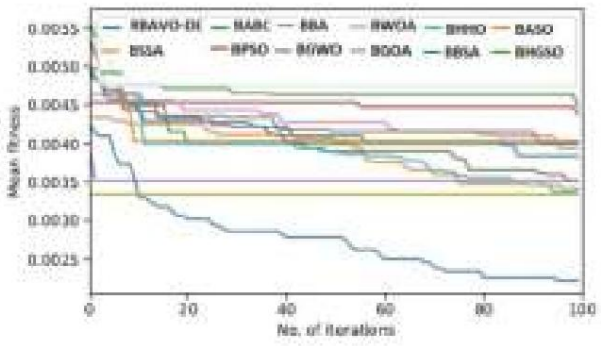


Рисунок A1. Продолжение

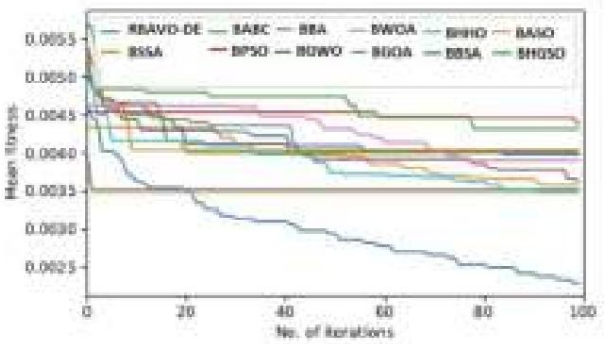
Machine Translated by Google



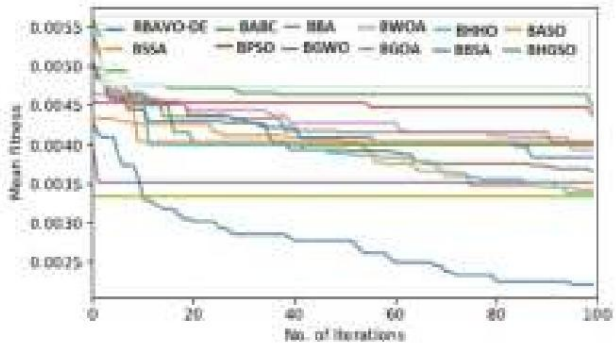
(e) ESCA



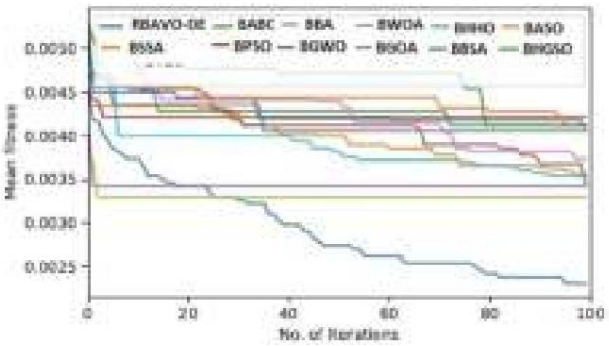
(f) GBM



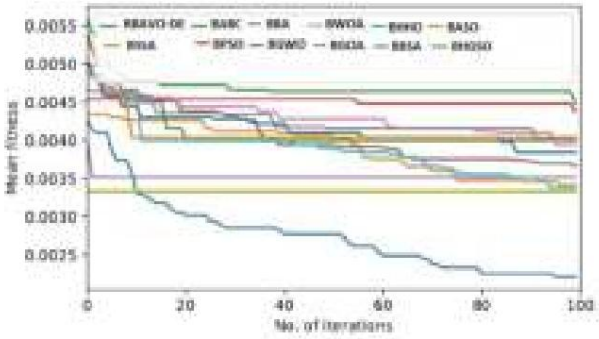
(g) HNSC



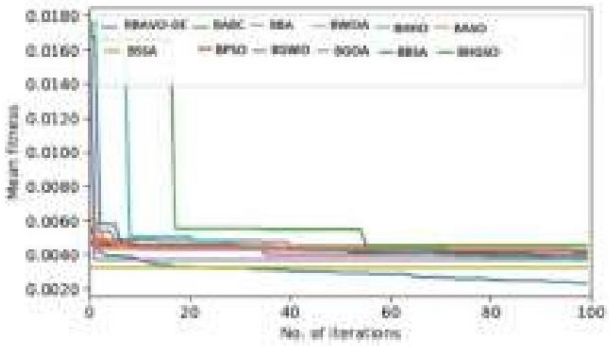
(h) KICH



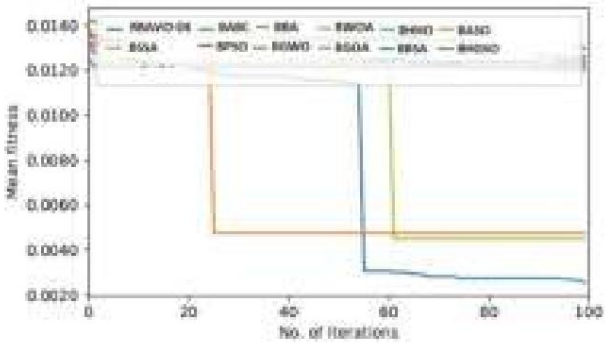
(i) KIRC



(j) KIRP



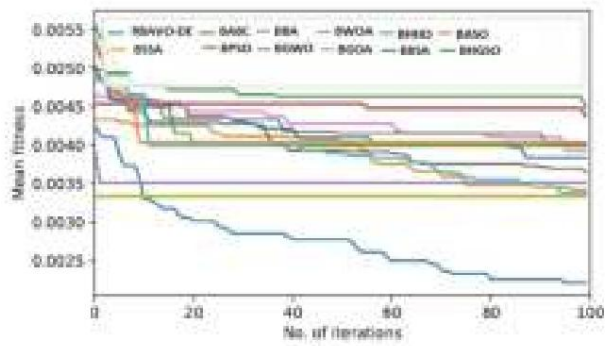
(k) LIHC



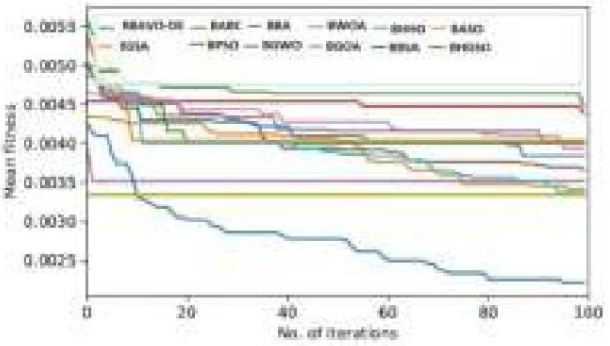
(l) LUAD

Рисунок А1. Продолжение

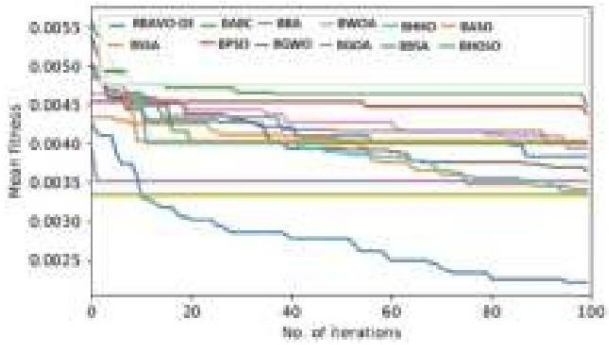
Machine Translated by Google



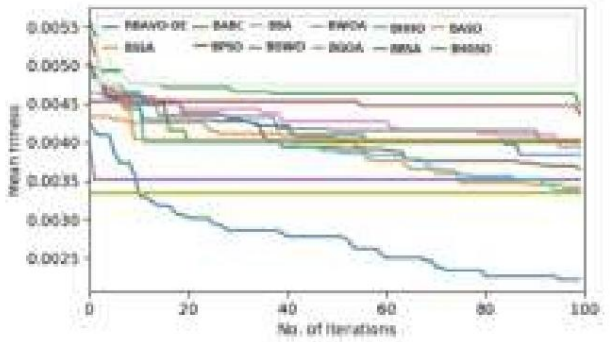
(m) LUSC



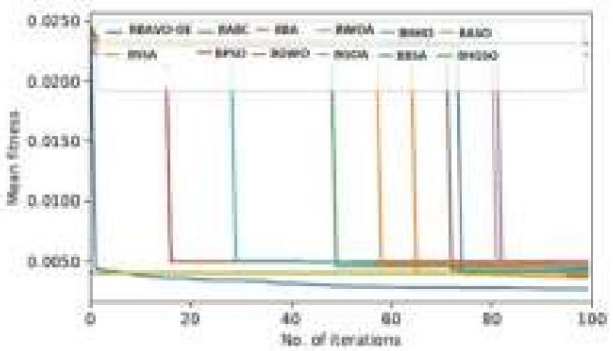
(n) PAAD



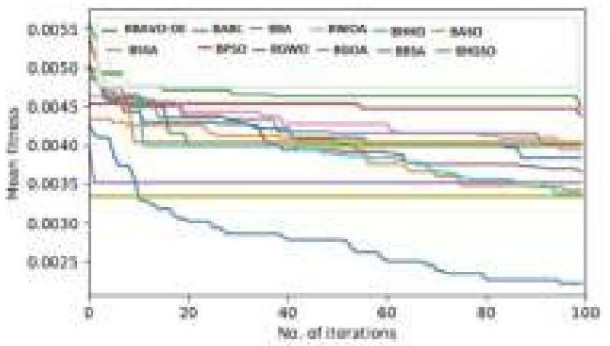
(o) PCPG



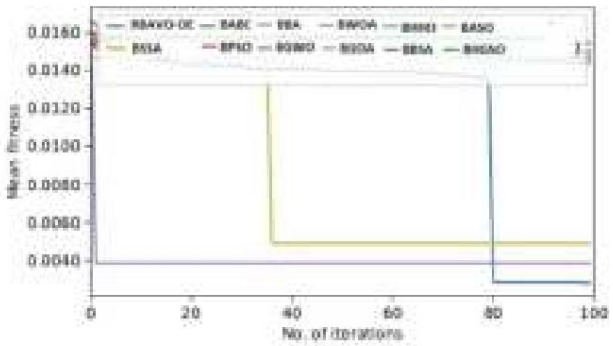
(p) READ



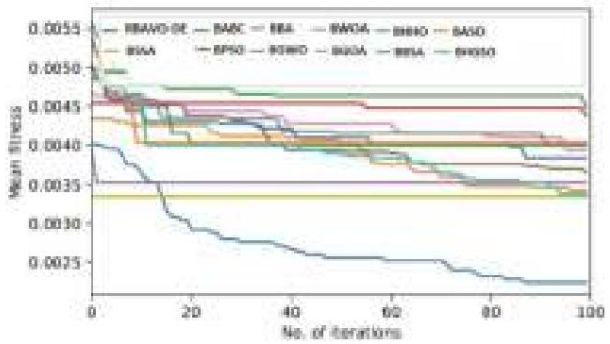
(q) SARC



(r) SKCM



(s) STAD



(t) THCA

Рисунок A1. Продолжение

Machine Translated by Google

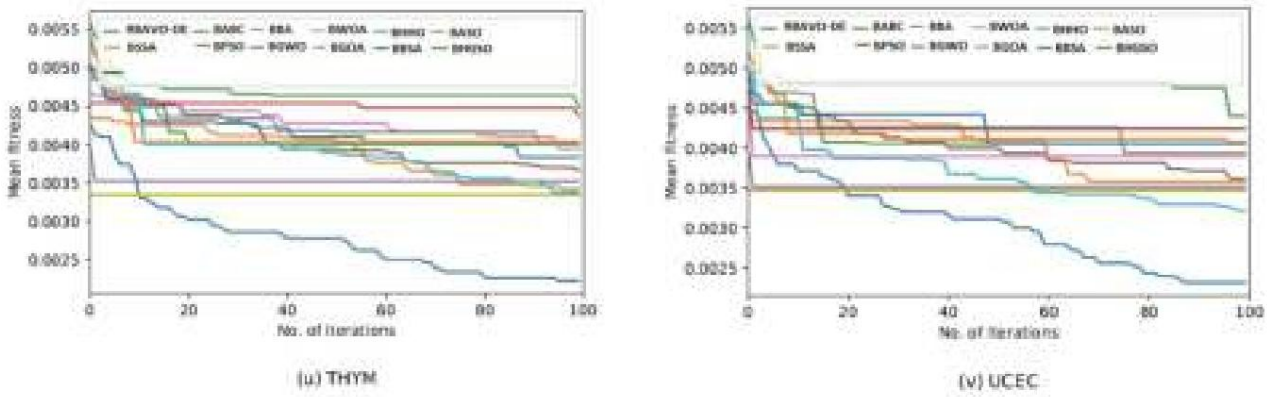


Рисунок A1. Графики сходимости предложенного алгоритма RBAVO-DE и конкурентных методов , использующих модель SVM, на общих тестах [57].

Приложение Б.2. Графики сходимости, использующие модель k-NN

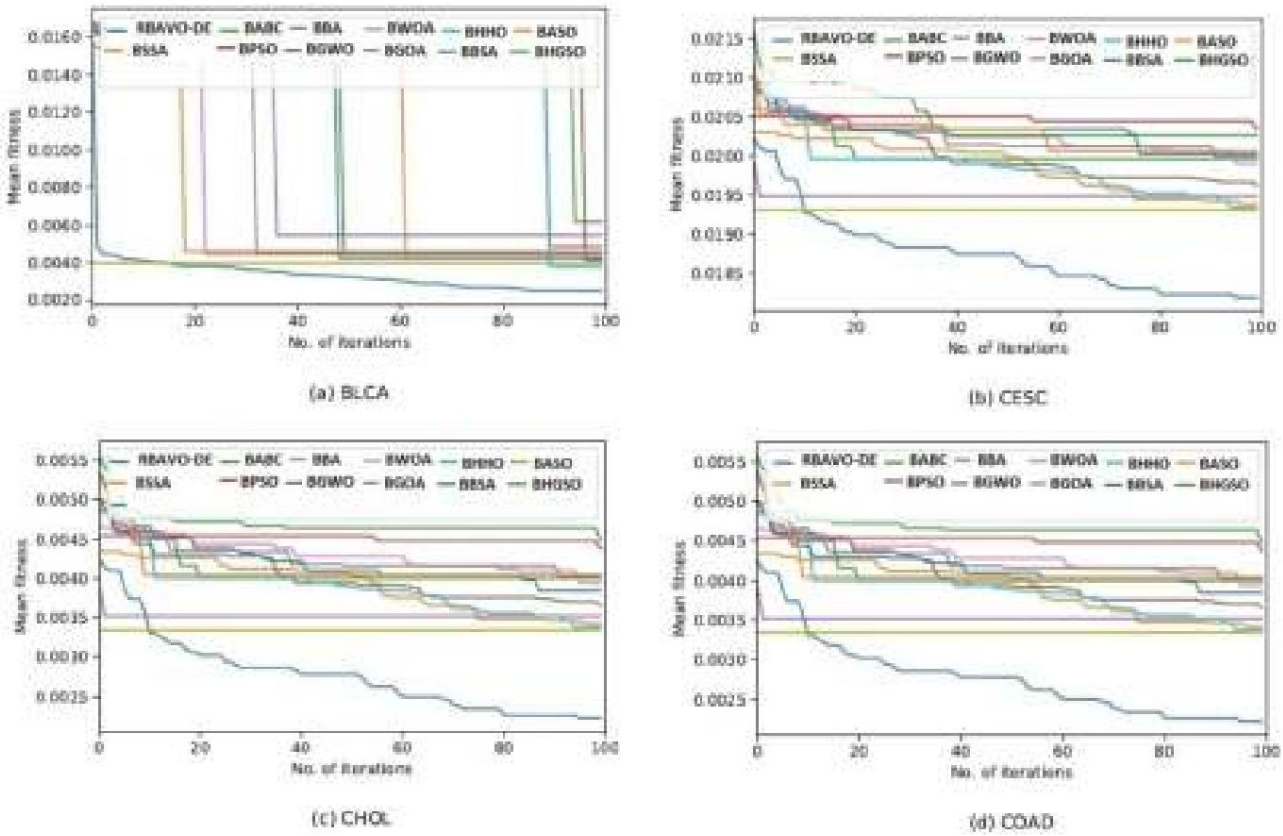


Рисунок A2. Продолжение

Machine Translated by Google

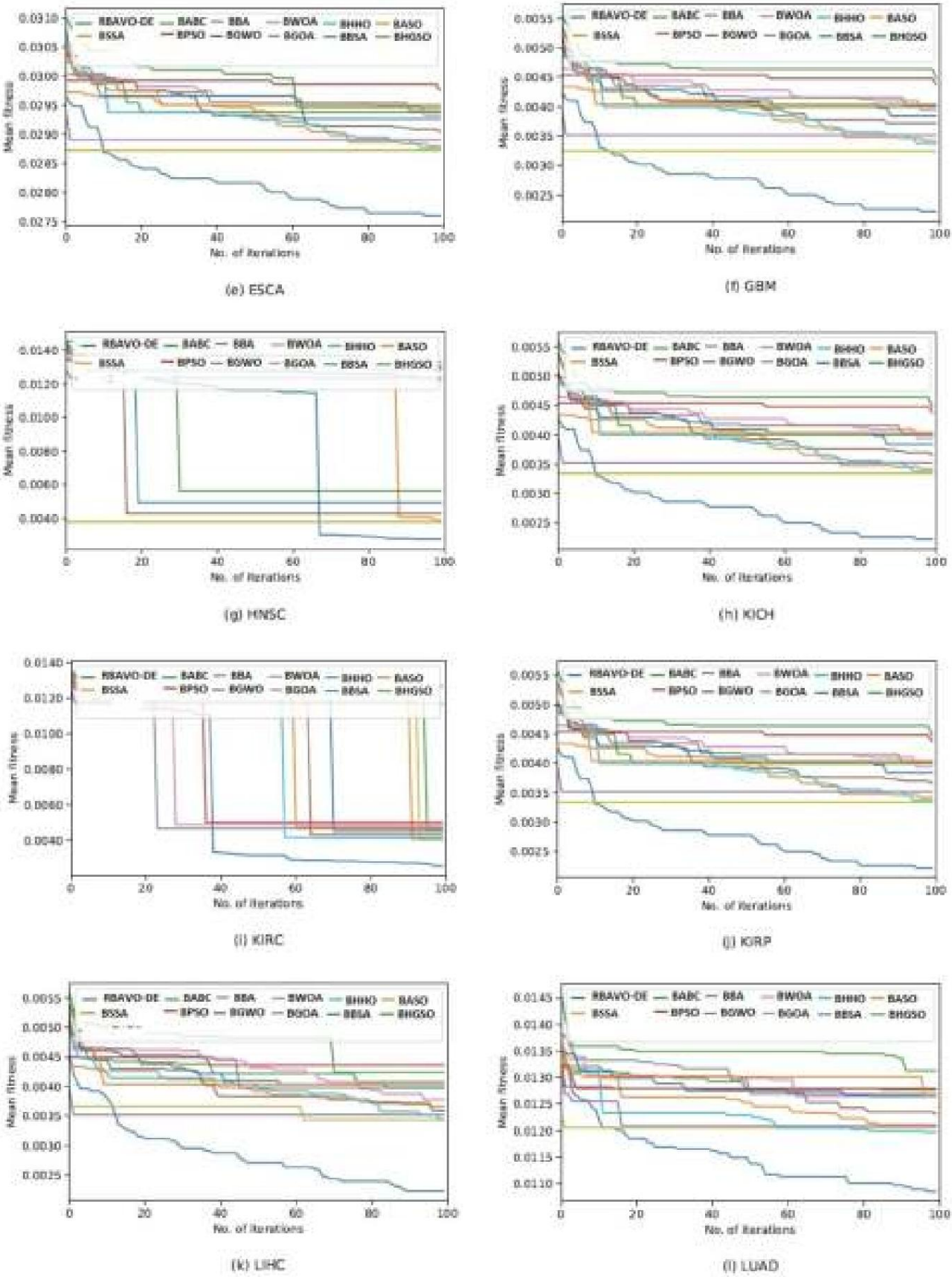
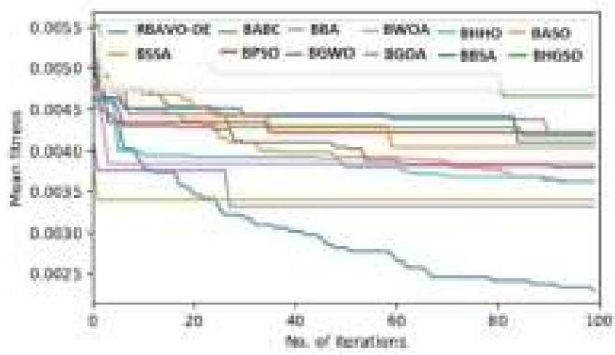
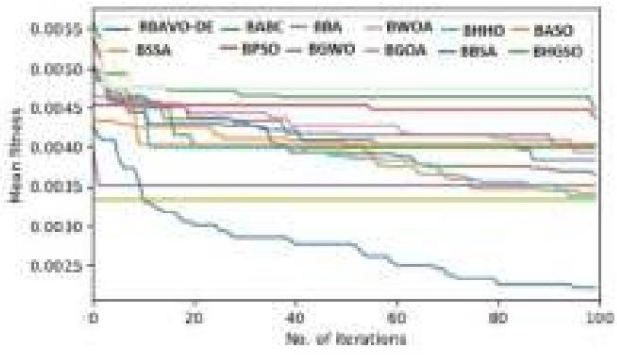


Рисунок А2. Продолжение

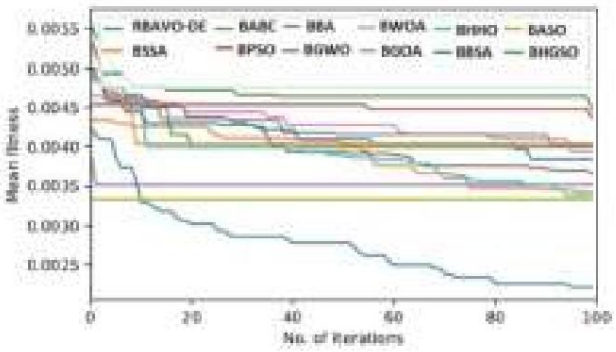
Machine Translated by Google



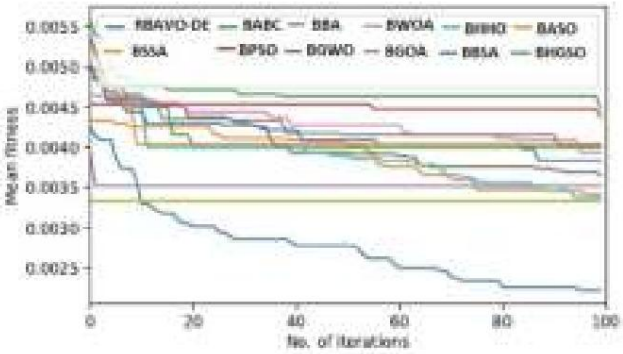
(m) LUSC



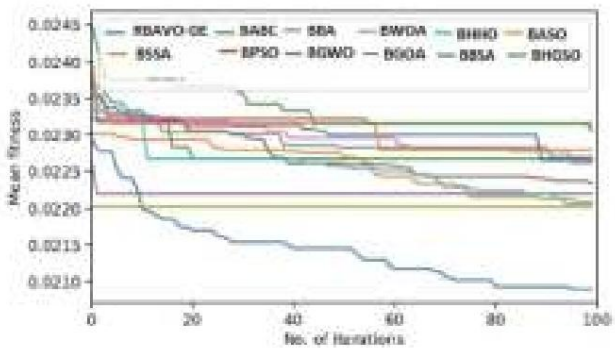
(n) PAAD



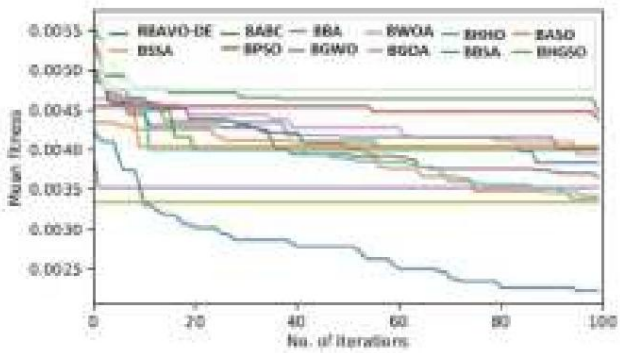
(o) PCPG



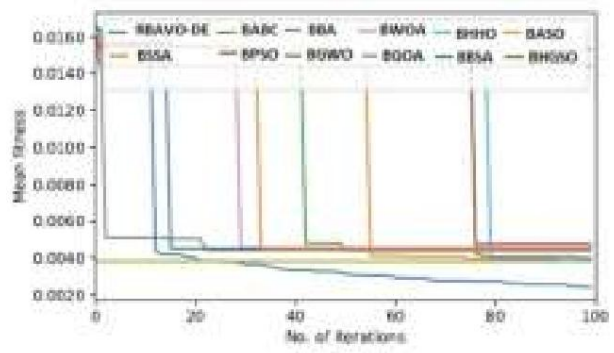
(p) READ



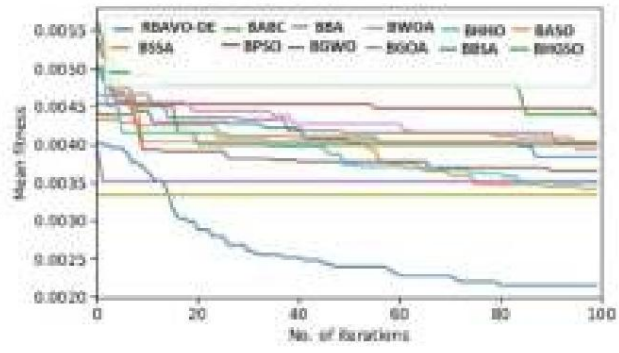
(q) SARC



(r) SKCM



(s) STAD



(t) THCA

Рисунок А2. Продолжение

Machine Translated by Google

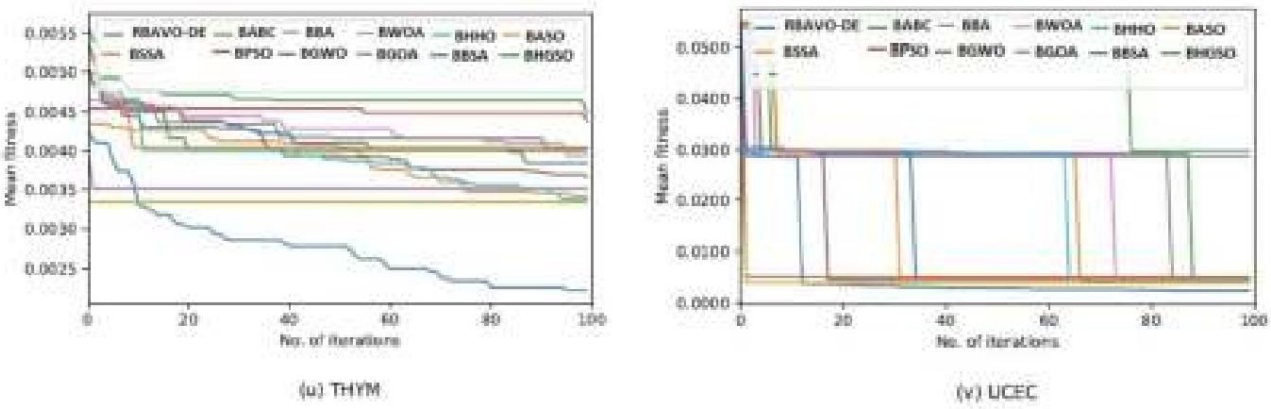


Рисунок А2. Графики сходимости предложенного алгоритма RBAVO-DE и конкурентных методов , использующих модель k-NN на общих тестах (продолжение) [57].

Ссылки

1. Эстрада-Меза, К.; Торрес-Копадо, А.; Лорети Гонсалес-Мельгоса, Л.; Руис-Манрикес, LM; Де Донато, М.; Шарма, А.; Патак, С.; Банерджи, А.; Пол, С. Недавние открытия в области микроРНК и длинной некодирующей РНК-опосредованной регуляции популяций стволовых клеток. 3 Биотех 2022, 12, 270. [CrossRef] [ПабМед]
2. Какати, Т.; Бхаттачарья, ДК; Калита, Дж.К.; Норден-Кричмар, Т.М. DEGnext: Классификация дифференциально экспрессируемых генов на основе данных секвенирования РНК с использованием сверточной нейронной сети с трансферным обучением. БМЦ Биоинформ. 2022, 23, 17. [CrossRef]
3. Чжао, С.; Фунг-Леунг, В.П.; Биттнер, А.; Нго, К.; Лю, Х. Сравнение RNA-Seq и микрочипов в профилировании транскриптома активированных Т-клеток. PLoS ONE 2014, 9, e78644. [Перекрестная ссылка]
4. Чен, З.; Луо, З.; Чжан, Д.; Ли, Х.; Лю, Х.; Чжу, К.; Чжан, Х.; Ван, З.; Чжоу, П.; Рен, Дж.; и др. ТИГР: Веб-портал, посвященный экспрессии генов иммунотерапии опухолей. Геном. Протеом. Биоинформ. 2023, 21, 337–348. [Перекрестная ссылка] [ПабМед]
5. Нуньес-Гарсия, Ж.; АбуОун, М.; Стори, Н.; Брауэр, М.; Дельгадо-Блас, Дж.; Мо, СС; Эллаби, Н.; Вельдман, К.; Хэнни, М.; Шатр, П.; и др. Гармонизация методов диагностики и наблюдения нового поколения, основанных на секвенировании in silico. наук. Отчет 2022, 12, 14372. [CrossRef]
6. Ван З.; Герштейн, М.; Снайдер, М. RNA-Seq: революционный инструмент для транскриптомики. Нат. Преподобный Жене. 2009, 10, 57–63. [Перекрестная ссылка]
7. Ким, В.Дж.; Чой, БР; Нет, Джей-Джей; Ли, ГГ; Ким, Ти Джей; Ли, Дж.В.; Ким, Б.Г.; Чой, СН Сравнение RNA-Seq и микрочипов в прогноз экспрессии белка и прогноз выживания. Передний. Жене. 2024, 15, 1342021. [CrossRef]
8. Ван, М.; Чен, Х.; Дай, Ю.; Ву, Д.; Лю, Ф.; Ян, З.; Песня, Б.; Се, Л.; Ян, Л.; Чжао, В.; и др. Согласованное исследование анализа геномного профилирования нового поколения на основе секвенирования 520 генов образцов тканей и плазмы. Мол. Диаг. Там. 2022, 26, 309–322. [Перекрестная ссылка]
9. Мецкер, ML. Технологии секвенирования — следующее поколение. Нат. Преподобный Жене. 2010, 11, 31–46. [Перекрестная ссылка] [ПабМед]
10. Панди, Д.; Онкара Перумал, П. Обзорный обзор глубокого обучения для RNA-Seq следующего поколения. анализ данных. Функц. Интегр. Геном. 2023, 23, 134. [CrossRef]
11. Лю, С.; Яо, В. Прогнозирование рака легких с использованием экспрессии генов и глубокого обучения с помощью выбора генов дивергенции KL. БМК Биоинформ. 2022, 23, 175. [CrossRef] [ПабМед]
12. Хуссейн, Э.Х.; Олива, Д.; Челик, Э.; Эмам, ММ; Гонием, Р.М. Усовершенствованный алгоритм оптимизации сажистой крачки для глобальной оптимизации и выбора функций. Экспертная система. Прил. 2023, 213, 119015. [CrossRef]
13. Джоши, А.А.; Азиз, Р.М. Двухфазный подход, основанный на поиске кукушки, для отбора генов и классификации онкологических заболеваний с использованием глубокого обучения с использованием данных об экспрессии генов с новой функцией приспособленности. Мультимед. Инструменты Прил. 2024, 83, 71721–71752. [Перекрестная ссылка]
14. Рамасвами, Р.; Кандхасами, П.; Паланисвами, С. Выбор признаков для данных об экспрессии генов болезни Альцгеймера с использованием модифицированного двоичного кода оптимизация роя частиц. IETE J. Res. 2023, 69, 9–20. [Перекрестная ссылка]
15. Кюи, Х.; Ли, Ю.; Фан, Дж.; Ван, Т. Новый алгоритм выбора характеристик фильтра, основанный на рельефе. Прил. Интел. 2022, 52, 5063–5081. [Перекрестная ссылка]
16. Альхенави, Э.; Аль-Сайед, Р.; Худаиб, А.; Мирджалили, С. Методы отбора признаков по данным микрочипов экспрессии генов для классификации рака: систематический обзор. Вычислить. Биол. Мед. 2022, 140, 105051. [CrossRef] [ПабМед]
17. Парлак, Б.; Уйсал, А.К. Новый метод выбора функций фильтра для классификации текста: расширенный селектор функций. Ж. Инф. наук. 2023, 49, 59–78. [Перекрестная ссылка]
18. Албулайхи, К.; Абу Аль-Хайджа, вопрос; Алсухибани, ЮАР; Джиллепалли, А.А.; Ашрафуззаман, М.; Шелдон, FT Обнаружение вторжений в Интернет вещей с использованием машинного обучения и нового высокопроизводительного метода выбора функций. Прил. наук. 2022, 12, 5015. [CrossRef]
19. Фатима А.; Назир, Т.; Назир, А.К.; Дин, AMU Эффективный метод выбора подмножества генов на основе дополнительной оболочки (IG на основе IWSSr) для распознавания опухолей. Мультимед. Инструменты Прил. 2024, 83, 64741–64766. [Перекрестная ссылка]
20. Каур, С.; Кумар, Ю.; Коул, А.; Кумар Камбой, С. Систематический обзор методов метаэвристической оптимизации для выбора признаков при диагностике заболеваний: открытые вопросы и проблемы. Арх. Вычислить. Методы англ. 2023, 30, 1863–1895. [Перекрестная ссылка]

Machine Translated by Google

21. Абд Эль-Магид, А.А.; Гад, АГ; Саллам, КМ; Мунасингхе, К.; Абохани, А.А. Улучшенный алгоритм двоичной адаптивной ветровой оптимизации на основе уменьшения размерности для контролируемой классификации. Вычислить. Индийский англ. 2022, 167, 107904. [Перекрестная ссылка]

22. Гад, А.Г.; Саллам, КМ; Чакраборти, РК; Райан, МЖ; Абохани, А.А. Улучшенный алгоритм двоичного поиска признаков выбор в классификации данных. Нейронный компьютер. Прил. 2022, 34, 15705–15752. [Перекрестная ссылка]

23. Хусейн, Р.М.; Абохани, А.А.; Абд Эль-Магид, А.А.; Хосни, К.М. Улучшенный алгоритм оптимизации двоичных суриката для эффективного выбора признаков классификации обучения с учителем. Информационная система. 2024, 292, 111616. [CrossRef]

24. Абд Эль-Магид, А.А.; Абохани, А.А.; Элашри, А. Эффективная стратегия выбора признаков для контролируемой классификации на основе Улучшенный алгоритм оптимизации двоичного Aquila. Вычислить. Индийский англ. 2023, 181, 109300. [CrossRef]

25. Инь, Ю.; Джанг-Жаккар, Дж.; Сюй, В.; Сингх, А.; Чжу, Дж.; Сабрина, Ф.; Квак, Дж. IGRF-RFE: Гибридный метод выбора функций для Обнаружение сетевых вторжений на основе MLP в наборе данных UNSW-NB15. Дж. Большие данные 2023, 10, 15. [CrossRef]

26. Накао, Х.; Имаока, М.; Хида, М.; Имаи, Р.; Накамура, М.; Мацумото, К.; Кита, К. Определение отдельных факторов, связанных при вальгусной деформации с помощью SVM-RFE. БМК Опорно-двигательный аппарат. Разлад. 2023, 24, 534. [CrossRef] [ПабМед]

27. Сарафрази, С.; Незамабади-Пур, Х. Столкновение с классификацией бинарных задач с помощью гибридной системы GSA-SVM. Математика. Вычислить. Модель. 2013, 57, 270–278. [Перекрестная ссылка]

28. Каденас, Дж. М.; Гарридо, МС; Мартинес, Р. Фильтр-обертка выбора подмножества функций на основе данных низкого качества. Экспертная система. Прил. 2013, 40, 6241–6252. [Перекрестная ссылка]

29. Ой, ЕСТЬ; Ли, Дж. С.; Мун, Б.Р. Гибридные генетические алгоритмы для выбора признаков. IEEE Транс. Паттерн Анал. Мах. Интел. 2004, 26, 1424–1437.

30. Абдоллахзаде, Б.; Гарехчопох, ФС; Мирджалили, С. Алгоритм оптимизации африканских стервятников: новый метаэвристический алгоритм, вдохновленный природой, для глобальных задач оптимизации. Вычислить. Индийский англ. 2021, 158, 107408. [CrossRef]

31. Эль-Шафей, Э.; Хассаниен, А.Е.; Саллам, КМ; Абохани, А. Подход к обучению квантовой нейронной сети для прогнозирования тяжести COVID-19 у пациентов. Вычислить. Матер. Продолжение. 2020, 66, 1745–1755. [Перекрестная ссылка]

32. Якуб А.; Верма, Северная Каролина; Азиз, Р.М. Оптимизация отбора генов и классификации рака с помощью гибридного синус-косинуса и алгоритма поиска кукушки. Дж. Мед. Сист. 2024, 48, 10. [CrossRef]

33. Джоши, А.А.; Азиз, Р.М. Подход глубокого обучения для классификации опухолей головного мозга с использованием метаэвристической оптимизации с использованием генов данные выражения. Межд. Дж. Система обработки изображений. Технол. 2023, 34, e23007. [Перекрестная ссылка]

34. Махто, Р.; Ахмед, С.У.; Рахман, Ру; Азиз, РМ; Рой, П.; Маллик, С.; Ли, А.; Шах, Массачусетс. Новая и инновационная система классификации рака, основанная на последовательном использовании гибридного отбора признаков. БМЦ Биоинформ. 2023, 24, 479. [CrossRef] [ПабМед]

35. Неггаз, Н.; Неггаз, И.; Абд Элазиз, М.; Хассиен, АГ; Абулай, Л.; Дамашевичюс, Р.; Ху, Г. Повышение эффективности оптимизатора поиска пищи скатами манты с помощью операторов тригонометрии: тематическое исследование на наборе медицинских данных. Нейронный компьютер. Прил. 2024, 36, 9405–9436. [Перекрестная ссылка]

36. Лю, Б.; Хак, А. Классификация типов опухолей на основе глубокого обучения с использованием данных об экспрессии генов. В материалах Международной конференции ACM по биоинформатике, вычислительной биологии и медицинской информатике 2018 г., Вашингтон, округ Колумбия, США, 29 августа – 1 сентября 2018 г.; стр. 89–96.

37. Халифа, НЕМ; Таха, МХН; Али, Делавэр; Словик, А.; Хассаниен, А.Е. Техника искусственного интеллекта для экспрессии генов по данным секвенирования РНК опухоли: новый оптимизированный подход к глубокому обучению. IEEE Access 2020, 8, 22874–22883. [Перекрестная ссылка]

38. Диллис, Массачусетс; Рау, А.; Обер, Дж.; Эннеке-Антье, К.; Жанмуген, М.; Слуга, Н.; Кейме, К.; Маро, Г.; Кастель, Д.; Эстель, Дж.; и др. Комплексная оценка методов нормализации для анализа данных высокопроизводительного секвенирования РНК Illumina. Брифинги Биоинформ. 2013, 14, 671–683. [Перекрестная ссылка]

39. Сяо, Ю.; Ву, Дж.; Лин, З.; Чжао, Х. Многомодельный ансамблевый метод прогнозирования рака на основе глубокого обучения. Вычислить. Методы Программы Биомед. 2018, 153, 1–9. [Перекрестная ссылка]

40. Лю, М.; Сюй, Л.; Йи, Дж.; Хуанг, Дж. Метод отбора генов по признакам, основанный на ReliefF и PSO. В материалах 10-й Международной конференции по измерительным технологиям и мехатронике (ICMTMA) 2018 г., Чанша, Китай, 7–8 января 2018 г.; IEEE: Нью-Йорк, штат Нью-Йорк, США, 2018 г.; стр. 298–301.

41. Кононенко И. Оценка атрибутов: Анализ и расширение РЕЛЬЕФ. В материалах Европейской конференции по машинам Обучение, Катания, Италия, 6–8 апреля 1994 г.; Springer: Берлин/Гейдельберг, Германия, 1994 г.; стр. 171–182.

42. Фарис, Х.; Мафарджа, ММ; Хейдари, А.А.; Альджара, И.; Ала'М, Аризона; Мирджалили, С.; Фудзита, Х. Эффективный алгоритм двоичного роя сальпов со схемой кроссовера для задач выбора функций. Информационная система. 2018, 154, 43–67. [Перекрестная ссылка]

43. Абдель-Бассет, М.; Дин, В.; Эль-Шахат, Д. Гибридный алгоритм оптимизации Харриса Хокса с имитацией отжига для выбора признаков. Артиф. Интел. Ред. 2020 г., 54, 593–637. [Перекрестная ссылка]

44. Сторн Р.; Прайс, К. Дифференциальная эволюция — простая и эффективная эвристика для глобальной оптимизации в непрерывных пространствах. Дж. Глоб. Оптим. 1997, 11, 341–359. [Перекрестная ссылка]

45. Карабога, Д.; Бастурк, Б. Мощный и эффективный алгоритм численной оптимизации функций: алгоритм искусственной пчелиной семьи (ABC). Дж. Глоб. Оптим. 2007, 39, 459–471. [Перекрестная ссылка]

46. Мирджалили С.; Гандоми, АХ; Мирджалили, СЗ; Сареми, С.; Фарис, Х.; Мирджалили, С.М. Алгоритм Salp Swarm: биооптимизатор для решения задач инженерного проектирования. Адв. англ. Программное обеспечение 2017, 114, 163–191. [Перекрестная ссылка]

47. Ханесар, Массачусетс; Тешнехлаб, М.; Шурехдели, М.А. Новая оптимизация роя бинарных частиц. В материалах Средиземноморской конференции по контролю и автоматизации 2007 г., Афины, Греция, 27–29 июня 2007 г.; IEEE: Нью-Йорк, штат Нью-Йорк, США, 2007 г.; стр. 1–6.

Machine Translated by Google

48. Мирджалили С.; Мирджалили, С.М.; Ян, XS Бинарный алгоритм летучей мыши. Нейронный компьютер. Прил. 2014, 25, 663–681. [Перекрестная ссылка]
49. Эмари, Э.; Заубаа, НМ; Хассаниен, А.Е. Подходы к оптимизации бинарного серого волка для выбора признаков. Нейрокомпьютинг 2016, 172, 371–381. [Перекрестная ссылка]
50. Хичем, Х.; Элькамель, М.; Рафик, М.; Месауд, Монтана; Уахиба, К. Новый алгоритм оптимизации двоичного кузнечика для задачи выбора признаков. Университет Дж. Короля Сауда. Вычислить. Инф. наук. 2022, 34, 316–328. [Перекрестная ссылка]
51. Хуссейн, А.Г.; Хассаниен, А.Е.; Хусейн, Э.Х.; Бхаттачарья, С.; Амин, М. S-образный алгоритм оптимизации двоичных китов для выбора признаков. В последних тенденциях в обработке сигналов и изображений: ISSIP 2017; Springer: Берлин/Гейдельберг, Германия, 2019 г.; стр. 79–87.
52. Чжао, В.; Ван, Л.; Чжан, З. Оптимизация поиска атомов и ее применение для решения задачи оценки гидрогеологических параметров. Информационная система. 2019, 163, 283–304. [Перекрестная ссылка]
53. Мэн, XB; Гао, XZ; Лу, Л.; Лю, Ю.; Чжан, Х. Новый алгоритм оптимизации, основанный на биотехнологиях: алгоритм роя птиц. Дж. Эксп. Теор. Артиф. Интел. 2016, 28, 673–687. [Перекрестная ссылка]
54. Хашим, Ф.А.; Хусейн, Э.Х.; Мабрук, М.С.; Аль-Атабани, В.; Мирджалили, С. Генри, оптимизация растворимости газов: новая физика. основанный алгоритм. Будущий генерал. Вычислить. Сист. 2019, 101, 646–667. [Перекрестная ссылка]
55. Хейдари, А.А.; Мирджалили, С.; Фарис, Х.; Альджара, И.; Мафарджа, М.; Чен, Х. Харрис занимается оптимизацией: алгоритм и приложения. Будущий генерал. Вычислить. Сист. 2019, 97, 849–872. [Перекрестная ссылка]
56. Набор данных по экспрессии генов PHK-Seq нормализованного уровня 3. Доступно онлайн: https://gdac.broadinstitute.org/. (доступ 20 декабрь 2023 г.).
57. Эль-Магид, ААА; Эльхули, А.Е.; Абохани, А.А.; Гафар, М. Отбор генов с помощью улучшенного алгоритма оптимизации ядерных реакций. для классификации рака в многомерных данных. J. Большие данные 2024, 11, 46. [CrossRef]
58. Мафарджа, М.; Мирджалили, С. Кит. Подходы к оптимизации выбора функций оболочки. Прил. Мягкий компьютер. 2018, 62, 441–453. [Перекрестная ссылка]
59. Тахер, Т.; Хейдари, А.А.; Мафарджа, М.; Донг, Дж.С.; Мирджалили, С. Бинарный оптимизатор Harris Hawks для многомерного выбора функций с малым размером выборки. В эволюционных методах машинного обучения; Шпрингер: Берлин/Гейдельберг, Германия, 2020 г.; стр. 251–272.
60. Деррак, Дж.; Гарсия, С.; Молина, Д.; Эррера, Ф. Практическое руководство по использованию непараметрических статистических тестов в качестве методологии сравнения алгоритмов эволюционного и роевого интеллекта. Рой Эвол. Вычислить. 2011, 1, 3–18. [Перекрестная ссылка]
61. Код Python для отбора генов посредством оптимизации рельефа бинарных африканских стервятников, интегрированной с дифференциальной эволюцией. Доступный онлайн: https://github.com/D-Amr-Atef/Gene_Selection_RBAVO-DE.git (по состоянию на 18 мая 2024 г.).

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Обзор

Обзор обучения с подкреплением при планировании производства: Инференциальная перспектива

Владимир Модрак 1,*, Ранджитарамасами Судхакарапандиан 2, Арунможи Баламуруган 2 и Зузана Солтысова 1

1 Факультет производственных технологий, Технический университет Кошице, 080 01 Прешов, Словакия;
2 zuzana.soltysova@tuke.sk Школа машиностроения, Технологический институт Веллора, Веллор 632014, Тамил Наду, Индия; sudhakarapandian.r@vit.ac.in (PC); arunmozhi.b@vit.ac.in (AB)
* Для переписки: vladimir.modrak@tuke.sk.

Аннотация: В этом исследовании был проведен систематический обзор планирования производства на основе методов обучения с подкреплением (RL) с использованием, в частности, библиометрического анализа. Цель этой работы, среди прочего, отметить растущий интерес к этой области и обрисовать влияние RL как типа машинного обучения на планирование производства. Для достижения этой цели в статье исследуется планирование производства с использованием RL путем исследования описательных метаданных соответствующих публикаций, содержащихся в базах данных Scopus, ScienceDirect и Google Scholar. Исследование сосредоточено на широком спектре публикаций, охватывающих период с 1996 по 2024 год. Результаты этого исследования могут послужить новым пониманием для будущих исследований в области планирования производства с использованием методов RL.

Ключевые слова: библиометрический анализ; планирование производства; обучение с подкреплением

check for updates

Цитирование: Модрак В.; Судхакарапандиан, Р.; Баламуруган, А.; Солтысова З. Обзор обучения с подкреплением в планировании производства : логический подход. Алгоритмы 2024, 17, 343. https://doi.org/10.3390/a17080343 .

Академические редакторы: Александр Долги, Давид Лемуан, Мария И. Рестрепо. и Фрэнк Вернер

Поступила: 6 июня 2024 г.
Пересмотрено: 28 июля 2024 г.
Принято: 2 августа 2024 г.
Опубликовано: 7 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария.
Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Планирование производства рассматривается как один из наиболее важных элементов управления производством, позволяющий привести производственную деятельность в соответствие с бизнес-целями, обеспечить бесперебойное движение товарных ресурсов и поддержать способность компании оставаться конкурентоспособной на рынке. Алгоритмы планирования играют важную роль в повышении эффективности и результативности производства и поэтому уже давно являются предметом обширных исследований в различных междисциплинарных областях, таких как промышленное проектирование, автоматизация и наука об управлении [1]. Задачи планирования производства могут быть решены с использованием трех основных типов пошаговых процедур, таких как точные алгоритмы, эвристические алгоритмы и метаэвристические алгоритмы [2,3]. Хотя точный алгоритм теоретически может гарантировать оптимальное решение, NP-трудность основных проблем делает невозможным их эффективное и результативное решение [4]. Эвристика использует набор правил для быстрого и эффективного создания решений по планированию без учета глобальной оптимизации. Более того, создание правил во многом зависит от тщательного понимания особен В то время как метаэвристика может создавать хорошие решения по планированию за разумное количество вычислительного времени, способ их создания операторами поиска существенно зависит от конкретной ситуации [9–13]. Кроме того, итеративный процесс поиска создает проблемы с точки зрения затрат времени и применимости в сценариях реального времени при решении крупномасштабных задач. Подходы к планированию, основанные на обучении с подкреплением (RL), оказались полезным инструментом в этом отношении. Обучение с подкреплением — это подобласть более широкой области машинного обучения. RL считается одним из наиболее перспективных подходов к созданию надежного совместного планирования, который позволяет руководителям производства взаимодействовать со сложной производственной средой, учиться на предыдущем опыте и выбирать оптимальные решения. Он включает в себя процесс, когда агент автономно выбирает и выполняет действия для выполнения заданной задачи. Агент учится на основе опыта

Machine Translated by Google

награды, которые он получает в определенных сценариях. Основная цель RL — оптимизировать совокупное вознаграждение, получаемое агентом за счет оценки и выбора действий в динамической среде [14–20]. Самые последние разработки в технологии искусственного интеллекта позволили успешно применить RL в задачах последовательного принятия решений с несколькими целями, которые можно использовать для планирования и управления роботами [21,22]. Исследования планирования производства с использованием RL с 1998 года развиваются как усовершенствованные методы оптимизации по сравнению с метаэвристикой. RL значительно повышает вычислительную эффективность решения задач планирования. С момента его создания в 1996 г. в журнале RL было опубликовано множество исследовательских статей по планированию производства, заложивших существенную и ценную основу (см. некоторые из них, например, [23–27]), но только две недавние обзорные статьи [28,29] принесли пользу. внимание на эту тему. Однако обе статьи, опубликованные в 2021 году, были посвящены другим точкам зрения на алгоритмы планирования на основе RL, чем представленные в этой статье. Другими словами, содержание предлагаемой статьи находится в дизъюнктивном и дополняющем отношении к двум упомянутым исследованиям.

Исследовательский вопрос этой статьи многогранен, включая множество особенностей этой области, которые требуют отдельных ответов. Основные темы, которые будут рассмотрены в этой статье: Каковы основные новые области исследований в этой области? Какие связанные темы охватываются планированием производства на основе RL? Каковы типичные области реализации в рамках RL, применяемые к планированию производства?

Его новизна заключается в предоставлении дополнительного анализа и оценок относительно RL при планировании производства. Он включает в себя, например, тенденцию цитирования RL по планированию производства, наиболее влиятельных авторов в этой области, наиболее важные источники в этой области, сравнение детерминированных типов методов планирования и неопределенных типов методов планирования.

2. Материалы и методы.

Используемый здесь библиометрический подход в качестве основного метода исследования представляет собой количественный инструмент мониторинга и представления научного прогресса путем изучения и визуализации научной информации. Растущее признание библиометрических методов в ряде академических дисциплин указывает на то, что их использование приносит ожидаемые результаты [30–33]. Настоящее исследование планирования производства на основе RL проводится с использованием процедуры, состоящей из пяти последовательных этапов, как показано на рисунке 1.

Более того, это исследование следует логическому подходу, при котором выборка населения исследуется для определения ее характеристик [34,35]. Более того, для формулирования результатов исследования здесь была применена инференциальная концепция научного представления, предложенная Суаресом [36] . Его суть заключается в использовании альтернативных рассуждений для достижения результатов, которые отличаются от изоморфного взгляда на научное представление в том смысле, что эмпирические знания играют важную роль в индуктивных рассуждениях [37,38].

Machine Translated by Google

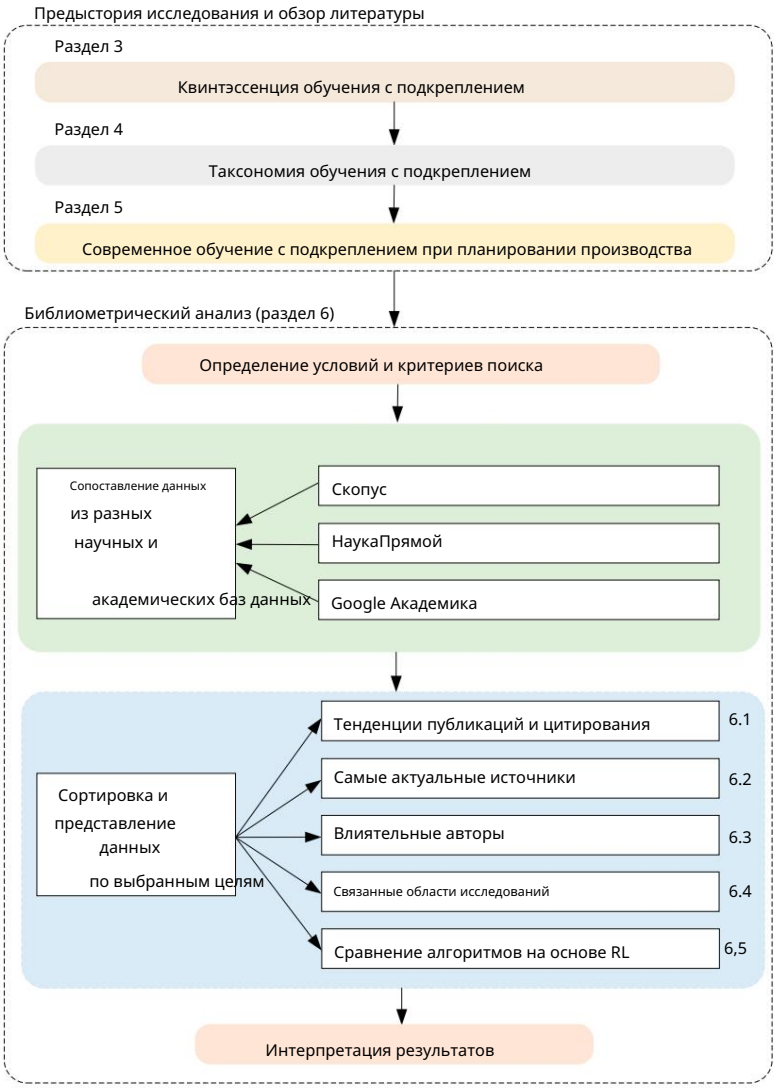


Рисунок 1. Структура методологии исследования.

3. Квинтэссенция обучения с подкреплением

Искусственный интеллект (ИИ) стал неотъемлемым и важным аспектом нашей повседневной жизни по мере приближения к концу этого десятилетия. Искусственный интеллект существует уже семь десятилетий, но за последние три десятилетия он набрал значительный оборот, что привело к обширным исследованиям в этой области [39]. Искусственный интеллект относится к различным методам , которые позволяют компьютерам приобретать знания и делать выбор путем анализа данных. Из этих методов машинное обучение (МО) добилось значительных успехов за последние два десятилетия, превратившись из простого любопытства в лабораториях в широко используемую практическую технологию в коммерческих приложениях. В сфере искусственного интеллекта машинное обучение стало предпочтительным подходом для создания функционального программного обеспечения в таких областях, как компьютерное зрение, обработка естественного языка, распознавание речи, управление роботами и ряд других приложений [40].

Многие разработчики ИИ теперь признают, что в некоторых случаях удобнее обучать систему, демонстрируя идеальное поведение ввода-вывода, а не программировать ее вручную, чтобы предвидеть желаемый ответ для каждого потенциального ввода. Машинное обучение существенно повлияло на различные области информатики и отрасли, занимающиеся проблемами, связанными с данными. Оно включает в себя обслуживание потребителей, диагностику дефектов сложных систем и управление логистическими цепочками. Подходы МО оказали различное влияние на несколько эмпирических наук, ффтаких как биология, космология и социальные науки. Эти подходы были творчески использованы для изучения обширных экспериментальных данных [41].

Machine Translated by Google

Тремя основными парадигмами МО являются обучение с учителем, обучение без учителя и обучение с подкреплением [15]. Обучение с учителем предполагает процесс обучения модели с использованием помеченного набора данных, что означает, что каждый вход связан с соответствующим точным выходом. Цель состоит в том, чтобы получить знания о преобразовании данных входных данных в соответствующие выходные данные, которые затем можно использовать для составления точных прогнозов на основе новых, ненаблюдаемых данных. Обучение без учителя предполагает выявление скрытых закономерностей или внутренних структур в неразмеченных входных данных. Модель направлена на получение знаний о фундаментальном распределении или группировке данных без явных указаний о желаемом результате [28]. RL сосредоточен на обучении агента делать широкий выбор, взаимодействуя с окружающей средой и получая вознаграждения или наказания в качестве обратной связи . Агент приобретает способность оптимизировать свою деятельность, чтобы максимизировать общее вознаграждение, которое он получает за период [42].

Благодаря своей динамической и адаптивной методологии RL очень эффективен для решения проблем в области исследований производственных систем и робототехники [22,28]. RL отличается от обучения с учителем, когда модель обучается на заранее определенном наборе данных, и обучения без учителя, когда модель обнаруживает скрытые закономерности в данных. При обучении с подкреплением агент обучается принимать ряд решений, взаимодействуя с окружающей средой. Целью агента является приобретение стратегии, иногда называемой политикой, которая оптимизирует общие вознаграждения, получаемые с течением времени [43].

RL — это процесс принятия решений, на который влияет поведенческая психология. Он предполагает, что агент учится достигать цели, действуя и получая обратную связь от окружающей среды посредством вознаграждений или наказаний. Этот метод проб и ошибок сродни тому, как люди и животные учатся у окружающей среды. Важнейшими компонентами RL являются агент, среда, действия, состояния и вознаграждения. Агент наблюдает за окружающей средой, предпринимает действия и получает вознаграждение, которое используется для руководства будущими действиями [14]. RL особенно эффективен в задачах с неопределенным результатом, таких как игры, управление роботами и управление ресурсами. В этих ситуациях агент должен сбалансировать исследование (пробование новых действий) и эксплуатацию (использование установленных действий, которые приводят к значительному вознаграждению), чтобы эффективно разработать оптимальную политику [42].

Математическая основа RL обычно представляется с использованием марковских процессов принятия решений (MDP), которые предлагают систематический подход к моделированию проблем принятия решений , включая как случайные, так и контролируемые агентом результаты. Решение MDP влечет за собой определение политики, которая предписывает оптимальные действия, которые необходимо предпринять на каждом этапе, с целью максимизировать прогнозируемые совокупные выгоды в будущем. Недавний прогресс в глубоком обучении значительно улучшил возможности RL [15,16]. Сочетание глубоких нейронных сетей и алгоритмов RL привело к появлению глубокого обучения с подкреплением (DRL), которое продемонстрировало исключительные результаты в сложных ситуациях. Яркие примеры включают AlphaGo, разработанную Google, которая одержала победу над чемпионами среди людей в игре Го, а также различные реализации в области автономного вождения, где алгоритмы DRL приобретают способность успешно ориентироваться в сложной и постоянно меняющейся среде. Обучение с подкреплением остается динамичной и быстро развивающейся областью машинного обучения, где текущие исследования постоянно расширяют границы возможностей искусственных агентов. Углубляясь в сложности RL, мы обнаруживаем новые возможности для интеллектуальных систем, способных приобретать знания и корректировать свое поведение в режиме реального времени. RL прокладывает путь в будущее, в котором компьютеры смогут беспрепятственно взаимодействовать с людьми и функционировать автономно более продвинутыми с

4. Таксономия алгоритмов обучения с подкреплением

Саттон и Барто [42] подчеркивают решающую роль трех основных компонентов в процессе RL: • Политика

определяет действия, которые необходимо предпринять в каждом состоянии окружающей среды. • Сигнал вознаграждения классифицирует эти действия как полезные или вредные в зависимости от непосредственного результата перехода между состояниями.

Machine Translated by Google

• Функция ценности оценивает долгосрочную эффективность действий, учитывая не только немедленное вознаграждение государства, но и ожидаемое вознаграждение в будущем.

Основная задача агента — оптимизировать совокупное вознаграждение. Следовательно, сигнал вознаграждения является основой для изменения политики и функции ценности. В конкретных системах RL существует дополнительный компонент, известный как модель среды, который не является обязательным. Он воспроизводит поведение климата или позволяет сделать более полные предположения о поведении окружающей среды. Модели используются для стратегического планирования, которое предполагает принятие решений, основанных на прогнозировании будущих сценариев до их фактического возникновения [24]. Методы, основанные на моделях, используются для решения задач обучения с подкреплением с использованием моделей и планирования, в отличие от более простых методов без моделей. Они демонстрируют очевидное поведение при обучении методом проб и ошибок и считаются противоположностью планирования [26,27]. Процессы RL можно представить и проанализировать с помощью математической структуры, называемой MDP. Стохастическая математическая модель может быть формализована с использованием пятикортежа (s, a, p, r, γ), в котором первый символ «s» представляет собой конечный набор всех возможных состояний среды, а st — состоян «a» представляет собой набор всех возможных действий, а действие, выполненное в момент времени «t», отмечено как . Символ «p» относится к матрице вероятностей перехода, которая определяет условную вероятность перехода в новое состояние «s» с наградой «r», учитывая текущее состояние «s» и действие «a» (для всех состояний и действия) [44]. На рисунке 2 показан RL цикла планирования производства.

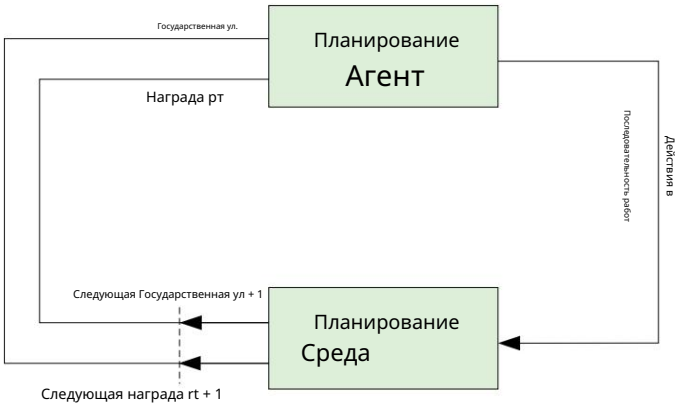


Рисунок 2. Цикл планирования производства RL.

Методологии обучения алгоритмов или агентов RL можно разделить на две отдельные категории: на основе моделей и без моделей. RL на основе моделей часто называют непрямым обучением. Агент использует прогнозирующую модель для изучения политики управления из среды посредством ограниченного количества взаимодействий. Затем агент применяет эту модель к последующим эпизодам для получения вознаграждений [15]. RL без модели, или прямое обучение, относится к обучению, при котором агент учится принимать решения без явного построения модели среды. Агент приобретает знания о политике контроля посредством экспериментального обучения в окружающей среде, используя методы проб и ошибок для оптимизации вознаграждений, не полагаясь на какую-либо ранее существовавшую модель. Этот подход демонстрирует адаптивность алгоритмов RL, позволяя им учиться и развиваться в динамических средах [45].

Поскольку существует множество различных алгоритмов RL, разумно понять разницу между ними. С этой целью были созданы системы классификации, позволяющие классифицировать их по различным критериям. Например, А.Л.Махамид и Гролингер [46] предложили классифицировать алгоритмы RL на основе типа среды. Алгоритмы RL также можно классифицировать с точки зрения политики: обучение в соответствии с политикой и обучение вне политики [47,48]. RL в основном классифицируются в соответствии подходами к обучению [15,45]. В связи с этим обновленная классификация алгоритмов RL с точки зрения используемых подходов к обучению представлена на рисунке 3.

Machine Translated by Google

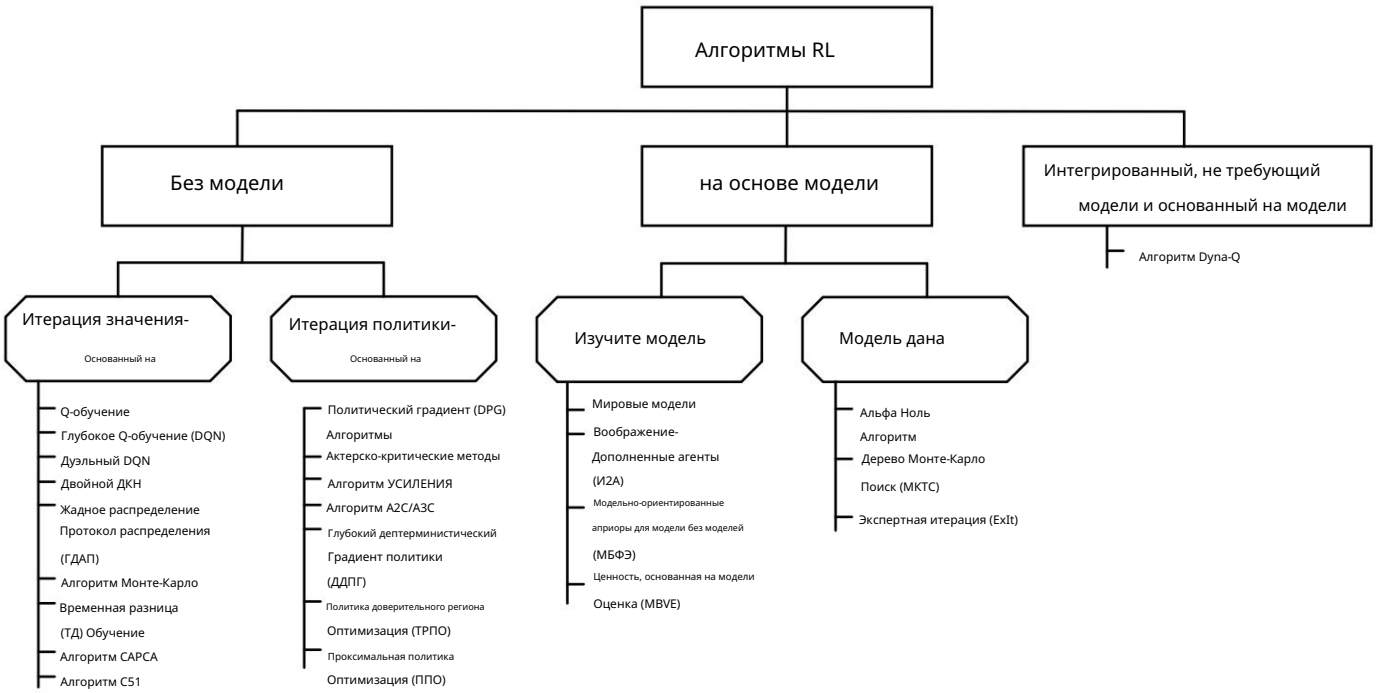


Рисунок 3. Классификация алгоритмов RL.

5. Современное состояние обучения с подкреплением при планировании производства

Детерминированное планирование включает в себя организацию и планирование задач при фиксированных параметрах и известных ограничениях. Его цель – оптимизировать эффективность производства и минимизировать сроки оборота. Обширная работа [49] исследует фундаментальные принципы и передовые методы детерминированного планирования, давая представление о различных алгоритмах.
и их приложения. Работа Пинедо [50] подробно рассматривает детерминированное планирование.
теория, алгоритмы и практические системы, подчеркивая их значение в производстве
и сервисные операции. Планирование Flowshop включает в себя последовательность операций на нескольких
машины в фиксированном порядке, чтобы минимизировать время изготовления или общее время завершения. Аллахверди [51]
обсуждает различные проблемы планирования потокового цеха, включая настройки, и рассматривает алгоритмы.
и подходы к их решению. Панвалкар и Смит [52] представили плодотворный обзор.
классических и современных исследований планирования потоков, охватывающих как точные, так и
эвристические методы.

Методы обучения с подкреплением (RL) приобрели популярность, подтвердив интерес к агентным моделям. Предыдущие исследования в основном были сосредоточены на использовании RL для
решить проблемы планирования рабочего времени. Производственный сектор сталкивается с такими проблемами, как
такие как удовлетворенность клиентов, деградация системы, устойчивость, запасы и эффективность,
влияющие на устойчивость и прибыльность предприятия. Индустрия 4.0 и интеллектуальное производство
решения для оптимизации операций и высококачественной продукции. Парашос и др. [53] интегрировать RL с бережливым
экологическим производством для создания устойчивой производственной среды,
снижение воздействия на окружающую среду за счет минимизации расхода материалов и жизненного цикла
расширение за счет вытягивающего производства, профилактического обслуживания и практики экономики замкнутого цикла.
Строгий экспериментальный анализ подтверждает его эффективность в повышении устойчивости и
повторное использование материала.

В последнее время значительный прогресс был достигнут в использовании RL для решения нескольких задач комбинаторной оптимизации, включая планирование производства, задачу маршрутизации транспортных средств и
Задача коммивояжера [54–59]. В RL задачу планирования производства можно рассматривать как
среду, в которой действует агент, разрабатывая политику посредством офлайн-обучения
взаимодействуя с этой средой. Этот подход предлагает новый подход к планированию.
проблемы, требующие строгих ограничений в реальном времени, таких как динамическое планирование цеха
проблемы [22,60–64].

Machine Translated by Google

В задачах планирования производства обычно используются алгоритмы RL на основе значений, включая Q-обучение, алгоритм временной разницы TD(λ) , SARSA, ARL, информированное Q-обучение, двойное Q-обучение, приблизительное Q-обучение, градиентный спуск TD(λ).) алгоритм, распределение доходов, обучение Q-III, реляционное RL, расслабленное SMART и TD(λ)-обучение. В глубоком обучении с подкреплением (DRL) используются многие подходы, основанные на ценности, такие как DQN (Deep Q-Learning Networks), слабосвязанный DRL, многоклассовый DQN и алгоритм Q-сети [43,44,65–73]. .

Цюй и др. [25] применили многоагентное приближенное Q-обучение, продемонстрировав его эффективность посредством численных экспериментов в статических и динамических средах и различных сценариях потокового цеха. Целью было создать и внедрить эффективный производственный график, учитывающий реалистичное взаимодействие между навыками рабочей силы и адаптивным оборудованием. Луо [74] использует метод DRL для динамического и гибкого планирования цеха, уделяя особое внимание сценариям сокращения опозданий. Луо и др. [75] впервые применили «иерархическую многоагентную оптимизацию проксимальной политики» (HMAPPO) для постоянно меняющейся многоцелевой задачи гибкого цеха с частичным отсутствием ожидания (MOFJSP) с добавлением новых заданий и поломками машин.

Износ оборудования в процессе производства увеличивает эксплуатационные расходы и вызывает сбои в рабочем процессе, что требует дорогостоящего корректирующего обслуживания. Профилактическое техническое обслуживание продлевает срок службы машины, но влечет за собой простои и затраты. Исследование [76] решает эти проблемы путем оптимизации графиков производства и технического обслуживания в системах с несколькими машинами, внедрения расширенного обучения с подкреплением для повышения эффективности RL в принятии производственных решений и содействии совместной работе машин. Сравнительные оценки в детерминированных и стохастических сценариях подчеркивают способность алгоритма максимизировать выгоды для бизнеса и предотвращать сбои.

По данным Ванга и др. [54] процесс планирования производства включает производство различных изделий с использованием гибридной модели производства, включающей модель многоагентного глубокого обучения с подкреплением (MADRL). Поппер и др. [60] предложили MADRL оптимизировать гибкие производственные предприятия с учетом таких факторов, как эффективность и экологические цели. Ду и др. [77] использовали метод Deep Q-Network (DQN) для решения проблемы гибкого планирования цеха задач (FJSP) на фоне изменения скорости обработки, времени настройки, времени простоя и транспортировки задач. Этот подход объединяет индикаторы состояния и действия для повышения эффективности компонента DQN. Кроме того, он включает в себя компонент исследовательского анализа данных (EDA), ориентированный на решение проблем, для улучшения исследования данных.

Ли и др. [78] использовали метод глубокого обучения с подкреплением (DRL) для решения дискретной гибкой задачи цеха с возможностью повторного использования разных инструментов (DFJSP-ITR), решая многокритериальную задачу оптимизации минимизации совокупного периода изготовления и общего энергопотребления . Предлагаемое решение включает в себя несколько общих характеристик состояния, пространство действий на основе генетического программирования и функцию вознаграждения. Чжоу и др. [79] предложили использовать стратегии онлайн-планирования на основе RL с составными функциями вознаграждения для повышения эффективности и надежности промышленных систем. В работе [18] использовался усовершенствованный алгоритм DRL для оптимизации планирования производства в сложных цехах, подчеркивая такие преимущества , как повышенная адаптируемость, глобальная видимость и оптимизация. Некоторые авторы сосредоточились на разработке различных алгоритмов DRL, способных формулировать сложные стратегии планирования производства. Например, Луо и др. [26] создали онлайн-структуру перепланирования для динамической многоцелевой задачи гибкого планирования цеха, позволяющую минимизировать общие опоздания или максимизировать коэффициент использования оборудования. Чжоу и др. [27] использовали технику глубокого Q-обучения для решения динамического планирования в интеллектуальном производстве. Другой метод динамического планирования с использованием глубокого RL предложен в [80] и использует оптимизацию проксимальной политики для определения идеальной политики планирования. Ван и Ашер [81] исследовали использование алгоритма Q-обучения с подкреплением для агентного планирования производства. Интеграция RL с планированием производства означает значительный прогресс в оптимизации производственных процессов. Поскольку исследования и технологии продолжают развиваться, это обещает еще большую эффективность и адаптируемость в о

Machine Translated by Google

предоставить ценную информацию о тенденциях исследований, влиятельных работах и ключевых участниках в этой быстро развивающейся области.

«Индустрия 4.0» и «умная индустрия» лидируют в современном производстве и производстве. Индустрия 4.0 объединяет передовые технологии, такие как Интернет вещей (IoT), искусственный интеллект (ИИ), большие данные и киберфизические системы, для создания высокоавтоматизированных и сетевых производственных сред. Концепция интеллектуальной промышленности делает упор на анализ данных в реальном времени, профилактическое обслуживание и адаптируемые производственные процессы для повышения эффективности, гибкости и производства [27]. Реализация алгоритмов RL имеет решающее значение в этих парадигмах, облегчая машинное обучение и прогнозирование будущего посредством взаимодействия в производственной среде. Это улучшает планирование, распределение ресурсов и контроль качества, позволяя фирмам достичь высокой автоматизации и точности принятия решений , что приводит к более устойчивому и оперативному современному производству [54]. Таблица 1 иллюстрирует вклад алгоритмов RL в планирование производства.

Таблица 1. Вклад обучения с подкреплением в планирование производства.

Тип планирования	Награда	Вклад
Категория: Окружающая среда		
Комплексный цех [18]	Штрафы	Усовершенствованный алгоритм DRL, оптимизирующий производственное планирование, подчеркивающий адаптивность, наглядность и оптимизацию.
Поточный цех [25]	Выполненный заказ и просроченный штраф	Эффективный график производства с учетом квалификации рабочей силы и адаптивного оборудования.
Гибкая мастерская [74]	Опоздания/коэффициент использования машины; стоимость энергии; поздние продукты	Онлайн-перепланирование, сводящее к минимуму опоздания или максимально увеличивающее использование машины. Метод MARL, оптимизирующий эффективность и экологические цели.
Магазин вакансий [54]	Штрафы	Метод DQN для минимизации времени выполнения задач при динамическом планировании. MADRL оптимизирует производительность на уровне системы с учетом взаимодействия деталей.
Категория: Метод		
Комплексный цех [18]	Штрафы	Алгоритм DRL для оптимизации планирования, улучшения адаптивности, видимости и оптимизации.
Поточный цех [25]	Выполненный заказ и просроченный штраф	Эффективный график с учетом навыков персонала и взаимодействия с техникой.
Гибкая мастерская [24]	Опоздания/коэффициент использования машины; стоимость энергии; поздние продукты	Обучение с подкреплением, позволяющее минимизировать время выполнения при динамическом планировании заданий.
Категория: Награда		
Комплексный цех [62]	Штрафы	Планирование заданий с использованием нескольких агентов в обучении по приоритетному использованию
Поточный цех [60]	Выполненный заказ против просроченного штрафа; стоимость энергии; просроченная	ресурсов. Оптимизация производственных предприятий для повышения эффективности и устойчивости.
Гибкая мастерская [19]	продукция Опоздание/коэффициент использования оборудования; стоимость энергии; поздние продукты	Решение алгоритма подкрепления при планировании с целью принятия решений в гибком магазине вакансий.

6. Библиометрический анализ исследований по RL в планировании производства.

Библиометрический анализ RL в контексте планирования производства фокусируется здесь на его теоретических основах и практических последствиях в период с 1996 по 2024 год. Первые два этапа включают сбор библиографических данных, которые были собраны через базу данных Scopus. Поиск был ограничен этой базой данных из-за ее важности как всеобъемлющего хранилища научной литературы и ее частого использования в академических оценках. Критерии включения для целей анализа, представленные в разделах 6.1–6.4 , охватывали публикации, содержащие термины «обучение с подкреплением», а также «производство » и «планирование». Кроме того, для анализа , проведенного в разделе 6.5, использовались термины «обучение с подкреплением» и «детерминированное планирование» или «обучение с подкреплением» и «неопределенное планирование». На следующих двух этапах сортир

Machine Translated by Google

Программное обеспечение Microsoft Excel и VOSviewer 1.6.20 использовалось для извлечения необходимой информации, такой как годовые научные результаты, наиболее важные источники, наиболее цитируемый автор и совпадение ключевых слов.

Процесс избранных исследований для указанной методологии исследования синтезирован в Рисунок 4, согласно рекомендациям PRISMA.

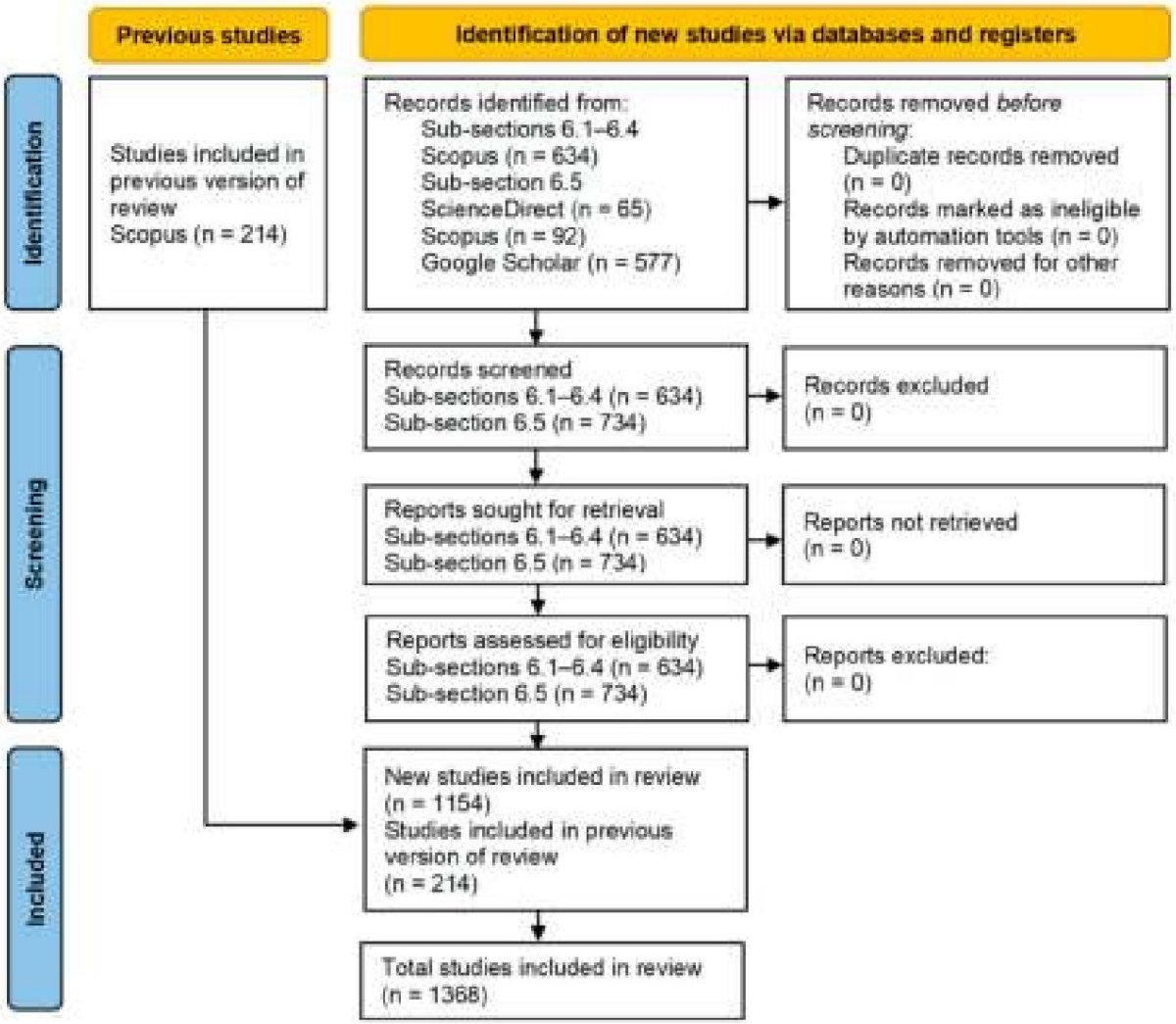


Рисунок 4. Блок-схема обновленных систематических обзоров PRISMA 2020, включающих поиск в базах данных.

По указанным выше ключевым словам и критериям поиска (по разделам 6.1–6.4 n = 634 публикации и по разделу 6.5 n = 734 публикации) всего было найдено 1368 статей в соавторстве до 27 мая 2024 г., когда все типы документов были включены в рецензию. анализ (новые исследования n = 1154 и исследования, включенные в предыдущую версию обзора n = 214).

Обзорный анализ разделов 6.1–6.4 включал 634 статьи, написанные в соавторстве, поиск по категориям «Название статьи, аннотация и ключевые слова». Основные области исследований, которым уделяется значительное внимание в статьях, включают информатику, инженерное дело, математику, науки о принятии решений , бизнес, менеджмент и подсчет, энергетику, химическую инженерию, материаловедение и другие темы. 634 публикации в нашей выборке были разделены на 19 различных тем исследований. Восемь основных областей исследований, а также распределение статей по ним показаны в таблице 2.

Machine Translated by Google

Таблица 2. Сопутствующие исследовательские дисциплины.

Области исследований	Количество публикаций	Процент
Информатика	429	32,7%
Инженерное дело	402	30,6%
Математика	159	12%
Науки о принятии решений	85	6,5%
Бизнес, менеджмент и счет	49	3,7%
Энергия	44	3,4%
Химическая инженерия	34	2,6%
Материаловедение	25	2%
Другие	85	6,5%

6.1. Тенденции публикаций и цитирования

Количество опубликованных статей и их цитирование обычно обеспечивают достаточно достоверную информацию.
информация для прогнозирования дальнейшего развития исследуемой области исследований. Учитывая
количество публикаций и цитирований в области планирования производства с использованием обучения
алгоритмы, хранящие данные за около 29 лет, был получен график анализа тенденций. Для
с этой целью применялись те же поисковые термины и ключевые слова, что и в случае идентификации
основных исследовательских категорий, но типы документов были распространены на все типы.
Причиной изменения типов документов было выяснение первоначальных исследовательских инициатив в этом
домен. Те же условия поиска были применены в остальной части статьи, чтобы включить
большую выборку публикаций для целей расследования. Годовое распределение
публикаций (из 634 наименований) и их цитирований из одной базы данных за
период с 1996 по 2024 год показан на рисунке 5.

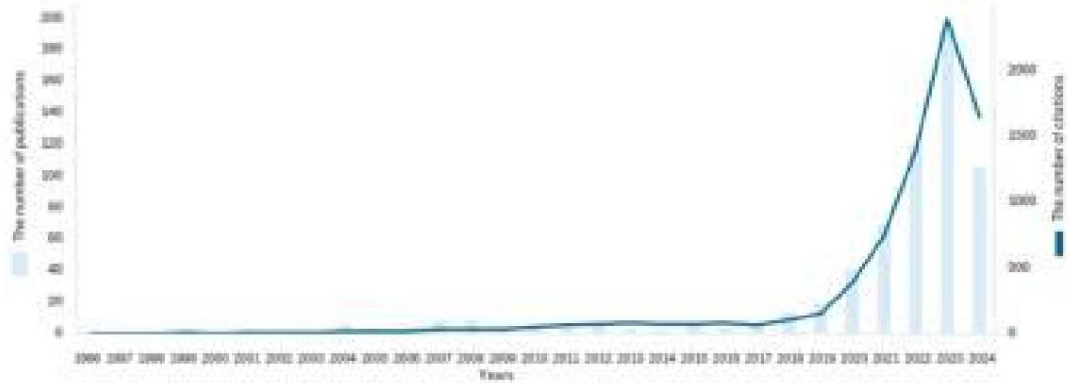


Рисунок 5. Тенденции публикаций и цитирования RL в области планирования производства с 1996 по 2024 год.

Анализ ежегодных научных результатов в период с 1996 по 2018 год убедительно продемонстрировал
относительно стабильно низкое количество статей, публикуемых ежегодно. На протяжении всего этого
две декады, необходимость РЛ в планировании производства в реальных условиях видимо
не появился. В период 2019–2023 годов наблюдался заметный рост числа
публикации, которые были зарегистрированы в наиболее признанной базе данных рецензируемого контента.
Это явление можно в первую очередь объяснить достижениями в области искусственного интеллекта.
Примечательно подчеркнуть, что если эта экспоненциальная тенденция увеличения числа
публикации продолжатся, как можно увидеть в 2024 году, тогда можно ожидать, что в течение
В следующем десятилетии важность RL в планировании производства значительно возрастет.

6.2. Наиболее релевантные источники

Идентификация наиболее релевантных публикаций из исходного набора данных представляет
общий подход в библиометрических исследованиях, поскольку такие источники обычно публикуют
влиятельные исследования, вызывающие широкий интерес. Как правило, наиболее продуктивные журналы
оказывают наибольшее влияние на развитие науки в той или иной области, поскольку они

Machine Translated by Google

публиковать больше статей и получать больше цитирований [77–79]. Что касается релевантности литературы в исследуемой области, то здесь определены десять лучших журналов, опубликовавших наибольшее количество статей. Также в этом подразделе упоминается десятка самых цитируемых научных журналов . На рисунках 6 и 7 журналы классифицируются в соответствии с этими двумя критериями, чтобы показать, что они представляют собой документы, которые имеют максимальную значимость для RL при планировании производства.

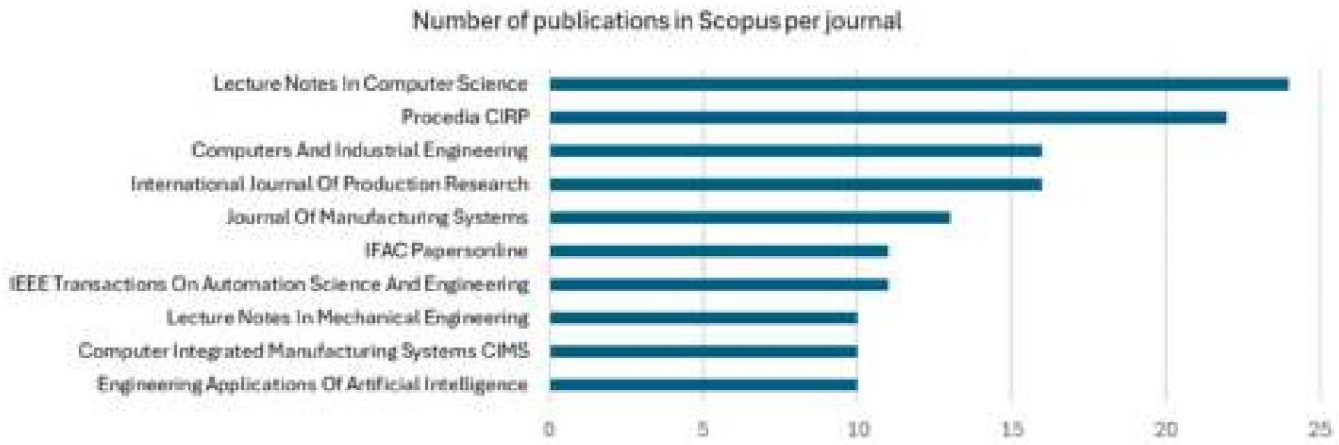


Рисунок 6. Десять крупнейших опубликованных журналов RL по вопросам планирования производства с 1996 по 2024 год.

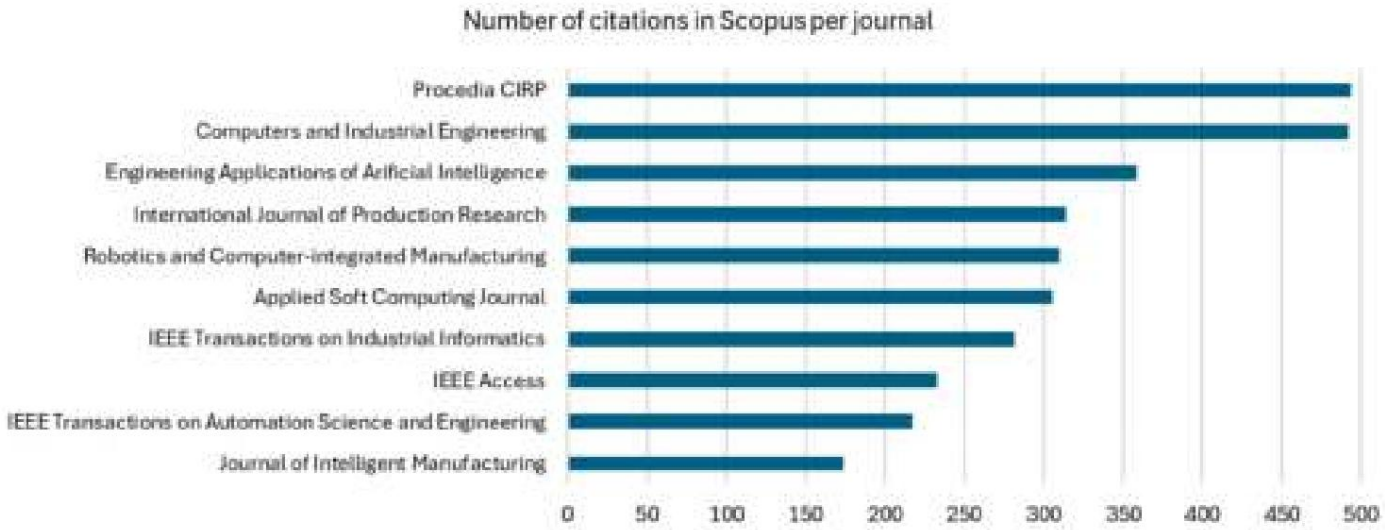


Рисунок 7. Десять наиболее цитируемых журналов RL по вопросам планирования производства с 1996 по 2024 год.

Конспекты лекций по информатике, Procedia CIRP и Computers and Industrial Engineering, а также двенадцать других журналов считаются наиболее значимыми научными публикациями в этой области. Также можно отметить, что все пятнадцать журналов, перечисленных на рисунках 6 и 7, независимо от результатов используемых показателей, эмпирически могут быть оценены как широко признанные за распространение передовых исследований в области обучения с подкреплением, применяемого к планированию производства. Кроме того, об их научной строгости также свидетельствует тот факт, что из пятнадцати выявленных журналов восемь соответствовали высоким стандартам качества, поскольку они индексируются для журналов Current Content Connect, что оказывает поддающееся проверке влияние на управление исследовательской практикой и поведением [80,81]. .

6.3. Самые цитируемые авторы

С 1996-х годов многие авторы внесли значительный вклад в развитие этой области. В этом подразделе мы хотим представить некоторых авторов, внесших значительный интеллектуальный вклад в исследование. Анализ наиболее цитируемых автор

Machine Translated by Google

было выполнено с использованием данных базы данных Scopus. Кроме того, для каждого источника публикации (из общего числа 634 наименований) был проведен анализ социтирования, чтобы выявить сеть между исследованиями. Для этой цели использовалось программное обеспечение VOSviewer [82] . Для получения актуальной информации и наглядного графического представления сложных отношений были использованы следующие фильтры. Фильтр 1: Максимальное количество авторов на документ — 15; Фильтр 2: Минимальное количество документов автора — 3; Фильтр 3: Минимальное количество цитирований автора — 10. Кроме того, был применен метод полного подсчета, что означает, что публикации, имеющие соавторов из нескольких стран, учитываются как полная публикация для каждой из этих стран. Сеть совместного цитирования выбранной выборки ученых, использующих эти настройки, представлена на рисунке 8.

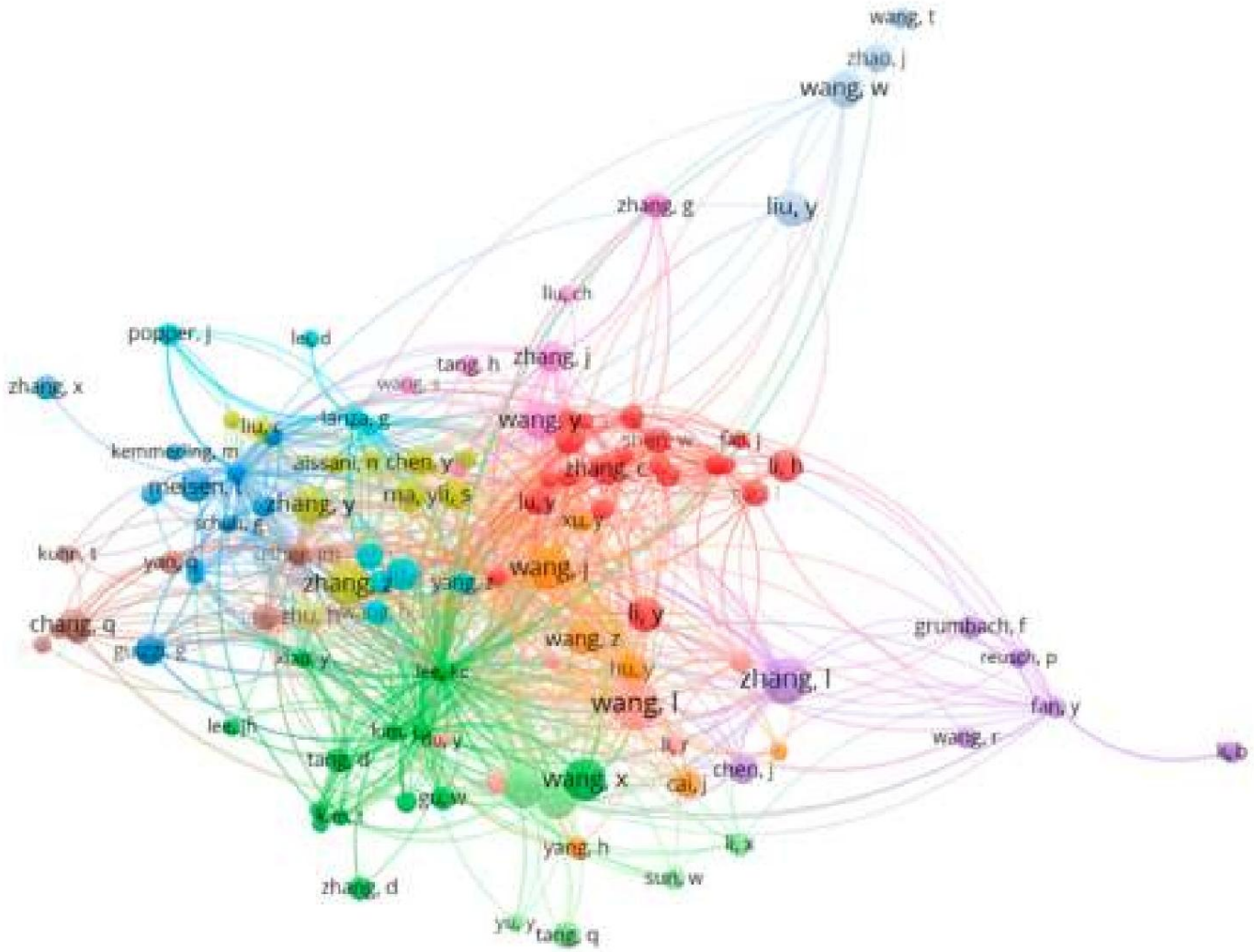


Рисунок 8. Сеть совместного цитирования авторов, внесших значительный вклад в развитие этой области с 1996 по 2024 год.

На этой карте сети совместного цитирования показаны, среди прочего, ученые, получившие наибольшее количество цитирований за последние 29 лет. Из 1376 цитируемых авторов вышеперечисленным критериям соответствуют 138. Каждый ученый из каждой включенной публикации представлен узлом в этой сети. Размер каждого узла указывает на частоту цитирования научных работ по теме. Граница проводится между двумя узлами, если два учёных цитировались в одном документе. Для ранжирования влиятельных ученых в данной области на основе их уровня цитируемости были выбраны десять наиболее цитируемых авторов. Эти десять авторов перечислены в Табл

Machine Translated by Google

Таблица 3. Наиболее влиятельные авторы.

Имя	Количество Цитаты	Страна	Имя	Количество Цитаты	Страна
Васченк, Б.	386	Германия	Ван, ЮК	265	Тайвань
Ван, Л.	375	Китай	Ван, Х.	224	Китай
Ван, Дж.	354	Китай	Лин, СС	216	Тайвань
Бауэрнхансль, Т.	342	Германия	Ли, С.	209	Китай
Ашер, Дж. М.	265	США	Бельджилали, Б.	207	Алжир

6.4. Определение смежных областей исследований

Целью данной части статьи является определение направлений исследований, в которых рассматривается проблема Представляет интерес планирование производства на основе RL. Для этого проводится анализ ключевых слов приложение было использовано, чтобы помочь разделить важные темы исследований, которые вызвал высокий интерес исследователей со стороны менее важных. Чтобы получить актуальные и репрезентативные категории, не включая менее значимые, была сделана следующая установка: использовано: Минимальное количество вхождений ключевых слов — 14. На основании этого ограничения 88 ключевые слова соответствуют пороговому значению из 3767 ключевых слов. Те основные ключевые слова, которые производились автоматически из заголовков в статьях по планированию производства на основе на RL вместе с их появлением показаны на рисунке 9.

Как видно, темы можно разделить на пять кластеров на основе компьютерного алгоритма: при этом каждый кластер имеет свой цвет, как показано на рисунке 9. Ключевые слова встречаются одновременно. На карте выделяются эти кластеры: чем темнее цвет, тем больше значение плотности. Это позволяет выявить актуальные темы исследований и их взаимные связи. На основании получили библиометрические результаты, извлеченные из VOSviewer, десять связанных тем, которые очень близкие к исследуемой области исследований были идентифицированы, как показано в Таблице 4.

Результаты Таблицы 4 указывают на их соответствие практической реальности и эксплуатации. цели исследования. Например, в последние годы заметно возрос интерес в использовании обучения с подкреплением для оптимизации задач планирования заданий в реальном времени [76–86]. Этот факт можно соотнести с продолжающейся тенденцией массовой кастомизации производства потребительских товаров [87,88]. Как известно, для массовой кастомизации характерна своевременно удовлетворять динамически меняющиеся требования пользователей, в то время как индивидуальные продукты нуждаются в должны быть выполнены в разные сроки. Соответственно, эффективное планирование заданий в режиме реального времени алгоритмы, основанные на DRL, становятся необходимыми. Следующий важный метод, который ранжируется В число 10 наиболее часто встречающихся ключевых слов входит контроль производства. Он использует другой контроль методы достижения производственных целей в отношении графиков производства и качества (см. например, [89–92]). Следующее важное ключевое слово, встречающееся одновременно в Таблице 3, — многоагентные системы. В целом, включение мультиагентных систем в обучение с подкреплением для производства планирование предлагает многочисленные преимущества с точки зрения гибкости, масштабируемости и адаптируемости [93–95]. Обеспечивая децентрализованное принятие решений и непрерывное обучение, эти системы могут эффективно справляться со сложностями и динамикой современной производственной среды, что приводит к более эффективным и устойчивым решениям для планирования [96]. Среди Ключевые слова совместного появления можно также выделить «умное производство», которое представляет собой область реализации планирования производства на основе RL. Несмотря на то умное производство также стало модным словом, которое также имеет свои недостатки, это концепция постепенно утверждается как новая производственная парадигма. На с другой стороны, сложность инфраструктур интеллектуальных производственных сетей становится выше. и выше, и неопределенность такой производственной среды становится серьезной проблемой. проблема [25]. Эти факты приводят к необходимости применения передовых решений динамического планирования, включающих в себя и планирование производства с использованием RL. Этот абзац одновременно ответы на основной вопрос исследования, сформулированный в разделе 1.

Далее, здесь будет проанализировано применение RL в планировании производства. библиометрические средства с точки зрения различных задач планирования.

Machine Translated by Google

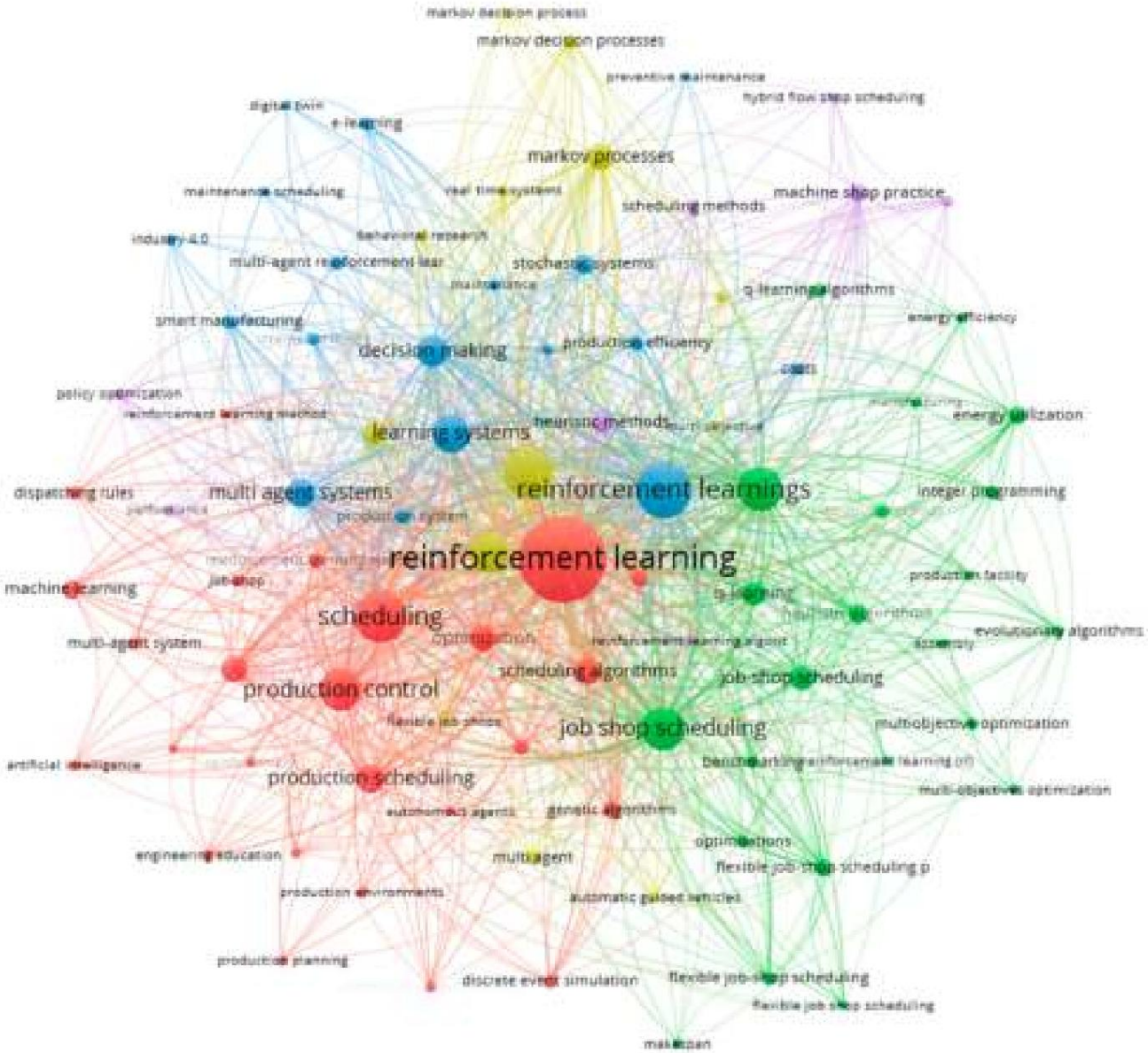


Рисунок 9. Ключевые слова совместного появления для RL в поле планирования производства с 1996 по 2024 год.

Таблица 4. Наиболее связанные методологии и области реализации планирования производства на основе на РЛ.

Связанные области исследований	Количество совпадений	Связанные термины/темы	Количество совпадений
Обучение с подкреплением	526	Производственный контроль	161
Планирование	220	Глубокое обучение с подкреплением	148
Глубокое обучение	215 и	Системы обучения ffi	134
Алгоритмы обучения	далее 178	Принятие решений	99
График работы цеха	168	Мультиагентные системы	86

6.5. Сравнение применения RL в различных задачах планирования

В целом методы планирования классифицируются в зависимости от времени принятия решения. разделены на динамические и статические. Динамическое планирование относится к системам реального времени, которые требуют реагирования на меняющиеся требования спроса, в то время как статическое планирование осуществляется в автономном режиме и фокусируется на краткосрочном временном горизонте, устанавливая фиксированные сроки завершения процесса. Это было обнаружено Вангом и др. [28] что планирование производства на основе подходов RL

Machine Translated by Google

в основном применяются для решения задач динамического планирования. Проблемы с планированием также могут классифицироваться по характеру среды планирования как детерминированная, при обработке параметры известны, неизменны и недетерминированы, когда входные параметры являются неопределенными [97]. Проблемы планирования производства в соответствии с этим вторым критерием могут быть классифицированы, как показано на рисунке 10.



Рисунок 10. Базовая классификация задач планирования.

Понимание различий между конкретными задачами планирования имеет решающее значение для выбора применение соответствующих методов оптимизации и планирования с целью улучшения эффективность и производительность в различных отраслях промышленности. Основное отличие планирования Проблемы ffi можно увидеть, например, в гибкости и сложности маршрутизации. Например, магазин вакансий задачи имеют высокую гибкость маршрутизации и сложность, в то время как для поточных и единичных Проблемы с машиной типичны для низкой гибкости, а проблема с параллельной машиной имеет умеренный характер. гибкость и сложность маршрутизации [98,99]. Их различия также заключаются в характере применения. домены, для которых они предназначены. Например, для индивидуального производства типичное планирование цеха [100,101], для сборочных линий в основном считается поточным цехом проблемы планирования [102,103], а планирование открытых цехов часто применимо в здравоохранение [104,105].

Классификация далее используется для изучения частоты применения RL в выявлены категории проблем планирования. Во-первых, детерминистический или неопределенный планирование сравнивается в соответствии с этой точкой зрения. Для этого условия поиска были определяется путем объединения следующих ключевых слов: «Обучение с подкреплением» и «Детерминированное планирование» или «Неопределенное планирование» соответственно. Данные вместе с Критерии включения были сопоставлены путем поиска: (i) База данных ScienceDirect — Все поля, Все годы и Все типы документов; (ii) База данных Scopus — Все поля, Все годы, Все документы. типы и (iii) база данных Google Scholar — все годы и все типы документов. Вместе Проанализировано 734 публикации . Web of Science не использовался в этой процедуре из-за низкая распространенность научных работ по данной тематике. Полученные таким образом данные представлено в таблице 5.

Таблица 5. Сравнение детерминированных и неопределенных методов планирования на основе RL.

Используемые поисковые запросы	Наука Прямая База данных	Скопус База данных	Google Академика База данных
«Детерминированное планирование» и «обучение с подкреплением»	53	68	474
«Неопределенное планирование» и «обучение с подкреплением»	12	24	103

Из таблицы 5 видно, что детерминированное планирование на основе RL находится в доминирующем положении. позиция против неопределенного планирования с использованием RL. Поэтому для выявления наиболее перспективных

Machine Translated by Google

Область(и) планирования производства в контексте RL, детерминистического планирования и подходов к неопределенному планированию рассматриваются ниже. Ключевые слова, использованные для этой цели, были:

(а) Планирование И «обучение с подкреплением» И конкретные детерминированные задачи планирования («цех работ», «цех потока», «открытый цех», «одна машина» и «параллельные машины»).

(б) «Обучение с подкреплением» И конкретные проблемы неопределенного планирования («нечеткое планирование » и «стохастическое планирование»).

Данные были получены из базы данных ScienceDirect с использованием фильтров «Исследовательские статьи» и «Все годы»; База данных Scopus с использованием фильтров «Названия статей», «Аннотации», «Ключевые слова», «Все годы» и «Все типы документов», а также база данных Google Scholar — «Все годы» и «Все типы документов». Полученные результаты представлены графически и сопоставлены в процентах друг с другом на рисунке 11.

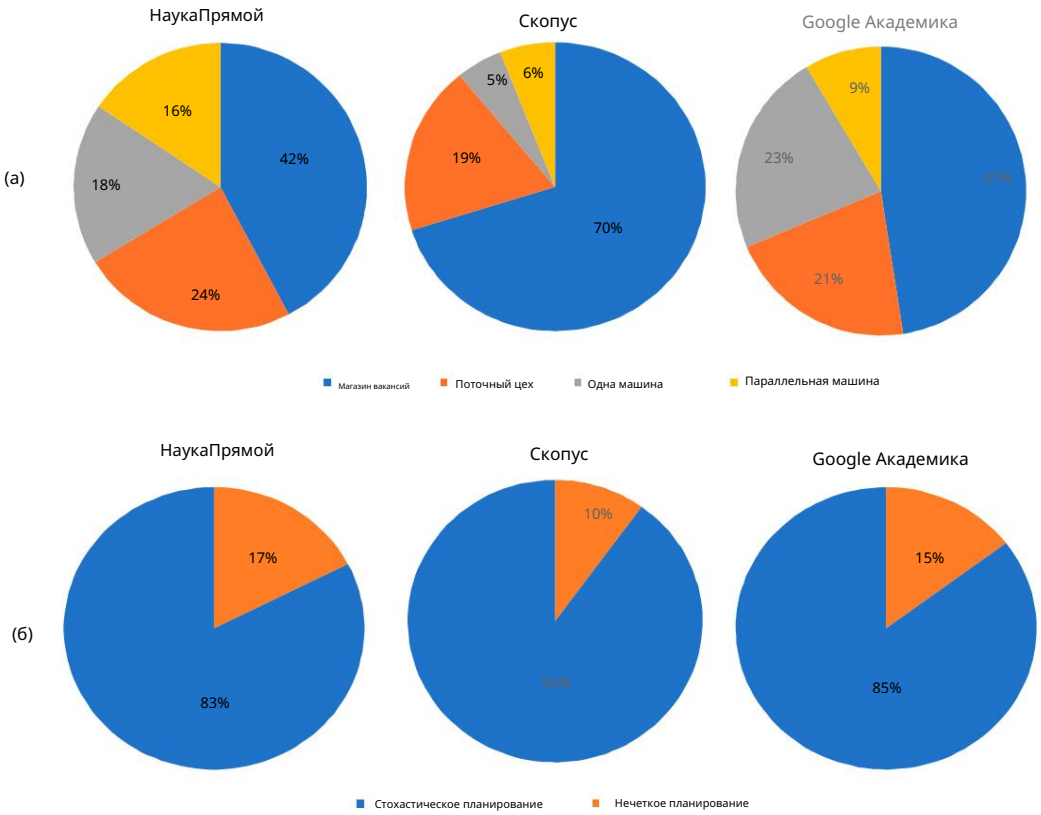


Рисунок 11. Сравнение (а) детерминированных методов планирования и (б) неопределенных методов планирования по их встречаемости в литературе.

На рисунке 11a видно , что алгоритмы планирования на основе RL в основном используются для решения задач планирования цеха [106], в то время как производственная среда поточного цеха является вторым по важности типом задач планирования, где применяется RL. Очевидно, что алгоритмы на основе RL нашли применение и в других типах задач планирования. В качестве наиболее заметных из них можно упомянуть энергоэффективное планирование, многокритериальное планирование и распределительное планирование. Последняя упомянутая проблема планирования особенно часто используется в интеллектуальных производственных системах [107]. Анализируя результаты рисунка 11b, можно заметить, что RL преимущественно используется в методах стохастического планирования по сравнению с его применением в методах нечеткого планирования. Более того, эффективность стохастического планирования с использованием RL также была продемонстрирована в ходе сравнительных исследований [29,108,109]. Хотя применение RL в нечетком планировании является многообещающим, оно остается менее распространенным из-за дополнительных сложностей, связанных с интег

Machine Translated by Google

методами RL [110,111]. Однако результаты этого количественного подхода для сравнения применения RL в различных задачах планирования дают некоторые полезные управленческие знания. понимание практики планирования. Тем не менее, дальнейшие исследования и исследования качественных аспектов планирования на основе RL, особенно в контексте умного производства, требуется.

7. Сравнение представленных исследований и предыдущих обзорных статей.

Как уже упоминалось во введении, данный обзорный доклад продолжает работу две более ранние статьи [28,29] путем анализа некоторых новых важных аспектов планирования производства. методы с использованием RL и предоставляет некоторую обновленную информацию о последних разработках. в поле. Цель этого раздела – предоставить четкое представление о различиях и сходство трех произведений с точки зрения их тематики. Следующая таблица 6 содержит сводное сравнение направлений исследований, включенных в указанные работы.

Таблица 6. Сравнение обзорных статей по RL в планировании производства.

В центре внимания расследования	Сравненные обзорные статьи		
	Ван и др., 2021 г. [28]	Кайхан и Йылдыз, 2021 г. [29]	Предложенное исследование
Тенденция цитирования RL в производстве планирование с 1996 по 2024 год			
Самые актуальные публикации в этой области			
Самые влиятельные авторы в этой области			
Сравнение детерминированного и неопределенного методы планирования на основе RL			
Сравнение стохастического и нечеткого методы планирования на основе RL			
Тенденция публикации RL в производстве планирование			
В наиболее значимых источниках опубликованы исследования по RL в планировании производства			
Сравнение самых частых детерминированные методы планирования на основе RL			
Сравнение ценностных и методы планирования на основе политик на основе RL			
Сравнение статического и динамического методы планирования на основе RL			
Категоризация литературы по RL в планирование производства в соответствии с несколькими аспекты			
Тенденция публикации RL в производстве планирование в зависимости от производства среда			
Сравнение разных типов RL алгоритмы планирования производства			
Сравнение стохастического и детерминистического методы планирования на основе RL			

Из этой таблицы видно, что сравниваемые статьи в основном дополняют друг друга. друг друга.

Machine Translated by Google

8. Выводы

Существующий объем исследований по планированию производства в основном состоит из исследований, проводимых в областях информатики, инженерии, математики, науки о принятии решений, бизнеса, управления и подсчета, энергетики, химической инженерии и материаловедения. Эти научные дисциплины, в частности «Инженерное дело» и «Информатика», оказали значительное влияние на развитие планирования производства на основе RL.

Изучение библиометрических данных зачастую указывает на то, что увеличение количества публикуемых статей связано с признанием прогрессивных тенденций в предмете. Кроме того, по этой причине библиометрический анализ становится все более полезным в различных академических областях, поскольку он делает картирование научной информации и анализ развития исследований объективными и повторяемыми. Использование этого метода позволяет нам выявить сети научного сотрудничества, установить связи между новыми темами исследований и направлениями исследований, а также показать связи между цитированием, социтированием и продуктивностью публикаций в этой области.

Основной вклад этой статьи можно резюмировать двумя этапами: (i) Этот обзор дает дополнительную информацию об использовании RL в планировании производства, предоставляя следующие новые функции: •

Была проведена тенденция цитирования RL в области планирования производства с 1996 по 2024 год (см. Рисунок 5), которая показывает, что тенденция увеличения, начиная с 2019 года по настоящее время, быстро продолжается в заметном росте их количества. • Анализ наиболее релевантных источников с точки зрения количества их цитирований (см. Рисунок 7) был проведен для выявления наиболее затронутых источников , в которых доступны новейшие знания в этой области. •

Были определены наиболее влиятельные авторы в этой области (см. Таблицу 3) для определения передового состояния планирования производства на основе RL. • Проведено количественное сравнение детерминированных и неопределенных методов планирования на основе RL (см. Таблицу 5). Оно показало, что детерминированные методы планирования с использованием RL занимают доминирующее положение по сравнению с методами неопределенного планирования на основе RL. • Количественное сравнение стохастического и нечеткого методов планирования представлено на рисунке 11b. В результате сравнения было замечено, что RL преимущественно используется в методах стохастического планирования по сравнению с его применением в методах нечеткого планирования.

(ii) Вклад данного документа можно также увидеть в следующих мероприятиях и их результатах: заявления:

• Тенденция публикации RL в области планирования производства была обновлена за период картирования с 1996 по 2024 год (см. Рисунок 5). Оно показало, что экспоненциальный рост публикаций, начиная с 2019 года [28], быстро продолжается в заметном росте их количества.

• Обновленный список наиболее релевантных источников, опубликованных в исследованиях по RL в планировании производства, представлен на рисунке 6. Было обнаружено, что пять из десяти источников, выявленных до 2021 года, входят в десятку крупнейших источников, выявленных до 2024 года. Это означает, что из-за растущего интереса к этой области исследований одновременно увеличивается и количество журналов, освещающих эту тему. •

Актуализация количественного сравнения наиболее частых детерминированных методов планирования на основе RL представлена на рисунке 11a. В этом контексте была подтверждена ранее выявленная тенденция: алгоритмы планирования на основе RL в основном используются для решения задач планирования цеха, в то время как поточное производство представляет собой второй по важности тип задач планирования. •

Обновленная категоризация недавно разработанных алгоритмов RL была разработана и представлена на рисунке 3 , что может обеспечить лучшую поддержку реализации для принятия решений в реальных проблемах.

Machine Translated by Google

• Обновленная классификация методов планирования представлена на рисунке 10. В принципе, ее можно использовать для облегчения выбора методов планирования для решения конкретных типов задач.

В дополнение к вышеупомянутым выводам необходимо будет сосредоточиться на других задачах, которые следует рассмотреть в будущем, таких как планирование производственного процесса с комплексной поддержкой обмена знаниями, растущий спрос на улучшения повсеместной «умности» в производстве. процессы, включая разработку и внедрение интеллектуальных алгоритмов, а также потребность в надежных инструментах планирования для гибких совместных производственных систем

Вклад автора: Концептуализация, VM; методология, CO и AB; валидация, ЗС; формальный анализ, ВМ и РС; расследование, ВМ и АБ; письменное – подготовка оригинального черновика, АБ и ВМ; написание-обзор и редактирование, РС и ВМ; визуализация, ЗС; администрация проекта В.М. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Данное исследование финансировалось в рамках проекта SME 5.0 за счет средств, полученных от исследовательской и инновационной программы Европейского Союза Horizon в рамках Соглашения о гранте Марии Склодовской-Кюри № 101086487 и проекта KEGA (Агентство грантов в области культуры и образования) № 101086487.

044TUKE-4/2023 выдан Министерством образования Словацкой Республики.

Заявление Институционального наблюдательного совета: Неприменимо.

Заявление об информированном согласии: Не применимо.

Заявление о доступности данных: данные содержатся в статье.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов. Спонсоры не играли никакой роли в разработке исследования; при сборе, анализе или интерпретации данных; в написании рукописи; или в решении опубликовать результаты.

Ссылки

1. Пинедо, М. Планирование и составление графиков в производстве и сфере услуг; Спрингер: Нью-Йорк, штат Нью-Йорк, США, 2005. 2.
Бехешти, З.; Шамсуддин, С.М.Х. Обзор метаэвристических алгоритмов на основе популяций. Межд. Дж. Адв. Мягкий компьютер. Прил. 2013, 5, 1–35.

3. Джафа, Ф.; Абрахам, А. (ред.) Метаэвристика для планирования в промышленных и производственных приложениях; Springer: Берлин/ Гейдельберг, Германия, 2008 г.; Том 128.

4. Абдель-Кадер, Р.Ф. Оптимизация роя частиц для планирования инструкций с ограничениями. СБИС Дес. 2008, 2008, 930610. [CrossRef]

5. Баламуруган А.; Ранджитарамасами, С.П. Модифицированная эвристика для оптимизации размера партии с комбинированным временем в Система массового производства. Межд. Дж. Индийский инж. Теория Прикл. Практика. 2023, 30, 1090–1115.

6. Гассеми Тари, Ф.; Ольфат, Л. Эвристические правила решения проблемы опозданий в поточном цехе с промежуточными сроками выполнения работ. Межд. Дж. Адв. Производитель Технол. 2014, 71, 381–393. [Перекрестная ссылка]

7. Модрак В.; Пандиан, Р.С. Алгоритм планирования цеха для минимизации времени завершения задачи n-работ и m-машин. Тех. Весн. 2010, 17, 273–278.

8. Тенарасу, М.; Рамешкумар, К.; Руссо, Дж.; Анбудаясанкар, С.П. Разработка и анализ правил принятия приоритетных решений с использованием подхода MCDM для гибкого планирования цеха: моделирование. Симул. Модель. Практика. Теория 2022, 114, 102416. [CrossRef]

9. Пандиан С.; Модрак В. Возможности, препятствия и проблемы генетического алгоритма при формировании производственных клеток. Адв. Логист. Сист. 2009, 3, 63–70.

10. Абдулреда, Миннесота; Бараа, А.А.; Джабир, А.Дж. Эвристические и метаэвристические модели оптимизации для планирования задач в облачном тумане системы: Обзор. Иракский Дж. Электр. Электрон. англ. 2020, 16, 103–112. [Перекрестная ссылка]

11. Модрак В.; Пандиан, RS; Семанко, П. Калибровка параметров ГА для задач оптимизации макета с использованием дизайна эксперименты. Прил. наук. 2021, 11, 6940. [CrossRef]

12. Кешанчи, Б.; Сури, А.; Навимипур, Нью-Джерси. Улучшенный генетический алгоритм планирования задач в облачных средах с использованием очередей с приоритетами: формальная проверка, моделирование и статистическое тестирование. Дж. Сист. Программное обеспечение 2017, 124, 1–21. [Перекрестная ссылка]

13. Янс Р.; Дегрейв, З. Метаэвристика для динамического определения размера партии: обзор и сравнение подходов к решению. Евро. Дж. Опер. Рез. 2007, 177, 1855–1875. [Перекрестная ссылка]

14. Хан, Б.А.; Ян, Дж. Дж. Решение на основе глубокого обучения с подкреплением для решения задачи гибкого планирования рабочего времени. Межд. Дж. Симул. Модель. 2021, 20, 375–386. [Перекрестная ссылка]

15. Шялика, К.; Сильва, Т.; Карунананда, А. Обучение с подкреплением при динамическом планировании задач: обзор. СН Компьютер. наук. 2020, 1, 306. [ПерекрестнаяСсылка]

16. Ван Х.; Чжан, Л.; Рен, Л.; Се, К.; Ван, К.; Йе, Ф.; Чен, З. Краткий обзор применения обучения с подкреплением для решения задач планирования цеха. Дж. Сист. Симул. 2022, 33, 2782–2791.

Machine Translated by Google

17. Дима И.С.; Габрара, Дж.; Модрак, В.; Петр, П.; Попеску, К. Использование экспертных систем в оперативном управлении производством. В материалах 11-й Международной конференции WSEAS по математике и компьютерам в бизнесе и экономике (MCBE'10), Яссы, Румыния, 13–15 июня 2010 г.; WSEAS Press: Стивенс Пойнт, Висконсин, США, 2010.

18. Вашнек, Б.; Рейхсталлер, А.; Бельзнер, Л.; Альтенмюллер, Т.; Бауэрнхансль, Т.; Кнапп, А.; Кек, А. Оптимизация глобальной планирование производства с глубоким обучением с подкреплением. Procedia CIRP 2018, 72, 1264–1269. [Перекрестная ссылка]

19. Ян Дж.; Лю, З.; Чжан, Т.; Чжан Ю. Автономный метод принятия решений в процессе транспортировки для задачи гибкого планирования цеха на основе обучения с подкреплением. В материалах Международной конференции по машинному обучению и интеллектуальным системным разработкам 2021 года, MLISE, Чунцин, Китай, 9–11 июля 2021 года; Институт инженеров по электротехнике и электронике : Пискатауэй, Нью-Джерси, США; стр. 234–238.

20. Модрак В.; Пандиан, RS , исследования в области управления операциями и системы сотового производства; IGI Global: Херши, Пенсильвания, США, 2010 г.

21. Хуан З.; Лю, К.; Чжу, Ф. Иерархическое обучение с подкреплением с адаптивным планированием для управления роботами. англ. Прил. Артиф. Интел. 2023, 126, 107130. [CrossRef]

22. Арвив, К.; Стерн, Х.; Эдан, Ю. Совместное обучение с подкреплением для решения задачи планирования потока операций с двумя роботами. Межд. Дж. Прод. Рез. 2016, 54, 1196–1209. [Перекрестная ссылка]

23. Вэнь Х.; Чжан, Х.; Син, Х.; Йе, Г.; Ли, Х.; Чжан, Ю.; Ван, Х. Улучшенный генетический алгоритм, основанный на обучении с подкреплением для проблемы планирования сборки самолетов. Вычислить. Индийский англ. 2024, 193, 110263. [CrossRef]

24. Айдын, МЭ; Озтемель, Э. Динамическое планирование цеха с использованием агентов обучения с подкреплением. Робот. Автон. Сист. 2000, 33, 169–178. [Перекрестная ссылка]

25. Цюй, С.; Ван, Дж.; Говил, С.; Леки, Дж. О. Оптимизированное адаптивное планирование системы производственного процесса с многопрофильной рабочей силой и несколькими типами машин: многоагентный подход к обучению с подкреплением на основе онтологий. Procedia CIRP 2016, 57, 55–60. [Перекрестная ссылка]

26. Луо, С.; Чжан, Л.; Фан, Ю. Динамическое многоцелевое планирование для гибкого цеха путем глубокого обучения с подкреплением. Вычислить. Индийский англ. 2021, 159, 107489. [CrossRef]

27. Чжоу, Л.; Чжан, Л.; Хорн, БКП Динамическое планирование на основе глубокого обучения с подкреплением в интеллектуальном производстве. Procedia CIRP 2020, 93, 383–388. [Перекрестная ссылка]

28. Ван, Л.; Пан, З.; Ван, Дж. Обзор интеллектуальной оптимизации на основе обучения с подкреплением для планирования производства. Комплексная система. Модель. Симул. 2021, 1, 257–270. [Перекрестная ссылка]

29. Кайхан, Б.М.; Йылдыз, Г. Приложения обучения с подкреплением для решения задач машинного планирования: подробный обзор литературы. Дж. Интелл. Производитель 2021, 34, 905–929. [Перекрестная ссылка]

30. Бродус Р.Н. К определению «библиометрии». Наукометрика 1987, 12, 373–379. [Перекрестная ссылка]

31. Арунможи, Б.; Судхакарапандиан, Р.; Султан Батча, Ю.; Раджай Ведарадж, И.С. Инференциальный анализ нержавеющей стали с добавками изготовление с использованием библиометрических показателей. Mater Today Proc. 2023, в печати. [Перекрестная ссылка]

32. Рандхава, К.; Уилден, Р.; Хобергер, Дж. Библиометрический обзор открытых инноваций: определение программы исследований. Дж. Прод. Иннов. Менеджер. 2016, 33, 750–772. [Перекрестная ссылка]

33. Ван Раан, А. Передовые библиометрические методы как количественное ядро оценок и прогнозирования, основанных на рецензировании. Наукометрика 1996, 36, 397–420. [Перекрестная ссылка]

34. Брэндом, Р.Б. Формулирование причин: введение в инференциализм; Издательство Гарвардского университета: Кембридж, Массачусетс, США, 2001.

35. Котари, CR Методология исследования: методы и техники; New Age International: Нью-Дели, Индия, 2004 г.

36. Суарес, М. Инференциальная концепция научного представления. Филос. наук. 2004, 71, 767–779. [Перекрестная ссылка]

37. Графиня Г. Научное представление, интерпретация и суррогатное рассуждение. Филос. наук. 2007, 74, 48–68. [Перекрестная ссылка]

38. Говье, Т. Проблемы анализа и оценки аргументов; Виндзорский университет: Виндзор, Онтарио, Канада, 2018 г.; Том 6.

39. Мунусами, Р.; Мукерджи, А.; Васудеван, К.; Венкатешваран, Б. Проектирование и моделирование мозга искусственного интеллекта (ИИ) для 2D-система навигации автомобиля. ИНКАС Бюлл. 2022, 14, 53–64. [Перекрестная ссылка]

40. Дунько В.; Бригель, Х.Дж. Машинное обучение и искусственный интеллект в квантовой области: обзор недавнего прогресса. Реп. прог. Физ. 2018, 81, 074001.

41. Хорвиц Э.; Маллиган, Д. Данные, конфиденциальность и всеобщее благо. Наука 2015, 349, 253–255. [Перекрестная ссылка]

42. Саттон, Р.С.; Барто, А.Г. Обучение с подкреплением: Введение; Массачусетский технологический институт: Нью-Йорк, штат Нью-Йорк, США, 2018.

43. Кунле, А.; Мэй, МС; Шефер, Л.; Ланца, Г. Объяснимое обучение с подкреплением в производственном контроле производственной системы цеха . Межд. Дж. Прод. Рез. 2022, 60, 5812–5834. [Перекрестная ссылка]

44. Эстесо, А.; Пейдро, Д.; Мула, Дж.; Диас-Мадроньеро, М. Обучение с подкреплением применительно к планированию и контролю производства. Межд. Дж. Прод. Рез. 2023, 61, 5772–5789. [Перекрестная ссылка]

45. Хан, МАМ; Хан, МРЖ; Тушил, А.; Сикдер, Н.; Махмуд, член парламента; Кузани, Аризона; Нахид, А.А. Систематический обзор робототехники, основанной на обучении с подкреплением, за последнее десятилетие. IEEE Access 2020, 8, 176598–176623. [Перекрестная ссылка]

46. Аль-Махамид, Ф.; Гролингер, К. Алгоритмы обучения с подкреплением: обзор и классификация. В материалах Канадской конференции IEEE по электротехнике и вычислительной технике (CCECE) 2021 года, Кингстон, Онтарио, Канада, 12–17 сентября 2021 года; IEEE: Пискатауэй, Нью-Джерси, США; стр. 1–7.

47. Акалин Н.; Лутфи, А. Подходы к обучению с подкреплением в социальной робототехнике. Датчики 2021, 21, 1292. [CrossRef]

48. Чжан Х.; Ю, Т. Таксономия алгоритмов обучения с подкреплением. В глубоком обучении с подкреплением: основы, исследования и Приложения; Спрингер: Сингапур, 2020 г.; стр. 125–133.

Machine Translated by Google

49. Бейкер, КР; Трич, Д. Принципы последовательности и планирования; John Wiley & Sons: Хобокен, Нью-Джерси, США, 2009 г.
50. Пинедо, М.Л. Планирование: теория, алгоритмы и системы; Шпрингер: Берлин/Гейдельберг, Германия, 2016.
51. Аллахверди А.; Нг, Коннектикут; Ченг, ТСЕ; Ковалёв М.Ю. Обзор проблем планирования со сроками и затратами на наладку. Евро. Дж. Опер. Рез. 2008, 187, 985–1032. [Перекрестная ссылка]
52. Панвалкар, СС; Смит, М.Л. Обзор исследований в области планирования цехов. Опер. Рез. 1977, 25, 45–84. [Перекрестная ссылка]
53. Парашос, П. Д.; Кулинас, ГК; Кулуриотис, Д.Э. Оптимизация на основе обучения для устойчивого и бережливого производства в контексте Индустрии 4.0. Алгоритмы 2024, 17, 98. [CrossRef]
54. Ван, С.; Ли, Дж.; Луо, Ю. Умное планирование для гибкого и гибридного производства с помощью многоагентного глубокого обучения с подкреплением. В материалах 2-й Международной конференции IEEE по информационным технологиям, большим данным и искусственному интеллекту 2021 года, ICIBA, Чунцин, Китай, 17–19 декабря 2021 года; Институт инженеров по электротехнике и электронике: Пискатауэй, Нью-Джерси, США; стр. 288–294.
55. Тан, Дж.; Хаддад, Ю.; Салонитис, К. Реконфигурируемое планирование производственной системы: подход к глубокому обучению с подкреплением. Procedia CIRP 2022, 107, 1198–1203. [Перекрестная ссылка]
56. Шахраби Дж.; Адиби, Массачусетс; Махутчи, М. Подход с подкреплением к оценке параметров при динамическом планировании цеха. Компьютерная индустрия, англ. 2017, 110, 75–82. [Перекрестная ссылка]
57. Ян Дж.; Ты, Х.; Ву, Г.; Хасан, ММ; Альмогрен, А.; Гуна, Дж. Применение обучения с подкреплением в задаче кластера БПЛА планирование. Будущий генерал. Вычислить. Сист. 2019, 95, 140–148. [Перекрестная ссылка]
58. Юань, Х.; Пан, Ю.; Ян, Дж.; Ван, В.; Хуанг, З. Исследование применения обучения с подкреплением при оптимизации работы системы HVAC. В строительном моделировании; Издательство Университета Цинхуа: Пекин, Китай, 2021 г.; Том 14, стр. 75–87.
59. Куринов И.; Ожеховский, Г.; Хямяляйнен, П.; Миккола, А. Автоматизированный экскаватор на основе обучения с подкреплением и многочастичества. системная динамика. IEEE Access 2020, 8, 213998–214006. [Перекрестная ссылка]
60. Поппер Дж.; Мотч, В.; Дэвид, А.; Петше, Т.; Русковски, М. Использование многоагентного глубокого обучения с подкреплением для гибкого планирования цеха с точки зрения устойчивого развития. В материалах Международной конференции по электротехнике, компьютерной технике, связи и мехатронике 2021, ICECCME, Маврикий, Маврикий, 7–8 октября 2021 г.; Институт инженеров по электротехнике и электронике: Пискатауэй, Нью-Джерси, США.
61. Сюн, Х.; Фан, Х.; Цзян, Г.; Ли, Г. Исследование правил диспетчеризации на основе моделирования в динамической задаче планирования цеха с пакетным выпуском и расширенными ограничениями технического приоритета. Евро. Дж. Опер. Рез. 2017, 257, 13–24. [Перекрестная ссылка]
62. Паласио, Х.К.; Хименес, Ю.М.; Шитгат, Л.; Ван Донинк, Б.; Нове, А. Алгоритм Q-Learning для гибкого планирования цеха в реальном производственном сценарии. Procedia CIRP 2022, 106, 227–232. [Перекрестная ссылка]
63. Чанг Дж.; Ю, Д.; Ху, Ю.; Он, В.; Ю, Х. Глубокое обучение с подкреплением для динамического гибкого планирования цеха со случайной Прибытие работы. Процессы 2022, 10, 760. [CrossRef]
64. Лю, Р.; Пиплани, Р.; Торо, К. Глубокое обучение с подкреплением для динамического планирования гибкого цеха. Межд. Дж. Прод. Рез. 2022, 60, 4049–4069. [Перекрестная ссылка]
65. Самсонов В.; Кеммерлинг, М.; Пагерт, М.; Люттик, Д.; Зауэрманн, Ф.; Гюцлафф, А.; Шух, Г.; Мейзен, Т. Управление производством в цехах с обучением с подкреплением. В материалах 13-й Международной конференции по агентам и искусственному интеллекту (ICAART 2021), онлайн, 4–6 февраля 2021 г.; SciTePress: Сетубал, Португалия; стр. 589–597.
66. Кунья, Б.; Мадурейра, АМ; Фонсека, Б.; Коэльо, Д. Глубокое обучение с подкреплением как средство планирования рабочего времени: обзор литературы. В материалах 18-й Международной конференции по гибридным интеллектуальным системам (HIS 2018), Порту, Португалия, 13–15 декабря 2018 г.; Мадурейра А.М., Абрахам А., Ганди Н., Варела М.Л., ред.; Springer International Publishing: Чам, Швейцария, 2018 г.
67. Ван, Х.; Чжан, Л.; Лин, Т.; Чжао, К.; Ван, К.; Чен З. Решение проблем планирования заданий в среде с высвобождением ресурсов с многоагентным обучением с подкреплением. Робот. Вычислить. Интегр. Производитель 2022, 77, 102324. [CrossRef]
68. Ох, Ш; Чо, Ю.И.; Ву, Дж. Х. Распределительное обучение с подкреплением с независимыми учащимися для решения задачи гибкого планирования рабочего времени с высокой изменчивостью. Дж. Компьютер. Дес. англ. 2022, 9, 1157–1174. [Перекрестная ссылка]
69. Чжан Ю.; Чжу, Х.; Тан, Д.; Чжоу, Т.; Гуй, Ю. Динамическое планирование цехов на основе глубокого обучения с подкреплением для многоагентных производственных систем. Робот. Вычислить. Интегр. Производитель 2022, 78, 102412. [CrossRef]
70. Лян, Ю.; Солнце, З.; Сонг, Т.; Чжоу, К.; Фан, В.; Фан, Дж.; Руи, Ю.; Чжоу, К.; Бай, Дж.; Ян, К.; и др. Lenovo планирует выпуск ноутбуков Производство с использованием глубокого обучения с подкреплением. Интерфейсы 2022, 52, 56–68. [Перекрестная ссылка]
71. Чен, Ю.; Го, В.; Лю, Дж.; Шен, С.; Лин, Дж.; Куи, Д. Подход к управлению охлаждением с несколькими заданными значениями для центров обработки данных с воздушным охлаждением с использованием алгоритм глубокой Q-сети. Измер. Контроль 2024, 57, 782–793. [Перекрестная ссылка]
72. Теат, Т.; Эрнст, Д. Применение глубокого обучения с подкреплением в алгоритмической торговле. Экспертная система. Прил. 2021, 173, 114632. [Перекрестная ссылка]
73. Санайе, С.; Саррафи, А. Новый метод управления энергопотреблением, основанный на алгоритме Deep Q Network, для низких эксплуатационных затрат интегрированная гибридная система. Energy Rep. 2021, 7, 2647–2663. [Перекрестная ссылка]
74. Луо, С. Динамическое планирование для гибкого цеха вакансий с добавлением новых вакансий посредством глубокого обучения с подкреплением. Прил. Мягкий компьютер. Дж. 2020, 91, 106208. [CrossRef]
75. Луо, С.; Чжан, Л.; Фан, Ю. Планирование в реальном времени для динамического многоцелевого гибкого цеха работ с частичным ожиданием без ожидания от Deep Обучение с подкреплением. IEEE Транс. Автомат. наук. англ. 2022, 19, 3020–3038. [Перекрестная ссылка]

Machine Translated by Google

76. Ху, Дж.; Ван, Х.; Тан, Х.-К.; Канадзава, Т.; Гупта, К.; Фарахат, А. Обучение с подкреплением знаний для интегрированного планирования производства и технического обслуживания на нескольких машинах. Вычислить. Индийский англ. 2023, 185, 109631. [CrossRef]
77. Ду, Ю.; Ли, JQ; Чен, XL; Дуань, ПЮ; Пэн, К.К. Обучение с подкреплением на основе знаний и оценка алгоритма распределения для задачи гибкого планирования цеха. IEEE Транс. Экстрен. Вершина. Вычислить. Интел. 2023, 7, 1036–1050. [Перекрестная ссылка]
78. Ли, Ю.; Гу, В.; Юань, М.; Тан, Ю. Динамическое планирование на основе данных в режиме реального времени для гибкого цеха с недостаточным транспортом. ресурсы с использованием гибридной сети Deep Q. Робот. Вычислить. Интегр. Производитель 2022, 74, 102283. [CrossRef]
79. Чжоу, Т.; Чжу, Х.; Тан, Д.; Лю, К.; Кай, К.; Ши, В.; Гуй, Ю. Обучение с подкреплением для онлайн-оптимизации планирования цеха на умной производственной фабрике. Адв. Мех. англ. 2022, 14, 16878132221086120. [CrossRef]
80. Ван, Л.; Ху, Х.; Ван, Ю.; Сюй, С.; Ма, С.; Ян, К.; Ван, В. Динамическое планирование цехов в интеллектуальном производстве с использованием глубокого обучение с подкреплением. Вычислить. Сеть. 2021, 190, 107969. [CrossRef]
81. Ван, Ю.; Ашер, Дж. М. Применение обучения с подкреплением для агентного планирования производства. англ. Прил. Артиф. Интел. 2005, 18, 73–82. [Перекрестная ссылка]
82. Кансино, Калифорния; Мериго, Х.М.; Коронадо, ФК Библиометрический анализ ведущих университетов в области инновационных исследований. Дж. Иннов. Знать. 2017, 2, 106–124. [Перекрестная ссылка]
83. Варин, К.; Каттелан, М.; Ферт, Д. Статистическое моделирование обмена цитированием между статистическими журналами. JR Стат. Соц. Сер. Стат. Соц. 2016, 179, 1–63. [Перекрестная ссылка] [ПабМед]
84. Мораль-Муньос, Х.А.; Эррера-Вьедма, Э.; Сантистебан-Эспехо, А.; Кобо, М.Дж. Программные средства для проведения библиометрического анализа в наука: Актуальный обзор. Проф. Де Ла Инф./Инф. проф. 2020, 29, e290103. [Перекрестная ссылка]
85. Карри, С. Давайте выйдем за рамки риторики: пришло время изменить то, как мы оцениваем исследования. Природа 2018, 554, 147–148. [Перекрестная ссылка] [ПабМед]
86. Аль-Хури, А.; Витта, Дж. П. Семь грехов исследований L2: обзор статистического качества 30 журналов и их показателей CiteScore, SJR, SNIP, Факторы воздействия JCR. Ланг. Учите. Рез. 2019, 23, 727–744. [Перекрестная ссылка]
87. Уолтман Л.; Ван Эк, Нью-Джерси; Нойонс, ECM. Единый подход к картированию и кластеризации библиометрических сетей. Ж. Информетр. 2010, 4, 629–635. [Перекрестная ссылка]
88. Ченг, Ф.; Хуанг, Ю.; Танпуре, Б.; Савалани, П.; Ченг, Л.; Лю, К. Экономичное планирование заданий для облачных экземпляров с использованием глубокого обучение с подкреплением. Класт. Вычислить. 2022, 25, 619–631. [Перекрестная ссылка]
89. Тайписутикул, Т.; Чен, Ю.-К.; Хуэй, Л.; Чен, Южная Каролина; Монгколват, П.; Ши, Т.К. Вопрос глубокого обучения с подкреплением для практических приложений ИИ. В материалах 12-й Международной конференции по Ubi-Media Computing, Бали, Индонезия, 5–8 августа 2019 г.; стр. 24–29.
90. Ян, Дж.; Хуанг, Ю.; Гупта, А.; Гупта, А.; Лю, К.; Ли, Дж.; Ченг, Л. Энергосберегающие системы для планирования заданий в режиме реального времени с использованием облачных данных центры: подход к глубокому обучению с подкреплением. Вычислить. Электр. англ. 2022, 99, 107688. [CrossRef]
91. Пиллер, Ф.Т. Массовая кастомизация: размышления о состоянии концепции. Межд. Дж. Флекс. Производитель Сист. 2004, 16, 313–334. [Перекрестная ссылка]
92. Сузич, Н.; Форза, К.; Трентин, А.; Анишич, З. Рекомендации по внедрению массовой кастомизации: текущие характеристики и предложения по улучшению. Прод. План. Контроль 2018, 29, 856–871. [Перекрестная ссылка]
93. Альтенмюллер, Т.; Штюкер, Т.; Вашнек, Б.; Кунле, А.; Ланца, Г. Обучение с подкреплением для интеллектуального и автономного управления производством в сложных цехах в условиях ограничений по времени. Прод. англ. 2020, 14, 319–328. [Перекрестная ссылка]
94. Чжао Ю.; Чжан Х. Применение машинного обучения и планирования правил в системе управления производством. Межд. Дж. Симул. Модель 2021, 20, 410–421. [Перекрестная ссылка]
95. Кунле, А.; Кайзер, JP; Тайсс, Ф.; Стрикер, Н.; Ланца, Г. Разработка адаптивной системы управления производством с использованием обучения с подкреплением. Дж. Интелл. Производитель 2021, 32, 855–876. [Перекрестная ссылка]
96. Панцер, М.; Бендер, Б.; Гронау, Н. Глубокое обучение с подкреплением в планировании и контроле производства: систематический обзор литературы. В материалах конференции по производственным системам и логистике, онлайн, 10–11 августа 2021 г.
97. Вояковски, П.; Варзолек Д. Классификация задач планирования в условиях производственной неопределенности. Рез. Логист. Прод. 2014, 4, 245–256.
98. Блэкстоун, Дж. Х.; Филлипс, DT; Хогг, Г.Л. Современный обзор правил диспетчеризации для производственных цехов. Межд. Дж. Прод. Рез. 1982, 20, 27–45. [Перекрестная ссылка]
99. Блажевич Дж.; Экер, К.Х.; Пеш, Э.; Шмидт, Г.; Вегларц, Дж. Справочник по планированию: от теории к приложениям; Springer: Берлин/Гейдельберг, Германия, 2007.
100. Иванов Д.; Долгий, А.; Соколов Б. Динамический подход к многоэтапному планированию цеха в гибкой сборочной системе на основе Индустрии 4.0. В достижениях в области систем управления производством. Путь к интеллектуальному, совместному и устойчивому производству: Международная конференция IFIP WG 5.7, APMS 2017, Гамбург, Германия, 3–7 сентября 2017 г., Материалы, Часть I; Springer International Publishing: Чам, Швейцария; стр. 475–482.
101. Модрак В. (ред.) Массовое производство по индивидуальному заказу: теоретические концепции и практические подходы; CRC Press: Бока-Ратон, Флорида, США, 2017.
102. Комаки, ГМ; Шейх, С.; Малакути, Б. Проблемы планирования производственного цеха со сборочными операциями: обзор и новые тенденции. Межд. Дж. Прод. Рез. 2019, 57, 2926–2955. [Перекрестная ссылка]
103. Ян, Ю.; Ли, Х. Конструктивный эвристический алгоритм, основанный на знаниях, для задачи планирования цеха с блокировкой распределенной сборки. Экспертная система. Прил. 2022, 202, 117269. [CrossRef]

Machine Translated by Google

104. Насири, ММ; Язданпараст, Р.; Джолай, Ф. Подход к оптимизации моделирования для планирования в реальном времени в условиях открытого цеха.
использование составного правила диспетчеризации. Межд. Дж. Компьютер. Интегр. Производитель 2017, 30, 1239–1252. [Перекрестная ссылка]
105. Абдельмагид, Т.Ф. Двухцелевое динамическое многопроцессорное планирование открытого цеха для технического обслуживания и медицинской диагностики. Эксперт
Сист. Прил. 2021, 186, 115777. [CrossRef]
106. Тремблет, Д.; Тевенин, С.; Долги, А. Оценка Makespan в гибкой среде планирования цеха с использованием машинного обучения.
Межд. Дж. Прод. Рез. 2024, 62, 3654–3670. [Перекрестная ссылка]
107. Фу, Ю.; Хоу, Ю.; Ван, З.; Ву, Х.; Гао, К.; Ван, Л. Проблемы распределенного планирования в интеллектуальных производственных системах.
Цинхуа наук. Технол. 2021, 26, 625–645. [Перекрестная ссылка]
108. Чжан Ю.; Чжан, Х.; Дин, Ю. Подход, основанный на обучении с подкреплением, к стохастическому планированию работы в магазине. IEEE Транс. Автомат.
наук. англ. 2020, 17, 72–83.
109. Ринчог А.; Мейер, А. На пути к стандартизации подходов к обучению с подкреплением для стохастического планирования производства. arXiv
2021, arXiv:2104.08196.
110. Цзэн, Б.; Ян, Ю. Гибридный подход к обучению с подкреплением и нечеткой логике для планирования работы цеха. Дж. Интелл. Производитель 2017,
28, 1189–1201.
111. Чжан Г.; Хуанг, Ю. Нечеткое обучение с подкреплением для многокритериального динамического планирования гибкой производственной системы. Дж.
Интел. Производитель 2005, 16, 293–304.

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Уменьшение смешивания на основе инверсии в области общей средней точки Использование радона высокого разрешения во временной области

Кай Чжуан [1], Дэниел Трад 1* и Амр Ибрагим [2]

1 Департамент Земли, энергетики и окружающей среды, факультет естественных наук, Университет Калгари, Калгари, AB T2N 1N4, Канада; kzhuang@ucalgary.ca
2 Кафедра физики, факультет естественных наук, Университет Бени-Суэфа, Бени-Суэф 62521, Египет;
amr.ibrahim@ucalgary.ca * Адрес для переписки: daniel.trad@ucalgary.ca

Аннотация: Мы реализуем метод устранения смешивания на основе инверсии в сборках общих средних точек (CMP) в качестве альтернативы стандартным методам доменов общих приемников (CRG). Основное преимущество устранения смешивания в области ОГТ заключается в том, что отражения от падающих слоев центрируются вокруг нулевых выносов. В результате сейсмограммы CMP имеют более простую структуру по сравнению с сейсмограммами обычных приемников (CRG), где эти отражения смещены по вершине. Следовательно, мы можем использовать гиперболический оператор Радона с нулевым выносом для обработки сейсмограмм ОСТ. Этот оператор является более эффективной в вычислительном отношении альтернативой гиперболическому радону со смещенной вершиной, необходимому для обработки сейсмограмм CRG. Разреженные преобразования, такие как преобразование Радона, могут суммировать отражения и создавать разреженные модели, способные разделять смешанные источники. Мы используем оператор Радона для разработки системы устранения смешивания на основе инверсии, которая включает ограничение разреженной модели. Включение ограничения разреженности в процесс инверсии повышает фокусировку преобразования и улучшает восстановление данных. Уменьшение смешивания на основе инверсии позволяет нам учитывать все наблюдаемые данные путем включения оператора смешивания в функцию стоимости. Наши примеры синтетических и полевых данных показывают, что уменьшение смешения на основе инверсии в области ОСТ может эффективно разделять смешанные источники.

Ключевые слова: деблендинг; редкий радон; инверсия

check for updates

Цитирование: Чжуан К.; Трад, Д.; Ибрагим, А. Инверсия Устранение смешивания в области общей средней точки с использованием радона высокого разрешения во временной области. Алгоритмы 2024, 17, 344. https://doi.org/10.3390/a17080344

Академический редактор: Фрэнк Вернер

Поступила: 16 мая 2024 г.
Пересмотрено: 25 июня 2024 г.
Принято: 5 августа 2024 г.
Опубликовано: 7 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Поиск эффективного сбора сейсмических данных является хорошо известной проблемой геофизических исследований. Одновременный сбор данных, также известный как смешанный сбор данных, представляет собой решение, предлагающее метод, который значительно сокращает время, затрачиваемое на поле, за счет одновременного включения нескольких источников. Этот метод, первоначально предложенный Гаротту [1] для вибросейсмических исследований, а затем расширенный Бисли и др. [2], Beasley [3] для морской среды, допускает взаимодействие источников во время сбора данных, которые впоследствии отделяются во время обработки. Несмотря на преимущества в экономии времени, существующие модели и алгоритмы устранения смешивания таких данных часто имеют ограничения.

Это исследование мотивировано необходимостью устранить эти ограничения и повысить точность и эффективность разделения сейсмических данных. Современные методы, включая случайное срабатывание и фазовое кодирование, проложили путь к новым подходам к сбору данных. Тем не менее, процесс уменьшения смешивания остается серьезным узким местом, причем преобладают две школы мысли: подход, основанный на шумоподавлении [4], который фокусируется на ослаблении неустойчивого и случайного шума , и подход, основанный на инверсии [5], который стремится реконструировать сигнал путем инвертирования процесса смешивания. В этой статье основное внимание будет уделено обработке данных методом случайной стрельбы.

Введение небольших случайных задержек во время срабатывания источника, как предложил Беркхаут [6], является критическим шагом в процессе устранения смешивания. Псевдо-уменьшение смешивания включает в себя переназначение смешанных кадров на нулевое время стрельбы и дублирование кадров в соответствии с количеством одновременных источников. Эта манипуляция из-за случайных задержек рендеринга

Machine Translated by Google

помехи от соседних источников в виде случайного шума в некоторых общих областях данных, оставляя только целевой сигнал для восстановления в виде когерентных событий. Затем псевдообработанные данные демонстрируют некогерентные структуры в доменах общей выборки средней точки (CMP), общей выборки приемника (CRG) и общей выборки смещений (COF), что облегчает процесс устранения смешивания. Существуют различные методы использования некогерентности смешанного шума, такие как работа Лина и Херрманна по разделению [7] и последующие методы, такие как шумоподавление Столта Радона [8], SSA-фильтрация [9], шумоподавление с использованием пространственно-изменяющегося медианного фильтра. [10], предварительная обработка с помощью дискретного кривлет-фильтра [11] и предварительная обработка с устойчивым пороговым регулированием в области FK [12], каждый из которых имеет свои преимущества и недостатки, часто борясь с любыми проблемами при эффективном различении сигнала и шума (источник помех) для подходов с шумоподавлением или очень дорогих и сложных операторов преобразования для подходов, основанных на инверсии.

В этой статье представлен подход устранения смешивания на основе инверсии с использованием домена CMP, который обещает высокоскоростное решение с использованием оператора преобразования Радона. Использование гиперболического преобразования Радона для устранения смешивания выполняет множество функций и существенно помогает в восстановлении сейсмического сигнала. Учитывая внутреннюю природу сбора сейсмических данных, акустические отклики, исходящие из недр Земли, проявляются как гиперболические приходы на поверхность. Гиперболическое преобразование Радона, объединяющее данные на основе гиперболических траекторий, идеально подходит для картирования сейсмических данных. Выбор домена CMP вместо традиционного метода домена CRG позволяет использовать более оптимизированную версию оператора преобразования Радона, что приводит к сокращению времени выполнения.

2. Теория

2.1. Оператор смешивания

Оператор смешивания позволяет нам переходить от смешанного домена, где источники перекрываются (так называемого супервыстрела), к несмешанному домену, где источники принадлежат соответствующим записям выстрелов, а время стрельбы для каждого выстрела скорректировано до нуля. Это можно представить как оператор смешивания Г, который содержит информацию о перекрытии и времени задержки . Затем мы можем представить наш несмешанный куб данных как D , а смешанные данные как b с помощью следующего соотношения [6,13]:

$$б = ГД. \quad (1)$$

Применяя сопряженный оператор смешивания к смешанным данным, мы получаем псевдоразмешанные данные в несмешанном пространстве, при этом каждый кадр занимает свой собственный кадр данных, а не несколько снимков на кадр данных. Это можно выразить следующим образом [6,1

$$D˜ = Г Hb. \quad (2)$$

Здесь D˜ представляет псевдоразбавленные данные в стандартных кадрах кадров. Роль матрицы смешивания заключается в объединении кадров со случайным временем задержки, эффективно накладывая отдельные данные записи кадров. Это также означает, что псевдо-устранение смешивания позволяет перераспределить данные по соответствующим кадрам. Это формирует чрезмерно полное представление данных с избыточной информацией, хранящейся в каждой псевдосмешанной исходной панели. Важно отметить, что сопряженный оператор не устраняет источник помех в данных; он просто копирует его.

Успешное разделение данных с использованием метода случайной активации зависит от включения случайных времен задержки. Эти задержки имеют решающее значение для преобразования помех источника в то, что напоминает случайный шум, при перераспределении данных с помощью оператора сопряженного смешивания. Очень важно, чтобы эти времена задержки были достаточно рандомизированы до такой степени, что в пересопоставленных областях данных не возникало никаких закономерностей, поскольку согласованные закономерности во временах задержки снижают эффективность нашего ограничения когерентности. Кроме того, задержки должны быть достаточно длительными, чтобы обеспечить разделение событий, придерживаясь минимальной длины задержки, определяемой функцией наименьшей частоты и кратности смешивания, как подробно опис

Machine Translated by Google

и Абма [14]. В то же время они должны быть достаточно короткими, чтобы мы могли получить преимущество в сокращении времени сбора данных.

Случайные задержки, вносимые в график смешивания и, следовательно, в оператор смешивания, имеют решающее значение для упрощения разделения. Когда к смешанному набору данных применяется оператор сопряженного смешивания (псевдо-уменьшение смешения), случайные задержки становятся решающими. Смешивание можно представить как операцию над стандартными данными сбора данных, которая вносит помехи от источника.

Влияние сложения на случайные времена задержки можно увидеть, когда данные сортируются в одну из других геометрических областей после псевдосглаживания. После применения оператора сопряженного смешивания можно увидеть когерентные события в дополнение к некогерентным, наблюдаемым до применения преобразования; эти когерентные события — это то, что мы стремимся извлечь, поскольку они представляют собой часть данных, время задержки которых правильно скорректировано, а спонтанные данные из-за мешающих выстрелов должны быть удалены.
Благодаря использованию метода инверсии и ограничения когерентности, такого как преобразование Радона с высоким разрешением, мы можем эффективно удалить мешающий сигнал и уменьшить смешивание наших данных.

2.2. Преобразование

Радона Чтобы решить недоопределенную проблему, наблюдаемую в задаче инверсии, мы используем гиперболическое преобразование Радона (HRT) для дальнейшего ограничения проблемы. Этот составной оператор, состоящий как из оператора смешивания, так и из оператора преобразования Радона, теперь переформулирует инверсию в задачу с разреженными ограничениями. HRT — это вариант преобразования Радона, который суммирует гиперболические базисные функции; в геофизике это уравнение определяется как функция времени прибытия, смещения источник-приемник и интервальной скорости, как показано Торсоном и Клербоутом [15]:

$$m(p, \tau) = \int_{h1}^{h2} d(h, t = \tau \sqrt{2} + p^2 h^2) dx. \quad (3)$$

Здесь m(p, τ) представляет собой данные пространства Радона, p обозначает медленность, τ — время двустороннего путешествия, h1 и h2 — нижний и верхний пределы смещения соответственно, а d — пространство данных, подлежащее преобразованию. . Медленность p определяется как обратная скорость 1/v. Картирование, основанное на гиперболических поступлениях, позволяет точно и разреженно представить сейсмические данные, поскольку сейсмические сигналы проявляются как гиперболические события в стандартных сейсмических исследованиях. Преобразование Радона использовалось в различных приложениях, включая анализ скоростей [15], подавление кратных волн [16], интерполяцию в частотной или временной области [17] и устранение смешивания посредством шумоподавления [8].

Хотя HRT хорошо подходит для сейсмических данных, он имеет несколько ограничений. Прежде всего, преобразование требует, чтобы гиперболические поступления были симметричными и центрированы в пространстве данных, что не обязательно имеет место в реальных данных. В действительности, наклон подземных структур приводит к смещению гиперболических вступлений вверх и вниз, что приводит к нецентрированным поступлениям. Симметрия подвергается сомнению двумя факторами: анизотропией и условиями съемки. Массивы поверхностных датчиков обычно не размещаются на идеально плоской поверхности, что приводит к отклонениям от гиперболических значений. Эту проблему можно легко решить с помощью коррекции статики трассировки, которая является частью стандартного процесса обработки. Для решения проблемы анизотропии можно использовать модифицированное уравнение для учета растяжения кривой в простых случаях. Смещение от централизованного расположения представляет собой довольно сложную проблему, имеющую множество потенциальных решений. Первое решение включает реализацию гиперболического преобразования Радона со сдвигом вер
В этом подходе для одного изображения данных создаются несколько панелей Radon, каждая из которых соответствует серии ожидаемых смещенных положений вершины. Хотя это решение является надежным и хорошо отображает сейсмические данные, оно значительно увеличивает время выполнения, поскольку нет быстрого способа создания панелей Radon, а время обработки умножается на количество сканируемых апексов. Второе решение проблемы центрирования включает преобразование данных в область CMP, где вершины центрируются даже для падающих отражений. Однако это

Machine Translated by Google

Решение имеет серьезный недостаток: домен CMP обычно имеет худшую выборку, чем другие домены данных, что делает наложение псевдонимов серьезной проблемой.

2.3. Деблендирование на основе инверсии

Учитывая наш уже определенный оператор преобразования, оператор отображения Г , который отображает пространство данных b в пространство модели D в уравнении (1), мы можем сформулировать проблему как задачу инверсии, которая обращает операцию [19]. Инвертируя эту операцию, нам нужно сформулировать целевую функцию, которая минимизирует несоответствие между данными и смоделированными данными, другими словами, минимизируя информацию модели, которая не соответствует данным . Эту целевую функцию часто представляют следующим образом [17]:

$$\begin{array}{l}\text{минимизировать } ||\\ D||2\ 2 \text{ С учетом } ||(b \quad \Gamma D)||2\ 2\ .\end{array} \tag{4}$$

Целевая функция для инверсии смешанных данных записывается как

$$J = ||b \quad \Gamma D||, \tag{5}$$

Прямое обращение оператора смешивания некорректно; таким образом, ни одно уникальное решение не может быть получено путем обращения только оператора смешивания Г. Такая некорректность обусловлена тем, что оператор смешивания служит только для сбора и сдвига данных, а сопряженный оператор служит только для их распределения. Чтобы обойти эту проблему, мы переформулировали схему инверсии, добавив оператор Радона в качестве ограничения когерентности.

$$J = ||b \quad \Gamma Lm|| + \mu||m||, \tag{6}$$

где L — преобразование Радона. Формулируя задачу таким образом, мы переходим от некорректной задачи к задаче с разреженными ограничениями.

2.4. Разреженная регуляризация

Мы используем разреженную регуляризацию для нашей схемы инверсии как способ определения нормы ℓ1 в модельном пространстве, как более надежное решение преобразования Радона [17] и, в свою очередь, схемы уменьшения смешивания. Для поиска разреженного решения в этой статье мы используем метод итеративно перевзвешенной инверсии наименьших квадратов (IRLS) [20]. IRLS — популярный способ поиска разреженных и/или надежных обратных решений из-за его простоты. По сути, IRLS переформулирует вычисление решения наименьшего абсолютного значения как повторение итераций метода наименьших квадратов с использованием весов в пространствах данных и модели. Этот подход более эффективен, чем прямое решение для решения наименьшего абсолютного значения, поскольку мы можем использовать более быстрые решатели наименьших квадратов, такие как сопряженный градиент, чтобы сократить время вычислений. Использование IRLS также позволяет выполнять инверсию смешанных норм; в этой статье мы используем норму ℓ2 в пространстве данных и норму ℓ1 в пространстве модели.

Как выяснили многие исследователи, применение разреженного ограничения к преобразованию Радона дает многочисленные преимущества и несколько недостатков при картировании сейсмических данных [17]. Эти улучшения решают такие проблемы, как отсутствие данных из-за наложения псевдонимов и ограниченность информации об апертуре при картировании и реконструкции. Преимущества также распространяются на использование преобразования Радона при уменьшении смешивания. Основное преимущество применения разреженного ограничения в пространстве модели для нашего метода заключается в том, что в качестве интеграла по пути неустойчивый и случайный шум от случайных времен задержки плохо отображается в пространстве Радона. Это плохое картографирование приводит к появлению размытий с низкой амплитудой в пространстве Радона, в отличие от фокальных точек с высокой амплитудой, наблюдаемых при когерентных гиперболических приходах. Это дополняется тем фактом, что в кадрах кадров , к которым принадлежат данные, сигналы суммируются и фокусируются правильно. Следовательно, решатель будет сходиться к самому разреженному решению, уменьшая вес размытого смешанного шума и увеличивая вес когерентных фокусных точек. Это также наиболее упрощенное решение, поскольку данные перераспределяются в те кадры, где они выглядят наиболее связными, а шум смешивания полностью объясняется смешиванием соседних кадров, что эффективно у

Machine Translated by Google

IRLS существует как способ косвенного решения разреженного решения задачи инверсии, представляя его как несколько итераций задачи наименьших квадратов, в данном случае сопряженного градиента (CG), при этом внешние итерации обновляют данные и веса модели, чтобы получить CG. Алгоритм решен, приведя решение к ℓ1-норме. Применение нормы ℓ1 в пространстве данных позволяет снизить чувствительность к выбросам, которые в противном случае отрицательно повлияли бы на инверсию, часто называемую устойчивой инверсией. Применение нормы ℓ1 в пространстве модели позволяет найти самое разреженное решение, в основном модель, которая объясняет данные с наименьшим количеством точек данных; это называется разреженным решением. Использование IRLS позволяет нам смешивать и сопоставлять нормы, которые мы хотим использовать, посредством расчета весов и, таким образом, можем найти решение смешанной нормы.

Добавляя веса модели и данных, мы можем затем продиктовать алгоритм, который будет сходиться к решению наименьшего абсолютного значения (ℓ1) или методу наименьших квадратов (ℓ2), переформулировав исходное уравнение наименьших квадратов путем минимизации модифицированного уравнения:

$$\text{Минимизировать } ||Wmm||_2^2 \text{ При условии } ||Wr(b - \Gamma Lm)||_2^2. \quad (7)$$

где Wr — оператор веса данных, а Wm — оператор веса модели. Эти веса позволяют пользователю отрегулировать предпочтение либо остатку, либо модели, где Wm можно изменить, чтобы получить определенное соответствие либо разреженной модели ℓ1 , либо гладкой модели ℓ2 , а Wr можно изменить, чтобы отдать предпочтение неспонтанным событиям или «весу». » все события одинаково во входных данных. Операторы обычно представляют собой диагональные матрицы. Например, Wr можно построить следующим образом:

$$\text{диаг}(Wr) = |r|^{(p-2)/2} \quad \text{где } r = \Gamma Lm - b. \quad (8)$$

где p равно 1 или 2 для измерения размера модели с использованием нормы данных ℓ1 или ℓ2 . Wm обычно строится как

$$\text{диаг}(Wm) = |m|^{(2-p)/2}. \quad (9)$$

где p равно 1 или 2 для инвертирования на основе нормы модели ℓ1 или ℓ2 . Поскольку веса данных создаются посредством деления, коэффициент демпфирования выбирается во избежание деления на ноль. Коэффициент демпфирования определяется как процентиль данных. Основное преимущество подхода инверсии для устранения смешивания состоит в том, что весь сигнал и шум смешивания можно объяснить в других областях, где они кажутся когерентными, в отличие от подхода шумоподавления, который стремится удалить шум смешивания кадр за кадром.

Мы также можем использовать разреженную инверсию, чтобы продемонстрировать подход к устранению шума с помощью преобразования Радона для удаления шума отдельных сейсмограмм приемников. Подход к шумоподавлению сначала применяет псевдоудаление смешанных данных для возврата набора данных в стандартных кадрах сбора данных, как показано на рисунке 1b, что затем позволяет разделить снимки посредством шумоподавления в других областях. Общий процесс шумоподавления с использованием разреженного/робастного Radon сначала применяет оператор сопряженного смешивания к смешанным данным, b:

$$\tilde{D} = b\Gamma^{\text{час}}, \quad (10)$$

Затем псевдообработанные данные используются при инверсии со следующей целевой функцией для шумоподавления с использованием преобразования Радона:

$$||Wr(\tilde{D} - Rm)||_p^p + \mu ||Wmm||_q^q, \quad (11)$$

где $\tilde{D}$ — псевдоразбавленные данные, R — оператор преобразования Радона, m — модель, μ — компромиссный параметр, а p и q — нормы данных и модели соответственно. Графическое представление шумоподавления радона можно увидеть на рисунке 2 с использованием стандартных

Machine Translated by Google

гиперболический Радон для уменьшения спонтанных событий в записи выстрелов в попытке восстановить когерентный сигнал.

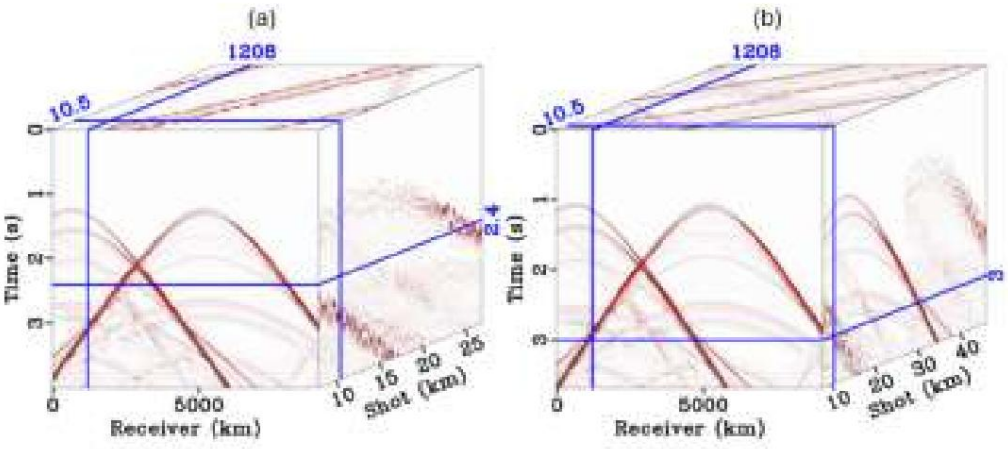


Рисунок 1. Применение сопряженного оператора: (a,b) – результат; в других геометрических областях можно увидеть , что наша цель когерентна, в то время как мешающие сигналы остаются некогерентными.

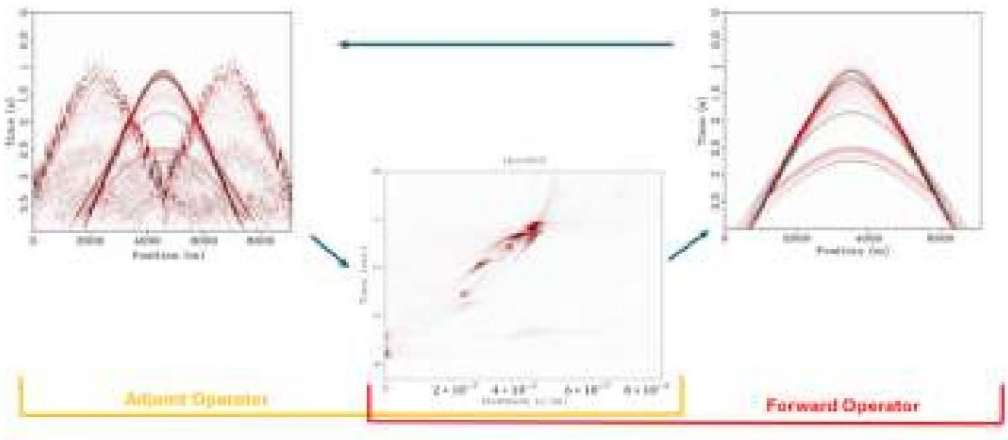


Рисунок 2. Подход к уменьшению смешивания с шумоподавлением. Подход к уменьшению шумоподавления использует разреженный/надежный Radon для удаления шума смешивания из каждого псевдо-облегченного кадра приемника.

Учитывая, что шум смешивания от одной записи выстрела является сигналом от другого, наш спорадический шум смешивания имеет те же амплитуды, что и сам выстрел. Эта проблема при высокой кратности смешения проявляется в том, что высокоамплитудный шум полностью подавляет участки сигнала с низкой амплитудой, что делает традиционные методы восстановления на основе фильтрации практически невозможными. Однако использование инверсии позволяет нам объяснить мешающие события с высокой амплитудой и эффективно переназначить только мешающую амплитуду, что позволяет нам восстановить сигналы с более низкой амплитудой. Схему подхода, основанного на инверсии, можно увидеть на рисунке. 3. Однако этот подход не лишен своих недостатков, в основном связанных с требованиями к памяти, поскольку весь набор данных смешивания плюс соответствующая модель должны храниться в памяти. Несмотря на первоначальные ожидания, этот метод не имеет существенной разницы во времени выполнения по сравнению с аналогичным подходом к шумоподавлению на основе инверсии, поскольку инверсия должна выполняться в обоих случаях. По сравнению с подходом шумоподавления Радона, подход инверсии использует другую целевую функцию, поскольку он стремится включить в инверсию как оператор с Мы можем сформулировать целевую функцию, используя уже обсуждавшиеся веса, а также уравнение (6):

$$J = \| Wr(b \quad \Gamma Lm) \| p + \mu \| Wmm \| q \quad _{q}. \tag{12}$$

Machine Translated by Google

Здесь входными данными являются не псевдоразбавленные данные, а смешанные данные, и оператор содержит как операторы смешивания, так и операторы Радона.

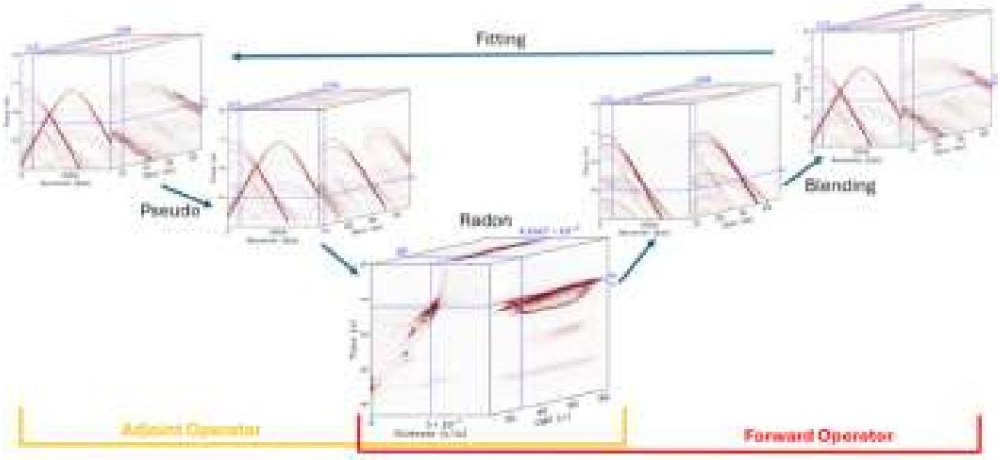


Рисунок 3. Схема устранения смешивания на основе инверсии: объединение оператора смешивания с оператором Radon Преобразование позволяет нам преобразовывать спорадические помехи смешивания обратно в когерентный сигнал в его правильное расположение, эффективно объясняющее, а не устраняющее помехи.

Реализовав устранение смешения как операцию, основанную на инверсии, используя метод смешивания и Радона вместе, система действует как переполненный словарь, где информация представлена и избыточна в псевдообъединенных кадрах. Это помогает в конвергенция, поскольку избыточная информация объясняет друг друга посредством переназначения данные обратно в соответствующие кадры кадров.

3. Вопросы высокопроизводительных вычислений (HPC)

Разреженное гиперболическое преобразование Радона используется в нашем алгоритме устранения смешивания для переформатирования некорректного оператора смешивания в задачу инверсии, использующую разреженную регуляризацию. Этот оператор Radon позволяет нам оценить производительность высокопараллелизуемой системы. операции с использованием ряда параллельных методов обработки. Чтобы правильно закодировать HRT для эффективно работать в openMP, мы отправляем внешние циклы смещения в отдельные потоки для вычисления преобразовать в сопряженный оператор; для прямого оператора вместо этого мы отправляем каждый внешний цикл медленности. Псевдокод преобразования Радона можно записать, как показано на рисунке.

Алгоритм 1.

Алгоритм 1. Псевдокод Радона.

```
1: функция РАДОН
2: #PRAGMA OMP PARALLEL FOR для q = медлительность          вставка для распараллеливания цикла
3:     do
4:         для h = смещение do
5:             перемещение = h 2 q
6:             для этого = от 0 до nt do
7:                 время = (это dt) 2 + переезд
8:                 модель[iqNt + it]+ = данные[ihNt + INT(время/dt)]
9:             конец для
10:        конец для
11: конец для
12: конечная функция
```

Для нашей реализации в CUDA мы настроили двумерную конфигурацию потоков. с одним измерением для каждого измерения времени и медленности в стандартном коде, затем вычислил преобразование независимо для каждой выходной ячейки для каждого потока. Для реализации HRT в openMPI, мы отправляем данные, относящиеся к каждому кадру преобразования, в отдельные

Machine Translated by Google

узлов, а затем вычислить преобразование независимо на каждом узле, используя методы CUDA или openMP, указанные выше. Затем мы собираем данные у мастера в конце расчета . Псевдокод CUDA можно увидеть в алгоритме 2.

Алгоритм 2. Псевдокод CUDA Radon 1: __global__ 2:

```
функция CUDA_
RADON int it = blockIdx.x * blockDim.x
    + threadIdx.x int iq = blockIdx.y * blockDim.y + threadIdx.y
    if (iq >= nq || it >= nt) return int iqNt = iq     nt double
    timemax = dt     nt for h=offset do int
    ihNt = ih     nt double
    moveout = h 2     q double time
    = (it     dt)
        model[iqNt + it] + =
        данные[ihNt + (INT)(время/dt)]
        13: конец для 14: завершение 2 + переезд
        функции
```

Поскольку значительное количество параллельных приложений используют многомерные данные, API CUDA группирует блоки потоков по нескольким измерениям, вплоть до трех. Это полезно для реализации Radon, поскольку обычно существует три измерения данных для преобразования: по одному для смещения, времени и скорости. Однако, присвоив каждой из наших переменных измерению , мы можем исключить соответствующие циклы, но это не самый эффективный способ реализации алгоритма. Поскольку модель представляет собой накопление по измерению смещения, мы можем столкнуться с гонкой данных, если измерение смещения h будет выполняться параллельно. Если бы смещение было установлено в качестве третьего измерения потока в CUDA, потребовался бы атомарный оператор; Первоначально он был протестирован и дал значительно более медленные результаты, чем алгоритм, представленный выше. Распараллеливая только два измерения сетки, мы избегаем необходимости использовать атомарные операторы накопления, которые могут вызвать значительное замедление, поскольку каждый поток выполняет работу для каждой ячейки сетки, что можно назвать оптимизацией с ориентацией на выходные данные.

Этот алгоритм интересен тем, что большую часть его можно оптимизировать параллельно, но при правильном распараллеливании существует множество проблем. Основная проблема оптимизации и распараллеливания алгоритма связана с большими объемами подкачек памяти, необходимыми для каждой итерации. Эти проблемы с памятью возникают из-за постоянной необходимости менять домены в операторе смешивания, что приводит к крупномасштабному непоследовательному доступу к данным. Другая проблема, с которой мы сталкиваемся, заключается в том, что этот алгоритм невозможно правильно распараллелить в системах с распределенной памятью (серверах) с использованием openMPI, поскольку расчет градиента на каждой итерации необходимо рассчитывать с использованием смешанного оператора, который требует объединения всех данных. Поскольку распределенная память в кластерах требует разделения набора данных по узлам вместо того, чтобы хранить их все в оперативной памяти главного узла, процесс сортировки создает очень сложную задачу для реализации openMPI. Из-за этого требования каждому узлу в кластере необходимо будет отправлять данные обратно после каждого расчета для расчета следующего шага, а затем мастеру необходимо будет повторно отправить результаты обратно на каждый узел. Для нашей реализации в CUDA вместо использования нескольких узлов мы используем одну высокопроизводительную машину, где как Radon, так и алгоритм спуска по методу наименьших квадратов переписаны с использованием ядер C

4. Результаты

Поскольку наши тестовые данные представляют собой в основном синтетические и морские данные, мы предполагаем, что пространство данных не содержит выбросов; таким образом, мы использовали ℓ2 в пространстве данных для инверсии смешанной нормы ℓ2-ℓ1 . Для нашего тестирования мы использовали три набора данных, два из которых были созданы синтетически с помощью конечной разницы, и один реальный набор морских данных из Мексиканского залива, который был численно смешан. Для синтетических наборов данных было проведено конечно-разностное моделирование с использованием 300 снимков и 300 приемников, равномерно расположенных на поверхности. Первый набор данных и самый упрощенный

Machine Translated by Google

— это набор данных модели клина, показанный на рисунке 4. Модель клина состоит из двух слоев, одного треугольного клина и набора жестких точек для проверки реконструкции дифракций низкой амплитуды. Данные клина были смешаны с двумя перекрывающимися снимками со временем задержки, рандомизированным между 0 и 400 выборками.

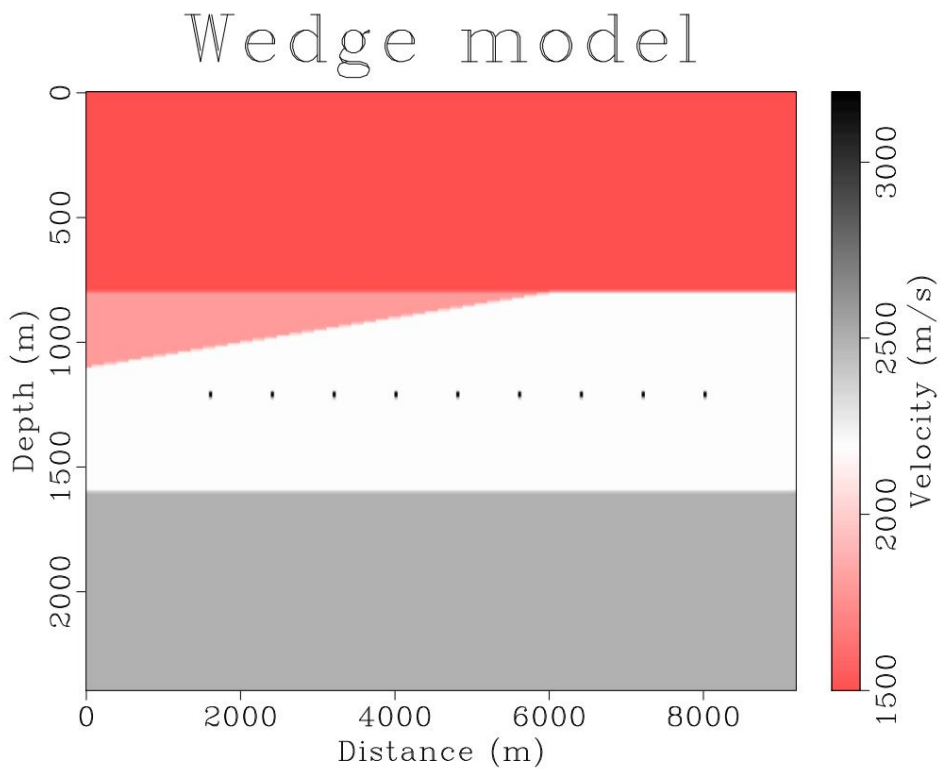


Рисунок 4. Скоростная модель 1: модель клина, состоящая из двух точек дифракции слоя 9 и одного клина.

Уменьшение смешивания можно увидеть на рисунке 5, где шум смешивания, видимый в псевдообработанных данных на рисунке 5b, удаляется посредством инверсии на рисунке 5c. Впоследствии результаты были преобразованы в область выстрелов, как показано на рисунке 6.

Несмотря на некоторые остаточные помехи в смешанном кадре, процесс уменьшения смешивания эффективно удалил вторичный мешающий кадр и хорошо сохранил целевой кадр, включая дифракции. Их можно наблюдать в верхней части рисунков 6b,c.

Сравнение нашего метода инверсии и метода шумоподавления показано на рисунке 7. Метод шумоподавления также использует разреженную HRT в качестве механизма шумоподавления для удаления случайного шума смешивания в псевдообработанных сейсмограммах. Основное отличие от метода инверсии заключается в том, что метод шумоподавления не включает оператор смешивания. В области CMP высокие амплитуды шума смешивания затрудняют прямое удаление с помощью обычных методов шумоподавления. Это связано с тем, что отличить шум от данных становится все труднее, поскольку шум смешивания имеет ту же амплитуду, что и сигнал, который мы стремимся восстановить. Без дополнительной информации, которую предоставляет оператор смешивания, это сходство затрудняет различение того, какие части данных являются шумом, а какие — когерентными сигналами. Как можно видеть на рисунке 7, результат шумоподавления успешно очистил данные, но за счет данных с более низкой амплитудой дифракционных вступлений, которые были удалены вместе с шумом смешивания.

Machine Translated by Google

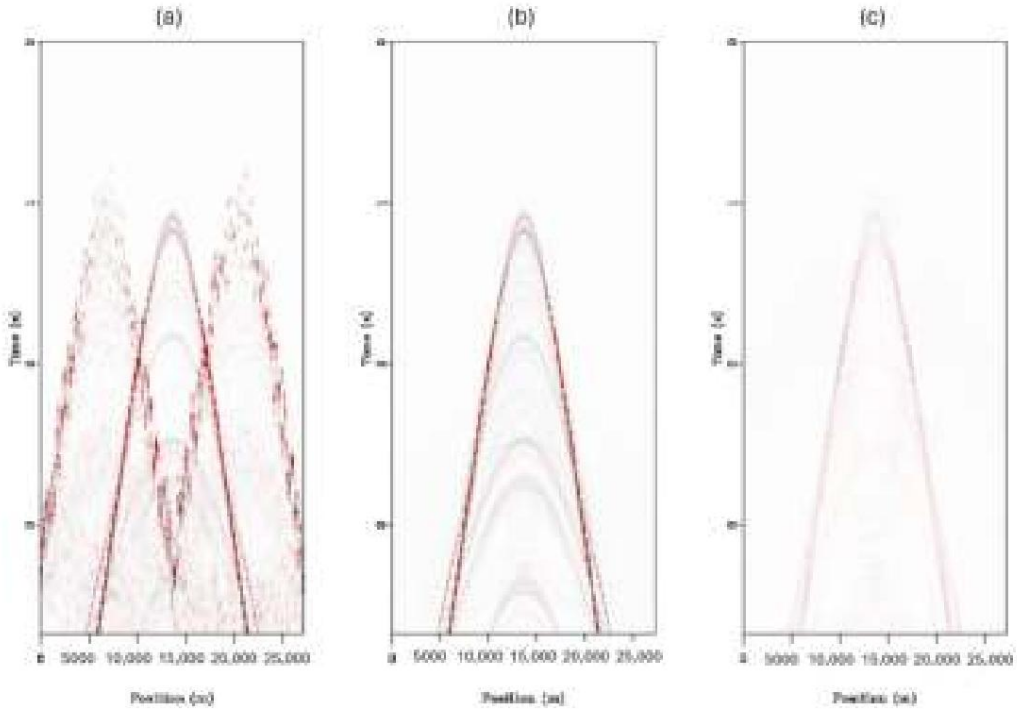


Рисунок 5. Результаты теста модели клина (CMP): тест на устранение смешивания 1 в области CMP с (a) псевдообъединенными данными, (b) очищенными данными и (c) показанной ошибкой реконструкции соответственно.

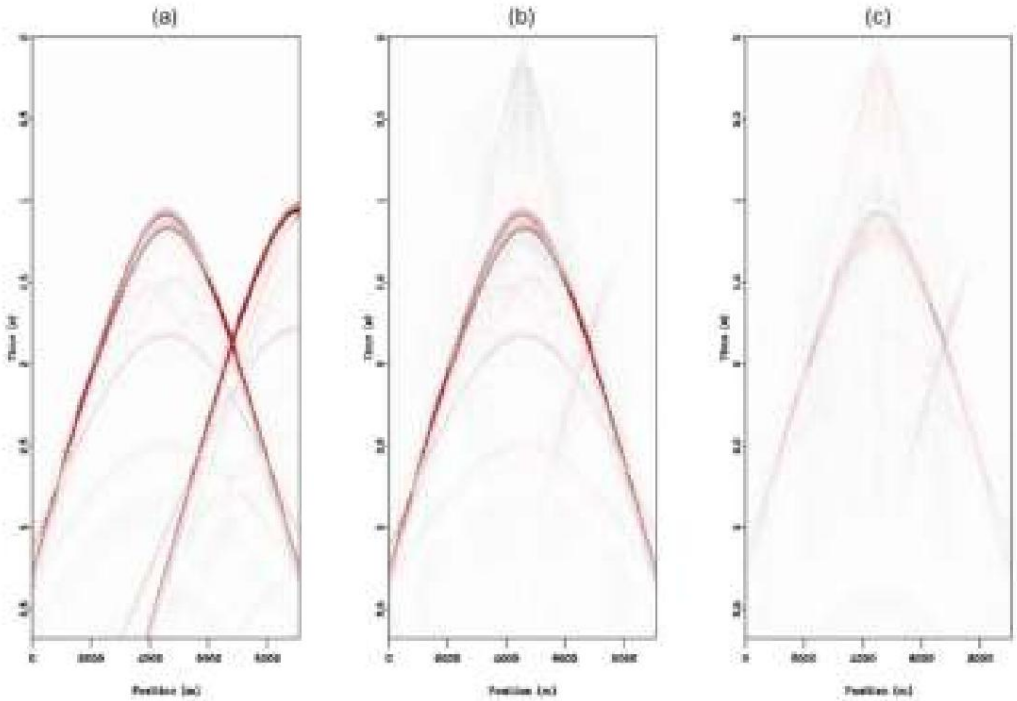


Рисунок 6. Результаты испытаний модели клина (область выстрела): Тест на разбавление 1 в области выстрела с (a) псевдообработанные данные, (b) очищенные данные и (c) показанная ошибка реконструкции соответственно.

Machine Translated by Google

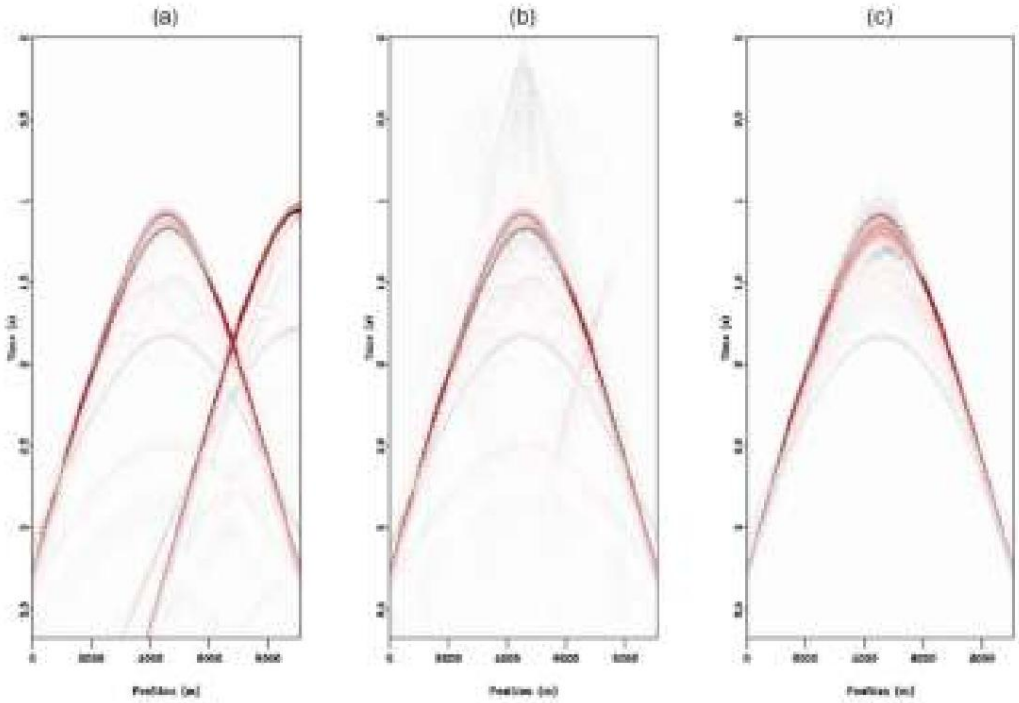


Рисунок 7. Сравнение результатов модели клина между подходом инверсии и шумоподавления: подходы инверсии и шумоподавления в области выстрела, с (a) смешанным, (b) инверсией с уменьшенным шумоподавлением и (c) уменьшенным шумоподавлением.

Синтетический тест номер 2 был сгенерирован посредством конечной разности с использованием стандартной модели Мар-Муси [21]. Поскольку тест 1 показал, что наш механизм уменьшения смешивания эффективно удаляет наши данные, мы решили сделать тест номер 2 скорее стресс-тестом, в котором метод удаления смешивания, не основанный на инверсии, столкнется с трудностями при эффективном уменьшении смешивания. Мы значительно увеличили сложность, сократив диапазон случайных времен задержки, а также увеличив количество перекрывающихся кадров с двух до пяти — условий, которые мы сейчас не ожидаем увидеть в реальном мире. На рисунке 8a показаны смешанные данные в области CMP; Легко представить, насколько сложно обычному шумоподавителю без оператора смешивания отличить данные от интерференции на этом изображении.

Анализируя результаты второго теста, мы видим на рисунке 8 , что наше восстановление отражений низкой амплитуды эффективно даже в том случае, когда они полностью подавлены высокоамплитудными помехами от других импульсов. Взглянув на ошибку реконструкции области выстрелов на рисунке 9, мы видим, что основные различия заключаются в уменьшении амплитуд событий с более высокой амплитудой и введении фонового шума с очень низкой амплитудой; мы полагаем, что ошибка амплитуды может возникнуть из-за слишком большой разреженности пространства модели, что обычно вызывает эти проблемы в Radon с высоким разрешением, или из-за неполного разделения снимков смешивания и размазывания более высоких амплитуд по данным в качестве фонового шума.

Нашим последним тестом был реальный набор данных, полученный из Мексиканского залива, известный как набор данных BP Tiber WATS [8,22], который затем был численно смешан. Схема смешивания была настроена иначе, чем другие, поскольку это также была проверка другого графика сбора данных, известного как непрерывное прослушивание. При непрерывном прослушивании у нас фактически есть один суперкадр, поскольку данные записываются до тех пор, пока не будут произведены все выстрелы и выстрелы не перекроются. Урезанный набор данных состоял из 90 снимков и 90 приемников с перекрытием между кадрами 70%, рандомизированных между 0 и 200 выборками. Результаты разделения набора данных Мексиканского залива можно увидеть в домене CMP на рисунке 10 и в домене выстрелов на рисунке 11. Анализируя результаты, мы видим, что алгоритм уменьшения смешивания исключительно эффективен при разделении выстрелов на реальных объектах. мировой набор данных с хорошим восстановлением всех событий, наблюдаемых в данных. Основная ошибка, которую мы можем наблюдать, заключае

Machine Translated by Google

В данной области выстрелы не полностью очищены, и некоторая амплитуда от заднего выстрела все еще видна в восстановленном кадре. Это можно наблюдать около 5,2 с на рисунке 11c.

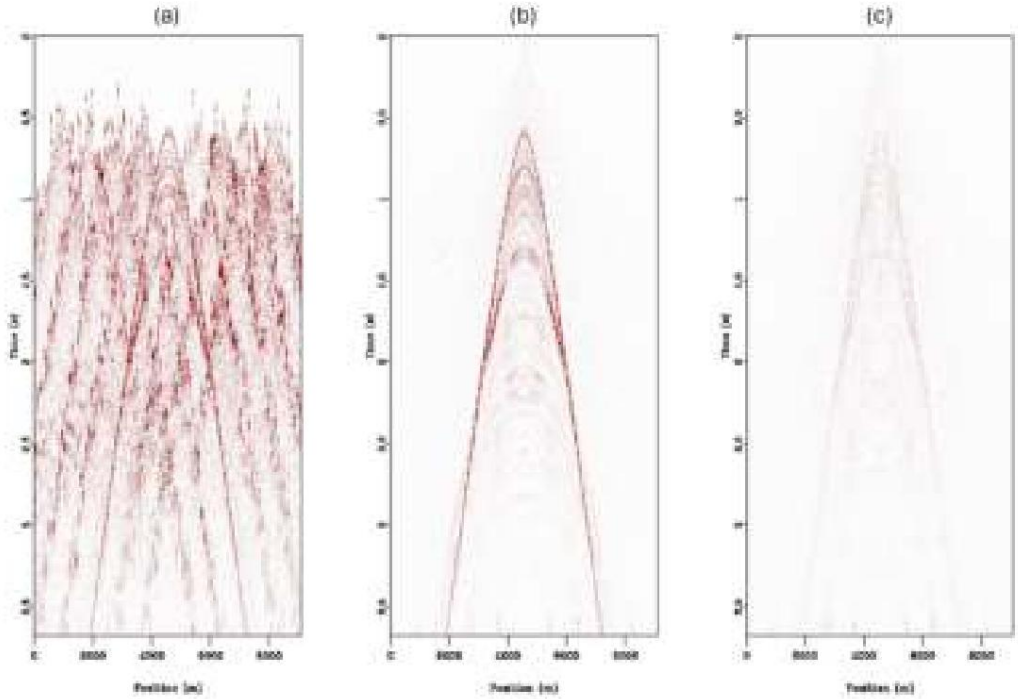


Рисунок 8. Результаты испытаний модели Мармуси (CMP): Тест на устранение смешивания 2 с 5 выстрелами, с (a) смешанной, (b) уменьшенной и (c) ошибкой реконструкции соответственно.

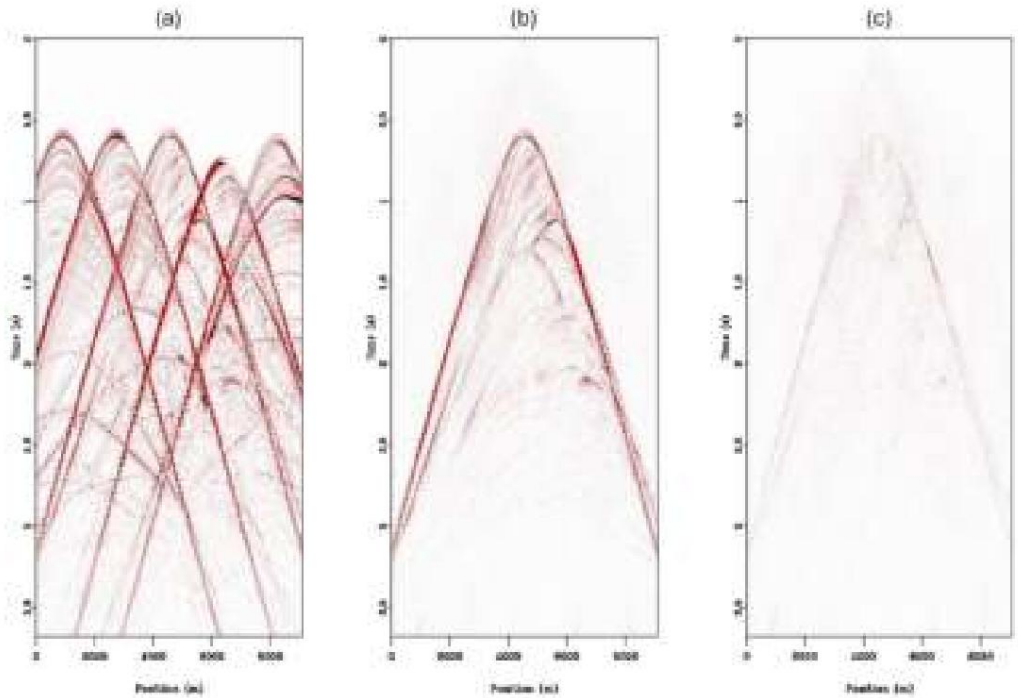


Рисунок 9. Результаты испытаний модели Мармуси (область выстрелов): тест на устранение смешивания 2 с 5 выстрелами, с (a) смешанным, (b) уменьшенным и (c) ошибками реконструкции соответственно.

Machine Translated by Google

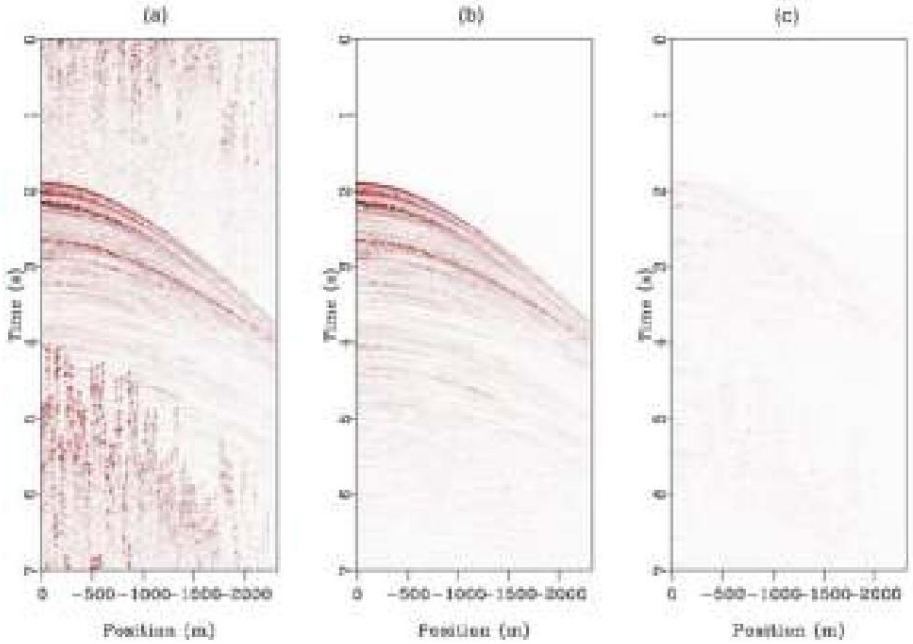


Рисунок 10. Данные Мексиканского залива в домене CMP: Устранение смешивания с использованием набора морских данных Мексиканского залива с результатами в домене CMP, со смешанными данными в (a), уменьшенными в (b) и ошибкой реконструкции в (c).

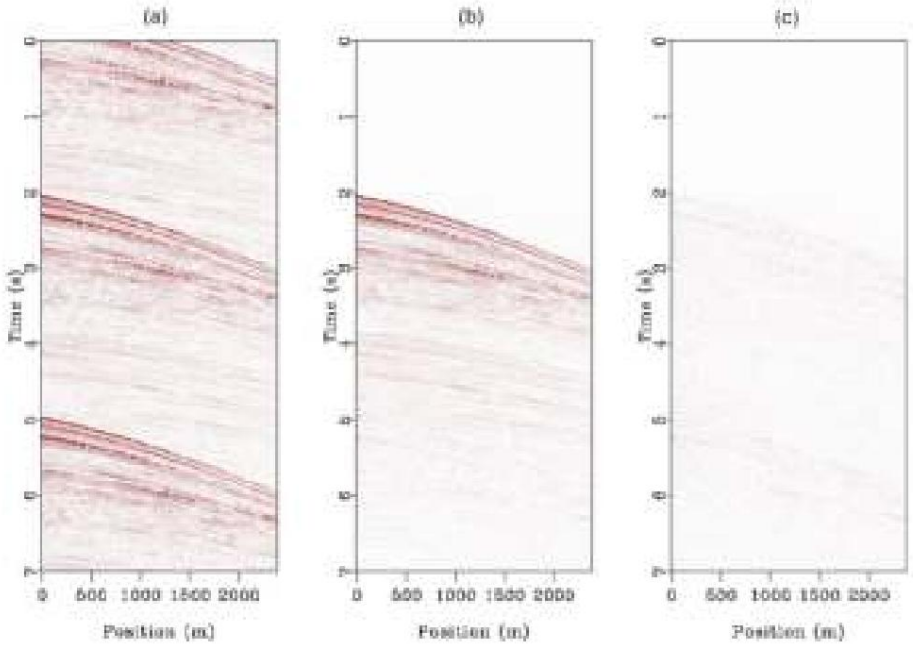


Рисунок 11. Данные Мексиканского залива: устранение смешивания с использованием набора морских данных Мексиканского залива с интерференцией времени выстрела 70%, со смешанными данными на (a), уменьшенными на (b) и ошибкой реконструкции на (c).

Ограничения этого подхода заключаются в предположениях, присущих использованию гиперболического преобразования Радона. Основным предположением этого подхода является гиперболическая траектория всех сейсмических сигналов, которые, как ожидается, будут точно представлены в пространстве Радона. Однако это предположение становится проблематичным при рассмотрении уникальных характеристик наземных сейсмических данных. Статика, присущая наземным данным, может вызвать вертикальные сдвиги в образцах, отклоняющиеся от их гиперболического выравнивания. Такие несоответствия приводят к неэффективному картированию области радона и потенциально ставят под угрозу целостн

Machine Translated by Google

Мы запустили алгоритм устранения смешивания и алгоритм преобразования Радона в диапазоне различное оборудование для демонстрации производительности алгоритмов с использованием разных методов распараллеливания в разных условиях: ноутбук, две рабочие станции и наша собственная кластер серверов. В ноутбуке используется 6-ядерный процессор Intel i7 8750H с видеокартой Nvidia RTX 2060. Графический процессор MAX-Q, первая рабочая станция оснащена AMD Threadripper 3960× 24- основной процессор с Nvidia RTX 3080Ti, а вторая рабочая станция оснащена Intel i7 9800×. 8-ядерный процессор, оснащенный графическим процессором Nvidia RTX 2060-super. Эти результаты можно увидеть в Таблице 1. Время выполнения между преобразованием Радона и алгоритмом устранения смешивания нельзя сравнивать напрямую, поскольку они выполняли разные задачи; относительная разница во времени выполнения на разных аппаратных средствах показывает, какое влияние оказывает сортировка на разные реализации.

Таблица 1. Таблица среднего времени работы для удаления радона и метода наименьших квадратов на разных машинах.

Процессор (API)	Время преобразования радона (с)	LS Время размешивания (с)
i7 6800k 6-ядерный (openMP)	4462	2590
i5 8600 6-ядерный (openMP)	3773	1748 г.
i7 8750H 6-ядерный (openMP)	4044	2203
i7 9800× 8-ядерный (openMP)	2715	1390
TR 3960 × 24 ядра (openMP)	1123	581
RTX 3060 (CUDA)	147	174
RTX 3080Ti (CUDA)	108	142
RTX 2060 Макс-Q (CUDA)	184	206
RTX 2060 Супер (CUDA)	157	198
TP 3960× (openMPI)	420	—

Видно, что переход от
От обработки процессора (openMP) до обработки на базе графического процессора (CUDA). Результаты openMPI показывают значительное ускорение по сравнению с результатами openMP и сокращение разрыва с CUDA результаты; это ускорение можно в основном объяснить более грубым характером openMPI. Реализация openMPI распределяет между потоками меньше ресурсов; таким образом, в целом меньше управление памятью должно выполняться системой за счет общего объема оперативной памяти. по мере создания большего количества копий данных. В стандартном тестировании преобразования Радона программа на основе CUDA код сократил время вычислений более чем на порядок по сравнению с openMP версия; после введения алгоритма наименьших квадратов преимущество уменьшилось на большая сумма. Благодаря распределению всей работы openMPI, включая оптимизацию LS между узлами, он может запускать оптимизацию LS параллельно, что ни ни openMP, ни код CUDA не могут, что приводит к общему сокращению времени вычислений по сравнению с кодом CUDA. openMP и подходы к реализации CUDA. Однако ослабление OpenMPI не может быть реализовано должным образом, поскольку главный узел должен собирать все данные с другого узла. узлы для выполнения расчета LS на каждой итерации; это приводит к большим передачам памяти по сети, что приводит к значительным задержкам и снижению скорости передачи данных. Накладные расходы связанный с реализацией openMPI, настолько велик, что мы не заметили преимущества в нашем прототипировании программы по сравнению даже с однопоточной средой выполнения. Это однако следует отметить, что openMPI использовал в этом упражнении самый большой объем оперативной памяти, около 122 ГБ по сравнению с реализациями openMP и CUDA, которые использовали 8 ГБ ОЗУ и 8 ГБ + 8 ГБ (зеркальной) видеопамяти соответственно. Главный компромисс между Подход на основе CUDA/GPU и подход OpenMP будут подходом скорости по сравнению с портативность; подход CUDA значительно быстрее, но требует графического процессора Nvidia, в то время как подход OpenMP будет работать на любом устройстве, поддерживающем распараллеливание OpenMP в компилятор, которым является любой современный процессор, существующий на рынке.

5. Выводы

Мы предложили метод устранения смешивания, который использует линейную инверсию составного оператор преобразования в домене CMP. Эта инверсия при использовании с наложением

Machine Translated by Google

позволяет нам прогнозировать и переназначать, а не устранять мешающую информацию о выстреле. Это, в свою очередь, облегчает точное восстановление событий малой амплитуды даже в сильно перемешанных данных. Реализация разреженного ограничения на алгоритм инверсии позволяет прийти к уникальному решению, которое повышает когерентность сигнала в гиперболических поступлениях. Кроме того, использование гиперболических преобразований Радона в области CMP дает нам оператор, который эффективен и хорошо соответствует нашим данным, обеспечивая согласованность в нашем восстановлении и обходя проблемы наложения псевдонимов. Следовательно, мы уверены , что наш подход инверсии является эффективной стратегией для уменьшения смешивания одновременно полученных данных, сохраняя при этом события малой амплитуды. Мы также изучили дополнительные методы ускорения этой платформы с помощью приложений как для ЦП, так и для графических процессоров, каждое из которых имеет свой набор преимуществ и недостатков.

Вклад авторов: Концептуализация, KZ и DT; методология, КЗ и ДТ; программное обеспечение, Казахстан; валидация, КЗ, ДТ и ИИ; формальный анализ, КЗ; расследование, КЗ; курирование данных, KZ и AI; письменное – подготовка оригинального черновика, КЗ; написание — рецензирование и редактирование, AI и DT; руководство, Д.Т. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Мы благодарим спонсоров Консорциума исследований в области сейсмологии упругих волн (CREWES) за их постоянную поддержку. Эта работа финансировалась промышленными спонсорами CREWES и NSERC (Совет по естественным наукам и инженерным исследованиям Канады) посредством гранта CRDPJ 543578-19.

Заявление о доступности данных: В этом исследовании не было создано или проанализировано никаких новых данных. Совместное использование данных не применимо к этой статье.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Сокращения

В данной рукописи использованы следующие сокращения:

COF	Общее смещение
CMP	Общая средняя точка
CRG	Общий приемник
HRT	Гиперболическое преобразование Радона
ASHRT	Гиперболическое преобразование Радона со сдвигом апекса
~~	Сопряженный градиент
ИРЛС	Итеративный перевзвешенный метод наименьших квадратов
Унифицированная архитектура устройств CUDA Compute (API программирования Nvidia)	
API	Интерфейс прикладного программирования

Ссылки

1. Гаротту, Р. Одновременная запись нескольких вибросейсмических сейсмических профилей. В расширенных тезисах технической программы SEG, 1983 г.; Общество геофизиков-разведчиков: Хьюстон, Техас, США, 1983 г.; стр. 308–310. [Перекрестная ссылка]
2. Бизли, CJ; Чемберс, RE; Цзян, З. Новый взгляд на одновременные источники. В расширенных тезисах технической программы SEG, 1998 г.; Общество геофизиков-разведчиков: Хьюстон, Техас, США, 1998 г.; стр. 133–135. [Перекрестная ссылка]
3. Бисли, К.Дж. Новый взгляд на морские одновременные источники. Вести. Край 2008, 27, 914–917. [Перекрестная ссылка]
4. Махдад А.; Дулгерис, П.; Блакьер, Г. Разделение смешанных данных путем итеративной оценки и вычитания смешанных данных. интерференционный шум. Геофизика 2011, 76, Q9–Q17. [Перекрестная ссылка]
5. Абма Р.; Хау, Д.; Фостер, М.; Ахмед, И.; Танис, М.; Чжан, К.; Арогунмати, А.; Александр Г. Независимый одновременный сбор и обработка источников. Геофизика 2015, 80, WD37–WD44. [Перекрестная ссылка]
6. Берхаут А.Дж. Изменение взглядов на сбор сейсмических данных. Вести. Край 2008, 27, 924–938. [Перекрестная ссылка] 7. Лин, Т.; Херрманн, Ф. Проектирование одновременного сбора данных с помощью компрессионного зондирования. В материалах 71-й конференции и выставки EAGE с участием SPE EUROPEC, Амстердам, Нидерланды, 8–11 июня 2009 г.; п. CP-127-00269. [Перекрестная ссылка]
8. Ибрагим А. Разделение одновременных сейсмических источников с использованием робастной инверсии операторов радона и миграции. доктор философии Диссертация, Университет Альберты, Эдмонтон, АВ, Канада, 2015 г.
9. Лю, Б.; Сакки, М. Интерполяция сейсмических записей с минимальной взвешенной нормой. Геофизика 2004, 69, 1560–1568. [Перекрестная ссылка]
10. Чен, Ю. Удаление смешивания с использованием пространственно-изменяющегося медианного фильтра. Исследовать. Геофиз. 2015, 46, 332–341. [Перекрестная ссылка]

Machine Translated by Google

11. Махмудян Ф.; Кертис, Дж.; Де Меерсман, К.; Кеннеди, Р.; Стейплс, Г. Удаление данных из одновременных источников. Доступно онлайн: https://geoconvention.com/wp-content/uploads/abstracts/2023/91390-deblending-of-simultainate-source-.pdf. (по состоянию на 25 июня 2024 г.).

12. Стэнтон, А.; Уилкинсон, К. Надежное разделение сейсмических данных из одновременных источников. arXiv 2018, arXiv:1812.06040.

13. Берхаут, А.Дж.; Блакьер, Ж.; Вершуур, DJ. Концепция двойного смешивания: сочетание бессвязной съемки с бессвязным ощущение. Геофизика 2009, 74, A59–A62. [Перекрестная ссылка]

14. Цзян З.; Абма, Р. Анализ одновременного отображения одновременных исходных данных. В SEG расширена техническая программа Тезисы 2010; Общество геофизиков-разведчиков: Хьюстон, Техас, США, 2010 г.; стр. 3115–3119. [Перекрестная ссылка]

15. Торсон, младший; Клербоут, Дж. Ф. Стохастическая инверсия суммирования скоростей и наклонного суммирования. Геофизика 1985, 50, 2727–2741. [Перекрестная ссылка]

16. Фостер, диджей; Мошер, К.С. Подавление множественных отражений с помощью преобразования Радона. Геофизика 1992, 57, 386–395. [Перекрестная ссылка]

17. Трад, Д.; Ульрих, Т.; Сакки, М. Последние взгляды на разреженное преобразование Радона. Геофизика 2003, 68, 386–399. [Перекрестная ссылка]

18. Трад, Д.; Харгривз, Н.; верВест, Б.; Уомбелл, Р. Множественное затухание с использованием преобразования Радона со сдвигом вершины. В расширенном SEG Тезисы; Общество геофизиков-разведчиков: Хьюстон, Техас, США, 2004 г. [CrossRef]

19. Клербоут, Дж. Ф. Анализ данных зондирования Земли: обработка или инверсия; Научные публикации Блэквелла: Лондон, Великобритания, 1992.

20. Весы, Дж. А.; Герстенкорн, А.; Трейтель, С. Решение Fast Ip для больших, разреженных линейных систем: применение к томографии сейсмического путешествия во времени. Дж. Компьютер. Физ. 1988, 75, 314–333. [Перекрестная ссылка]

21. Бругуа, А.; Бурже, М.; Лайли, П.; Пуле, М.; Рикарте, П.; Верстег, Р. Мармуси, модель и данные. В трудах ЕЭГ семинар-практикум по инверсии сейсмических данных, Копенгаген, Дания, 28 мая – 1 июня 1990 г.

22. Наранхо, Дж. К.; Экстранд, Э.Дж.; Этген, Дж.; Харгроув, КЛ; Брютон, Дж. Г.; Гарсия, О.; Аствацатуров Г.; Хайт, Д.; Хауисон, Б.; Бай, Б.; и др. Разработка исследования и реализация стратегии сбора сейсмических данных с использованием нескольких широкоазимутальных буксируемых кос на месторождении Tiber Discovery; глубокая вода Мексиканского залива, США. В расширенных тезисах технической программы SEG 2011; Общество геофизиков-разведчиков: Хьюстон, Техас, США, 2011 г.; стр. 132–136. [Перекрестная ссылка]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Алгоритм построения таблицы приоритетов для CFG независимо от того, являются ли они OPG

Леонардо Лискано, Эдуардо Ангуло и Хосе Маркес *

Факультет компьютерных наук и инженерии Северного университета, Барранкилья 081007, Колумбия; ldlizcano@uninorte.edu.co (LL); edangulo@uninorte.edu.co (EA)
* Переписка: jmarquez@uninorte.edu.co

Аннотация: Грамматики приоритета операторов (OPG) — это контекстно-свободные грамматики (CFG), которые характеризуются отсутствием двух соседних нетерминальных символов в теле каждой продукции (правая часть). Языки приоритета операторов (OPL) являются детерминированными и контекстно-свободными. Для этих языков установлены три возможных отношения старшинства между парами терминальных символов. Многие CFG не являются OPG, поскольку к ним нельзя применить приоритет оператора, поскольку они не соответствуют основному правилу. Чтобы решить эту проблему, мы провели тщательное переопределение левого и правого наборов терминалов, которые являются основой для расчета отношений предшествования, и определили новый набор «Самый левый». Также подробно описаны алгоритмы их расчета. Самый значительный вклад нашей работы заключается в том, что мы устанавливаем отношения приоритета между терминалами, преодолевая основное правило отсутствия двух последовательных нетерминалов, используя алгоритм, который позволяет построить таблицу приоритета операторов для CFG независимо от того, является ли это OPG. В статье показаны сложности предложенных алгоритмов и возможные исключения из предложенных правил. Мы представляем примеры с использованием OPG и двух не-OPG, чтобы проиллюстрировать работу предлагаемых алгоритмов. С их помощью строится таблица приоритета операторов и корректно выполняется восходящий синтаксический анализ .

Ключевые слова: язык приоритета операторов; грамматика приоритета операторов; таблица приоритетов; отношения предшествования; алгоритм

check for updates

Цитирование: Лискано, Л.; Ангуло, Э.; Маркес, Дж. Алгоритм построения таблицы приоритетов для CFG, независимо от того, являются ли они OPG. Алгоритмы 2024, 17, 345. https://doi.org/10.3390/a17080345

Академический редактор: Димитрис Фотакис

Поступила: 24 июня 2024 г.
Пересмотрено: 29 июля 2024 г.
Принято: 30 июля 2024 г.
Опубликовано: 7 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

В данной статье рассматривается семейство языков приоритета операторов (OPL), созданное Р. Флойдом [1], которое используется для эффективного восходящего синтаксического анализа [2,3]. Ключевым выводом является вдохновение автора структурой арифметических выражений, где приоритет отдается мультипликативным операторам (* и /) над аддитивными операторами (+ и -). В процессе синтаксического анализа снизу вверх левая часть (нетерминальный символ) полностью заменяет идентифицированную правую часть (тело) в контекстно-свободной грамматике (CFG), не оставляя места для двусмысленности. Приоритет операторов был распространен на грамматики, состоящие из терминалов с правой стороны, даже если их разделяли нетерминалы. CFG обычно считается грамматикой приоритета операторов (OPG), если она не содержит произведений со смежными нетерминальными символами в правой части.

OPL очень помогли в выводе контекстно-свободных грамматик. Пример описан в [4–6], где представлен метод вывода грамматик без приоритета операторов из выборок в скобках. Метод основан на построении деревьев разбора и использовании функций, создающих грамматики. Правила приоритета используются для обозначения скобок, окружающих конечные элементы строки в образце, указывая порядок, в котором строка должна оцениваться.

Исследования формальных языков, управляемых вводом (IDL) [7–9] (позже переименованных в Visible Pushdown Languages (VPL) [10]) пришли к выводу [11] , что свойства замыкания OP подразумевают свойства замыкания ID и что языки ID представляют собой особый тип. языков ОП, характеризующихся ограниченными отношениями ОП [12].

Machine Translated by Google

С развитием параллельных вычислений появились свойства локальной разборчивости. из OPL были использованы для создания быстрых параллельных парсеров [13].

Главным вкладом этой статьи является ее новый подход к построению отношений приоритета между терминальными символами в не-OPG. Этот подход позволяет создавать таблицы приоритета операторов и анализировать снизу вверх строку, полученную не-OPG, с поддержкой полученной таблицы. Новизна работы является важным аспектом, отличающим ее от предыдущих исследований в этой области. • Наборы левых и правых терминалов, предложенные в

предыдущих работах [1,11,14] для построения отношений приоритета между терминалами OPG, были переопределены. • Был определен новый набор терминалов под названием «Самый левый» для поддержки решения отношений приоритета, когда на правой стороне продукции находятся два или более соседних терминала. Этот набор важен, поскольку он систематически обрабатывает отношения приоритета в таких сложных сценариях, расширяя восходящий процесс анализа за счет не-OPG.

• Был установлен алгоритм построения таблицы приоритетов с поддержкой трех предложенных наборов терминалов.

• Наконец, применение предложенных алгоритмов позволяет получить из не- OPG таблицу отношений и приоритета операторов, ранее ограниченную только OPG. Кроме того, восходящий анализатор приоритета операторов можно использовать для анализа любой строки, сгенерированной не-OPG, - метод, который можно применять только к OPG.

Как показано в данной работе, предлагаемое решение по построению таблицы приоритета операторов является не просто теоретической концепцией, а практическим инструментом, который можно применять к ОПГ без каких-либо ограничений. В конце работы функционал демонстрируется путем применения его к OPG и двум не-OPG и проведению соответствующего восходящего разбора. Эта практическая демонстрация должна вселить уверенность в эффективность предлагаемого решения и обеспечить его практическую применимость.

Статья организована следующим образом: В разделе 2 приведены необходимые определения для понимания этой работы. В разделе 3 представлена предыдущая работа, проведенная в области CFG. В разделе 4 представлен наш подход, подробно описана проблема, которую мы стремимся решить, а также переопределения концепций предыдущих работ, использованных для разработки наших алгоритмов. В разделе 5 представлены примеры применения предложенного алгоритма к одному OPG и двум не-OPG, демонстрируя применение восходящего анализа с таблицами приоритетов для проверки их использования. В разделе 6 описано несколько исключений, когда правила, описанные в предлагаемых алгоритмах, не применяются. Наконец, в разделе 7 представлены выводы нашей работы.

2. Основные определения

2.1. Контекстно-свободные грамматики и языки

В рамках иерархии Хомского [15] контекстно-свободные грамматики играют важную роль в приложениях языков программирования и компиляции, обращаясь к синтаксической структуре языков программирования.

Контекстно -свободная грамматика (CFG) G генерирует язык L(G), обычно называемый контекстно-свободным языком (CFL), который состоит из строк, сгенерированных CFG. CFG определяется как четырехкортеж G(S) =< T, N, S, P >, где: • T представляет собой

набор терминальных символов, которые создают строки в CFL, генерируемые CFG. • N представляет собой набор

нетерминальных символов или синтаксических переменных, которые определяют строки, ведущие к генерации CFL. • S — начальный

нетерминальный символ, из которого извлекаются все строки, определенные GFC. генерируется.

• P — это набор продукций или правил, которые группируют терминалы и нетерминалы для генерации строк. Производство принимает форму A α, где A представляет собой нетерминал , а α — комбинацию терминалов и нетерминалов. Символ произносится как «продукт». Таким образом, продукция читается как « Продукция альфа».

Machine Translated by Google

Будут использоваться следующие обозначения: строчные буквы с начала алфавита обозначают терминальные символы (a, b, c, ... T); заглавные буквы с начала алфавита обозначают нетерминальные символы (A, B, C, ... N); строчные буквы в конце алфавита обозначают терминальные строки (u, v, w, x, y, z T +); Греческие буквы в нижнем регистре обычно обозначают терминальные и/или нетерминальные строки (α, β, γ, ... (T N)), а пустая строка будет обозначаться ε (эта греческая буква будет исключением из правила). вышеуказанное соглашение).

2.2. Грамматики приоритета операторов (основное правило)

CFG считается операторной грамматикой, если ни одна из продукций в P не имеет соседних нетерминальных символов в правой части. Другими словами, для продукции A α в G α не принимает вид uBCv.

2.3. Выводы

Строка η (T N) может быть получена за k шагов из нетерминала A N, если существует последовательность строк вида η1, η2, . . . , ηκ . С помощью символа = , означающего одношаговый вывод, последовательность строк оформляется в виде: η1 = η2 = . . . = ηk .

В этой последовательности η1 соответствует A , а ηk — η. Каждое промежуточное ηi в последовательности имеет вид δBγ, где B N и существует продукция B β, такая, что подстановкой ее в правой части получается ηi+1(= δβγ) . Вывод в один или несколько шагов обозначается оператором += . В общем случае утверждается, что каждая строка терминальных символов w L(G) может быть установлена как S += w или разложением вывода в один или несколько шагов: S = η1 = η2 = . . . = ηk = w. Для грамматики G форма предложения η (T N) = η, что означает, что η может быть получено с нулем или более выводами из строки S, которая S. Если две грамматики порождают это идентичные языки, они считаются эквивалентными.

2.4. Виды дифференцирований

Согласно [16], S = η1 = η2 = . . . = ηk = w — крайнее левое дифференцирование строки w T, когда каждое ηi , 2 i k, 1, имеет вид xiAiβi, где xi T Ai N и βi (T N) . Более того, Ai αi , принадлежит P, и тогда в каждом ηi символ Ai заменяется на αi , что приводит к форме предложения ηi+1 . Каждая ηi называется левосентенциальной формой.

Если каждый ηi , 2 i k 1, принимает форму βiAixi с xi T βi (T N) , Ai N, и тогда право- , будет формироваться крайнее правое дифференцирование. Каждое ηi называется сентенциальной формы.

2.5. Деревья разбора

Согласно [16], дерево вывода для КФГ G(S) =< T, N, S, P > представляет собой помеченное и упорядоченное дерево, в котором каждый узел получает метку, соответствующую символу из множества N T {ε}. Если нелистовой узел помечен A , а его непосредственные потомки помечены X1, X2, . . . , Xn, то A X1X2 . . . Xn является продукцией в P. Помеченное упорядоченное дерево D является деревом вывода для КФГ G(A) =< T, N, A, P > , если

1. Корень D обозначается A.
2. Если Д1, . . . , Dk — поддеревья прямых потомков корня, а корень Di обозначается Xi , тогда A X1 . . . Xk находится в P.
3. Di — дерево вывода для G(Xi) =< T, N, Xi , P > , если Xi находится в N или Di — единственный узел. помеченный Xi , если Xi находится в T или Di представляет собой одиночный узел, помеченный ε.

Деревья разбора могут быть построены на основе любого типа вывода.

Machine Translated by Google

2.6. Синтаксический анализ снизу вверх

Синтаксический анализ синтаксического анализа «снизу вверх» [2], также известный как синтаксический анализ с сокращением сдвига, пытается построить дерево синтаксического анализа для входной строки, которая начинается с листьев (нижней части дерева) и перемещается в корень (верхушку дерева). Этот процесс рассматривается как приведение строки w к начальный символ S CFG. На каждом этапе сокращения синтаксического анализа определенная подстрока форма предложения, соответствующая правой стороне произведения, заменяется нетерминальной формой символ левой стороны этого производства. Если на каждом шаге подстрока выбрана правильно, крайнее правое дифференцирование прослеживается обратно.

2.7. Ручка

Ручкой [2,17] правосентенциальной формы γ является продукция A β и позиция γ , где строку β можно найти и заменить на A , чтобы получить предыдущую правопредложенную форму в крайнем правом выводе γ. Как правило, если C = αAw = αβw, тогда A β , если позиция рядом с α является дескриптором αβw. Если грамматика неоднозначна, более чем получится одна ручка. Если грамматика однозначна, то каждое правое предложение форма имеет ровно один дескриптор. Обычно строка β считается дескриптором αβw , когда условия для этого ясны.

2.8. Реализация восходящего анализа

Синтаксический анализ снизу вверх путем сдвига и сокращения использует стек для хранения грамматических символов. и буфер для обработки входной строки, подлежащей анализу. Символ $ также используется для разделения нижняя часть стека и правая часть входного буфера. Формат модели распознавания начинается со стека в $ и строки w$ на входе, как показано в таблице 1.

Таблица 1. Таблица синтаксического анализа «снизу вверх».

Куча	Вход	Действие
$	ш$	
...	...	Сдвиг/Уменьшение
...	...	Сдвиг/Уменьшение
$C	$	Принимать

Синтаксический анализ снизу вверх сдвигает ноль или более входных символов в стек до тех пор, пока не будет найден дескриптор. наверху. Затем синтаксический анализ сводит дескриптор к левой стороне соответствующей продукции. чтобы получить форму предложения, соответствующую предыдущему шагу крайнего правого вывода. При уменьшении синтаксический анализ распознает, что правый конец дескриптора находится вверху куча. Таким образом, синтаксический анализ предпринимает действия, чтобы найти его левый конец в стеке и определить нетерминал, который заменит его по правой стороне какого-нибудь производства, где он соответствует ручке. Эти шаги повторяются до тех пор, пока не будет обнаружена ошибка или пока стек не будет содержит $S , а входные данные имеют символ $, после чего синтаксический анализ заканчивается и входная строка считается допустимой для CFG.

3. Предыдущая работа

Согласно [1,11,14] для КФГ G левый терминал устанавливает LG(A) , а правый терминал множества RG(A) определяются следующим образом:

$$LG(A) = \{a \mid A \Rightarrow^{+} a\alpha, \alpha \in (N \cup \{\varepsilon\})^{*}\}.$$

$$RG(A) = \{a \mid A \Rightarrow^{+} \alpha a, \alpha \in (N \cup \{\varepsilon\})^{*}\}.$$

Отношения предшествования возникают в результате установления в дереве разбора бинарных отношений между последовательными терминалами или теми, которые становятся последовательными после восходящей связи. процесс к нетерминальному символу S.

Machine Translated by Google

Отношения старшинства устанавливаются следующим образом: a .= b

1. тогда и только тогда, когда A αaγbβ P, γ N {ε}.
2. a b тогда и только тогда, когда A αaDβ P, b LG(D). 3. a b, если и только если A αDbβ P, a RG(D).

Согласно [2], отношения старшинства используются для разграничения дескриптора в правопредложенной форме. То есть в форме предложения αβw подстрока β является дескриптором, если в CFG существует продукция A β ; следовательно, его можно ограничить отношениями старшинства, получив α β w. Отношение .= между символами, используемыми, когда дескриптор имеет более одного последовательного терминального символа. То есть, если β = a1a2 . . . an, то соотношение ai .= i = ai+1 , 1, . . . , n 1 должно быть установлено. Предположим, что правопредложение имеет вид A0a1A1a2 . . . anAn, где каждый Ai является нетерминалом. В этом случае отношения могут быть установлены как ai Aiai+1 с максимум одним нетерминалом (принцип операторной грамматики) между терминалами. Нетерминалы можно было бы исключить в правильной форме предложения, оставив только отношения между терминалами.

Короче говоря, каждый раз, когда получается дескриптор, его можно заключить между символами и . Кроме того, два последовательных символа терминала могут находиться внутри дескриптора (нетерминалы посередине), и связь должна быть установлена .=. Наконец, нетерминальные символы могут быть удалены из правопредложенной формы, когда дескриптор найден и зафиксирован между отношениями старшинства.

Согласно [2], концепциям, обсуждаемым в разделе 2.8, и отношениям предшествования, алгоритм 1 анализа предшествования строится следующим образом:

Алгоритм 1. Алгоритм анализа приоритета. Требование:

```
Набор отношений приоритета. процедура
    PrecedenceParsing(w) stack     [$] end     f
        alse pos     0
        do a
        stack[top]
        b
            w$[pos], если a =
            $ и b = $ then

                print("Действие: принять") end
                    true
            еще
                если a     b или a .= b then
                        pos + +
                        push(stack, b)
                        print(“Действие: Shift”) else
                if a
                        b then
                            делать
                                e     pull(stack) while
                            not (stack[top]     e)
                            print(“Действие: Уменьшить”)
                        else
                            print(“Действие: Ошибка”)
                            end     true
                        end if
                    end if
            end if
        while not end end
```

Machine Translated by Google

4. Подход

В этой работе предлагается переопределить наборы левого и правого терминалов, соответствующие LG (A) и RG(A) соответственно. Кроме того, представлено определение нового левого множества . Новые правила определения отношений старшинства демонстрируются, даже если CFG не соответствует фундаментальному правилу операторной грамматики, означающему, что CFG может иметь более одного соседнего нетерминального символа в правой части своих продуктов.

Представлены алгоритмы построения трех предложенных множеств: Левого, Крайнего левого и Правого. Кроме того, мы демонстрируем алгоритм, включающий вновь установленные правила приоритета.

4.1. Определения

Существуют следующие символы:

• Строка терминалов x T +. • Строка нетерминалов β N . • Предложенная форма γ (T N) Определены ·

следующие множества:

• Левая часть, обозначаемая как L(A), представляет собой набор терминальных символов, которые могут появляться в левой части любой формы предложения как продукт крайне правого вывода из A.

$$L(A) = \{a \in T \mid A = \beta a\gamma\}.$$

• Крайний левый, обозначаемый как Lm(A), представляет собой набор терминальных символов, которые могут появляться в конце левой части при любом выводе из A, в отсутствие других крайних левых грамматических символов (или которые происходят от ε в их отсутствие).).

$$Lm(A) = \{a \quad T \mid A = \quad a\gamma\}.$$

• Право, обозначаемое как R(A), представляет собой набор терминальных символов, которые могут появляться в правом конце самого правого вывода из A или появляться дальше вправо в любом произведении A.

$$R(A) = \{a \quad T \mid A = \quad \gamma a \quad A \quad \gamma a\beta\}.$$

Согласно вышеизложенному определены следующие правила старшинства:

- Если C γ1aβBγ2 , b L(B), тогда a b.
- C γ1AβBγ2 , = ε,то aR(A)b. b Lm(B) β. Если
- Если C γ1Aβbγ2 , a R(A) β = ε, то a b.
- Если C γ1aβbγ2, то a .= b.

4.2. Анализ проблемы

4.2.1. Грамматика абсолютного

оператора. Учитывая крайний правый ряд вывода вида:

$$S \mathrel{+}= \gamma 1Ax1 = \gamma 1\omega 1x1, \text{ где } A \quad \omega 1 \text{ и } \omega 1 = \gamma 2B\gamma 3. \tag{1}$$

Отметим, что ω1 является дескриптором согласно предыдущему определению в разделе 2.7. Правопредложенную форму (1) можно переписать следующим образом:

$$\gamma 1\omega 1x1 = \gamma 1\gamma 2B\gamma 3x1 \mathrel{+}= \gamma 1\gamma 2Bx2x1 = \gamma 1\gamma 2\omega 2x2x1, \text{ где } B \quad \omega 2. \tag{2}$$

Из последней формы предложения в (2) следует, что ω2 является дескриптором согласно определению в разделе 2.7.

Machine Translated by Google

В связи с вышеизложенным анализ будет сосредоточен на произведениях, рожденных от A, которые имеют по крайней мере один нетерминальный B справа, как видно из (1). Произведения в A будут пройдены через грамматику, чтобы найти отношения старшинства нетерминала B, расположенного с правой стороны.

В контексте ω1, когда B ω2 и ω2 является дескриптором, следующие формы можно получить из его определения в (1): (a) ω1 =

γ4aB, отсюда следует, что a L(B). Здесь γ2 = γ4a и γ3 = ε. (б) ω1 = Bbγ5, то R(B) b.
Здесь γ2 = ε и γ3 = bγ5. (в) ω1 = γ4aBbγ5, то a L(B) и R(B) b. Здесь γ2 = γ4a и γ3 = bγ5.

Для нетерминального стартового символа S определяется начальная фиктивная продукция E S и применяется правило (c).

Для произведения A γ1aXbγ2, X N {ε}, отсюда следует, что a .= b.

4.2.2. Общая грамматика с двумя или более последовательными нетерминальными символами.

Переопределение левого, правого и определение крайнего левого.
Слева(B):

Учитывая самый правый вывод:

$$\gamma 1aBx1 = \gamma 1a\omega 1x1. \quad (3)$$

Замечено, что ω1 является дескриптором восходящего анализа (обратный крайний правый вывод), поскольку B = ω1; поэтому из (3) можно установить, что γ1a ω1 x1. Кроме того, ω1 может иметь следующий

вид: (a) ω1 = β1bγ2.
(б) ω1 = β1Cγ3.

Продолжая самый правый вывод в (3) и рассматривая каждый из вариантов ω1, мы получаем, что:

(a) γ1aBx1 = γ1aβ1bγ2x1, поскольку ω1 — ручка, следовательно, γ1a β1bγ2 x1; там-
следовательно, a b , где B β1bγ2 и b L(B). (б)
γ1aBx1 = γ1aβ1Cγ3x1 = γ1aβ1Cx2x1 = γ1aβ1β2Dx3x2x1 = γ1aβ1β2β3bγ4 x3x2x1. Замечено, что C +=
β2Dx3 и D β3bγ4; следовательно, β3bγ4 — ручка, и, следовательно, γ1aβ1β2 β3bγ4 x3x2x1 и
a b.

Короче говоря, γ1aBx1 += γ1aβ1β2β3bγ4x3x2x1; взяв β = β1β2β3 и γ5 = γ4x3x2, получим суммарный вывод γ1aBx1 += γ1aβbγ5x1, где B += βbγ5 и b L(B).

Справа(B):

Учитывая самый правый вывод:

$$\gamma 1Bax1 = \gamma 1\omega 1ax1. \quad (4)$$

Замечено, что ω1 является дескриптором при синтаксическом анализе снизу вверх (обратный крайний правый вывод), поскольку B = ω1; поэтому из (4) можно установить, что γ1 ω1 ax1. Кроме того, ω1 может

иметь следующий

вид: (a) ω1 = γ2bβ. (б) = ε.

ω1 = γ3Cβ1, β1 Продолжая самый правый вывод в (4) и беря каждый из вариантов ω1, мы получаем, что:

(a) γ1Bax1 = γ1γ2bβax1, поскольку ω1 — ручка, следовательно, γ1 γ2bβ ax1; следовательно, b a , где B
γ2bβ и b R(B). (б) γ1Bax1 = γ1γ3Cβ1ax1
= γ1γ3γ4Dax1 = γ1γ3γ4γ5bβ2ax1. Отмечается, что C += γ4D и D γ5bβ2; следовательно, γ5bβ2 является
дескриптором и, следовательно,

Machine Translated by Google

γ1γ3γ4 γ5bβ2 ax1 и b a.

Короче говоря, γ1Bax1 += γ1γ3γ4γ5bβ2ax1; взяв γ6 = γ3γ4γ5, получим суммарный вывод γ1Bax1 += γ1γ6bβ2ax1, где B += γ6bβ2 и b R(B).

Крайний левый (B):

Учитывая продукцию с двумя нетерминальными символами справа (тело):

$$S \quad AB \tag{5}$$

$$A \quad \gamma 1 b 1 \beta 1 \tag{6}$$

Самый правый вывод из S в (5) :

S = AB += Aay += Aax1 = γ1b1β1ax1 = γ2Cax1 += γ2γ3γ4b2β2ax1. (i) (iii) (v)
(ii) (iv) (ви) (vii)

В итоге:

- В (v) соотношение γ1b1β1 ax1 может быть установлено, поскольку правая часть продукция (6) — это ручка. Следовательно, b1 a и b1 R(A).
- В (vi) известно, что β1 = C0C1 . . . CnC, где каждый Ci N и C — нетерминал , сначала полученный справа. Отмечается, что γ2 = γ1b1C0 . . . Cn. • В (vii) C += γ3γ4b2β2 является γ4b2β2 результатом последнего крайнего правого вывода и, в свою очередь, дескриптором; следовательно, соотношение γ2γ3 γ4b2β2 ax1 установлено , поскольку γ4b2β2 — ручка, следовательно, b2 a. • Из (iv) Aax1 += γ2γ3γ4b2β2ax1.

Установив γ5 = γ2γ3γ4, тогда A += γ5b2β2, следовательно, b2 R(A). • Из (ii) устанавливается B += aγ , откуда мы определяем новое самое левое множество B как Lm(B) = {a T | Б += aγ}.

4.2.3. Интуитивный способ наблюдения за элементами каждого множества

Видение Left(·):
Учитывая CFG: A
CDE
E eh D
d C
cF F
f

Самый правый вывод строки cf deh приводит к следующему способу создания Le f t(A) или L(A).

A = CDE = CDeh = Cdeh = cFdeh = cfdeh.
e д c

Терминальные символы e, d и c рассматриваются слева и записываются в L(A). Символ e является первым , который виден слева, и записывается в L(A), а символ h блокируется символом e , поскольку оба появляются при замене E. Затем D заменяется на d, что наблюдается и записывается в L(A). Когда cF заменяет C, мы наблюдаем и записываем c в L(A).
Наконец, когда осуществляется замена F на f , она уже не может наблюдаться (блокируется c) и не может быть зарегистрирована. Следовательно:

$$L(A) = \{e, d, c\}.$$

Видение права(·):
Учитывая CFG:
A abBC
Б эФ

Machine Translated by Google

C cD D
d F
f

Самый правый вывод строки abe f cd предполагает следующий способ создания Right(A) или R(A).

A = abBC = abBcD = abBcd = abeFcd = abefcd.
б с д

Терминальные символы b, c и d рассматриваются справа и записываются в R(A). Символ b является первым, наблюдаемым и записанным в R(A). Символ a блокируется символом b, поскольку оба появляются при выведении A. Затем C заменяется на cD , и c наблюдается и записывается в R(A). Когда D заменяется на d, мы наблюдаем и записываем d в R(A). Тогда B получается с помощью eF, но e не может быть записано, поскольку предыдущее включение c и d в R(A) делает его невидимым или блокирует его наблюдение справа. Наконец, при замене F на f его также нельзя наблюдать справа (заблокировано c и d) и невозможно записать. Следовательно:

R(A) = {b, c, d}.

Видение крайнего левого(·):
Учитывая CFG:

S дАБ
A фа
B Cb
C c

Крайнее правое производное строки df acb имеет следующий способ создания Le f tmost(B) или Lm(B), как показано на рисунке 1.

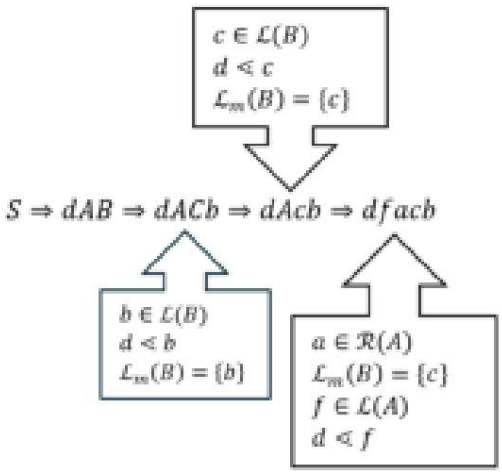


Рис. 1. Вывод цепочки dfacb и последовательности создания набора Le f tmost(B) .

Видно, что L(B) = {b, c}. Когда B выводит Cb, мы наблюдаем, что Lm(B) = {b}, потому что до этого момента больше не видно терминальных символов. В следующем выводе C заменяется на c; следовательно, c находится левее подстроки cb, поэтому Lm(B) = {c}.

Machine Translated by Google

4.2.4. Краткое изложение отношений между грамматическими символами из произведений и наборов

В таблице 2 показаны связи между грамматическими символами произведений и наборами терминалов.

Таблица 2. Полученные общие производства и отношения старшинства.

Производство	Отношение
A αβб	a .= б
A Cβb, β = ε	R(C) ba
A αβC	L(C)
A BβC, β = ε	R(B) Lm(C)

4.3. Алгоритмы

Следующие алгоритмы были разработаны на основе определений, представленных в разделе 4.1.

Алгоритм 2 получает нетерминал A , продукция которого известна; для каждого из них проверяется, соответствуют ли они форме βBγ; если да, то алгоритм 2 выполняется снова, отправляя нетерминал B в качестве параметра (при условии, что B≠ A), и его ответ включается в множество L(A).

В дальнейшем проверяется, соответствует ли продукция форме βaγ, и если да, то терминал a добавляется в набор L(A).

После завершения проверки всех произведений A набор L(A) возвращается в качестве результата алгоритма 2.

Алгоритм 2 Левый алгоритм нетерминала A

```
функция влево(A)
    L(A)      для
    каждой продукции A do, если A
            βBγ , то
            L(A)   Left(B) // L(A) содержит левый конец(B) , если
        если A
            βaγ , то a   L(A) //
            a находится в конце L(A) , если
        конец
    для
    вернуть Л(A)
конечная функция
```

Алгоритм 3 получает нетерминал A , продукция которого известна; для каждого из них проверяется, соответствуют ли они форме βBγ и β = ε; если да, то алгоритм 3 выполняется снова, отправляя нетерминал B в качестве параметра (при условии, что B≠ A), и его ответ включается в множество Lm(A).

Впоследствии проверяется, соответствует ли продукция форме βaγ и β. Если да, то терминал = ε. a добавляется в множество Lm(A).

После завершения проверки всех продукций A возвращается набор Lm(A). как результат алгоритма 3.

Machine Translated by Google

Алгоритм 3. Крайний левый алгоритм нетерминала A

```
функция Крайняя левая(A)
    Lm(A)       для
    каждой продукции A делать, если
        A       βBγ и β =   ε , то
            Lm(A)   Крайний левый(B)
        конец,
        если если A       βaγ и β =   ε , то a
            Lm(A) конец,
        если
    конец
    для возврата
Lm(A) конечная функция
```

Алгоритм 4 получает нетерминал A , продукция которого известна; для каждого из них проверяется, соответствуют ли они виду γBβ и β = ε; если да, то алгоритм 4 выполняется снова, отправляя нетерминал B в качестве параметра (при условии, что B ≠ A) и его ответ включен в множество R(A).

В дальнейшем проверяется, соответствует ли продукция форме γaβ, и если да, то терминал a добавляется в набор R(A).

После того как все произведения A проверены, набор R(A) возвращается в качестве результата алгоритма 4.

Алгоритм 4. Правый алгоритм нетерминала A

```
функция вправо(A)
    R(A)       для
    каждой продукции A делать, если
        A       γBβ и β =   ε , то
            R(A)   Правый(B)
        конец, если
        если A       γaβ , то a
                R(A) end
        if end
    для
    возврата R(A)
end function
```

Алгоритм 5 получает грамматику G(S) =< T, N, S, P > и алгоритмы 2–4. Первоначально отношения старшинства добавляются между разграничивающим символом $ и наборами Левого и Правого нетерминального символа начала S согласно правилам (A.1) и (A.2) соответственно.

Алгоритм 5 работает итерационным способом. Он обходит все произведения множества P и на каждой итерации инициализирует две временные переменные u и l значениями null и пустым списком соответственно. Затем он выполняет ряд процессов для каждой пары смежных грамматических символов X и Y в произведении.

1. Алгоритм проверяет, имеет ли XY форму ab; если да, то он добавляет соответствующий приоритет взаимосвязь между a и b согласно правилу (B).
2. Алгоритм проверяет, имеет ли XY форму aB; если да, то он присваивает u значение a и добавляет соответствующие отношения старшинства между a и терминалами в Left(B), следуя правилу (C).
3. Алгоритм проверяет, имеет ли XY форму Ab; если да, то он добавляет соответствующие отношения старшинства между b и терминалами в Right(A), следуя правилу (D.1). Затем проверьте, является ли u ≠ null, и если да, добавьте соответствующее отношение старшинства между u и b, следуя правилу (D.2), и сбросьте u до нуля. Наконец, проверяет, есть ли в списке l элементы, и если да, добавляет соответствующие отношения старшинства между b и терминалами в Right(B), следуя правилу (D.3), и сбрасывает список l как пустой.

Machine Translated by Google

4. Алгоритм проверяет, имеет ли XY форму AB; если да, то он добавляет соответствующие отношения старшинства между терминалами в правом (A) и терминалами в крайнем левом (B), следуя правилу (E.1). Затем он проверяет, является ли u≠ null, и если да, добавляет соответствующее отношение старшинства между u и терминалами в Left(B), следуя правилу (E.2). Наконец , он проверяет, есть ли в списке l элементы, и если да, то добавляет соответствующие отношения старшинства между терминалами в Right(C), где C l , и терминалами в крайнем левом (B), следуя правилу (E.3), и при необходимости сбрасывает список l .

Алгоритм 5 Алгоритм получения таблицы приоритета операторов G- грамматики

```
Требуется: G(S) =< T, N, S, P > table (G,
    Left(·), Right(·), Крайний левый(·))
        вычислить Left(A), Right(A), Крайний левый(A), A N добавить
        $ a, a Left(S) добавить (A.1)
        a $, a Right(S) для (A.2)
        каждого A γ в P do u null

            l [ ] // пустой список
            для каждого XY, смежного в γ, do if XY =
                ab then
                    добавить a .= б (Б)
                конец, если
                если XY = aB , то
                    ты a
                    добавить a b, b Left(B) (C)
                end,
                если XY = Ab , затем
                    добавить a b, a Right(A), (Д.1)
                    если u≠ null , то
                        добавить (Д.2)
                        u .= bu
                    null
                    end, если if l≠ [ ]
                        тогда для B
                            l добавьте a b, a Right(B) (Д.3)
                        end for l
                            [ ]
                    end if
                end if if
                XY = AB то
                    добавить a b, a Right(A) b Left(B), если u≠ (E.1)
                    null , то добавить u
                        b, b Left(B) (E.2)
                    конец, если
                    если l≠ [ ] , то для
                        C l выполните
                            добавьте c b, c Right(C) b Крайний левый(B) (E.3)
                        конец для
                    конец, если
                    если B = ε , то
                        добавьте A
                    к l
                        else l
                    [ ] end
                if end if
            end for
        end for
    end процедура
```

Machine Translated by Google

4.4. Анализ сложности

Для расчета сложности будут установлены следующие переменные:
четыре предложенных алгоритма:

- t : Мощность набора терминальных символов.
- p : Мощность множества продукций. g :
- Количество грамматических символов в произведениях.
- n : Мощность набора нетерминальных символов.

4.4.1. Сложность алгоритма 2 (слева)

Функция Left проверяет, в каких постановках слева появляется символ A.
В первом if-блоке наблюдается рекурсивный вызов Left , что означает, что все нетерминальные символы G могут быть пройдены в конце выполнения при поиске Left(A).
Следовательно, все порождения наихудшего случая будут пройдены, что даст время вычислений O(p). Второй if-блок имеет время вычисления O(1), что не влияет на общее время вычисления O(p).

4.4.2. Сложность алгоритма 3 (крайний левый)

Самая левая функция имеет структуру, аналогичную предыдущей; следовательно, время выполнения вычислений равно O(p).

4.4.3. Сложность алгоритма 4 (справа)

Функция Right имеет структуру, аналогичную Left; следовательно, время выполнения вычислений равно O(p).

4.4.4. Сложность алгоритма 5 (таблица)

Согласно первому циклу for, количество раз, которое он будет выполняться, будет равно p. Сразу после предыдущего внутренний цикл for выполняется в худшем случае g раз, предполагая, что общее количество грамматических символов встречается в каждом произведении. Следовательно, вычислительная стоимость двух циклов равна O(p × g).

Вычислительные затраты на построение Left(A), Right(A) и Leftmost(A), A N, уже рассчитаны. При вычислении сложности таблицы будет учитываться длина каждого сгенерированного набора, которая в худшем случае равна t.

Анализ четырех if-блоков внутри второго for-цикла заключается в следующем: •

В первом if-блоке после проверки выполняемая инструкция выполняется в константном режиме. время и обозначается O(1).

- Во втором if-блоке после проверки первая инструкция выполняется за постоянное время O(1), а вторая — за вычислительное время O(t); следовательно, время вычисления if-блока равно O(t).
- В третьем if-блоке после проверки выделяются три подблока; первый соответствует перемещению Right(A), выполняемому за вычислительное время O(t). После проверки u≠ null второй подблок представляет две инструкции, выполняемые за постоянное время O(1). Третий подблок после проверки l≠ [ ] представляет вычислительную стоимость O(n × t), определяемую циклом for, выполненным в худшем случае O(n) раз, что соответствует числу нетерминальных символов в G и его путешествие из Right(B) со стоимостью O(t). Таким образом, вычислительные затраты третьего if-блока составляют O(n × t), что соответствует наихудшему случаю.
- В четвертом if-блоке после проверки выделяются четыре подблока. Первый соответствует комбинациям, образованным путем пересечения правого(A) и самого левого(B) для получения отношений предшествования a b. Поскольку максимальный размер каждого набора равен t, вычислительная стоимость этих t^2). После проверки u≠ null , комбинаций равна O(t). Второй подблок представляет инструкцию, в которой выполняется обход Left(B) ; следовательно, вычислительная стоимость равна O(t). В третьем подблоке -блок, после проверки l≠ [ ], представляет for-цикл, который выполняет не более n раз комбинации, с

Machine Translated by Google

проходя вправо(C) и самое левое(B) для получения отношений старшинства с b. Эти комбинации проходятся максимально за $O(t^2)$ вычислительного времени, поскольку t — размер обоих наборов и, следовательно, вычислительная стоимость третьего подблока равна $O(n \times t^2)$. Последний подблок будет иметь стоимость O(1). Таким образом, вычислительные затраты четвертого блока составляют $O(n \times t^2)$. Таким образом, вычислительные затраты четырех if-блоков во втором цикле for Алгоритма 5 составляет $O(n \times t^2)$, это худший случай; следовательно, вычислительные затраты t, основная часть алгоритма 5 , образованная двумя вложенными циклами for и четырьмя if-блоками, равна $O(p \times g \times n \times t^2)$, что больше, чем стоимость вычисления любого из трех наборов терминальных символов Left(A), Right(A) и самый левый(A), A N, при запуске алгоритма и соответствующий O(p).

5. Примеры

5.1. Пример 1

5.1.1. Грамматика

Учитывая следующую ОПГ: E
E + T | TT
T F | ФФ (E)
| id Наборы
строятся шаг за шагом по алгоритмам.

5.1.2. Левый

Левые наборы рассчитываются, как подробно описано в Таблице 3 .

Таблица 3. Расчет левых наборов для CFG в примере 1.

Левый(E)			
Для E E + T			
1. Вычисление первого условия алгоритма 2.	A = E β = ε Б = E γ = +T Л(E) Л(E)	2. Вычисление второго условия алгоритма 2.	A = E β = E a = + γ = T L(E) {+}
Для E T			
3. Вычисление первого условия алгоритма 2.	A = E β = ε Б = T γ = ε L(E) = {+} L(T)		

Левый(T)			
Для T T F			
1. Вычисление первого условия алгоритма 2.	A = T β = ε Б = T γ = F L(T) L(T)	2. Вычисление второго условия алгоритма 2.	A = T β = T a = * γ = F L(T) { }
Для T F			
3. Вычисление первого условия алгоритма 2.	A = T β = ε Б = Ф γ = ε L(T) = { } L(F)		

Machine Translated by Google

Таблица 3. Продолжение.

Левый(Ф)			
Для F (E)		Для F id	
1. Вычисление второго условия алгоритма 2.	A = Ф β = ε a = (γ = E) L(F) {(}	2. Вычисление второго условия алгоритма 2.	A = Ф β = ε a = идентификатор γ = ε L(F) = {(, id}

Краткое изложение левых наборов нетерминалов:

• Left(E) = {+} L(T). • Left(T) = { } L(F). • Левый(F) = {(, id}.

Подстановка множеств: • Left(E) = {+, ,(, id}. • Left(T) = { , (, id}. • Left(F) = {(, id}.

5.1.3. Right

Наборы Right рассчитываются, как подробно описано в Таблице 4.

Таблица 4. Расчет правых множеств для CFG в примере 1.

Вправо (E)			
Для E E + T			
1. Вычисление первого условия алгоритма 4.	A = E γ = E+ Б = T β = ε P(E) P(T)	2. Вычисление второго условия алгоритма 4.	A = E γ = E a = + β = T R(E) R(T) {+}
Для E T			
3. Вычисление первого условия алгоритма 4.	A = E γ = ε Б = T β = ε R(E) = R(T) {+}		

Вправо(T)			
Для T T F			
1. Вычисление первого условия алгоритма 4.	A = T γ = T Б = Ф β = ε R(T) R(F)	2. Вычисление второго условия алгоритма 4.	A = T γ = T a = * β = Ф R(T) R(F) { }
Для T F			
3. Вычисление первого условия алгоритма 4.	A = T γ = ε Б = Ф β = ε R(T) = R(F) { }		

Machine Translated by Google

Таблица 4. Продолжение.

Вправо(Ф)			
Для F (E)		Для F id	
1. Оценка второго условие алгоритма 2.	A = Ф γ = (E a =) β = ε R(F) {)}	2. Оценка второго условие алгоритма 2.	A = Ф γ = ε a = идентификатор β = ε R(F) = {, идентификатор}

Краткое изложение правильных наборов нетерминалов:

- Right(E) = R(T) {+}.
- Right(T) = R(F) { }.
- Right(F) = {, id}.

Замена наборов:

- Right(E) = {, id, , +}.
- Right(T) = {, id, }.
- Right(F) = {, id}.

5.1.4. Крайний левый

Крайние левые рассчитываются, как подробно описано в Таблице 5.

Таблица 5. Расчет крайних левых наборов для CFG в примере 1.

Крайний левый (E)			
Для E E + T		Для E T	
1. Оценка первого условие алгоритма 3.	A = E β = ε Б = E γ = +T Lm(E) Lm(E)	2. Оценка первого условие алгоритма 3.	A = E β = ε Б = T γ = ε Лм(E) = Лм(T)

Крайний левый(T)			
Для T T F		Для T F	
1. Оценка первого условие алгоритма 3.	A = T β = ε Б = T γ = F Lm(T) Lm(T)	2. Оценка первого условие алгоритма 3.	A = T β = ε Б = Ф γ = ε Лм(T) = Лм(Ф)

Крайний левый(F)			
Для F (E)		Для F id	
1. Оценка второго условие алгоритма 3.	A = Ф β = ε a = (γ = E) Lm(F) {(}	2. Оценка второго условие алгоритма 3.	A = Ф β = ε a = идентификатор γ = ε Lm(F) = {(, id}

Краткое описание крайних левых наборов нетерминалов:

- Крайний левый(E) = Lm(T).

Machine Translated by Google

• Крайний левый(T) = Lm(F).
• Крайний левый(F) = {(, id}.

Замена наборов:
• Крайний левый(E) = {(, id}.
• Крайний левый(T) = {(, id}.
• Крайний левый(F) = {(, id}.

5.1.5. Тематические исследования

Для алгоритма 5 показано применение некоторых отношений старшинства.

1. Для символа-разделителя $:

(а) $ L(E) Правило (П.1).
• $ +
• $
•
• $ (
•
$ идентификатор

(б) R(E) $ Правило (П.2).
•) $
•
идентификатор $
• $
• + $

2. Для E E + T

(а) X = E, Y = + Правило (D.1).
P(Э) +
•) +
•
идентификатор +
• +
• + +

и = ноль, л = []
(б) X = +, Y = T Правило (C).
+ Л(Т)
• +
• + (
• + идентификатор

ты = +, л = [ ]

3. Для F (E)

(а) X = (, Y = E Правило (C).
(Л(Е)
• (+
• (
• ((
•
(идентификатор

ты = (, l = [ ]
(б) X = E, Y =) Правило (D.1).
P(E))
•))
•
идентификатор)
•)
• +)
u≠ null u .= b (.=) Правило (D.2).

и = ноль, л = []

Machine Translated by Google

4. В следующих случаях форма XY не выполняется в правой части продукции ; следовательно, никакое правило не может быть применено:

(а) E T

(б) T F

(c) F id

Остальные отношения предшествования выводятся с использованием того же метода, что и в ранее продемонстрированных случаях.

5.1.6. Таблица приоритета операторов

В конце выполнения алгоритма получается таблица приоритетов операторов 6 .

Таблица 6. Таблица приоритетов примера 1.

	+		(	)		$
+						
				.=		
идентификатор $						

5.1.7. Синтаксический анализ снизу вверх

Пусть id + ((id + id) (id)) id — строка, полученная путем крайнего правого вывода, тогда применение алгоритма 1 приводит к восходящему анализу, подробно описанному в таблице 7.

Таблица 7. Анализ примера 1 «снизу вверх».

Куча	Введите	Действие
$	id + ((id + id) (id)) id$ +((id + id)	Сдвиг
	(id)) id$ +((id + id) (id))	Уменьшать
$	id$ ((id + id) (id)) id$ (id + id)	Сдвиг
$+	(id)) id$ id + id) (id))	Сдвиг
$ +	id$ +id) (id)) id$ +id)	Сдвиг
($ +	(id)) id$ id) (id)) id$)	Сдвиг
(($ +	(id)) id$) (id))	Уменьшать
((id $	id$) (id)) id$	Сдвиг
+ (($ +	(id)) id$ (id))	Сдвиг
((+ $ +	id$ (id)) id$ id))	Уменьшать
((+id $	id$)) id$))	Уменьшать
+ ((+	id$	Сдвиг
$ +		Уменьшать
(($		Сдвиг
+ (() $		Сдвиг
+ ($ +		Сдвиг
($ +)		Уменьшать
(*($ + ((id $ + ((		Сдвиг

Machine Translated by Google

Таблица 7. Продолжение.

Куча	Ввод)	Действие
$ + (()	id$)	Уменьшать
$ +	id$)	Уменьшать
($	id$	Сдвиг
+ ($ + ()	id$	Уменьшать
$+	id$	Сдвиг
$ +		Сдвиг
$ + id	$	Уменьшать
$ +	$	Уменьшать
$+	$	Уменьшать
$	$	Принимать

5.2. Пример 2

5.2.1. Грамматика

Учитывая следующие не-ОПГ:

C CД; | Д;
D Тид (L)
T T * | интервал
Л Я | ε
Я T | T, я

Наборы строятся поэтапно, следуя алгоритмам.

5.2.2. Левый

Левые наборы рассчитываются, как подробно описано в Таблице 8 .

Таблица 8. Расчет левых наборов для CFG в примере 2.

Левый(S)			
Для S SD;			
1. Оценка первого условие алгоритма 2.	A = C β = ε B = S γ = D; L(S) L(S)	2. Оценка первого условие алгоритма 2.	A = C β = C Б = Д γ =; L(S) L(D)
3. Оценка второго условие алгоритма 2.	A = C β = SD a =; γ = ε L(S) L(D) {; }		
Для S D;			
4. Оценка первого условие алгоритма 2.	A = C β = ε Б = Д γ =; L(S) L(D) {; } L(D)	5. Оценка второго условие алгоритма 2.	A = C β = Д a =; γ = ε L(S) = L(D) {; }

Machine Translated by Google

Таблица 8. Продолжение.

Левый(Д)			
Для D Tid(L)			
1. Вычисление первого условия алгоритма 2.	A = Д β = ε Б = Т γ = идентификатор(L) L(D) L(T)	2. Вычисление второго условия алгоритма 2.	A = Д β = T a = идентификатор γ = (Л) L(D) = L(T) {id}
Левый(Т)			
Для Т Т			
1. Вычисление первого условия алгоритма 2.	A = T β = ε Б = Т γ = L(T) L(T)	2. Вычисление второго условия алгоритма 2.	A = T β = T a = * γ = ε L(T) { }
Для Т int			
3. Вычисление второго условия алгоритма 2.	A = T β = ε a = int γ = ε L(T) = { , int}		
Левый(Л)			
Для L I			
1. Вычисление первого условия алгоритма 2.	A = L β = ε Б = Я γ = ε Л(Л) = Л(Я)		
Левый(я)			
Для I Т			
1. Вычисление первого условия алгоритма 2.	A = I β = ε Б = Т γ = ε L(I) L(T)		
Для I Т, я			
2. Вычисление первого условия алгоритма 2.	A = я β = ε Б = Т γ =, я L(I) L(T)	3. Вычисление второго условия алгоритма 2.	A = я β = T a =, γ = I L(I) = L(T) {, }

Краткое изложение левых наборов нетерминалов: • Left(S) = L(D) {; }. • Left(D) = L(T) {id}. • Left(T) = { , int}. • Влево(L) = L(I). • Left(I) = L(T) {, }.

Machine Translated by Google

Подстановка множеств: •

Left(S) = {; , идентификатор, *,
интервал}. • Left(D) = {id, , int}. •
Left(T) = { , int}. • Left(L) = {, ,
*, int}. • Left(I) = {, , *, int}.

5.2.3. Right

Наборы Right рассчитываются, как подробно описано в Таблице 9.

Таблица 9. Расчет правых множеств для CFG в примере 2.

Справа(С)			
Для S SD;		Для S D;	
1. Вычисление второго условия алгоритма 4.	A = C γ = SD a =; β = ε P(C) {; }	2. Вычисление второго условия алгоритма 4.	A = C γ = Д a =; β = ε P(C) = {; }

Вправо(Д)	
Для D Tid(L)	
1. Вычисление второго условия алгоритма 4.	A = Д γ = Tid(L a =) β = ε P(Д) = {)}

Вправо(Т)			
Для T T		Для T int	
1. Вычисление второго условия алгоритма 4.	A = T γ = T a = * β = ε R(T) { }	2. Вычисление второго условия алгоритма 4.	A = T γ = ε a = int β = ε R(T) = { , int}

Вправо (Л)	
Для L I	
1. Вычисление первого условия алгоритма 4.	A = Л γ = ε Б = Я β = ε P(Л) = P(Я)

Machine Translated by Google

Таблица 9. Продолжение.

Верно (я)			
Для I T			
1. Вычисление первого условия алгоритма 4.	A = я γ = ε Б = Т β = ε Р(И) P(T)		
Для I Т, я			
2. Вычисление первого условия алгоритма 4.	A = я γ = Т, Б = Я β = ε R(I) R(T) R(I)	3. Вычисление второго условия алгоритма 4.	A = я γ = Т а =, β = я R(I) = R(T) {, }

Краткое описание правильных наборов нетерминалов: • Right(S) = {; }. • Вправо(D) = {)}. • Right(T) = { , int}. • Вправо(Л) = П(I). • Right(I) = R(T) {, }.

Подставляя множества:

• Right(S) = {; }. • Вправо(D) = {)}. • Right(T) = { , int}. • Right(L) = {, , *, int}. • Right(I) = {, , *, int}.

5.2.4. Крайний левый

Крайние левые наборы рассчитываются, как подробно описано в Таблице 10.

Таблица 10. Расчет крайних левых наборов для CFG в примере 2.

Крайний левый(S)			
Для S SD;		Для S D;	
1. Вычисление первого условия алгоритма 3.	A = C β = ε Б = С γ = Д; Lm(S) Lm(S)	2. Вычисление первого условия алгоритма 3.	A = C β = ε Б = Д γ =; Лм(С) = Лм(Д)

Крайний левый(D)		
Для D Tid(L)		
1. Вычисление первого условия алгоритма 3.	A = Д β = ε Б = Т γ = идентификатор(L) Лм(Д) = Лм(Т)	

Machine Translated by Google

Таблица 10. Продолжение.

Крайний левый(T)			
Для T T		Для T int	
1. Оценка первого условие алгоритма 3.	A = T β = ε Б = T γ = Lm(T) Lm(T)	2. Оценка второго условие алгоритма 3.	A = T β = ε a = целое число γ = ε Lm(T) = {int}

Крайний левый (Л)			
Для L I			
1. Оценка первого условие алгоритма 3.	A = Л β = ε Б = Я γ = ε Лм(Л) = Лм(Я)		

Крайний левый(I)			
Для I T		Для I T, я	
1. Оценка первого условие алгоритма 3.	A = я β = ε Б = T γ = ε Lm(I) Lm(T)	2. Оценка первого условие алгоритма 3.	A = я β = ε Б = T γ =, я Lm(I) = Lm(T)

Краткое описание крайних левых наборов нетерминалов:

- Крайний левый(S) = Lm(D).
- Крайний левый(D) = Lm(T).
- Крайний левый(T) = {int}.
- Крайний левый(L) = Lm(I).
- Крайний левый(I) = Lm(T).

Замена наборов:

- Крайний левый(S) = {int}.
- Крайний левый(D) = {int}.
- Крайний левый(T) = {int}.
- Крайний левый(L) = {int}.
- Крайний левый(I) = {int}.

5.2.5. Тематические исследования

Для алгоритма 5 показано применение некоторых отношений старшинства.

1. Для символа-разделителя $:

(а) $ L(S) Правило (П.1).

- $;
- $ идентификатор
- $
- $ целое

(б) R(S) $ Правило (П.2).

- ; $

2. Для S SD;

Machine Translated by Google

(а) X = S, Y = D Правило (E.1).

R(S) Lm(D)

- ; интервал

и = ноль, л = []

(б) X = D, Y = ; Правило (D.1).

P(Д) ;

-) ;

и = ноль, л = []

3. Для D Tid(L)

(а) X = id, Y = (Правило (B).

- идентификатор .= (

и = ноль, л = []

(б) X = (, Y = L Правило (C).

(Л(Л)

- ,
- ((
- (целое

ты = (, l = [ ]

(c) X = L, Y =) Правило (D.1).

P(Л))

- ,)
-)
- интервал)

u≠ null u .= b (.=) Правило (D.2).

и = ноль, л = []

4. В следующих случаях форма XY не выполняется в правой части продукции ; следовательно, никакое правило не может быть применено:

(а) T int

(б) Л Я

(c) L ε

(г) Я T

Остальные отношения предшествования выводятся с использованием того же метода, что и в ранее продемонстрированных случаях.

5.2.6. Таблица приоритета операторов

В конце выполнения алгоритма получается таблица приоритетов операторов 11 .

Таблица 11. Таблица приоритетов примера 2.

	;	идентификатор	(	)		интервал	,	$
			.=					
идентификатор				.=				
)								
интервал								
,								
$								

Machine Translated by Google

5.2.7. Синтаксический анализ снизу вверх

Пусть int id(); интервал идентификатор (интервал, интервал); быть строкой, полученной путем крайнего правого вывода, а затем применения Алгоритм 1 приводит к восходящему анализу, подробно описанному в Таблице 12.

Таблица 12. Анализ примера 2 «снизу вверх».

Куча	Введите int	Действие
$	id(); интервал идентификатор (интервал, интервал); $	Сдвиг
$интервал	идентификатор(); интервал идентификатор (интервал,	Уменьшать
$	интервал); $ идентификатор(); интервал	Сдвиг
$идентификатор	идентификатор (интервал, интервал); $ ();	Сдвиг
	интервал идентификатор (интервал,	Сдвиг
	интервал); $); интервал идентификатор	Уменьшать
$id($id() $	(интервал, интервал); $; интервал	Сдвиг
$;	идентификатор (интервал, интервал); $;	Уменьшать
$	интервал идентификатор (интервал,	Сдвиг
$интервал	интервал); $ int id(интервал,	Уменьшать
$	интервал); $ int id(интервал,	Сдвиг
$идентификатор	интервал); $ идентификатор	Сдвиг
	(интервал, интервал); $	Сдвиг
$id($id(int	идентификатор	Уменьшать
	(интервал,	Сдвиг
	интервал); $	Сдвиг
$id($id(,		Уменьшать
$id(,		Уменьшать
int		Сдвиг
$id(, $id($id())	(интервал, интервал); $ интервал, интервал); $, интервал); $ интервал); $ инт); $); $); $); $; $	Уменьшать
$	; $	Сдвиг
$;	$	Уменьшать
$	$	Принимать

5.3. Пример 3

5.3.1. Грамматика

Учитывая следующие не-ОПГ:

S ABC

A aA | a

Б бБ | б | ε

C CDc | c

Д д

Наборы строятся поэтапно, следуя алгоритмам.

5.3.2. Левый

Левые наборы рассчитываются, как подробно описано в Таблице 13 .

Machine Translated by Google

Таблица 13. Расчет левых наборов для CFG в примере 3.

Левый(S)			
Для S ABC			
1. Вычисление первого условия алгоритма 2.	A = S β = ε Б = А γ = до нашей эры L(S) L(A)	2. Вычисление первого условия алгоритма 2.	A = S β = А Б = Б γ = С L(S) L(A) L(B)
3. Вычисление первого условия алгоритма 2.	A = C β = AB Б = С γ = ε L(S) = L(A) L(B) L(C)		

Левый(А)			
Для А aA			
1. Вычисление второго условия алгоритма 2.	A = A β = ε a = a γ = A L(A) {a}	2. Вычисление первого условия алгоритма 2.	A = A β = a Б = А γ = ε L(A) {a} L(A)
Для А a			
3. Вычисление второго условия алгоритма 2.	A = A β = ε a = a γ = ε L(A) = {a}		

Левый(Б)			
Для В bB			
1. Вычисление второго условия алгоритма 2.	A = B β = ε a = b γ = B L(B) {b}	2. Вычисление первого условия алгоритма 2.	A = Б β = б Б = Б γ = ε L(B) {b} L(B)
Для В b			
3. Вычисление второго условия алгоритма 2.	A = B β = ε a = б γ = ε L(B) = {b}		

Machine Translated by Google

Таблица 13. Продолжение.

Слева(C)			
Для C CDc			
1. Вычисление первого условия алгоритма 2.	A = C β = ε B = C γ = Dc Л(C) Л(C)	2. Вычисление первого условия алгоритма 2.	A = C β = C Б = Д γ = c L(C) L(D)
3. Вычисление второго условия алгоритма 2.	A = C β = CD a = c γ = ε L(C) L(D) {c}		
Для C c			
4. Вычисление второго условия алгоритма 2.	A = C β = ε a = c γ = ε L(C) = L(D) {c}		

Левый(Д)			
Для D d			
1. Вычисление второго условия алгоритма 2.	A = D β = ε a = d γ = ε L(D) = {d}		

Краткое описание левых множеств нетерминалов: • Left(S) = L(A) L(B) L(C). • Левый(A) = {a}. • Левый(B) = {b}. • Left(C) = L(D) {c}. • Влево(D) = {d}.

Подставляя множества: • Left(S) = {a, b, d, c}. • Левый(A) = {a}. • Левый(B) = {b}. • Левый(C) = {d, c}. • Влево(D) = {d}.

5.3.3. Right

Наборы Right рассчитываются, как подробно описано в Таблице 14.

Machine Translated by Google

Таблица 14. Расчет правых множеств для CFG в примере 3.

Справа(С)			
Для S ABC			
1. Вычисление первого условия алгоритма 4.	A = S γ = AB B = C β = ε P(C) = P(C)		

Справа(А)			
Для А aA			
1. Вычисление первого условия алгоритма 4.	A = A γ = a B = A β = ε P(A) P(A)	2. Вычисление второго условия алгоритма 4.	A = A γ = ε a = a β = A R(A) {a}
Для А a			
3. Вычисление второго условия алгоритма 4.	A = A γ = ε a = a β = ε R(A) = {a}		

Справа(Б)			
Для В bB			
1. Вычисление первого условия алгоритма 4.	A = Б γ = б Б = Б β = ε P(B) P(B)	2. Вычисление второго условия алгоритма 4.	A = Б γ = ε a = b β = В R(B) {b}
Для В b			
3. Вычисление второго условия алгоритма 4.	A = Б γ = ε a = b β = ε R(б) = {б}		

Вправо(С)			
Для С CDc		Для С c	
1. Вычисление второго условия алгоритма 4.	A = C γ = CD a = c β = ε R(C) {c}	2. Вычисление второго условия алгоритма 4.	A = C γ = ε a = c β = ε R(C) = {c}

Machine Translated by Google

Таблица 14. Продолжение.

Вправо(Д)	
Для D d	
1. Вычисление второго условия алгоритма 4.	A = Д γ = ε a = d β = ε R(D) = {d}

Краткое изложение правильных наборов нетерминалов: • Right(S) = R(C). • Право(A) = {a}. • Вправо(B) = {b}. • Вправо(C) = {c}. • Вправо(D) = {d}.

Замена множеств: • Right(S) = {c}. • Право(A) = {a}. • Вправо(B) = {b}. • Вправо(C) = {c}. • Вправо(D) = {d}.

5.3.4. Крайний левый

Крайние левые наборы рассчитываются, как подробно описано в Таблице 15.

Таблица 15. Расчет крайних левых наборов для CFG в примере 3.

Крайний левый(S)	
Для S ABC	
1. Вычисление первого условия алгоритма 3.	A = S β = ε Б = А γ = до нашей эры Lm(S) = Lm(A)

Крайний левый(A)			
Для A aA		Для A a	
1. Вычисление второго условия алгоритма 3.	A = A β = ε a = a γ = A Lm(A) {a}	2. Вычисление второго условия алгоритма 3.	A = A β = ε a = a γ = ε Lm(A) = {a}

Крайний левый(Б)			
Для B bB		Для B b	
1. Вычисление второго условия алгоритма 3.	A = B β = ε a = b γ = B Lm(B) {b}	2. Вычисление второго условия алгоритма 3.	A = B β = ε a = б γ = ε Lm(B) = {b}

Machine Translated by Google

Таблица 15. Продолжение.

Крайний левый(C)			
Для C CDc		Для C c	
1. Вычисление первого условия алгоритма 3.	A = C β = ε B = C γ = Dc Lm(C) Lm(C)	2. Вычисление второго условия алгоритма 3.	A = C β = ε a = c γ = ε Lm(C) = {c}

Крайний левый(D)		
Для D d		
1. Вычисление второго условия алгоритма 3.	A = Д β = ε a = d γ = ε Lm(D) = {d}	

Краткое описание самых левых наборов

нетерминалов: • Крайний
левый(S) = Lm(A). • Крайний
левый(A) = {a}. • Крайний
левый(B) = {b}. • Крайний
левый(C) = {c}. • Крайний левый(D) = {d}.

Подстановка

множеств: • Крайний
левый(S) = {a}. • Крайний
левый(A) = {a}. • Крайний
левый(B) = {b}. • Крайний
левый(C) = {c}. • Крайний левый(D) = {d}.

5.3.5. Тематические исследования

Для алгоритма 5 показано применение некоторых отношений старшинства.

1. Для символа-разделителя $:

(а) $ L(S) Правило (П.1). $
- а $
- б $
- г $
- в

(б) R(S) $ Правило (П.2). с
- $

2. Для S ABC

(а) X = A, Y = B Правило (E.1).
R(A) Lm(B) a
- b

u = null, l = [A]

(b) X = B, Y = C Правило (E.1).
R(B) Lm(C) b
- c

Machine Translated by Google

A l≠ [ ] Правило (Д.3).
R(A) Lm(C)
• а в

и = ноль, л = []

3. Для A aA

(а) X = a, Y = A Правило (C).
a L(A)
• а а

ты = а, л = [ ]

4. Для B bB

(а) X = b, Y = B Правило (C).
б Л(В)
• б б

ты = б, л = [ ]

5. Для C CDc

(а) X = C, Y = D Правило (E.1).
R(C) Lm(D)
• в г

и = ноль, л = []

(б) X = D, Y = c Правило (D.1).
R(D) c
• д c

и = ноль, л = []

6. В следующих случаях форма XY не выполняется в правой части продукции ; следовательно, никакое правило не может быть применено:

(а) A a
(б) Б б
(с) В ε
(г) С с
(д) D d

Остальные отношения предшествования выводятся с использованием того же метода, что и в ранее продемонстрированных случаях.

5.3.6. Таблица приоритета операторов

В конце выполнения алгоритма получается таблица приоритетов операторов 16 .

Таблица 16. Таблица приоритетов примера 3.

	а	б	с	д	$
а					
б					
с					
д					
$					

5.3.7. Синтаксический анализ снизу вверх

Пусть aabbcdc — строка, полученная путем крайнего правого вывода, затем применяя алгоритм 1. приводит к восходящему анализу, подробно описанному в таблице 17.

Machine Translated by Google

Таблица 17. Анализ примера 3 «снизу вверх».

Куча	Вход	Действие
$	ааббкдк$	Сдвиг
$a	abbcdc$	Сдвиг
$aa	bbcdc$	Уменьшать
$a	bbcdc$	Уменьшать
$	bbcdc$	Сдвиг
$б	bcdc$	Сдвиг

Таблица 17. Продолжение.

Куча	Введите	Действие
$бб	cdc$	Уменьшать
$б	cdc$	Уменьшать
$	cdc$	Сдвиг
$c	постоянный ток $	Уменьшать
$	постоянный ток $	Сдвиг
$д	c$	Уменьшать
$	c$	Сдвиг
$c	$	Уменьшать
$	$	Принимать

5.4. Обсуждение

В примерах рассматриваются три CFG: один OPG и два не-OPG. В первый CFG, замечено, что он соответствует основному правилу, тогда как последующие соответствуют нет, поскольку они представляют произведения с последовательными нетерминальными символами справа сторона. Независимо от характера CFG, предложенные алгоритмы играют решающую роль в получение таблицы приоритета операторов. Использование алгоритма 1 , примененного к определенной строке для каждого CFG выполняется анализ снизу вверх, проверяя, что строки генерируются их соответствующие грамматики. Этот процесс подчеркивает функциональность предложения по выполнять синтаксический анализ снизу вверх на основе приоритета операторов, даже если CFG не выполнить основное требование ОПГ.

6. Исключения

Если CFG имеет любую из следующих форм, правила, изложенные в алгоритмах этого бумага не будет применяться:

1. A aBб

   B βa

   Наборы для Б во второй постановке таковы:

   $L(B) = \{a\}.$
   $R(B) = \{a\}.$

   Отношения предшествования, основанные на разделах алгоритма 5 , демонстрируют:

   • Для правила (C): a a и u = a.
   • Для правила (D.1): a b.
   • Для правила (D.2): a .= b.

   Применение правил (D.1) и (D.2) порождает конфликт.

Machine Translated by Google

2. A aBb B

aβ Так как

β = ε β = γa.

Согласно установленным условиям множества для B во второй продукции таковы: L(B) = {a}.

R(B) = {a}.

Отношения предшествования, основанные на разделах алгоритма 5, демонстрируют:

• Для правила (C): a a и u = a. • Для правила (D.1): a b. • Для правила (D.2): a .= b.

Применение правил (D.1) и (D.2) порождает конфликт.

Если R(B) не содержит a, то конфликта не возникает.

3. A bBa B

aβ

Множествами для B во второй продукции являются:

L(B) = {a}.

R(B) = {a}.

Отношения предшествования, основанные на разделах алгоритма 5, демонстрируют:

• Для правила (C): b a и u = b. • Для правила (D.1): a a. • Для правила (D.2): b .= a.

Применение правил (C) и (D.2) порождает конфликт.

4. A bBa

B βa

Поскольку β = ε β = aγ.

Согласно установленным условиям множества для B во второй продукции таковы: L(B) = {a}.

R(B) = {a}.

Отношения предшествования, основанные на разделах алгоритма 5, демонстрируют:

• Для правила (C): b a и u = b. • Для правила (D.1): a a. • Для правила (D.2): b .= a.

Применение правил (C) и (D.2) порождает конфликт.

Если L(B) не содержит a, то конфликта не возникает.

Когда продукция CFG примет следующую форму, конфликтов не будет:

A aБб

B γ ,

где γ не имеет ни a в конце правой части, ни b в конце левой части.

7. Выводы

Предыдущая работа была посвящена применению OPG для решения проблем, связанных с анализом различных форм языков; однако мало внимания уделялось определению левого и правого наборов, а также созданию алгоритма для поиска таблицы приоритета операторов.

В этой работе мы переопределили наборы «Справа» и «Слева», ранее представленные другими авторами, и представили новый набор под названием «Самый левый». Эти наборы составляют основу алгоритма, создающего таблицу приоритета операторов. Каждый набор построен на основе алгоритма, который облегчает его получение, добавляя новое измерение к существующим исследованиям.

Предложенный алгоритм построения таблицы приоритетов, нарушая (за некоторыми исключениями) базовое определение ОПГ, открывает новые возможности. Он позволяет построить таблицу приоритета операторов из CFG, не относящегося к OPG, открывая путь восходящему алгоритму синтаксического анализа для распознавания строки, сгенерированной CFG.

Machine Translated by Google

Показаны три примера CFG: один OPG и два не-OPG. Предложенные алгоритмы систематически применяются к этим примерам, шаг за шагом, пока не будут получены таблицы приоритета операторов, независимо от природы CFG. Кроме того, алгоритм синтаксического анализа «снизу вверх» применяется к конкретным строкам для каждой грамматики, обеспечивая проверку того, что они их сгенерировали.

Важно отметить, что хотя алгоритм и определение новых наборов терминальных символов являются значительным достижением, у них есть ограничения. Не всегда удается решить все случаи, когда два или более соседних терминальных символа находятся справа . Есть исключения, и мы должны о них знать.

Вклад авторов: концептуализация, LL и JM; методология, Дж.М.; программное обеспечение, LL; валидация, LL и JM; формальный анализ, LL, EA и JM; расследование, LL и JM; ресурсы, Дж.М.; курирование данных , EA; написание — подготовка оригинального проекта, EA и JM; написание — обзор и редактирование, EA и JM; визуализация, советник; надзор, Дж.М.; администрация проекта, JM; приобретение финансирования, JM; Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Это исследование не получило внешнего финансирования, а APC финансировался Университетом дель Норте.

Заявление о доступности данных. Совместное использование данных не применимо к этой статье.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Ссылки

1. Флойд Р.В. Синтаксический анализ и приоритет операторов. Дж. ACM (JACM) 1963, 10, 316–333. [Перекрестная ссылка] 2.
Альфред, В.; Моника, С.; Рави, С.; Джеффри, Принципы и методы компиляторов DU; Pearson Education: Лондон, Великобритания, 2007 г.
3. Грюн, Д.; Джейкобс, CJH « Техники синтаксического анализа» (монографии по информатике); Шпрингер: Берлин, Гейдельберг, 2006.
4. Креспи-Регицци, С. Эффективная модель грамматического вывода. В обработке информации 71; Гилкрист, Б., Ред.; Эльзевир: Северная Голландия, Нью-Йорк, 1972; стр. 524–529.
5. Креспи-Регицци, С. Сокращение перечисления при усвоении грамматики. В материалах 2-й Международной совместной конференции по искусственному интеллекту, Сан-Франциско, Калифорния, США, 1–3 сентября 1971 г.; стр. 546–552.
6. Креспи-Регицци, С.; Мелканов, Массачусетс; Лихтен, Л. Использование грамматического вывода для разработки языков программирования. Коммун. АКМ 1973, 16, 83–90. [Перекрестная ссылка]
7. фон Браунмюль, Б.; Вербек, Р. Языки, управляемые вводом, распознаются в пространстве log n. В разделах теории вычислений; Математические исследования Северной Голландии; Карплнски М., ван Леувен Дж., ред.; Elsevier: Северная Голландия, Нидерланды, 1985 г.; Том 102, стр. 1–19.
8. Мельхорн, К. Галечные горные хребты и их применение для распознавания DCFL. В автоматах, языках и программировании; де Баккер Дж., ван Леувен Дж., Ред.; Шпрингер: Берлин/Гейдельберг, Германия, 1980 г.; стр. 422–435.
9. Мандриоли, Д.; Праделла, М. Обобщение языков, управляемых вводом: теоретические и практические преимущества. Вычислить. наук. Версия 2018 г., 27, 61–87. [Перекрестная ссылка]
10. Алур Р.; Мадхусудан, П. Добавление вложенной структуры к словам. Дж. ACM 2009, 56, 1–43. [Перекрестная ссылка]
11. Креспи Региззи, С.; Мандриоли, Д. Приоритет операторов и свойство визуального нажатия. Дж. Компьютер. Сист. наук. 2012, 78, 1837–1867. [Перекрестная ссылка]
12. Креспи Региззи, С.; Праделла, М. За пределами грамматик и языков с приоритетом операторов. Дж. Компьютер. Сист. наук. 2020, 113, 18–41. [Перекрестная ссылка]
13. Баренги, А.; Креспи Региззи, С.; Мандриоли, Д.; Панелла, Ф.; Праделла, М. Параллельный анализ стал практичным. наук. Вычислить. Программа. 2015, 112, 195–226. [Перекрестная ссылка]
14. Креспи-Регицци, С.; Мандриоли, Д.; Мартин Д.Ф. Алгебраические свойства языков приоритета операторов. Инф. Контрольный 1978 г., 37, 115–133. [Перекрестная ссылка]
15. Хомский, Н. Три модели описания языка. ИРЭ Транс. Инф. Теория 1956, 2, 113–124. [Перекрестная ссылка]
16. Ахо, А.В.; Уллман, Дж. Д. Теория синтаксического анализа, перевода и компиляции; Прентис-Холл Энглвуд: Клиффс, Нью-Джерси, США, 1973 г.; Том 1.
17. Лауден Кеннет, К. Создание компилятора: принципы и практика. В курсе технологии; PWS Publishing Co.: Монтгомери, Иллинойс, США, 1997 год.

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Выбор канала ЭЭГ при реабилитации больных с инсультом Оптимизатор НДТ

Мохаммед Азми Аль-Бетар 1,*, Заид Абди Алькарим Альяссери [2,3,*], Нур Камаль Аль-Каззаз [4], Шариф Насер Махадме 1,5, Набиль Салих Али [2] и Кристоф Гугер [6]

1 Центр исследований искусственного интеллекта (AIRC), Колледж инженерии и информационных технологий, Университет Аджмана, Аджман, а/я 346, Объединенные Арабские Эмираты;
2 sharif.makhadmeh@uop.edu.jo Центр исследований и разработок информационных технологий (ITRDC), Университет Куфы,
3 Наджаф 54001, Ирак; nabeel@uokufa.edu.iq Инженерный колледж, Университет
4 Варит Аль-Анбия, Кербела 56001, Ирак, Факультет биомедицинской инженерии, Инженерный колледж Аль-Хорезми, Багдадский университет, Багдад 47146, Ирак;
5 noorbme@kecbu.uobaghdad.edu.iq Кафедра обработки данных и искусственного интеллекта, факультет информационных
6 технологий, Университет Петры, Амман 1196, Jordan G.Tec Medical Engineering GmbH, 4521 Шидльберг, Австрия; guger@gtec.at * Адрес для переписки: m.albetar@ajman.ac.ae (MAA-B.); zaid.alyasseri@uokufa.e

Аннотация: Инсульт является основной причиной смертности во всем мире, нарушает мозговой кровоток, приводя к тяжелым повреждениям головного мозга. Гемиплегия, распространенное последствие, приводит к потере двигательной активности на одной стороне тела. Многие люди, пережившие инсульт, сталкиваются с долговременными двигательными нарушениями и требуют серьезной реабилитаци Электроэнцефалограммы (ЭЭГ) представляют собой неинвазивный метод мониторинга активности мозга и используются в интерфейсах мозг-компьютер (BCI), чтобы помочь в реабилитации. Задачи двигательного воображения (МИ), обнаруженные с помощью ЭЭГ, имеют решающее значение для разработки ИМК, которые помогают пациентам восстановить двигательную цель. Однако интерпретация сигналов ЭЭГ для задач ИМ остается сложной задачей из-за их сложности и низкого отношения сигнал/шум. Основная цель этого исследования — сосредоточиться на оптимизации выбора каналов в BCI на основе ЭЭГ специально для реабилитации после инсульта. Определение наиболее информативных каналов ЭЭГ имеет решающее значение для регистрации нервных сигналов, связанных с двигательными нарушениями у пациентов, перенесших инсульт.

В этой статье предлагается метод оптимизации на основе бинарного алгоритма летучей мыши (BA) для выбора наиболее релевантных каналов с учетом уникальных нейрофизиологических изменений у пациентов с инсультом. Этот подход способен повысить производительность BCI за счет повышения точности классификации и снижения размерности данных. Мы используем атрибуты время-энтропия-частота (TEF), обрабатываемые посредством автоматического анализа независимых компонентов с шумоподавлением вейвлет-преобразованием (AICA-WT), чтобы повысить четкость сигнала. Выбранные каналы и функции подтверждаются с помощью классификатора k-ближайшего соседа (KNN) с использованием общедоступных наборов данных BCI, что демонстрирует улучшенную классификацию задач MI и потенциал для лучших результатов реабилитации.

Ключевые слова: ЭЭГ; извлечение признаков; выбор канала; пациенты, перенесшие инсульт; Алгоритм НДТ

check for updates

Ссылка: Аль-Бетар, Массачусетс; Альяссери, ЗАА; Аль-Каззаз, Северная Каролина; Махадме, С.Н.; Али, Н.С.; Гугер, К. Канал ЭЭГ Отбор для пациентов с инсультом Реабилитация с использованием BAT Optimizer. Алгоритмы 2024, 17, 346. https://doi.org/10.3390/a17080346 .

Академический редактор: Габриэлла Трукко

Поступила: 21 мая 2024 г.
Пересмотрено: 14 июля 2024 г.
Принято: 15 июля 2024 г.
Опубликовано: 8 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария.
Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Всемирная организация здравоохранения (ВОЗ) называет инсульт второй по значимости причиной смертности в мире из-за ущерба, который он наносит мозгу вследствие нарушений мозгового кровообращения [1,2]. Одной из наиболее изнурительных форм повреждения нервной системы является гемиплегия, которая описывает частичный или полный паралич одной стороны тела, включая руку, ногу, стопу и кисть. Ишемический инсульт, наиболее распространенный тип инсульта, вызывает нарушение перфузии головного мозга, что приводит к быстрой потере функции мозга [3,4]. Шестьдесят и более процентов людей, переживших инсульт, нуждаются в реабилитации из-за постоянного нарушения двигательных функций [5]. Однако люди, пережившие инсульт, страдают от широкого спектра нарушений, включая зрительные и когнитивные нарушения, которые оказывают кумулятивное разрушительное воздействие на

Machine Translated by Google

выполнять даже самые элементарные задачи. Соответственно, исследования по эффективному лечению и реабилитации жертв инсульта были в центре внимания на протяжении многих лет [6].

Электроэнцефалограмма (ЭЭГ) — это неинвазивный метод, который оценивает электрическую активность мозга с хорошим временным разрешением, чтобы определить результаты различных умственных задач. Преобразуя сигналы ЭЭГ мозга в управляющие инструкции, интерфейсы мозг-компьютер (ИМК) помогают людям с физическими нарушениями взаимодействовать с внешним сенсорным миром [7]. Воображение движений (МИ) — это мысленное воображение движения тела без реального движения мышц, а сопровождающая его ритмическая активность мозга может быть записана и использована в качестве входных сигналов систем ИМК. Событийная десинхронизация /синхронизация (ERD/ERS) относится к обнаружению ритмических изменений мощности в сенсомоторной области в пределах Роландического мю (9–13 Гц) и бета (13–30 Гц) частотных диапазонов, которые могут быть используется для различения различных типов задач МИ [8]. В связи с тем , что ИМ не требует каких-либо посторонних стимулов, ИМК на основе ИМ имеют широкий спектр потенциальных применений [9].

Однако изменчивость и сложность сигналов ЭЭГ создают трудности для традиционного декодирования MI-BCI [7]. Важными и сложными остаются вопросы повышения способности декодирования ЭЭГ, извлечения различительной информации из ЭЭГ с низким соотношением сигнал/шум и реализации надежной классификации различных задач ИМ [8].

Выбор каналов в исследованиях на основе ЭЭГ важен для определения подмножества каналов, которое наиболее информативно с точки зрения захвата правильных нейронных сигналов для выбранного приложения [10]. Тем не менее, такие соображения и соответствующие им проблемы могут отличаться в других контекстах применения, и это можно увидеть при сравнении реабилитации после инсульта с обработкой эмоций и обнаружением деменции с использованием набора данных на основе ЭЭГ [11]. Например, реабилитация после инсульта характеризуется тем, что при неврологических дефектах, возникших в результате инсульта, возникают определенные изменения в деятельности головного мозга и функционировании нейронных сетей. Данные ЭЭГ также могут характеризоваться специфическими особенностями, отличающими их от здоровых людей или других пациентов, такими как изменения топологической организации сетей, аберрантная активность и появление новых регуляторных стратегий [12].

Следовательно, подход к выбору наиболее значимых каналов ЭЭГ для реабилитации после инсульта, возможно, должен быть более избирательным и специфичным. Идея состоит в том, чтобы выяснить, какой из каналов обеспечивает лучшую запись нейронных характеристик конкретного двигательного, когнитивного или поведенческого нарушения, которое может возникнуть у пациента с инсультом, и это не будет то же самое, что каналы, которые были бы идеальны для другие приложения [13]. Однако существуют отдельные подходы, такие как обработка эмоций и обнаружение деменции, которые могут использовать разные нейрофизиологические механизмы и/или спектрально-временные паттерны. Например , при обработке эмоций полоса частот, связанная с обработкой эмоций, может быть распределена по различным областям коры, и целью выбора канала может быть захват этой распределенной частотной активности. Аналогичным образом, в случае выявления деменции изменение структуры и функции мозга, определяющее когнитивные нарушения, может представлять собой другое пространственное распределение нагрузки, что требует других правил выбора каналов [14]. Предлагаемое исследование, ориентированное непосредственно на проблему выбора канала в контексте реабилитации после инсульта, призвано предложить эффективное решение, которое может улучшить эффективность BCI и, таким образом, улучшить процесс реабилитации.

Выбор канала является основной задачей данного исследования. Многие электроды используются в медицинских и диагностических целях. Для практических применений BCI для точности классификации необходимо множество электродов; однако установка большого количества электродов на кожу головы требует много времени и может привести к переобучению сигнала ЭЭГ. Чтобы решить эти проблемы, исследователям необходимо идентифицировать электроды, которые невозможно классифицировать. Выбор электродов может оценить неврологические знания людей из-за различных реакций и идеального размещения электродов, зависящего от субъекта [15]. Задание классификации, зависящее от пользователя, требует автоматического определения релевантности электродов. Эту проблему можно решить, рассмотрев широкий спектр электродов и используя множество методов для определения подходящих каналов для каждого пациента. Метод оптимизации на основе частиц BA использовался для выбора наиболее точн

Machine Translated by Google

ЭЭГ-каналы для реабилитации пациентов с инсультом. Каждый канал данных ЭЭГ обеспечивает уникальные функции классификации. Выбор канала сокращает хранение и обработку данных, ускоряет обучение классификатора за счет использования более простых данных и уменьшает «проклятие размерности».

Многие исследования изучали выбор каналов у здоровых людей [16,17], тогда как настоящее исследование сосредоточено на реабилитации после инсульта, где основные нейрофизиологические нарушения и ограничения пациентов с инсультом могут потребовать индивидуальной стратегии выбора каналов.

Выбор каналов ЭЭГ для обработки эмоций, когнитивной оценки и клинической диагностики часто предполагает знание нервных механизмов и областей мозга, связанных с желаемой деятельностью. Общие закономерности активности мозга и оптимальный выбор каналов хорошо известны для здоровых людей. Однако инсульт может изменить структуру, возможности подключения и функции мозга. Эти изменения могут вызывать аномальные структуры мозга и компенсаторные процессы [18,19]. Таким образом, лучшие каналы ЭЭГ для реабилитации после инсульта могут отличаться от тех, которые рекомендуются здоровым людям.

Чтобы уловить нейронные корреляты двигательных, когнитивных и поведенческих нарушений у пациентов с инсультом, выбор каналов должен быть более специализированным и персонализированным из-за нейропластичности, вызванной инсультом. Нацеливание на пациентов с инсультом индивидуального решения, которое может правильно распознавать нейронные отпечатки и модели активности мозга, призвано связать недостатки, связанные с инсультом. Этот этап имеет решающее значение для разработки методов лечения инсульта на основе BCI, которые улучшают результаты реабилитации [20].

Литература по выбору каналов у здоровых людей может быть полезной, но это исследование признает необходимость решения уникальных проблем и соображений людей, переживших инсульт. Ученые изучили различия в выборе каналов между пациентами, перенесшими инсульт, и здоровыми людьми, чтобы лучше понять нейрофизиологические изменения и оптимальные настройки BCI для реабилитации после инсульта.

Поэтому в данном исследовании метод оптимизации на основе алгоритма бинарной летучей мыши (БА) используется для выявления наиболее важных каналов ЭЭГ для реабилитации после инсульта. Таким образом, этот метод подходит для нужд людей, переживших инсульт, за счет оптимизации технологии BCI и эффективности реабилитации.

Необработанные данные ЭЭГ сегментируются и фильтруются с использованием стандартных фильтров и автоматизированного анализа независимых компонентов с использованием метода шумоподавления с вейвлет-преобразованием (AICA-WT) с целью решения упомянутых выше проблем. Затем атрибуты время-энтропия-частота (TEF) создаются путем объединения эффективных характеристик из временной, энтропийной и частотной областей. В результате метод оптимизации на основе БА используется для оптимизации характеристик ТЭФ с целью выбора эффективных каналов, улучшающих стационарность и устойчивость системы. После того, как мы обучим методу классификатора KNN, мы проводим серию экспериментов для оценки предложенной нами структуры, используя 25 сеансов BCI на основе MI с последующими визитами для оценки для изучения функциональных изменений до и после нейрореабилитации ЭЭГ на основе общедоступных наборов данных конкурса BCI. Гибридные атрибуты время-энтропия-частота (TEF), используемые в структуре AICA-WT-TEF-Chs для классификации MI-BCI, предназначены для эффективного использования базовой информации временной, энтропийной и частотной областей для повышения эффективности классификации.

Подход, предложенный в этой статье, дает следующие преимущества. Во-первых, рассматривается, как переменные время-энтропия-частота (TEF) влияют на эффективность классификации.
Во-вторых, он использует эффективные каналы ЭЭГ, генерируемые методом оптимизации на основе БА, для обнаружения изменений как до, так и после реабилитации. В-третьих, он проверяет эффективность структуры AICA-WT-TEF-Chs с использованием различных моделей классификации и передовых методов. Устраняя посторонние измерения, эффективный выбор каналов уменьшает объем данных, которые необходимо хранить и обрабатывать, ускоряет обучение классификатора за счет использования более простых данных и предотвращает переобучение классификатора, уменьшая влияние «проклятия размерности».

2. Сопутствующие работы

ИМК на основе ЭЭГ, в частности, имеют хорошее временное разрешение [21], но записанная волновая активность может быть искажена различными способами в зависимости от возникших артефактов, поэтому предварительная обработка необработанных сигналов ЭЭГ имеет решающее значение для классификации сигналов на основе

Machine Translated by Google

В большинстве случаев эти артефакты могут имитировать или накладывать ненормальное поведение мозга. В исследованиях изучалось влияние артефактов на сигналы ЭЭГ, включая моргание глаз, движения глаз, сердечные артефакты и мышечную активность, а шум от линий электропередачи также может пересекать частоты ЭЭГ [22,23]. В результате оценка эффективности сигналов ЭЭГ при наличии фонового шума может быть затруднена. В этой ситуации пространственные фильтры могут генерировать более локализованный сигнал для каждого электрода, поскольку BCI на основе ЭЭГ, в частности, имеют плохое пространственное разрешение.

Чтобы выбрать наилучшие характеристики для анализа ЭЭГ на основе BCI-MI, извлечение признаков выполняется для каждого записанного канала в нескольких доменах. Наиболее популярным методом является использование общих пространственных паттернов (CSP) для получения дискриминантных характеристик из многомерных сигналов ЭЭГ, которые можно использовать для идентификации пространственных характеристик, связанных с ERD, создаваемых различными задачами ИМ [24]. Однако на эффективность CSP легко влияет шум и она очень чувствительна к выбору полос частот [16]. Для решения проблемы эффективности было разработано несколько методов извлечения признаков для анализа данных ЭЭГ во временной, частотной и частотно-временной областях [25]. Кроме того, в [26] были предложены характеристики биспектра вейвлет-когерентности (WC) для различения правого и левого ИМ.

Размеры особенностей, полученных из сигналов ЭЭГ, обычно довольно велики для каждого канала и становятся больше при переходе к следующему каналу. Классификация нескольких признаков требует дополнительных вычислений и времени. Чтобы решить эту проблему, необходимо получить наиболее эффективный набор функций, выбрав подмножество каналов ЭЭГ, которые более тесно связаны с умственной работой, чем другие. Четыре методологии оценки: фильтрация, упаковка, внедрение и гибрид могут использоваться для категоризации алгоритмов выбора признаков [27]. На основе конкретных статистических критериев учитывается метод фильтрации, используемый для двигательной активности ЭЭГ, основанной на намерении. Например, прежде чем классифицировать двигательное намерение, He et al. [28] разработали статистический метод границы Бхаттачарьи для выбора канала. С помощью этого метода выбиралось лучшее подмножество кандидатов, в котором также использовались стратегия CSP и последовательного поиска. Там и др. [29] предложили другой статистический метод, основанный на ранге CSP, для сортировки коэффициентов фильтра с абсолютными значениями и последующего выбора признаков с наибольшим последующим коэффициентом. Хотя метод фильтрации имеет низкую точность, он работает быстро и не зависит от выбора субъекта или классификатора.

С другой стороны, классификатор и предмет весьма важны в процессе выбора обертки [30]. Таким образом, кандидаты подмножества оцениваются с использованием точности классификации и, таким образом, могут делать более точные прогнозы, чем методы фильтрации. В результате стратегия обертки требует больше вычислительных затрат, чем метод фильтрации, и подвержена проблеме переобучения [31], которой можно избежать, используя меры перекрестной проверки для прогнозирования. Для выбора лучшего канала большинство стратегий-оболочек используют последовательный прямой поиск, стратегию обратного исключения [32] и метод эвристического поиска [33,34]. Не говоря уже об условиях остановки, процедуры фильтрации и упаковки также были объединены в целях выбора. Эта гибридная стратегия была создана для работы с огромными наборами данных. Например, Гаур и др. [35] предложили наилучший метод выбора канала на основе CSP. Прежде чем ранжировать все каналы в соответствии с их оценками и использовать точность классификации для оценки выбранной идеальной комбинации каналов, они сначала применили норму L1 CSP для группировки оценок вклада каналов. Наконец, используя определенные критерии для процесса обучения конкретного классификатора, был использован встроенный подход для выбора значимых каналов.

Чтобы выбрать каналы с лучшими результатами классификации, Ши и др. [36] использовали методы выбора признаков, рекурсивное удаление признаков и оптимизацию с нулевой нормой, обученные с помощью машин опорных векторов. Использование этих методик позволило существенно сократить каналы ЭЭГ [37].

Решающим шагом в классификации данных ЭЭГ является использование соответствующих функций, которые существенно влияют на способность системы эффективно классифицировать данные [38–40]. Следовательно, для достижения наилучшей эффективности классификации необходимо использовать достоверные признаки сигналов ЭЭГ [41].

Machine Translated by Google

Кроме того, методы глобальной оптимизации и классификация машинного обучения могут быть объединены для решения таких проблем в области BCI, включая извлечение и выбор признаков [42]. Несколько алгоритмов вызвали большой интерес благодаря своим приложениям при выборе признаков, например, дифференциальная эволюция (DE), имитация отжига (SA), оптимизация роя частиц (PSO), искусственная пчелиная колония (ABC) и оптимизация муравьиной колонии (ACO). [43].

Были проведены исследования по выбору каналов для различных парадигм BCI, включая исключение рекурсивных каналов для задач воображения движений [44]. В исследовании, проведенном Rakotomamonjy et al. [45] исследователи использовали технику выбора канала , встроенную в обучение классификатора SVM. Пространство признаков искалось с использованием ГА в ссылках [46,47], а функция приспособленности представляла собой взвешенную линейную комбинацию количества признаков и точности SVM. Удхая и др. в [48] предложили подход к выбору грубого набора признаков на основе PSO для определения наилучшего подмножества признаков, а точность классификатора окрестности использовалась в качестве критерия оценки подмножества признаков. DE был использован Baig et al. [49], а PSO и GA использовались Atyabi et al. [50] для того, чтобы найти лучшее подмножество функций соответственно. В [51] в процессе выбора характеристик для идентификации эмоций на основе ЭЭГ использовалась оптимизация муравьиных колоний, моделирование отжига, генетические алгоритмы, оптимизация роя частиц и дифференциальная эволюция. Эффективность этих подходов к выбору функций, основанных на EA, весьма обнадеживает.

Однако размер области поиска в этих алгоритмах предопределен; поэтому на каждой итерации важно принимать решение относительно того, следует ли выбирать каждую функцию. В этих обстоятельствах вычислительные усилия будут потрачены впустую на некоторые функции, которые являются избыточными, ненужными или тривиальными по своей природе. Кроме того, эволюционные алгоритмы (ЭА) легко сходятся к локальному оптимуму благодаря широкому пространству поиска, а также вмешательству избыточных, нерелевантных и тривиальных функций. В этом исследовании мы разрабатываем методы выбора признаков на основе EA для воображения движений BCI посредством уменьшения размерности.

Производительность классификатора резко снизится по мере увеличения количества признаков из-за проклятия размерности, которое возникает, когда количество признаков превышает определенный порог. Кроме того, процедура обучения занимает больше времени, чем выше становится размерность признака [46].

Чтобы найти большое количество существующих каналов, требуется выбор функций с тщательным поиском для извлечения связанных каналов [10,52]. Исследователи сосредоточились на вопросах выбора функций и предложили подходы, основанные на полном поиске, жадном поиске и случайном поиске [11]. Высокие вычислительные затраты и сходимость к локальному оптимуму остаются проблемами этих методов выбора [53].

3. Алгоритм BAT B

этом разделе представлен и проиллюстрирован алгоритм оптимизации летучих мышей с точки зрения его вдохновения, летучих мышей и их образа жизни в природе, а также общая процедура оптимизации метода и математическая формулировка.

3.1. Вдохновение для

BAT Алгоритм оптимизации летучих мышей был впервые предложен Синь-Ше Яном в 2010 году [54]. Синь -Ше Ян предложил алгоритм летучих мышей, основанный на характеристиках эхолокации летучих мышей, используемых для охоты на добычу и передвижения в темноте. Микролетучая мышь издает чрезвычайно громкие звуковые импульсы и слышит их эхо от объектов и добычи вокруг себя. По звуку эха летучие мыши могут определять тип добычи и стратегию охоты на нее. Летучие мыши обычно производят от 10 до 20 импульсов каждую секунду, когда ищут добычу. Как только они приближаются к добыче, количество пульсаций увеличивается до 200 импульсов в секунду, а звуковой пульс становится тише.

В природе летучие мыши обладают коллективным охотничьим поведением, когда они летают вместе с определенной скоростью, чтобы одновременно охотиться на добычу, используя характеристику эхолокации, как показано на рисунке 1. На рисунке показано движение летучих мышей, основанное на эхолокации.

Machine Translated by Google

механизм, при котором летучие мыши обновляют свое местоположение и движутся к добыче на основе эхо-звука и местоположения других летучих мышей, определяемого путем взаимодействия друг с другом. Более того, скорость летучих мышей становится выше, а пульс снижается, когда летучие мыши приближаются к месту добычи.

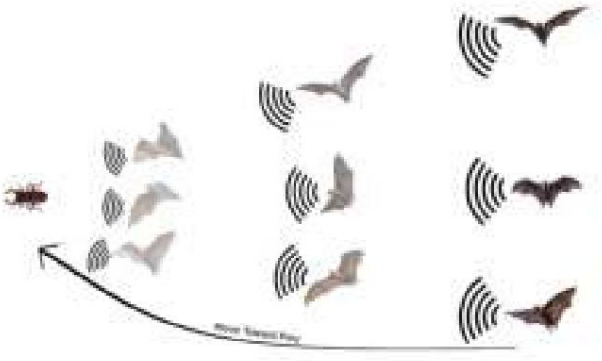


Рисунок 1. Движение летучей мыши к добыче.

В алгоритме BAT каждая летучая мышь рассматривается как решение в популяции, положение добычи является оптимальным решением для конкретной задачи оптимизации, местоположение представлено в виде вектора x с диапазоном между fmin и fmax, скорость microbats представлена в векторе v, а длина волны равна λ с частотой пульса r [0, 1] и диапазоном громкости A, где наибольшая из них представлена как Ao , а наименьшая — Amin.

3.2. Процедура BAT

В этом разделе представлена общая процедура оптимизации алгоритма BAT. Эта процедура оптимизации содержит шесть основных шагов, которые представлены на рисунке 2 и подробно обсуждаются ниже:

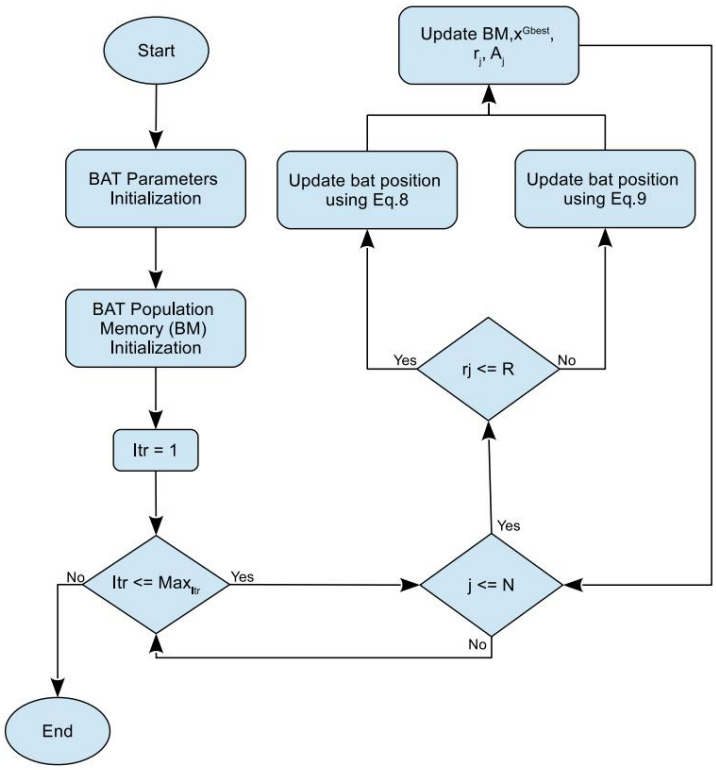


Рисунок 2. Блок-схема алгоритма Bat.

• Шаг 1: Инициализация параметров алгоритма BAT. На этом этапе параметры BAT инициализируются начальными значениями. Этими параметрами являются численность летучих мы

Machine Translated by Google

(решений) (N), максимальное число итераций (MaxItr), максимальный (fmax) и минимальный (fmin) диапазон частот, вектор скорости (v), скорость громкости (A), частоту пульса (r), начальную частоту ^{0}), два параметра (λ и α) с постоянными значениями в диапазоне [0, 1] и полоса пропускания пульса (r) диапазон (ε) в диапазоне [1, 1].

- Шаг 2: Инициализация памяти совокупности BAT. На этом этапе решения НДТ генерируются случайным образом с использованием уравнения (1) с учетом алгоритма НДТ и конкретных параметров и ограничений задачи оптимизации.

$$x_{ij} = lbi + (ubi \quad lbi) \times R, \tag{1}$$

где i = 1, 2, . . . , d, j = 1, 2, . . . , N и R — случайное число от 0 до 1.

Полученные решения вместе создают совокупность, как показано в уравнении (2), и сохраняются в памяти BAT (BM) в порядке возрастания на основе их пригодности Gbest и сохраняются в значениях. наиболее подходящими значениями присваивается первой позиции x BM. Лучшее решение с

$$\text{БМ} = \begin{matrix} x_1^1 & x_2^1 & \cdots & x_д^1 \\ x_1^2 & x_2^2 & \cdots & x_д^2 \\ \vdots & \vdots & \cdots & \vdots \\ x_1^H & x_2^H & \cdots & x_д^H \end{matrix}, \tag{2}$$

- Шаг 3: Интенсификация популяции летучих мышей. Теперь все летучие мыши x летают и меняют свое положение с учетом скорости v, определяемой частотой f , которая генерируется случайным образом, как показано в уравнениях (3) и (4). Соответственно, позиции бит обновляются с использованием уравнения (5).

$$fi = fmin + (fmin \quad fmax) \times R, \tag{3}$$

$$v_я^{'j} = v_я^{jj} + (x_я \quad x\ Gbest) \times fi, \tag{4}$$

$$x_я^{'j} \quad x\ ii, \quad + v = {}^{'j} \tag{5}$$

Примечательно, что микробы обновляют свои местоположения, чтобы быть ближе к лучшим в мире (x Gbest). Таким образом, новое положение летучих мышей усиливает положение в направлении Gbest к x .

- Шаг 4: Диверсификация популяции летучих мышей.

На этом этапе позиции микробатов обновляются на основе случайных параметров, чтобы попытаться найти лучшие глобальные решения. Новое решение выбирается из BM на основе метода выбора и присваивается x Gbest. Впоследствии текущее решение обновлено с помощью $x^{\text{Лучше}}$ всего следующим образом:

$$x_я^{'j} = x^{\text{Гбест}} + \varepsilon \times A\ j, \quad ' \tag{6}$$

где A ' это средняя громкость. Поскольку шаги 3 и 4 являются противоречивыми шагами, алгоритм BAT выбирает для выполнения один из них за раз. Этот выбор основан на следующем уравнении:

$$x^{'j} = \begin{cases} x^{\text{Гбест}} + \varepsilon \times A\ j \quad ' & \text{если } rj \quad R \\ x\ j + v\ 'j & \text{в противном случае} \end{cases}, \tag{7}$$

- Шаг 5: Обновление BM. Положение микробат в BM будет обновлено на этом этапе Gbest , если новое местоположение будет более подходящим, чем старое, и R Aj . Более того, x

Machine Translated by Google

обновляется, если новый BM содержит решение с лучшим значением пригодности. Впоследствии значения rj и Aj будут обновлены в соответствии с уравнениями (8) и (9).

$$0 \text{ гдж} = \text{рдж}(1 \quad e(\quad \gamma \times itr)), \tag{8}$$

$$Aj = \alpha \times Aj, \tag{9}$$

где itr обозначает текущий номер итерации.

• Шаг 6: Критерий остановки. Шаги 3, 4 и 5 будут повторяться до тех пор, пока алгоритм не достигнет критерий остановки.

4. Предлагаемый метод

В этом разделе представлен основной вклад данной статьи. Предлагаемый метод включает пять этапов, которые представлены на рисунке 3 и подробно обсуждаются ниже.

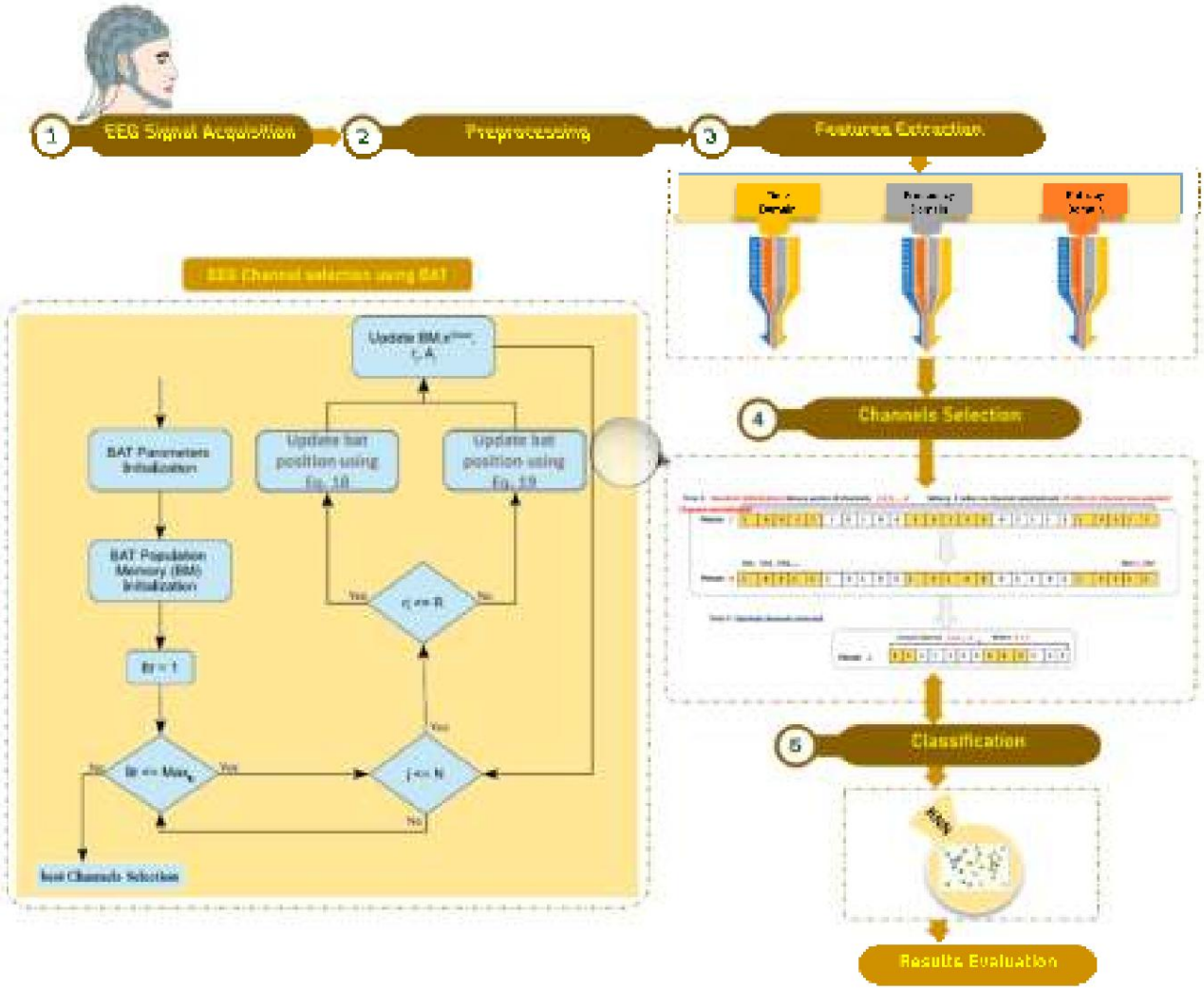


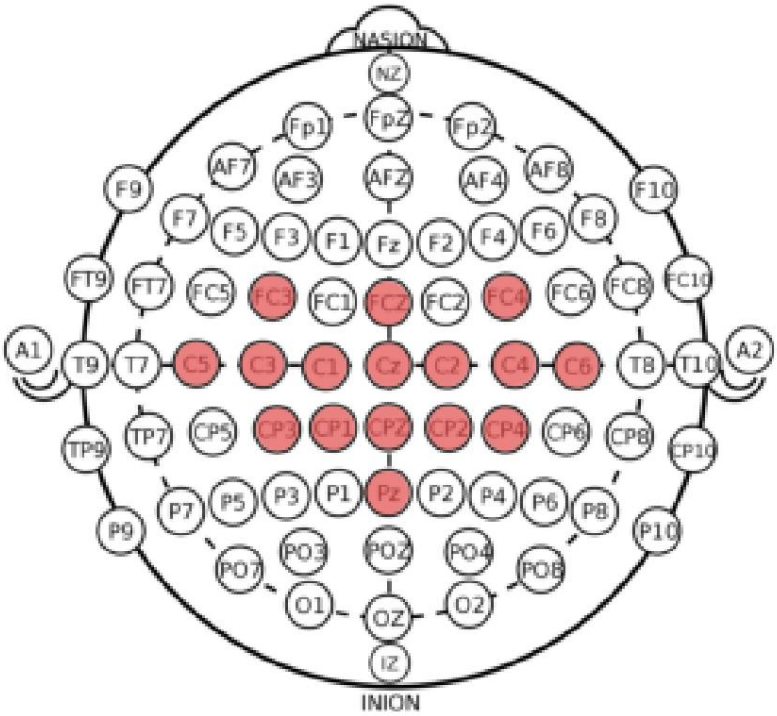
Рисунок 3. Предлагаемый метод выбора канала электроэнцефалограммы.

4.1. Фаза I: Сбор сигналов ЭЭГ. В этом

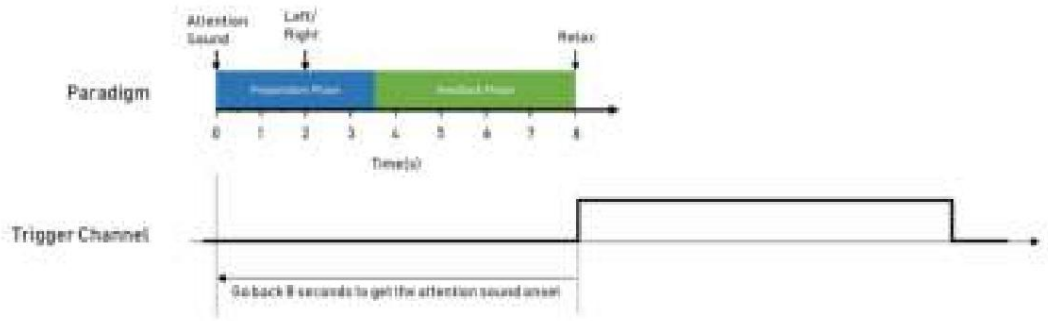
исследовании источником набора данных является стандартный набор данных ЭЭГ компании g.tec Medical Engineering GmbH; Подробности об этом наборе данных описаны в [8]. Изучены данные ЭЭГ постинсультных больных с гемипарезом верхних конечностей. В этом исследовании приняли участие 8 пациентов после инсульта, получавших лечение с помощью системы RecoveryX (медицинская инженерия g.tec).

Machine Translated by Google

GmbH), средний возраст 22 года (SD = 4,582). Каждый участник проходил обучение по ИМ на базе BCI в течение трех месяцев, по 2 учебных занятия в неделю (всего 25 учебных занятий). Исследовательская группа провела и проанализировала две оценки (до и после обучения).
Оценка перед обучением была запланирована за 30–35 дней до вмешательства, а оценка после обучения проводилась через несколько дней после вмешательства (см. Рисунок 4).
Этиккомиссия земель Оберостеррайха в Австрии (№ D-42-17) утвердила этот протокол исследования, и каждый пациент подписал форму информированного согласия перед предварительной
Наконец, этот набор данных записывается с частотой дискретизации 256 Гц.



(а)



(б)

Рис. 4. (а) Распределение электродов ЭЭГ по системе 10–20; (б) – принципиальная схема протокола записи ЭЭГ.

В соответствии с системными показаниями к мысленной задаче при ИМ пациентам было предложено визуализировать движение запястья тыльной стороны. До и после нейрореабилитации ЭЭГ пациенты участвовали в 25 сеансах BCI на основе ИМ с последующими посещениями для оценки функциональных изменений. Каждая сессия состояла из 240 повторений МИ обеими руками, разбитых на три пробежки по 80 попыток. Каждый сеанс длился примерно один час, включая время, необходимое для настройки и очистки. Задачи BCI на основе MI были проиллюстрированы рандомизированными интервалами между испытаниями в псевдослучайном порядке.

Machine Translated by Google

Для регистрации всех различных нервных процессов, происходящих во время реабилитации после инсульта, включая когнитивные, аффективные и двигательные компоненты, использовались шестнадцатиканальные активные ЭЭГ-капсулы . Колпачки изготовлены австрийской компанией g.tec Medical Engineering GmbH.

По международной системе 10–20 электроды ЭЭГ располагались следующим образом: (FC5, FC1, FCz, FC2, FC6, C5, C3, C1, Cz, C2, C4, C6, CP5, CP1, CP2, CP6.).
К испытуемому прикрепляли заземляющий электрод на уровне FPz , а референтный электрод - на мочке правого уха.

Эти каналы связаны с различными когнитивными и сенсомоторными процессами, которые в той или иной степени нарушаются у пациентов после инсульта. В частности, лобно-центральные каналы (FC5, FC1, FCz, FC2, FC6) связаны с эмоциональной регуляцией, рабочей памятью и вторичной двигательной областью. Кроме того, важно отметить, что центральные каналы (C5, C3, C1, Cz, C2, C4 и C6) связаны с сенсорно-моторной интеграцией и представлением движений тела. Кроме того, описано , что центро-теменные каналы (CP5, CP1, CP2, CP6) участвуют в процессе интеграции экстероцептивных и проприоцептивных входов, а также в регуляции произвольных движений [22].

4.2. Фаза II: Предварительная

обработка. В каждом канале записанного набора данных ЭЭГ первоначально использовались два стандартных фильтра. Во-первых, полосовой фильтр (BPF) с частотами от (8 до 30 Гц) использовался для ограничения полосы записываемых данных ЭЭГ, а во-вторых, режекторный фильтр Баттерворта (BW) на частоте (50 Гц) использовался для уменьшения шума. от помех в линии электропередачи. (ИК) s(t) = [s1(t), . . . ,sn(t)], используя алгоритм FastICA , предложенный в [55]. Набор s(t) из n неизвестных компонентов , которые были линейно смешаны в матрице A, и набор x(t) из n наблюдений, где x(t) = [x1(t), . . . , xn(t)] [56–58] представляют собой ЭЭГ и связаны с s(t), t – временем; следовательно, уравнение ICA имеет вид

$$x(т) = Ac(т) \quad (10)$$

Затем для оценки артефактных компонентов, IC, использовались три метрики : эксцесс (Kurt), асимметрия (Skw) и энтропия выборки (SampEn). Если эти параметры превышали порог pm1,2 для каждой микросхемы, микросхема отмечалась как критическая и удалялась с помощью WT. Практическая ценность порога была определена методом проб и ошибок и предыдущих исследований [59–61]. Пороговое значение ±1,2 не является недостатком метода AICA-WT, поскольку артефактные микросхемы не отбраковывались, а были очищены от шума с использованием метода WT. В результате WT обесшумил помеченные IC, прежде чем вернуть расширенные компоненты в исходный набор данных ЭЭГ [22,23].

4.3. Фаза III: Извлечение признаков ЭЭГ

В этой работе используются три различных метода извлечения признаков. Эти методы включают функции временной области, функции энтропии и функции частотной области, как показано в [8].

4.4. Фаза IV: Выбор канала ЭЭГ с помощью BAT Optimizer

Этот этап является основным и включает в себя несколько шагов для достижения оптимального подмножества каналов ЭЭГ, которое может обеспечить максимальную точность. Следующие шаги показывают, как мы адаптируем оптимизатор BAT для задачи выбора канала ЭЭГ: • Шаг 1:

Инициализация параметров алгоритма BAT. На этом этапе параметры BAT инициализируются начальными значениями. Этими параметрами являются количество микробатов (решений) (N), максимальное количество итераций (MaxItr), максимальный (fmax) и минимальный (fmin) диапазон частот, вектор скорости (v), скорость громкости (A), частота пульса (r).), начальная ^{0}), два параметра (λ и α) с постоянными значениями в диапазоне [0, 1] и полоса пропускания частота импульсов (диапазон r (ε) в диапазоне [-1, 1], как показано в таблице 1.

Machine Translated by Google

Таблица 1. Метаэвристические параметры.

Алгоритм	Параметры	Численность населения	Бежит
CS	β = 1,5, Па = 0,25 с1 =	20	30
ПСО	с2 = 2 А = 0,5, г	20	30
НДТ	= 0,5 α = 0,5, γ = 1,	20	30
ФФА	β0 = 0,2	20	30
ГВО	-	20	30

• Шаг 2: Инициализация памяти совокупности BAT. На этом этапе решения НДТ генерируются случайным образом с использованием уравнения (11) с учетом алгоритма НДТ и конкретных условий. параметры и ограничения задачи оптимизации.

$$x_i^{old} = lbi + (ubi - lbi) \times R, \quad (11)$$

где $i = 1, 2, \ldots, d$, $j = 1, 2, \ldots, N$ и R — случайное число от 0 до 1.

Полученные решения вместе генерируют совокупность, как показано в уравнении (12): и хранятся в памяти BAT (BM) в порядке возрастания в зависимости от их пригодности. ценности. Лучшее решение с наиболее подходящими значениями присваивается x_{Gbest} и хранится в первая позиция БМ.

$$БМ = \begin{matrix} x_1^1 & x_2^1 & \cdots & x_d^1 \\ x_1^2 & x_2^2 & \cdots & x_d^2 \\ \vdots & \vdots & \cdots & \vdots \\ x_1^N & x_2^N & \cdots & x_d^N \end{matrix}, \quad (12)$$

• Шаг 3: Интенсификация популяций НДТ. Теперь все летучие мыши x летают и меняют свои положение с учетом скорости v, определяемой частотой f , которая генерируется случайным образом, как показано в уравнениях (13) и (14). Соответственно, позиции бит обновляются с помощью Уравнение (15).

$$fi = fmin + (fmin - fmax) \times R, \quad (13)$$

$$v_i^j = v_i^{old} + (x_i^{old} - x_i^{Г\text{лучшее}}) \times фи, \quad (14)$$

$$x_{iii}^j = x^{old} + в^j \quad (15)$$

Примечательно, что летучие мыши обновляют свои местоположения, чтобы быть ближе к лучшим в мире. (х Гбест). Таким образом, новое положение микробатов усиливает положение в направлении в сторону $x^{Гбест}$.

• Шаг 4: Диверсификация популяции НДТ.
На этом этапе позиции микробатов обновляются на основе случайных параметров. попытаться найти лучшие глобальные решения. Новое решение выбирается из БМ на основе по методу отбора и присвоено x лучшему. Впоследствии текущее решение обновлено с использованием x best следующим образом:

$$x_i^j = x_{\text{лучший}} + \varepsilon \times A^{'}_{ж.} \quad (16)$$

где $A^{'}$ это средняя громкость. Поскольку Шаги 3 и 4 являются противоречивыми, НДТ алгоритм выбирает один из них для выполнения. Этот выбор основан на следующее уравнение:

$$x^{j} = \begin{cases} x^{\text{лучший}} + \varepsilon \times A^{'}_{ж.} & \text{если } rj < R \\ xj + v^{j} & \text{в противном случае} \end{cases}, \quad (17)$$

Machine Translated by Google

• Шаг 5: Обновление BM. Положение микробат в BM будет обновлено на этом этапе Gbest , если новое местоположение будет более подходящим, чем старое, и R Aj . Более того, X обновляется, если новая БМ содержит решение с лучшим значением пригодности. Впоследствии значения rj и Aj будут обновлены в соответствии с уравнениями (18) и (19).

$$\text{гдж} = \text{рдж}^{0}(1 \quad e^{(\quad \gamma \times itr)}, \tag{18}$$

$$Aj = \alpha \times Aj, \tag{19}$$

где itr обозначает текущий номер итерации.

• Шаг 6: критерий остановки. Шаги 3, 4 и 5 будут повторяться до тех пор, пока алгоритм не достигнет критерий остановки.

4.5. Этап V: Классификация B

этой работе используется классификатор KNN. В этом предложенном методе набор данных разделен на 60% обучения, 20% проверки и 20% тестирования.

5. Эксперименты и результаты.

В этом разделе подробно объясняется эффективность предлагаемого метода (НДТ) каналов ЭЭГ для реабилитации пациентов, перенесших инсульт. Поскольку предлагаемый подход недетерминирован, вычисляется средний уровень точности за 20 прогонов, чтобы избежать систематических результатов. В экспериментах использовался ПК Lenovo Ideapad 310, Китай. Процессор Intel Core i7® 2,59 ГГц, 16 ГБ оперативной памяти и официальная Windows 10. Производительность предлагаемого метода оценивается по двум показателям: точности и количеству выбранных каналов ЭЭГ, с учетом стабильности алгоритмов. путем вычисления лучшей, худшей и средней точности. Кроме того, применяется статистический тест, основанный на точности классификации набора данных моторной/визуальной ЭЭГ с использованием суммы рангов метаэвристических алгоритмов. Предлагаемый метод тестируется с использованием трех типов характеристик ЭЭГ: характеристик временной области, характеристик частотной области и характеристик энтропийной области. Кроме того, результаты производительности предлагаемого метода оцениваются с использованием статистического теста, чтобы показать значимость предлагаемого метода по сравнению с другими метаэвристическими алгоритмами.

В таблице 1 показаны параметры, используемые для выбранных метаэвристических алгоритмов, используемых в данной работе.

5.1. Результаты во временной области

В таблице 2 представлены результаты семи характеристик во временной области: фрактальная размерность Хигучи (HFD); параметры Хьорта: активность Хьорта (HjAc), сложность Хьорта (Hj-Comp) и подвижность Хьорта (HjMo); Херст (Гур); эксцесс (Курт); и асимметрия (Skw).
Точность и количество выбранных каналов ЭЭГ — это два измерения для оценки семи функций, выбранных для набора данных ЭЭГ двигательной/визуализации. Метрики оценки подразделяются на три категории: лучшая, худшая, средняя точность и отсутствие. каналов, выбранных для оценки устойчивости алгоритмов.

Согласно таблице 2, результаты производительности метаэвристических алгоритмов PSO, GWO, FFA, CS и BAT с несколькими функциями достигли разной и замечательной точности в отношении лучшей, худшей и средней точности и количества выбранных каналов. Что касается функции HFD, алгоритм BAT достиг более высокой точности — 99, 97,5 и 92,8 для лучшей, средней и худшей точности и выбора канала соответственно. Для функции HjAc алгоритм BAT достиг высокой точности — 97,7 по точности и нет. каналов, выбранных в лучшем случае, в то время как PSO достиг ценного соотношения 94,4 и 85,3 по точности и количеству каналов, выбранных в среднем и худшем случаях, соответственно. Для функции HjComp алгоритм PSO достиг точности 88,6 и 84,7 в среднем и наихудшем случае, тогда как алгоритм BAT достиг наивысшей точности 93,8 в лучшем случае. Для функции HjMo BAT получила 97,8, 95,4 и 88,6 баллов за лучший, средний и худший среди других метаэвристических алгоритмов. Для функции «Hur» PSO получила высокую точность — 72,8, 65,2 и 56,6 для лучшего, среднего и худшего соответственно. Для функции Курта BAT

Machine Translated by Google

Алгоритм получил ценную точность 73,2, 68,8 и 63,1 для лучшего, среднего и худшего
точность и количество выбранных каналов соответственно.

Более того, для Skw BAT достигла высокой точности — 79,2 и 72,5 для лучшего и среднего значения.
соответственно, по сравнению с ПСО, который получил точность 58 в худшем случае
точности. На рисунке 4 показаны результаты анализа характеристик во временной области для сходимости.
скорость и распределение каналов, сравнение алгоритмов BAT и PSO, которые достигли
самая высокая точность по сравнению с другими метаэвристическими алгоритмами, т. е. GWO, FFA и CS.
На рисунке 4a скорость сходимости показывает каналы и количество итераций для
алгоритмы BAT и PSO. Кроме того, на рисунке 4б представлены эффективные и эмоциональные
каналы при распределении каналов по алгоритмам BAT и PSO.

Таблица 2. Производительность метаэвристических алгоритмов с несколькими подходами к выделению признаков для
Набор данных моторной/визуальной ЭЭГ.

Функция/алгоритм			PSO	GWO	FFA	CS	НДТ
HFD	Лучший	Точность	97,9	90,7	66,7	54,9	99
		Количество каналов	8	3	1	1	6
	Худший	Точность	81,5	67,4	45,1	43,2	92,8
		Количество каналов	12	6	3	2	13
	Иметь в виду	Точность	94,4	76,2	56,7	50,6	97,5
		Количество каналов	9	4	2	2	10
HjAc	Лучший	Точность	97,1	88,4	68,8	55,1	97,7
		Количество каналов	5	3	1	1	4
	Худший	Точность	85,3	70,4	41,7	42,1	77,9
		Количество каналов	9	5	3	2	14
	Иметь в виду	Точность	94,4	82,1	60,7	48,7	90,8
		Количество каналов	8	4	3	2	8
HjComp	Лучший	Точность	91,2	78,9	58,8	48,7	93,8
		Количество каналов	7	3	2	1	5
	Худший	Точность	84,7	58,1	47,4	40,9	74,7
		Количество каналов	15	6	3	2	13
	Иметь в виду	Точность	88,6	69,9	53,9	44,2	88,2
		Количество каналов	9	4	3	1	9
ХджМо	Лучший	Точность	96,5	89,3	67,9	53,5	97,9
		Количество каналов	6	2	2	1	5
	Худший	Точность	86,4	51,8	53,9	43,8	88,6
		Количество каналов	11	6	3	2	12
	Иметь в виду	Точность	92,3	80,6	61,2	47,9	95,4
		Количество каналов	9	4	3	1	9
Гур	Лучший	Точность	72,8	57,9	47,2	44,1	70,1
		Количество каналов	4	2	1	1	5
	Худший	Точность	56,6	44,1	41,5	41,3	53,9
		Количество каналов	10	6	3	2	11
	Иметь в виду	Точность	65,2	51,7	44,7	42,6	64,9
		Количество каналов	8	4	3	2	9

Machine Translated by Google

Таблица 2. Продолжение.

Функция/алгоритм			ПСО	ГВО	ФФА	CS	НДТ
Курт	Лучший	Точность	71,2	61	48,4	43,4	73,2
		Количество каналов	7	3	2	1	7
	Худший	Точность	56,5	48,2	43,3	40,3	63,1
		Количество каналов	12	6	3	2	12
	Иметь в виду	Точность	65,6	54,9	45,4	42	68,8
		Количество каналов	9	5	3	2	10
Скв	Лучший	Точность	78,7	63,7	50	44,4	79,2
		Количество каналов	4	1	2	1	3
	Худший	Точность	58	41	43,2	40,4	48,2
		Количество каналов	11	6	3	2	13
	Иметь в виду	Точность	69,6	53,3	45,9	42,3	72,5
		Количество каналов	7	4	2	2	9

5.2. Результаты энтропийной области

В таблице 3 показаны результаты точности производительности с использованием семи энтропийных функций: а именно: ConFuzEn, FuzEn, impe, MFEmu, RCMFEmu, SampEn и Цен. На основании точность и количество выбранных каналов для лучшего, худшего и среднего случаев, эти энтропии функции используются для сравнения пяти метаэвристических алгоритмов: PSO, GWO, FFA, CS и БАТ соответственно. По сравнению с другими семью энтропийными признаками лучшая классификация производительность достигла высочайшей точности при использовании ConFuzEn с алгоритмом PSO по сравнению с алгоритмами GWO, FFA, CS и BAT. Другие функции энтропии производятся выше Результаты по точности с алгоритмом BAT выше, чем с алгоритмами PSO, GWO, FFA и CS. Алгоритм BAT зарегистрировал точность 95,5, 95,6, 88,1, 87,2, 91,4 и 84,6 с использованием FuzEn. impe, MFEmu, RCMFEmu, СампЭн и ЦЭн соответственно. Кроме того, на рисунке 5 представлены результаты оценки производительности алгоритмов BAT и PSO относительно Скорость конвергенции и распределение каналов.

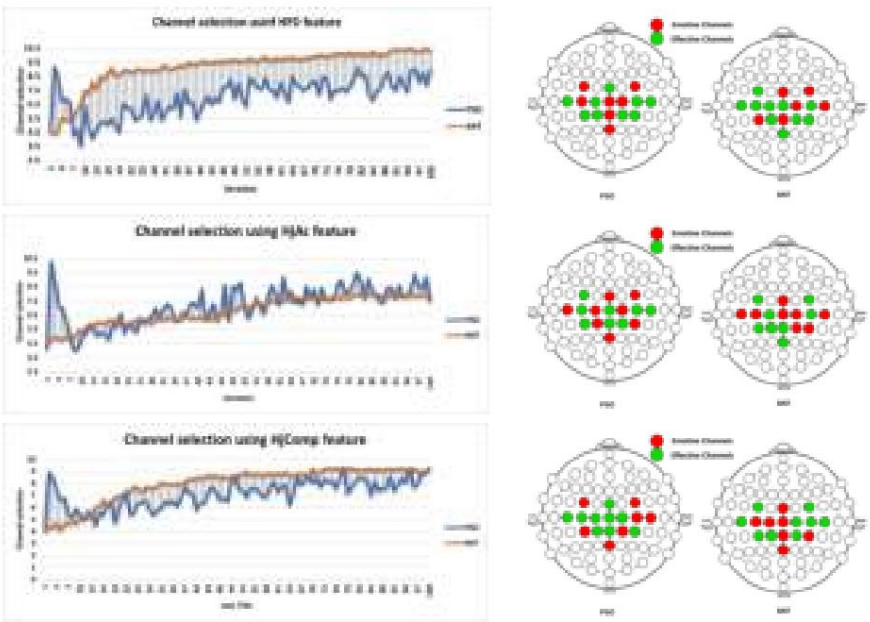


Рисунок 5. Продолжение.

Machine Translated by Google

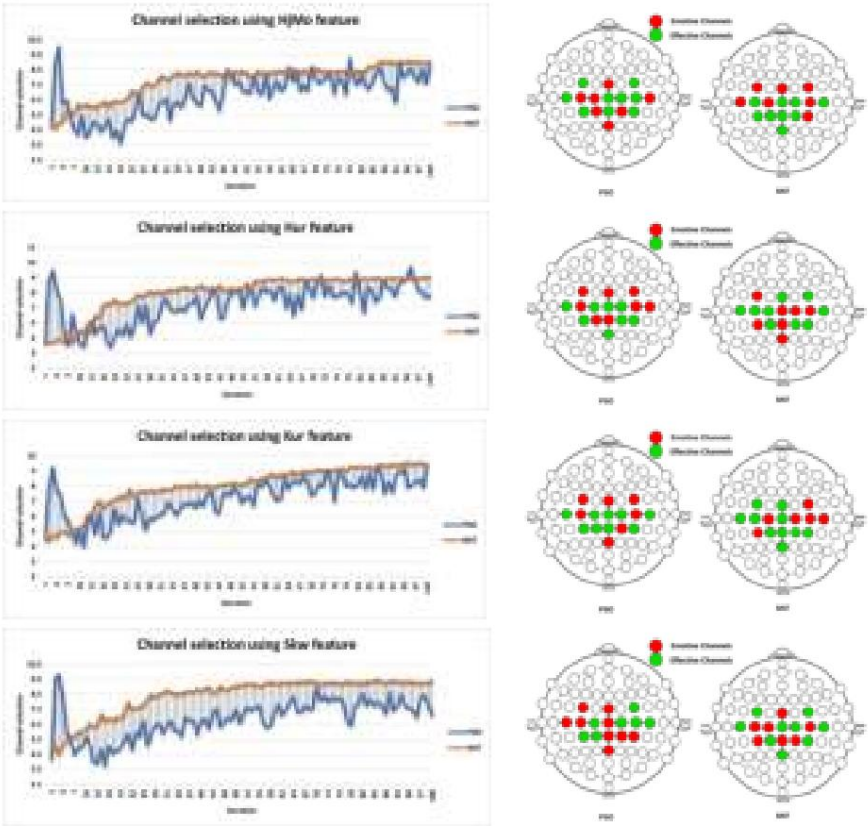


Рисунок 5. Скорость конвергенции и распределение каналов.

Таблица 3. Производительность метаэвристических алгоритмов с несколькими подходами к выделению признаков для Набор данных моторной/визуальной ЭЭГ.

Функция/алгоритм			ПСО	ГВО	ФФА	CS	НДТ
ConvFuzEn	Лучший	Точность	96,6	82,7	61,3	52	96,5
		Количество каналов	3	3	1	1	7
	Худший	Точность	76,4	64,7	45,5	42,5	87,1
		Количество каналов	11	6	3	2	14
	Иметь в виду	Точность	90,2	73,6	54,9	45,8	93,2
		Количество каналов	8	5	2	1	11
ФузЭн	Лучший	Точность	93,6	81,6	60,6	49,1	95,5
		Количество каналов	2	4	2	1	8
	Худший	Точность	82,2	68,7	50,5	41,8	88,9
		Количество каналов	10	6	3	2	12
	Иметь в виду	Точность	88,6	74,4	55,7	45,5	92,1
		Количество каналов	7	5	3	1	10
импен	Лучший	Точность	94,7	78,6	59,3	49,8	95,6
		Количество каналов	5	3	2	1	7
	Худший	Точность	75,8	56,1	47,9	44,3	84,9
		Количество каналов	10	6	3	2	12
	Иметь в виду	Точность	85,3	69,8	52,3	46,6	90,8
		Количество каналов	8	5	2	2	10

Machine Translated by Google

Таблица 3. Продолжение.

Функция/алгоритм			PSO	GWO	FFA	CS	НДТ
МФЕМу	Лучший	Точность	85,2	73,6	54,3	46,8	88,1
		Количество каналов	7	3	2	1	7
	Худший	Точность	70,8	55,9	45,1	40,5	75,5
		Количество каналов	12	6	3	2	12
	Иметь в виду	Точность	80,7	66,1	49,4	44,8	84
		Количество каналов	9	5	2	2	9
RCMFEmu	Лучший	Точность	84,3	67,7	51,9	46,6	87,2
		Количество каналов	4	2	2	1	8
	Худший	Точность	63,7	47	44,9	41,2	75,3
		Количество каналов	12	6	3	2	13
	Иметь в виду	Точность	77,3	56,5	47,7	43	80,3
		Количество каналов	8	4	2	1	10
СампЭн	Лучший	Точность	87,9	71,5	52,9	51,9	91,4
		Количество каналов	6	3	1	1	7
	Худший	Точность	75,4	53,2	42,6	40,4	75,9
		Количество каналов	10	6	3	3	14
	Иметь в виду	Точность	81,4	63,7	48,2	44,3	83,6
		Количество каналов	8	5	2	2	10
ЦЭН	Лучший	Точность	83,9	72,5	52,3	46,6	84,6
		Количество каналов	7	3	1	1	4
	Худший	Точность	65,9	54,4	41,2	40,2	61,3
		Количество каналов	12	7	3	2	13
	Иметь в виду	Точность	78,7	64,5	47,2	42,8	79,8
		Количество каналов	10	5	2	1	10

5.3. Результаты в частотной области

На основании Таблицы 4 характеристики частотной области средних F и medF получили высокие значения. точности 96,7 и 83,4 для алгоритмов PSO и BAT соответственно. В лучшем, худшем, и средних случаях, среднее значение F приобрело значительную точность по сравнению с функцией medF на общие результаты по точности для алгоритмов PSO и BAT. Наилучшая точность medF была получено при 96,7 с PSO, в то время как наихудшая и средняя достигнутая точность составила 90,3 и 93,8. с алгоритмом BAT соответственно. Более того, алгоритм BAT показал высокую точность. результаты классификации 83,4 и 78,4 относительно лучшей и худшей точности с особенность medF соответственно, тогда как PSO получила точность 69,5. Скорость сходимости и распределение каналов показано на рисунке 6.



Рисунок 6. Продолжение.

Machine Translated by Google

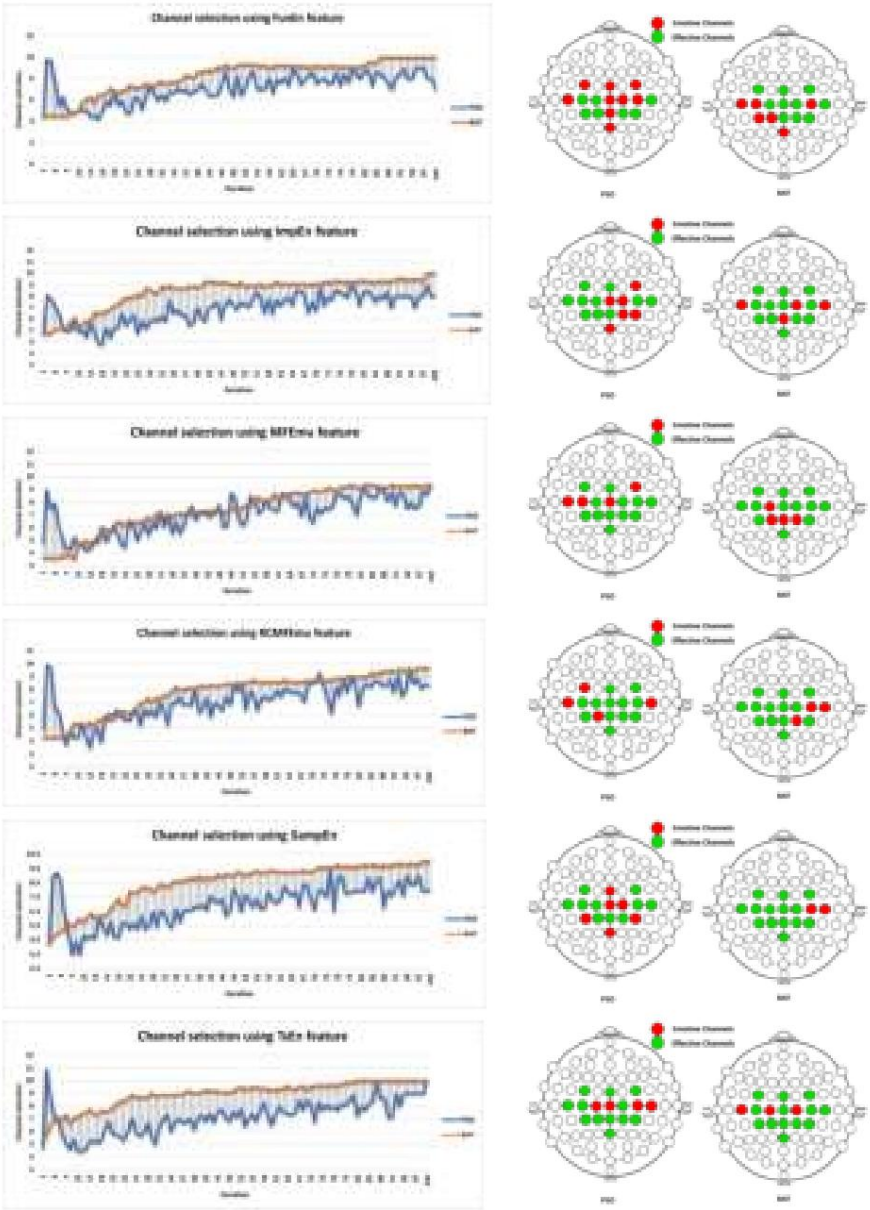


Рисунок 6. Скорость конвергенции и распределение каналов.

Как уже упоминалось, в таблице 4 показана производительность предлагаемого метода (BAT) по сравнению с алгоритмами PSO, GWO, FFA и CS с использованием функций частотной области. Алгоритм BAT показал лучшие результаты, чем все другие алгоритмы, за исключением некоторых случаев , когда алгоритм PSO немного превосходил его, например, в среднем F, где лучшая точность составила 96,7 для PSO и 96,3 для BAT. Однако в целом алгоритм BAT показал лучшие результаты с показателями точности, равными 93,8, 91,2, 76,4, 54 и 50 для алгоритмов BAT, PSO, GWO, FFA и CS соответственно. Скорость сходимости и распределение каналов показаны на рисунке 7.

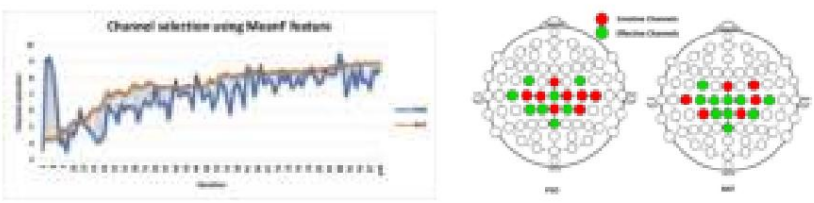


Рисунок 7. Продолжение.

Machine Translated by Google



Рисунок 7. Скорость сходимости и распределение каналов.

Таблица 4. Производительность алгоритма BAT по сравнению с несколькими метаэвристическими алгоритмами, использующими особенности частотной области.

Функция/алгоритм			ПСО	ГВО	ФФА	CS	НДТ
среднее F	Лучший	Точность	96,7	87,3	64,1	52,5	96,3
		Количество каналов	5	3	2	1	7
	Худший	Точность	65,4	66	50,2	41,9	90,3
		Количество каналов	10	6	3	2	12
	Иметь в виду	Точность	91,2	76,4	54	50	93,8
		Количество каналов	8	4	2	2	10
МедФ	Лучший	Точность	82,7	69,2	53,6	46,6	83,4
		Количество каналов	6	3	2	1	6
	Худший	Точность	69,5	55,5	41	20	68,7
		Количество каналов	10	6	3	2	14
	Иметь в виду	Точность	76,3	60,6	46,1	36,5	78,4
		Количество каналов	9	4	2	2	10

5.4. Результаты статистического анализа

Прежде чем представить наши выводы на предмет точности и нет. каналов, выбранных для улучшая эффективность классификации анализа временных, энтропийных и частотных характеристик ударных сигналов ЭЭГ, мы хотели бы изучить влияние точности классификации в наборе данных моторной/визуальной ЭЭГ путем принятия суммы рангов метаэвристики алгоритмы. Таким образом, функция, которая улучшила точность ранжирования выбранных каналов можно идентифицировать.

Как показано в таблице 5, мы выполнили суммирование рангов PSO, GWO, FFA, CS, и метаэвристические алгоритмы BAT в соответствии с отдельной областью функций. В во временной области мы протестировали семь функций. Алгоритм BAT значительно улучшил ранг точности, особенно в HFD и Hjac, для ранга точности 99 и 97,7 соответственно. Более того, другие функции во временной области, такие как HjComp, HjMo, Kurt и Skw, показало существенное влияние на результаты точности классификации относительно суммы ранга алгоритма BAT, за исключением функции Hur, которая оказала большое влияние на алгоритм PSO. Однако на этапе статистического тестирования алгоритм BAT имел низкие показатели. результаты суммирования рангов среди других метаэвристических алгоритмов.

С другой стороны, сумма рангов алгоритма BAT показала значимое значение. влияние на результаты точности классификации по семи признакам энтропии домена, за исключением функции ConvFuzEn, которая оказала заметное влияние на PSO. алгоритм с разницей в 0,1 по сравнению с BAT относительно результатов суммы рангов в статистическом анализе. Однако алгоритм BAT имел низкие результаты суммирования рангов. на этапе статистического тестирования среди других метаэвристических алгоритмов.

Однако точность классификации обусловлена особенностями частотной области: «meanF» продемонстрировал заметное улучшение точности алгоритма PSO относительно до ранга точности 96,7 по сравнению с функцией medF, точность которой составляла 83,4.

Machine Translated by Google

ранжироваться по алгоритму BAT. Более того, алгоритмы BAT и PSO имели низкие результаты суммирования рангов среди других метаэвристических алгоритмов.

Таблица 5. Сумма рангов метаэвристических алгоритмов, основанных на точности классификации на наборе данных двигательной/визуальной ЭЭГ.

Функция/алгоритм			PSO	GWO	FFA	CS	НДТ
ТДФ ХйМо	HFD	Точность	97,9	90,7	66,7	54,9	99
		Классифицировать	2	3	4	5	1
	HjAc	Точность	97,1	88,4	68,8	55,1	97,7
		Классифицировать	2	3	4	5	1
	HjComp	Точность	91,2	78,9	58,8	48,7	93,8
		Классифицировать	2	3	4	5	1
		Точность	96,5	89,3	67,9	53,5	97,9
		Классифицировать	2	3	4	5	1
	Гур	Точность	72,8	57,9	47,2	44,1	70,1
		Классифицировать	1	3	4	5	2
	Кур	Точность	71,2	61	48,4	43,4	73,2
		Классифицировать	2	3	4	5	1
	Скв	Точность	78,7	63,7	50	44,4	79,2
		Классифицировать	2	3	4	5	1
	Суммирование рангов		13	21	28	35	8
ФДФ МедФ	среднее F	Точность	96,7	87,3	64,1	52,5	96,3
		Классифицировать	1	3	4	5	2
		Точность	82,7	69,2	53,6	46,6	83,4
		Классифицировать	2	3	4	5	1
	Суммирование рангов		3	6	8	10	3
ЭДФ МФЕМу	ConvFuzEn	Точность	96,6	82,7	61,3	52	96,5
		Классифицировать	1	3	4	5	2
	ФузЭн	Точность	93,6	81,6	60,6	49,1	95,5
		Классифицировать	2	3	4	5	1
	импен	Точность	94,7	78,6	59,3	49,8	95,6
		Классифицировать	2	3	4	5	1
		Точность	85,2	73,6	54,3	46,8	88,1
		Классифицировать	2	3	4	5	1
	РЦМФЕМу	Точность	84,3	67,7	51,9	46,6	87,2
		Классифицировать	2	3	4	5	1
	СампЭн	Точность	87,9	71,5	52,9	51,9	91,4
		Классифицировать	2	3	4	5	1
	ЦЭН	Точность	83,9	72,5	52,3	46,6	84,6
		Классифицировать	2	3	4	5	1
	Суммирование рангов		13	21	28	35	8

На рисунке 8 показаны результаты скорости метаэвристических алгоритмов в секундах.

Machine Translated by Google

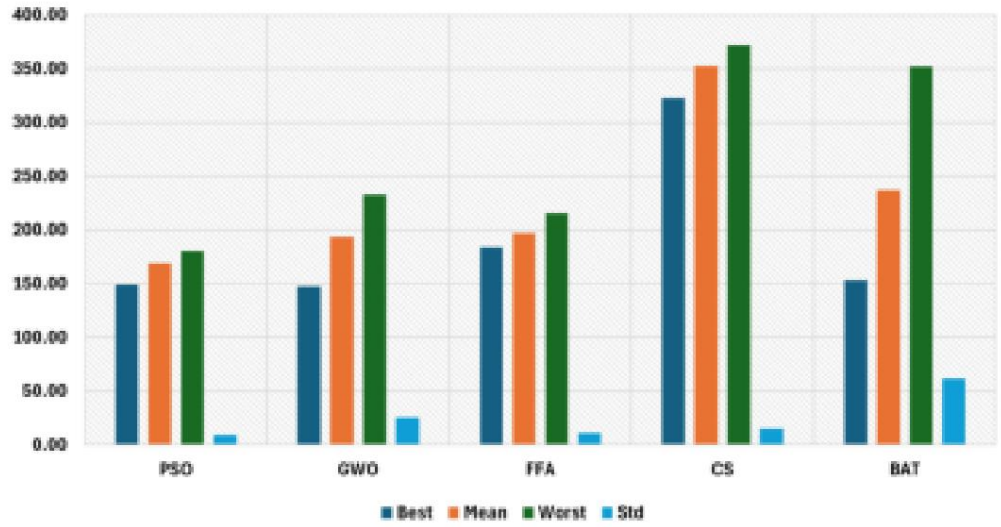


Рисунок 8. Скорость метаэвристических алгоритмов в секундах.

5.5. Обсуждение результатов

Целью данного исследования является выбор наиболее подходящих каналов ЭЭГ, которые обеспечивают эффективную степень точности реабилитации пациентов, перенесших инсульт, с использованием метода оптимизации на основе алгоритма двоичной летучей мыши для оптимизации характеристик TEF. Для дальнейшего улучшения эффективности классификации сигналов ЭЭГ инсульта в качестве основы для задачи классификации использовались три типа характеристик (а именно, характеристики времени, энтропии и частотной области). Для оценки эффективности предлагаемого метода были применены два измерения: точность и нет. выбранных каналов ЭЭГ. Кроме того, мы провели статистический тест, используя сумму рангов набора данных двигательной/визуальной ЭЭГ, чтобы изучить влияние точности классификации и значимости предлагаемого метода по сравнению с современными метаэвристическими алгоритмами, такими как PSO, GWO, FFA и CS. Во-первых, точность и нет. выбранных каналов ЭЭГ оценивались путем вычисления наилучшего, наихудшего и среднего значений точности для определения стабильности реализованного алгоритма. Во временной области было протестировано семь функций, и функция HFD оказала наиболее существенное влияние на улучшение задачи классификации в оптимизаторе BAT среди других алгоритмов оптимизации для лучшего, среднего и худшего случаев точности, а также для нет. каналов выбрано соответственно на основе классификатора kNN. Кроме того, результаты BAT превзошли тестовые значения для случая лучшей точности по сравнению с PSO, который показал самые высокие результаты по сравнению с алгоритмами GWO, FFA и CS в функциях HjAc и HjComp. Более того, результаты оптимизатора BAT показали высокие значения лучшей, средней и худшей точности, а также нет. каналов, выбранных соответственно с помощью функций HFD, HjMo и Kurt соответственно, в то время как PSO оказал наиболее существенное влияние на улучшение задачи классификации по сравнению с BAT с функцией Hur, основанной на лучшей, средней и худшей то Для лучших значений оптимизатор BAT превзошел по ценным и самым высоким значениям среди PSO и других метаэвристических алгоритмов с шестью функциями HFD, HjAc, HjComp, HjMo, Kurt и Skw соответственно (см. Таблицу 2). Кроме того, по результатам точности производительности с использованием семи энтропийных признаков на основе классификатора KNN, лучшая производительность классификации была достигнута с наибольшей точностью при использовании ConFuzEn с алгоритмом PSO по сравнению с алгоритмами Другие функции энтропии дали более высокие результаты точности с помощью алгоритма BAT, чем алгоритмы PSO, GWO, FFA и CS, в соответствии с лучшими, средними и худшими значениями точности, а также нет. выбранных каналов соответственно. Алгоритм BAT зарегистрировал значения точности в наилучшем случае 95,5, 95,6, 88,1, 87,2, 91,4 и 84,6 с использованием функций FuzEn, impe, MFEmu, RCMFEmu, SampEn и TsEn соответственно, в то время как PSO превзошел BAT по значениям точности для Функция ConvFuzEn для классификатора KNN (см. таблицу 3).

Machine Translated by Google

Алгоритмы BAT и PSO получили высокую точность 96,7 и 83,4 соответственно по сравнению с другими алгоритмами, описанными в литературе, для функций частотной областиmeanF и medF. Алгоритм BAT показал результаты классификации с высокой точностью 83,4 и 78,4 относительно лучшей и худшей точности с функцией medF соответственно, тогда как PSO получил точность 69,5; кроме того, алгоритмы PSO и BAT значительно повысили точность по сравнению с функцией medF по общим результатам точности в случаях лучшей, худшей и средней точности. Основываясь на результатах классификации на основе признаков трех различных типов доменов, алгоритмы BAT и PSO оказали значительное влияние на производительность классификации по сравнению с результатами алгоритмов оптимизации GWO, FFA и CS в отношении точности и отсутствия ошибок. . выбранных каналов соответственно. Следовательно, они демонстрируют значительное улучшение средней точности по сравнению с результатом классификации на основе отдельных областей признаков, как видно на рисунке 5, который объясняет сравнение скорости сходимости и распределения каналов для алгоритмов BAT и PSO.

Во-вторых, мы исследовали влияние точности классификации на набор данных моторной/визуальной ЭЭГ, используя принятую сумму рангов метаэвристических алгоритмов. Таким образом, можно определить признак, который улучшил точность ранжирования выбранных каналов. Алгоритм BAT значительно улучшил ранг точности, особенно в функциях HFD и Hjac, с рангом точности 99 и 97,7 соответственно по сравнению с алгоритмами PSO, GWO, FFA и CS согласно результатам суммы рангов с уважение к временной области.
Кроме того, такие функции, как HjComp, HjMo, Kurt и Skw, оказали влияние на результаты точности классификации относительно суммы рангов алгоритма BAT, за исключением функции Hur, которая оказала сильное влияние на алгоритм PSO. Однако на этапе статистического тестирования алгоритм BAT показал низкие результаты суммирования рангов среди других метаэвристических алгоритмов. Кроме того, алгоритм BAT продемонстрировал значимое влияние на результаты точности классификации для суммы рангов относительно семи признаков энтропийной области, за исключением признака ConvFuzEn, который оказал заметное влияние на алгоритм PSO с разницей в 0,1 по сравнению с НДТ в отношении суммы ранговых результатов статистического анализа. Однако на этапе статистического тестирования алгоритм BAT показал низкие результаты суммирования рангов среди других метаэвристических алгоритмов.
Более того, алгоритмы BAT и PSO показали низкие результаты суммирования рангов среди других метаэвристических алгоритмов в функциях частотной области. Функция «meanF» продемонстрировала заметное улучшение точности при использовании алгоритма PSO по сравнению с рангом точности 96,7 по сравнению с функцией medF, которая имела ранг точности 83,4 при использовании алгоритма BAT.

6. Выводы и будущая работа

В этой статье предлагается новый метод выбора каналов ЭЭГ, основанный на алгоритме оптимизации, называемом алгоритмом на основе BAT. Основная цель предложенного алгоритма — выбор наиболее релевантных каналов ЭЭГ, которые могут обеспечить более высокую точность реабилитации пациентов с инсультом.

Предлагаемый метод был протестирован на стандартном наборе данных ЭЭГ, собранном у восьми пациентов с постинсультным гемипарезом верхних конечностей. Данные ЭЭГ регистрировались с помощью ЭЭГ-шапочек с 16 активными электродами от g.Nautilus PRO, g.tec Medical Engineering GmbH, Австрия.

Предлагаемый метод использовал несколько методов извлечения признаков ЭЭГ, таких как временная область, частотная область и энтропийная область. Кроме того, предложенный метод (BAT) сравнивался с четырьмя метаэвристическими алгоритмами, а именно: оптимизацией роя частиц (PSO), оптимизатором серого волка (GWO), поиском кукушки (CS) и алгоритмом светлячка (FFA). Эффективность предложенного метода оценивалась по показателю точности и количеству выбранных каналов. Предложенный метод показал наилучшие результаты по сравнению со всеми методами извлечения признаков и показал значительное улучшение при использовании стандартного теста статистического анализа, называемого тестом суммирования рангов.

Machine Translated by Google

Наконец, предложенный метод позволил сократить количество каналов ЭЭГ менее чем вдвое при сохранении точности.

Что касается будущих работ, предлагаемый метод страдает некоторыми проблемами; например, предлагаемый метод не способен генерировать новые решения, способные повысить точность (т. е. застрявшие в локальных минимумах). Для этого предлагаемый метод необходимо будет улучшить, чтобы преодолеть эту проблему, изменив механизм алгоритма BAT или гибридизировав алгоритм BAT с другим метаэвристическим алгоритмом, чтобы иметь возможность найти оптимальное решение. Что касается второй проблемы, текущая версия алгоритма BAT ориентирована на поиск оптимального решения, обеспечивающего максимальную точность и не учитывающего количество каналов ЭЭГ в целевой функции. Поэтому в дальнейшей работе мы будем работать над предлагаемой многокритериальной версией БАТ, используя высочайшую точность и наименьшее количество каналов ЭЭГ одновременно.

Вклад авторов: концептуализация, MAA-B., ZAAA и NKA-Q.; методология, MAA-Б., ЗААА и НКА-Q.; программное обеспечение, MAA-Б. и ЗААА; валидация, MAA-Б. и CHM; расследование, MAA-Б. и ЗААА; ресурсы, MAA-Б., ЗААА и КГ; письменная форма — первоначальный проект, MAA-B., ZAAA, NKA-Q., SNM, NSA и CG; Визуализация, MAA-B., ZAAA, SNM, NSA и CG. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Работа выполнена при поддержке деканата исследований и последипломного образования (DRG) Университета Аджмана, Аджман, ОАЭ (грант № 2022-IRG-ENIT-7).

Заявление о доступности данных: данные содержатся в статье.

Конфликт интересов: Автор Кристоф Гугер работал в компании g.tec Medical Engineering GmbH. Остальные авторы заявляют, что исследование проводилось при отсутствии каких-либо коммерческих или финансовых отношений, которые могли бы быть истолкованы как потенциальный конфликт интересов.

Ссылки

1. Ли, Ф.; Фан, Ю.; Чжан, Х.; Ван, К.; Ху, Ф.; Цзя, В.; Хуэй, Х. Многофункциональный метод слияния на основе сигнала ЭЭГ и его применение в классификации инсульта. Дж. Мед. Сист. 2020, 44, 39. [CrossRef]
2. Хайрунизам, В.; Йен, CW; Муругаппан, М.; Юно, АК; Разлан, З.М.; Шахриман, А.; Мустафа, WAW; Ибрагим, З.; Нурхафиза, С. Экспериментальная основа оценки эмоций пациентов, перенесших инсульт, с помощью электроэнцефалограммы (ЭЭГ). Дж. Физ. Конф. Сер. 2020, 1529, 052072. [CrossRef]
3. Карино-Эскобар, Род-Айленд; Каррильо-Мора, П.; Вальдес-Кристерна, Р.; Родригес-Барраган, Массачусетс; Эрнандес-Аренас, К.; Кинзаньос-Фреснедо, Дж.; Галисия-Альварадо, Массачусетс; Кантильо-Негрете, Дж. Продольный анализ ритмов мозга пациентов с инсультом во время вмешательства с использованием интерфейса мозг-компьютер. Нейральный пласт. 2019, 2019. [CrossRef] [ПабМед]
4. Грефкес, К.; Финк, Г.Р. Подходы, основанные на связности, при инсульте и восстановлении функции. Ланцет Нейрол. 2014, 13, 206–216. [Перекрестная ссылка]
5. Хачинский В.; Доннан, Джорджия; Горелик, ПБ; Хаке, В.; Крамер, Южная Каролина; Касте, М.; Фишер, М.; Брайнин, М.; Бьюкен, АМ; Ло, ЭХ; и др. Инсульт: Работа над приоритетной мировой повесткой дня. Инсульт 2010, 41, 1084–1099. [Перекрестная ссылка] [ПабМед]
6. Гао, В.; Кюи, З.; Ю, Ю.; Мао, Дж.; Сюй, Дж.; Джи, Л.; Кан, Х.; Шен, Х.; Ли, Х.; Чжу, С.; и др. Применение системы интерфейса мозг-компьютер со зрительной и двигательной обратной связью при функциональной реабилитации конечностей и мозга после инсульта: клинический случай. Наука о мозге. 2022, 12, 1083. [CrossRef]
7. Джин, Дж.; Цюй, Т.; Сюй, Р.; Ван, Х.; Цихоцкий, А. Классификация ЭЭГ по двигательным образам, основанная на римановой разреженной оптимизации и слиянии Демпстера-Шафера многочастотных паттернов. IEEE Транс. Нейронная система. Реабилитация. англ. 2022, 31, 58–67. [Перекрестная ссылка]
8. Аль-Каззаз, Северная Каролина; Альяссери, ЗАА; Абдулкарим, К.Х.; Али, Н.С.; Аль-Микани, Миннесота; Гугер, К. Объединение функций ЭЭГ для воображения движений: новая надежная основа реабилитации пациентов, перенесших инсульт. Вычислить. Биол. Мед. 2021, 137, 104799. [CrossRef]
9. Аль-Тимеми, Ах; Бугманн, Г.; Аутрам, Н.; Эскудеро, Дж. Снижение ошибок классификации миоэлектрического контроля движений рук с помощью независимого анализа компонентов. В материалах 5-й Международной конференции по информационным технологиям, ICIT, Бали, Индонезия, 26–28 октября 2011 г.; Том 2011.
10. Аль-Каззаз, Северная Каролина; Али, ШБМ; Ахмад, С.А.; Эскудеро, Дж. Оптимальный выбор канала ЭЭГ для выявления сосудистой деменции с использованием улучшенного алгоритма двоичного гравитационного поиска. В материалах Международной конференции по инновациям в биомедицинской инженерии и науках о жизни, Пенанг, Малайзия, 10–13 декабря 2017 г.; стр. 125–130.
11. Аль-Каззаз, Северная Каролина; Сабир, МК; Али, ШБМ; Ахмад, С.А.; Грэммер, К. Многоканальная оптимизация с использованием гибридных спектрально-энтропийных маркеров для улучшения гендерной идентификации ЭЭГ, основанных на эмоциях. IEEE Access 2021, 9, 107059–107078. [Перекрестная ссылка]
12. Веласко И.; Сиполс, А.; Де Блас, CS; Пастор Л.; Байона, С. Классификация сигналов ЭЭГ по двигательным образам с использованием многомерного подхода временных рядов. Биомед. англ. OnLine 2023, 22, 29. [CrossRef]

Machine Translated by Google

13. Джузизаде, М. Особенности сети и вариабельности сигналов, оцениваемые по ЭЭГ, у мужчин и женщин во время зрительно-пространственной задачи. доктор философии Диссертация, Университет Оттавы/Университет Оттавы, Оттава, Онтарио, Канада, 2024.
14. Аль-Каззаз, Северная Каролина; Сабир, МК; Али, С.; Ахмад, С.А.; Грэммер, К. Эффективные каналы ЭЭГ для идентификации эмоций в областях мозга с использованием алгоритма дифференциальной эволюции. В материалах 41-й ежегодной международной конференции Общества инженерии в медицине и биологии IEEE (EMBC) 2019 г., Берлин, Германия, 23–27 июля 2019 г.; стр. 4703–4706.
15. Дорнхеге, Г.; Миллан, младший аргентинец; Хинтербергер, Т.; МакФарланд, диджей; Мюллер, К.Р. На пути к интерфейсу «мозг-компьютер»; МТИ Пресс: Кембридж, Массачусетс, США, 2007 г.; Том 63.
16. Кингфай, К.; Мошфеги, Ю. О выборе каналов для классификации умственной нагрузки на основе ЭЭГ. В материалах Международной конференции по машинному обучению, оптимизации и науке о данных, Грасмер, Великобритания, 22–26 сентября 2023 г.; стр. 403–417.
17. Лю, Т.; Ву, Ю.; Да.; Цао, Л.; Цао, Ю. Двухэтапный разреженный многокритериальный эволюционный алгоритм для оптимизации выбора канала в БКИ. Передний. Хм. Неврологии. 2024, 18, 1400077. [CrossRef] [ПабМед]
18. Сулеймани, Б.; Далласта, И.; Дас, П.; Куласингем, Япония; Гирдженти, С.; Саймон, Джей Зи; Бабади, Б.; Марш, Е.Б. Измененная направленная функциональная связь лежит в основе когнитивного восстановления после инсульта. Мозговое сообщество. 2023, 5, fcad149. [Перекрестная ссылка] [ПабМед]
19. Сеген, К.; Спорнс, О.; Залесский, А. Мозговая сетевая коммуникация: концепции, модели и приложения. Нат. Преподобный Нейроски. 2023, 24, 557–574. [Перекрестная ссылка] [ПабМед]
20. Манг Дж.; Сюй, З.; Ци, Ю.; Чжан Т. Поддержка когнитивно-моторного процесса в замкнутом цикле восстановления двигательных функций после инсульта, опосредованного BCI: проблемы и подходы. Передний. Нейробототехника 2023, 17, 1271967. [CrossRef] [ПабМед]
21. Нуньес, Польша; Шринивасан, Р.; Вестдорп, АФ; Виджесингхе, РС; Такер, DM; Зильберштейн, РБ; Кадуш, П.Дж. Когерентность ЭЭГ: I: статистика, референтный электрод, объемная проводимость, лапласианы, кортикальная визуализация и интерпретация в нескольких масштабах. Электроэнцефалограф. Клин. Нейрофизиология. 1997, 103, 499–515. [Перекрестная ссылка] [ПабМед]
22. Аль-Каззаз, Северная Каролина; Хамид бин Мохд Али, С.; Ахмад, С.А.; Ислам, MS; Эскудеро, Дж. Автоматическое удаление артефактов на ЭЭГ нормальных людей. и люди с деменцией, использующие ICA-WT во время задач на рабочую память. Датчики 2017, 17, 1326. [CrossRef] [ПабМед]
23. Аль-Каззаз, Северная Каролина; Али, ШБМ; Ахмад, С.А.; Ислам, MS; Эскудеро, Дж. Дискриминация легких когнитивных нарушений, связанных с инсультом, и сосудистой деменции с использованием анализа сигналов ЭЭГ. Мед. Биол. англ. Вычислить. 2018, 56, 137–157. [Перекрестная ссылка] [ПабМед]
24. Чжан Ю.; Нам, CS; Чжоу, Г.; Джин, Дж.; Ван, Х.; Чихоцкий, А. Пространственные паттерны разреженных групп с временными ограничениями для моторики изображения BCI. IEEE Транс. Киберн. 2018, 49, 3322–3332. [Перекрестная ссылка] [ПабМед]
25. Чаудхари, С.; Таран, С.; Баджадж, В.; Сиули, С. Гибкий аналитический подход, основанный на вейвлет-преобразовании, для задач двигательного воображения. классификация в приложениях BCI. Вычислить. Методы Программы Биомед. 2020, 187, 105325. [CrossRef]
26. Дуань, Л.; Хунсинь, З.; Хан, М.С.; Фанг, М. Распознавание задач воображения движений для BCI с использованием CSP и хаотического двойника PSO SVM. Дж. Китайский университет. Сообщения Телекоммуникации. 2017, 24, 83–90. [Перекрестная ссылка]
27. Алотайби Т.; Абд Эль-Сами, FE; Альшебейли, SA; Ахмад И. Обзор алгоритмов выбора каналов для обработки сигналов ЭЭГ. EURASIP J. Adv. Сигнальный процесс. 2015, 2015, 66. [CrossRef]
28. Он, Л.; Ю, З.; Гу, З.; Ли, Ю. Бхаттачария, выбор каналов на основе границ для классификации образов движений в сигналах ЭЭГ. В материалах Китайской конференции по контролю и принятию решений 2009 г., Гуйлинь, Китай, 17–19 июня 2009 г.; стр. 2353–2356.
29. Тэм, В.К.; Ке, З.; Тонг, Кентукки. Характеристики общего пространственного паттерна под меньшим набором электродов ЭЭГ в интерфейсе мозг-компьютер у пациентов с хроническим инсультом: многосессионное исследование набора данных. В материалах ежегодной Международной конференции Общества инженерии в медицине и биологии IEEE 2011 г., Бостон, Массачусетс, США, 30 августа – 3 сентября 2011 г.; стр. 6344–6347.
30. Гомес-Лопес, Х.К.; Эскобар, Джей-Джей; Диас, А. Ф.; Дамас, М.; Гил-Монтойя, Ф.; Гонсалес, Дж. Повышение сходимости оболочки на основе GA для задач выбора признаков на многомерных данных. В материалах конференции по генетическим и эволюционным вычислениям, Бостон, Массачусетс, США, 9–13 июля 2022 г.; стр. 431–434.
31. Шен, К.; Чжан, К. Двухэтапный улучшенный алгоритм оптимизации Grey Wolf для выбора признаков при многомерной классификации. Комплекс Интел. Сист. 2022, 8, 2769–2789. [Перекрестная ссылка]
32. Тивари, А.; Чатурведи, А. Автоматический выбор каналов ЭЭГ для многоклассовой классификации интерфейса мозг-компьютер с использованием многокритериальный улучшенный алгоритм светлячка. Мультимед. Инструменты Прил. 2023, 82, 5405–5433. [Перекрестная ссылка]
33. Миллан, младший аргентинец; Франце, М.; Моуриньо, Дж.; Цинкотти, Ф.; Бабилони, Ф. Значимые особенности ЭЭГ для классификации спонтанных двигательные задачи. Биол. Киберн. 2002, 86, 89–95. [Перекрестная ссылка] [ПабМед]
34. Мартинес-Кахигаль, В.; Сантамария-Васкес, Э.; Хорнеро, Р. Оптимизация выбора канала интерфейса мозга и компьютера с использованием метаэвристики и эволюционных алгоритмов. Прил. Мягкий компьютер. 2022, 115, 108176. [CrossRef]
35. Гаур, П.; Гупта, Х.; Чоудхури, А.; Маккриди, К.; Пачори, РБ; Ван, Х. Общий пространственный паттерн скользящего окна для улучшения классификации образов движений в ЭЭГ-BCI. IEEE Транс. Инструмент. Измер. 2021, 70, 4002709. [CrossRef]
36. Ши, Б.; Ван, К.; Инь, С.; Юэ, З.; Хуай, Ю.; Ван, Дж. Алгоритм поиска двоичной гармонии как метод выбора канала для двигателя. BCI на основе изображений. Нейрокомпьютинг 2021, 443, 12–25. [Перекрестная ссылка]
37. Гаеми, А.; Рашеди, Э.; Пуррахими, АМ; Камандар, М.; Рахдари, Ф. Автоматический выбор канала в сигналах ЭЭГ для классификации движений левой или правой руки в компьютерных интерфейсах мозга с использованием улучшенного алгоритма бинарного гравитационного поиска. Биомед. Сигнальный процесс. Контроль 2017, 33, 109–118. [Перекрестная ссылка]
38. Лотте, Ф.; Конгедо, М.; Лекюйер, А.; Ламарш, Ф.; Арнальди, Б. Обзор алгоритмов классификации мозг-компьютера на основе ЭЭГ. интерфейсы. Дж. Нейронная инженерия. 2007, 4, Р1. [Перекрестная ссылка]
39. Хасан, М.М.; Алам, MGR; Уддин, МЗ; Худа, С.; Альмогрен, А.; Фортино, Г. Распознавание человеческих эмоций с использованием сетевой архитектуры глубоких убеждений. Инф. Фьюжн 2019, 51, 10–18. [Перекрестная ссылка]

Machine Translated by Google

40. Хосла А.; Ханднор, П.; Чанд, Т. Сравнительный анализ методов обработки и классификации сигналов для различных приложения на основе сигналов ЭЭГ. Биокиберн. Биомед. англ. 2020, 40, 649–690. [Перекрестная ссылка]

41. Алтан Г.; Кутлу, Ю.; Аллахверди, Н. Классификация активности мозга на основе сетей глубоких убеждений с использованием ЭЭГ по медленным корковым потенциалам. в инсульте. Межд. Дж. Прил. Математика. Электрон. Вычислить. 2016, 4, 205–210. [Перекрестная ссылка]

42. Гарсиа-Лаэнсина, П.Дж.; Родригес-Бермудес, Г.; Рока-Дорда, Дж. Исследование уменьшения размерности функций ЭЭГ в моторике. классификация задач изображений. Экспертная система. Прил. 2014, 41, 5285–5295. [Перекрестная ссылка]

43. Алер Р.; Гальван И.М. Оптимизация количества электродов и пространственных фильтров для интерфейсов мозг–компьютер с помощью эволюционный многокритериальный подход. Экспертная система. Прил. 2015, 42, 6215–6223. [Перекрестная ссылка]

44. Лал, Теннесси; Шредер, М.; Хинтербергер, Т.; Уэстон, Дж.; Богдан, М.; Бирбаумер, Н.; Шолькопф, Б. Выбор векторного канала поддержки в БКИ. IEEE Транс. Биомед. англ. 2004, 51, 1003–1010. [Перекрестная ссылка] [ПабМед]

45. Ракотомамонджи, А.; Гиг, В. Конкурс BCI III: набор данных II SVM для орфографии BCI P300. IEEE Транс. Биомед. англ. 2008, 55, 1147–1154. [Перекрестная ссылка] [ПабМед]

46. Тан, П.; Ван, Х.; Ван, Ю. Уменьшение размерности при выборе признаков на основе эволюционных алгоритмов для изображений движений интерфейс мозг-компьютер. Рой Эвол. Вычислить. 2020, 52, 100597. [CrossRef]

47. Цай, К.Ф.; Эберле, В.; Чу, К.Ю. Генетические алгоритмы при выборе признаков и экземпляров. Информационная система. 2013, 39, 240–247. [Перекрестная ссылка]

48. Удхая Кумар, С.; Ханна Инбарани, Х. Выбор признаков на основе PSO и приблизительная классификация на основе наборов окрестностей для BCI Многоклассовая задача по двигательному воображению. Нейронный компьютер. Прил. 2017, 28, 3239–3258. [Перекрестная ссылка]

49. Байг, МЗ; Аслам, Н.; Шум, НР; Чжан, Л. Алгоритм дифференциальной эволюции как инструмент выбора оптимального подмножества признаков в двигательное воображение ЭЭГ. Экспертная система. Прил. 2017, 90, 184–195. [Перекрестная ссылка]

50. Атьяби А.; Люерсен, М.; Фитцгиббон, С.; Powers, DM Эволюционный выбор функций и уменьшение количества электродов для классификации ЭЭГ. В материалах Конгресса IEEE по эволюционным вычислениям 2012 г., Брисбен, Квинсленд, Австралия, 10–15 июня 2012 г.; стр. 1–8.

51. Накиса, Б.; Растгу, Миннесота; Тьондронегоро, Д.; Чандран, В. Эволюционные алгоритмы вычислений для выбора признаков Распознавание эмоций на основе ЭЭГ с помощью мобильных датчиков. Экспертная система. Прил. 2018, 93, 143–155. [Перекрестная ссылка]

52. Аль-Каззаз, Северная Каролина; Али, SHM; Ахмад, С.А. Алгоритм выбора канала на основе сигнала ЭЭГ на основе дифференциальной эволюции для раннего выявления сосудистой деменции среди людей, перенесших инсульт. В материалах конференции IEEE-EMBS по биомедицинской инженерии и наукам (IECBES) 2018 г., Саравак, Малайзия, 3–6 декабря 2018 г.; стр. 239–244.

53. Альяссери, ЗАА; Хадер, АТ; Аль-Бетар, Массачусетс; Жоао, ПП; Осама, А.А. Аутентификация личности на основе ЭЭГ с использованием многокритериального алгоритма опыления цветов. В Proceedings of the Evolutionary Computation (CEC), Рио-де-Жанейро, Бразилия, 8–13 июля 2018 г.; стр. 15–22.

54. Ян, XS. Новый метаэвристический алгоритм, вдохновленный летучей мышью. В «Стратегиях сотрудничества по оптимизации, вдохновленных природой» (NICSO 2010); Springer: Берлин/Гейдельберг, Германия, 2010 г.; стр. 65–74.

55. Хиваринен А. Быстрые и надежные алгоритмы с фиксированной точкой для анализа независимых компонентов. IEEE Транс. Нейронная сеть. 1999, 10, 626–634. [Перекрестная ссылка] [ПабМед]

56. Эскудеро, Дж.; Хорнеро, Р.; Абасоло, Д.; Фернандес, А. Слепое разделение источников для улучшения спектральных и нелинейных характеристик записей магнитоэнцефалограммы. Применение при болезни Альцгеймера. Мед. англ. Физ. 2009, 31, 872–879. [Перекрестная ссылка] [ПабМед]

57. Джеймс, СЈ; Гессен, К.В. Независимый компонентный анализ биомедицинских сигналов. Физиол. Измер. 2005, 26, P15. [Перекрестная ссылка] [ПабМед]

58. Аль-Каззаз, Северная Каролина; Хамид бин Мохд Али, С.; Ахмад, С.А.; Ислам, MS; Эскудеро, Дж. Выбор материнских вейвлет-функций для анализа многоканального сигнала ЭЭГ во время задачи рабочей памяти. Датчики 2015, 15, 29015–29035. [Перекрестная ссылка] [ПабМед]

59. Эскудеро, Дж.; Хорнеро, Р.; Абасоло, Д.; Фернандес, А. Количественная оценка удаления артефактов на реальной магнитоэнцефалограмме сигналы со слепым разделением источников. Энн. Биомед. англ. 2011, 39, 2274–2286. [Перекрестная ссылка] [ПабМед]

60. Барбати, Г.; Поркаро, К.; Заппасоди, Ф.; Россини, премьер-министр; Теккио, Ф. Оптимизация подхода к анализу независимых компонентов для идентификации и удаления артефактов в магнитоэнцефалографических сигналах. Клин. Нейрофизиология. 2004, 115, 1220–1232. [Перекрестная ссылка]

61. Маммоне, Н.; Ла Фореста, Ф.; Морабито, Ф.К. Автоматическое отклонение артефактов из многоканальной ЭЭГ кожи головы с помощью вейвлета ICA. IEEE Sens. J. 2011, 12, 533–542. [Перекрестная ссылка]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Классификация и регрессия точечной коррозии трубопроводов

На основе сигналов утечки магнитного потока с использованием сверточного метода Нейронные сети

Юфэй Шен и Вэньсин Чжоу *

Департамент гражданского строительства и экологической инженерии, Университет Западного Онтарио, Лондон, ON N6A 5B9, Канада * Адрес для переписки: wzhou@eng.uwo.ca

Аннотация: Точечную коррозию на нефте- и газопроводах трудно обнаружить и определить ее размер, и, следовательно, они представляют собой серьезную проблему для практики управления целостностью трубопроводов. В этом исследовании разрабатываются две модели сверточной нейронной сети (CNN) для идентификации точечных отверстий и прогнозирования размеров и местоположения точечной коррозии в соответствии с сигналами утечки магнитного потока, генерируемыми с помощью магнитостатического анализа методом конечных элементов. Для обучения и проверки двух моделей CNN создаются обширные примеры трехмерного параметрического анализа методом конечных элементов . Кроме того, комплексный анализ алгоритмов оценивает производительность модели, предоставляя представление о практическом применении моделей CNN для управления целостностью трубопроводов. Показано, что предложенная классификационная модель CNN очень точна при классификации точечных отверстий и точечных коррозионных дефектов в целом. Показано, что предложенная регрессионная модель CNN обеспечивает высокую точность прогнозирования местоположения точечного отверстия и обеспечивает достаточно высокую точность оценки глубины и диаметра точечного отверстия даже при наличии шумов измерений. Это исследование показывает эффективность использования алгоритмов глубокого обучения для улучшения практики управления целостностью корродированны

Ключевые слова: алгоритм глубокого обучения; сверточная нейронная сеть; трубопровод; коррозия; пинхол; анализ методом конечных элементов; сигнал утечки магнитного потока

check for updates

Цитирование: Шен, Ю.; Чжоу, В. Классификация и регрессия точечной коррозии на трубопроводах на основе сигналов утечки магнитного потока с использованием сверточных нейронных сетей. Алгоритмы 2024, 17, 347. https://doi.org/10.3390/a17080347.

Академические редакторы: Чи-Лунг Линь, Бор-Джюнн Хван, Шао-Ган Мяоу и Чи-Хун Чуан

Поступила: 24 июня 2024 г.
Пересмотрено: 1 августа 2024 г.
Принято: 6 августа 2024 г.
Опубликовано: 8 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Трубопроводы являются экономичным и безопасным средством транспортировки и распределения больших объемов нефтегазовой продукции на большие расстояния [1]. Однако отказы трубопроводов могут привести к серьезным последствиям, таким как гибель людей, экономические потери и ущерб окружающей среде [2]. Среди различных механизмов разрушения, угрожающих структурной целостности трубопроводов, ведущую угрозу представляет коррозия [3]. Инструмент внутритрубного контроля (ILI) широко используется для обнаружения и измерения коррозии с целью оценки целостности трубопровода [4], а утечка магнитного потока (MFL) является преобладающей технологией ILI для выявления внутренних и внешних коррозионных дефектов как в трубопроводах для жидкости, так и в газопроводах. [5]. Форум операторов трубопроводов (POF) предполагает, что единичную коррозионную аномалию на внутренней или внешней поверхности трубопровода можно отнести к одному из семи классов на основе сообщаемой ILI длины и ширины коррозии, как показано на рисунке 1 [6], т.е. , осевые прорези, окружные прорези, осевые канавки, круговые канавки, общие, точечные и точечные, чтобы лучше количественно оценить точность инструментов ILI и облегчить выбор подходящих моделей для оценки разрывной способности трубопроводов, содержащих коррозионные дефекты [6,7].

Оценка размеров коррозионных аномалий по сигналам МФЛ относится к области обратного моделирования. В трубопроводной отрасли часто используется итеративная обратная модель из-за ее замечательной точности [8–10]. Однако итеративная обратная модель требует больших вычислительных затрат. Обратные модели, основанные на машинном обучении, предоставляют многообещающие альтернативы определению размеров коррозионных дефектов на основе сигналов MFL с высокой точностью, но являются более эффективными в вычислительном отношении по сравнению с итерационными обратными моделям

Machine Translated by Google

Кандрооди и др. [14] передали оцененную алгоритмом ширину и длину дефекта, а также извлеченные размах сигналов MFL в нейронную сеть с гауссовой радиальной базисной функцией для прогнозирования глубины коррозии. Фэн и др. [15] предложили методологию корректировки ошибок для восстановления профилей коррозии по сигналам MFL на основе нейронной сети с радиальной базисной функцией. Однако эти модели имеют определенные ограничения: обучение этих моделей основано на выбранных вручную функциях сигналов MFL, что потенциально упускает ценные данные, необходимые для точного прогнозирования профиля Кроме того, большинство этих методов фокусируются исключительно на одном измерении сигнала MFL, игнорируя важную информацию, присутствующую в двух других измерениях.

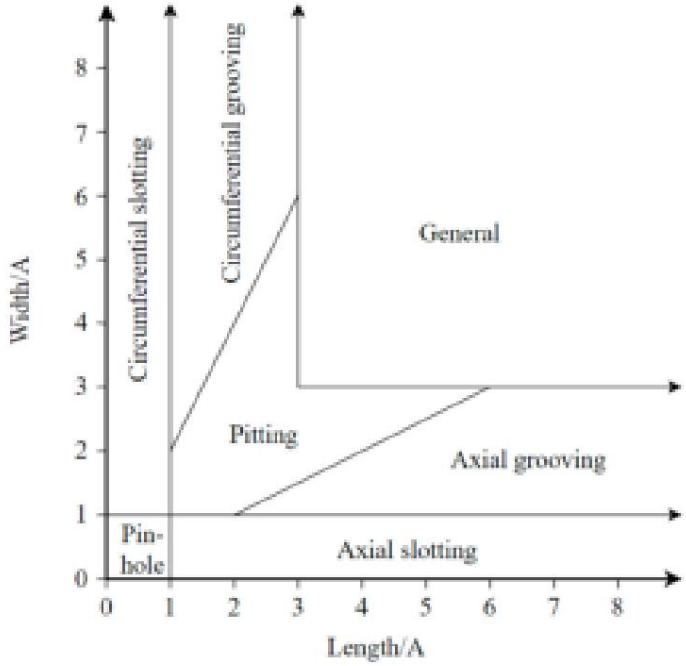


Рисунок 1. Семь категорий коррозионных аномалий основаны на длине и ширине аномалии. Примечание: A = макс. {10 мм, толщина стенки трубы}.

За последнее десятилетие сверточные нейронные сети (CNN) стали революционным алгоритмом глубокого обучения, дающим замечательные результаты в различных областях распознавания образов, включая обработку изображений и голоса [16,17]. Инструмент CNN может автоматически извлекать глубокие признаки из входных данных вместо созданных вручную экстракторов признаков; поэтому он хорошо подходит для решения обратного моделирования на основе сигналов MFL. В литературе сообщалось о применении CNN для обратного анализа сигналов MFL . Лу и др. [18] представили визуальное преобразование CNN, которое прогнозирует длину, ширину и глубину аномалий, принимая во внимание две компоненты (т.е. окружную и радиальную) сигнала MFL, игнорируя при этом продольную составляющ Шен и Чжоу [19] использовали CNN для оценки местоположения и размеров коррозии на стальных трубопроводах на основе трех компонентов сигналов MFL. Ван и Чен [20] предложили архитектуру CNN, состоящую из двух основных модулей: один для классификации аномалий , а другой — для регрессии размера аномалий. Входные данные модуля классификации состоят из всех трех компонентов сигнала MFL, а выходные данные представляют собой семь категорий аномалий, как показано на рисунке 1. Модуль регрессии включает семь отдельных CNN, каждая из которых адаптирована к определенному типу аномалии, прогнозируя такие параметры, как ширина, длина и глубина дефекта.

Согласно отчету Европейской группы по сбору данных о происшествиях на газопроводах (EGIG) [21], точечная коррозия считается основным фактором, способствующим отказам трубопроводов, из-за ее потенциального перехода от небольшого начального объема потери металла к более серьезной потере металла. учитывая сложность обнаружения с помощью мониторинга давления, особенно когда утечка продуктов не превышает уровня земли [22,23]. Несмотря на влияние точечных отверстий на целостность трубопровода,

Machine Translated by Google

Литература, специально посвященная обнаружению и определению размеров точечной коррозии, ограничена , что создает пробел в знаниях. Кроме того, небольшие коррозионные аномалии, такие как точечные отверстия, могут иногда перекрываться с относительно большой зоной коррозии [23–25], создавая сложный дефект, известный в этом исследовании как точечный дефект коррозии в целом (PIC). В литературе также мало исследований, посвященных обнаружению и определению размеров дефектов PIC .

Целью данного исследования является разработка алгоритмов глубокого обучения, то есть моделей CNN, для классификации и определения размеров точечных отверстий и дефектов PIC на основе большого набора сигналов MFL. Классификационная модель CNN рассматривает шесть сценариев, т.е. точечные отверстия на внутренней или внешней поверхности трубы, общую коррозию на внутренней или внешней поверхности и дефекты PIC на внутренней или внешней поверхности. Сигналы о точечных отверстиях и дефектах PIC, выявленных с помощью модели классификации, затем передаются в регрессионную модель CNN для прогнозирования размеров и местоположения точечных отверстий и дефектов PIC. На практике нет необходимости последовательно применять две модели CNN. В зависимости от конкретного приложения данные могут вводиться непосредственно в регрессионную модель, если классификация уже была проведена другими способами и основной целью является прогнозирование размера дефекта. Поскольку реальные сигналы MFL обычно являются собственностью и не являются общедоступными, сигналы MFL, используемые в этом исследовании, генерируются численно путем применения трехмерного (3D) магнитостатического анализа методом конечных элементов (FEA) с использованием коммерческого программного обеспечения COMSOL (версия 5.6). Трехмерные (т.е. продольные, окружные и радиальные) карты сигналов MFL служат входными данными для обучения и проверки моделей CNN для классификации дефектов и прогнозирования размеров, а также окружного и продольного положения отверстий на трубопроводе. Также исследуется влияние шумов измерений, связанных с сигналом MFL, на точность регрессионной модели CNN. Это исследование заполняет пробел в знаниях в литературе, применяя CNN для обнаружения, определения местоположения и определения размера точечных отверстий и дефектов, вызванных коррозией. Отмечается, что точное определение положения точечного отверстия в пределах общей коррозии имеет решающее значение, поскольку относительное положение точечного отверстия в пределах общей коррозии оказывает заметное влияние на взрывоопасность дефекта PIC и, следовательно, имеет большое значение для точного определения. прогнозирование пропускной способности. Это исследование демонстрирует эффективность CNN для обнаружения и определения размеров точечной коррозии, а также жизнеспособность CNN для применения в практике управления целостностью трубопроводов.

Остальная часть статьи структурирована следующим образом. В разделе 2 представлены принципы технологии MFL и основная справочная информация о взаимосвязи между сигналами MFL и аномальными особенностями. В разделе 3 представлена предлагаемая модель FEA и параметры моделирования сигналов MFL. В разделе 4 описывается структура разработанных моделей CNN и их применение для классификации и регрессии точечных отверстий. Соответствующая точность прогнозирования моделей CNN также обсуждается в разделе 4. В разделе 5 суммированы заключительные замечания исследования.

2. Принципы техники MFL.

Магнитостатический FEA используется для численной генерации сигналов MFL в исследовании. Соответствующее основное уравнение, полученное из уравнений Максвелла, показано в уравнении (1):

$$\times \quad \frac{1}{\mu} \times A = 0 \quad (1)$$

где × обозначает операцию завивки; μ представляет магнитную проницаемость ферромагнитного образца (N/A2), а A – вектор магнитного потенциала. Уравнение (1) решается при заданных граничных условиях для получения A, которое затем используется для получения плотности магнитного потока B следующим образом [26,27]:

$$B = \times A \quad (2)$$

Метод MFL основан на принципе измерения, при котором ферромагнитный материал демонстрирует MFL вблизи дефекта, как показано на рисунке 2 [18]. Процедура измерения следующая: в процессе ИЛИ инструмент МФЛ приводится в движение.

Machine Translated by Google

по трубопроводу, а магниты внутри инструмента генерируют магнитное поле. В идеальной трубе весь магнитный поток остается внутри стенки трубы [28], что приводит к равномерному распределению магнитного поля [29]. Однако при наличии дефектов некоторая часть магнитного потока «утекает» за пределы трубы, образуя поле магнитной рассеяния вблизи дефекта [30]. Это поле утечки обнаруживается датчиками Холла, генерирующими электрические сигналы [29]. Сила и распределение утечки флюса изменяются в зависимости от геометрии дефекта на трубопроводе. Путем захвата сигнала MFL, который обычно состоит из трехмерного векторного поля плотности магнитного потока (B), представленного Bx в окружном направлении трубы, By в радиальном направлении и Bz в продольном (осевом) направлении, можно оценить размеры и особенности расположения коррозионного дефекта.

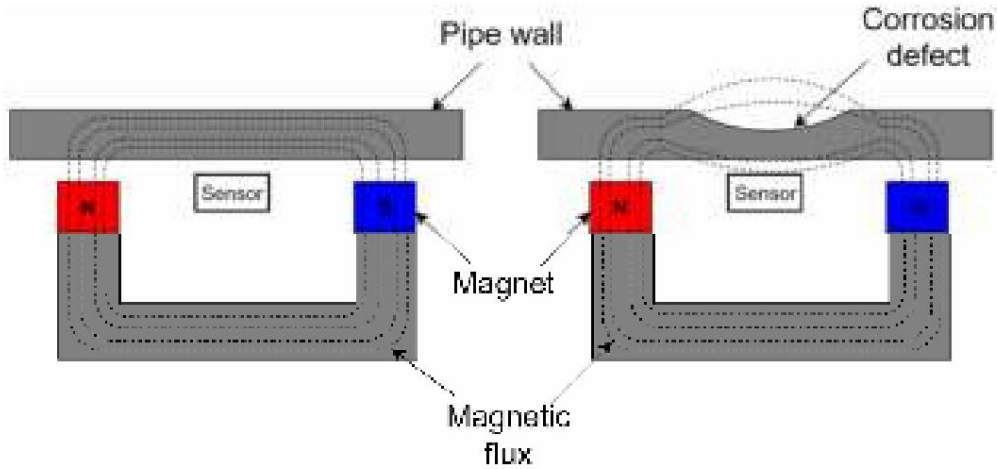


Рисунок 2. Принцип метода MFL.

Осевые (AMFL) и кольцевые (CMFL) инструменты MFL — это два типа инструментов MFL, которые классифицируются по расположению магнитов внутри инструмента MFL. В инструментах AMFL постоянные магниты ориентированы параллельно продольной оси трубы. Эти инструменты превосходно обнаруживают и измеряют дефекты в окружном направлении трубы, но менее эффективны при определении размеров элементов, ориентированных в осевом направлении [31]. И наоборот, инструменты CMFL имеют магниты, расположенные в направлении окружности трубопровода. Они более точны при измерении объектов, выровненных в продольном направлении, по сравне Инструмент CMFL является основным предметом данного исследования, поскольку он более точен, чем инструмент AMFL, для определения размеров продольно ориентированных коррозионных аномалий, которые перпендикулярны окружному напряжению, возникающему в результате внутреннего давления трубы, и, следовательно, могут иметь большое влияние на разрывная мощность трубопровода.

3. Моделирование сигналов MFL с использованием FEA

3.1. Параметры моделирования

Инструмент CMFL, рассматриваемый в данном исследовании, включает в себя три пары постоянных магнитов. NdFeB выбран в качестве магнитного материала с высокой коэрцитивной силой 895 000 А/м, стабильными магнитными свойствами и высокой магнитной энергией [32]. Предполагается, что для моделирования трубы используется сталь марки X52, характеризующаяся остаточной плотностью магнитного потока (Br) 1 Тл, электропроводностью 5,882 × 106 См/м, напряженностью коэрцитивного поля (Hc) 415 А/м и плотность потока насыщения (Bs) 1,9Тл. На рисунке 3 показаны схемы поперечного сечения инструмента CMFL и секции трубы, а геометрические свойства и свойства материала каждого компонента сведены в таблицу 1. Отмечается, что величина отрыва принимается равной 5 мм, что соответствует рис. — расстояние между датчиком и внутренней поверхностью трубы [33].

Machine Translated by Google

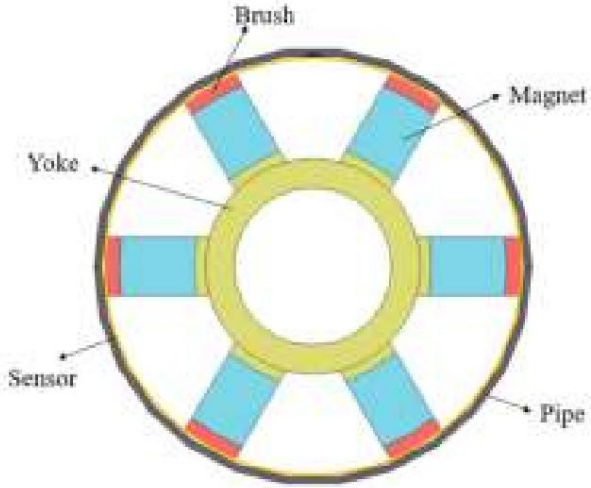


Рисунок 3. Иллюстрация инструмента CMFL.

Таблица 1. Геометрия каждого компонента предлагаемой 3D-модели CMFL.

Элементы	Секция трубы	Магниты	Кисти	ярмо
геометрия (мм)	600 × 610 × 10 а Длина	б 300 × 81 × 105	× 81 × 20	б 300 × 300 × 40 300 а

× наружный диаметр × толщина стенки. б Длина × ширина × толщина.

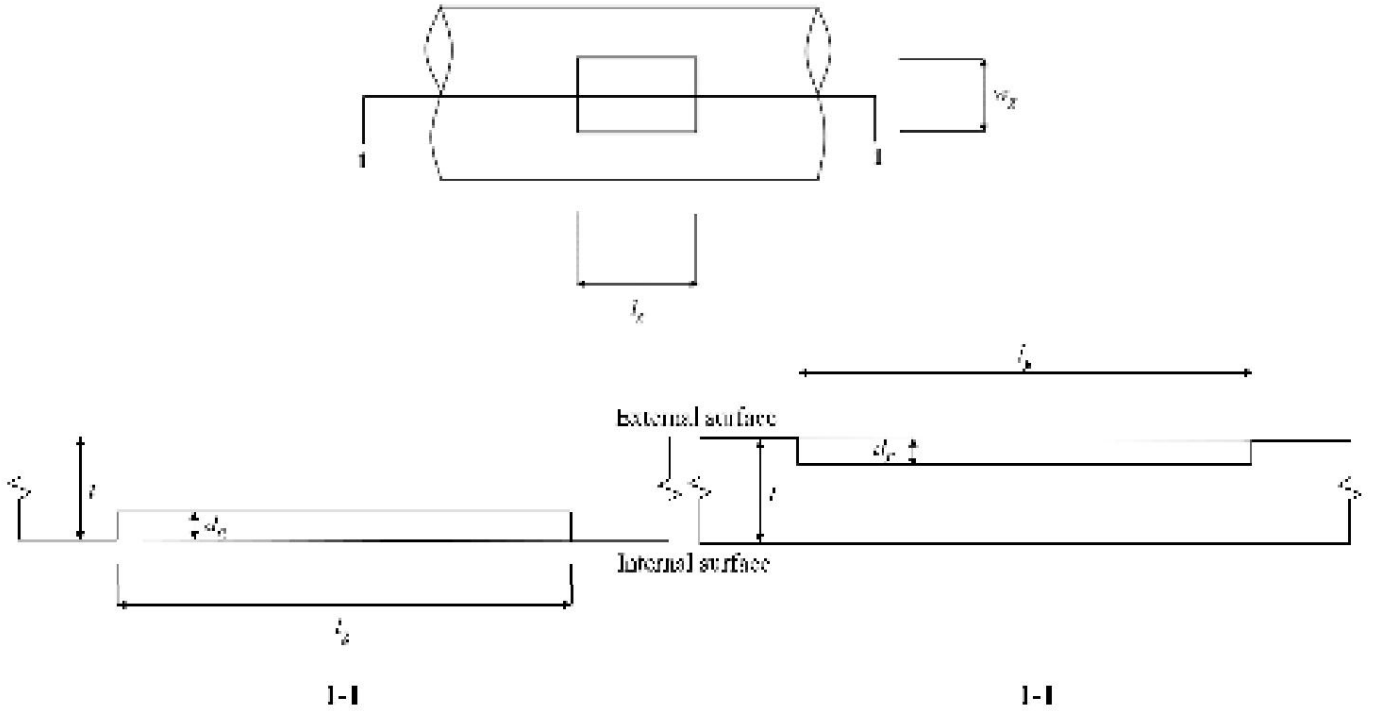
3.2. Дела ВЭД

Создается и анализируется серия моделей FEA с одинаковыми условиями намагничивания и атрибутами труб, но они различаются по типам коррозии, геометрическим параметрам и характеристикам. места в этом исследовании. Моделирование включает в себя 9600 анализируемых случаев, из которых 1600 случаев отнесены к каждому из шести сценариев, как показано на рисунке 4, а именно: внутренняя общая коррозия. (Джин), внешняя общая коррозия (Gex), внутреннее точечное отверстие (Pin), внешнее точечное отверстие (Pex), внутреннее Дефект PIC (PICin) и внешний дефект PIC (PICex). Общая коррозия идеализируется как кубовидной формы, определяемой шириной (wg), глубиной (dg) и длиной (lg), а точечное отверстие имеет вид идеализированный как цилиндрический, определяемый радиусом (rp) и глубиной (dp). Обратите внимание на центр Предполагается, что точечное отверстие в дефекте PIC совпадает с таковым при общей коррозии, который определяется двумя координатами, т. е. его продольным расположением (h) и окружным место (φ) в пределах обозначенной области размером 160 × 160 мм, как показано на рисунке 5. Эта область отмечен голубым цветом, центр которого задан координатами (r, φ = 0, h = 0), где r представляет внутренний радиус трубы. Эта особая квадратная площадка размером 160 × 160 мм на была выбрана внутренняя стенка трубы, поскольку она охватывает наибольшую площадь и демонстрирует постоянную профиль плотности магнитного потока между соседними магнитами.

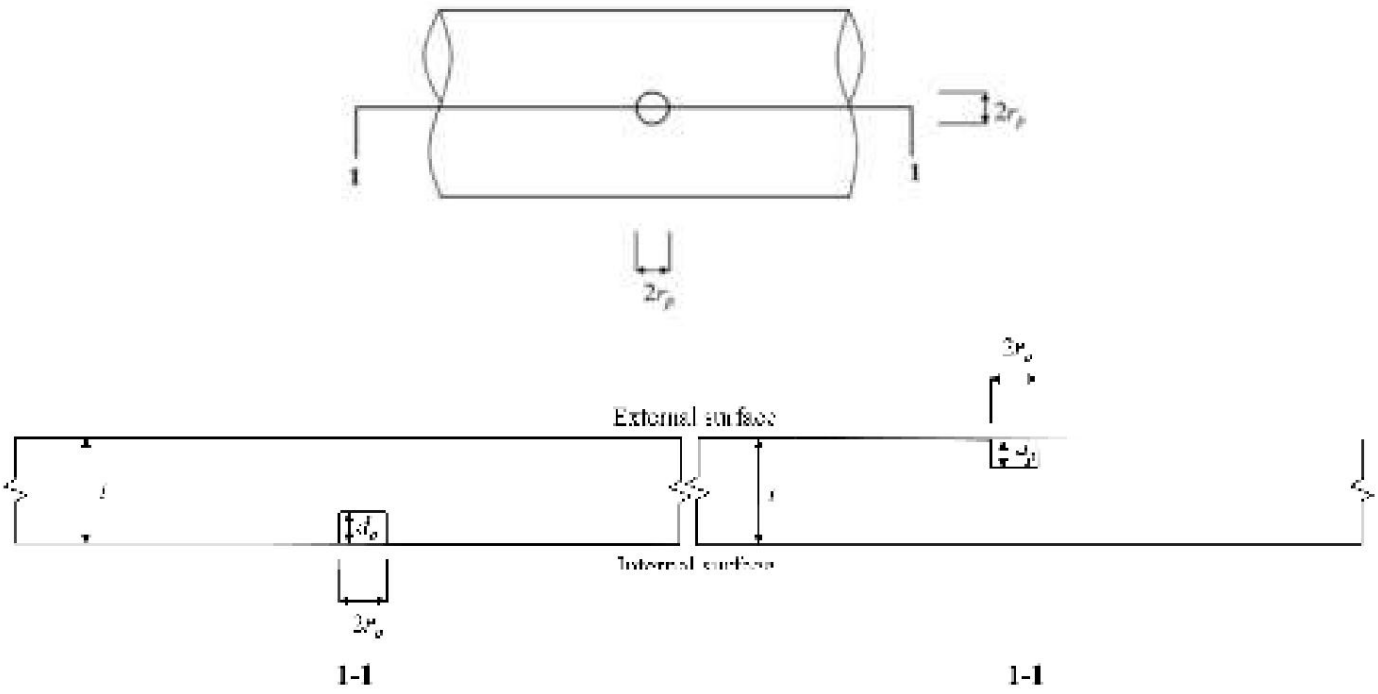
Случаи анализа, включающие только общую коррозию, т. е. Gin и Gex, включают четыре значения wg, т. е. 30, 50, 70 и 90 мм, четыре значения dg/t, т. е. 20, 40, 60 и 80 %, где t обозначает толщину стенки трубы и четыре значения lg, т.е. 30, 50, 70 и 90 мм. Это приводит к всего 64 общих коррозии разного размера. По локализации коррозии пять значения h (-50, -25, 0, 25 и 50 мм) и пять значений φ (-10, -5, 0, 5 и 10 градусов) считаются. Перестановки 64 дефектов и 25 мест приводят к 1600 случаям для каждого. Джина и Гекса .

Случаи анализа, включающие только точечную коррозию, т. е. Pin и Pex, включают восемь значения rp, т.е. 1, 1,5, 2, 2,5, 3, 3,5, 4 и 4,5 мм, и восемь значений dp/t, т.е. 10, 20, 30, 40, 50, 60, 70 и 80%. В результате всего образуется 64 точечных коррозии разного размера. По локализации коррозии пять значений h (-50, -25, 0, 25 и 50 мм) и пять учитываются значения φ (10, 5, 0, 5 и 10 градусов). Перестановка 64 дефектов а 25 локаций также приводят к 1600 случаям для каждого из Pin и Pex.

Machine Translated by Google



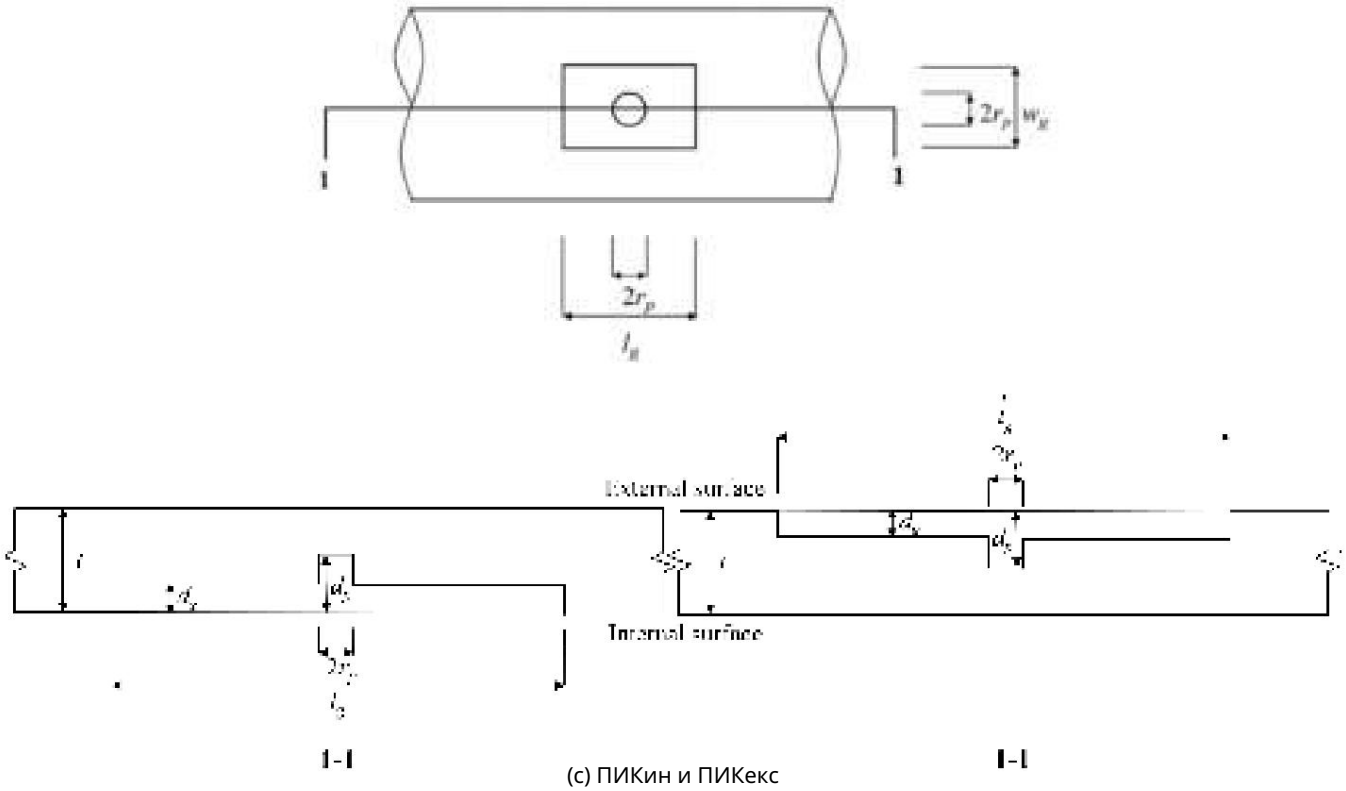
(а) Джин и Гекс



(б) Штифт и Рех

Рисунок 4. Продолжение.

Machine Translated by Google



(c) ПИКин и ПИКекс

Рисунок 4. Шесть типов коррозионных ситуаций: (а) Джин и Gex; (б) Штифт и Рех; (в) ПИКин и ПИКекс.

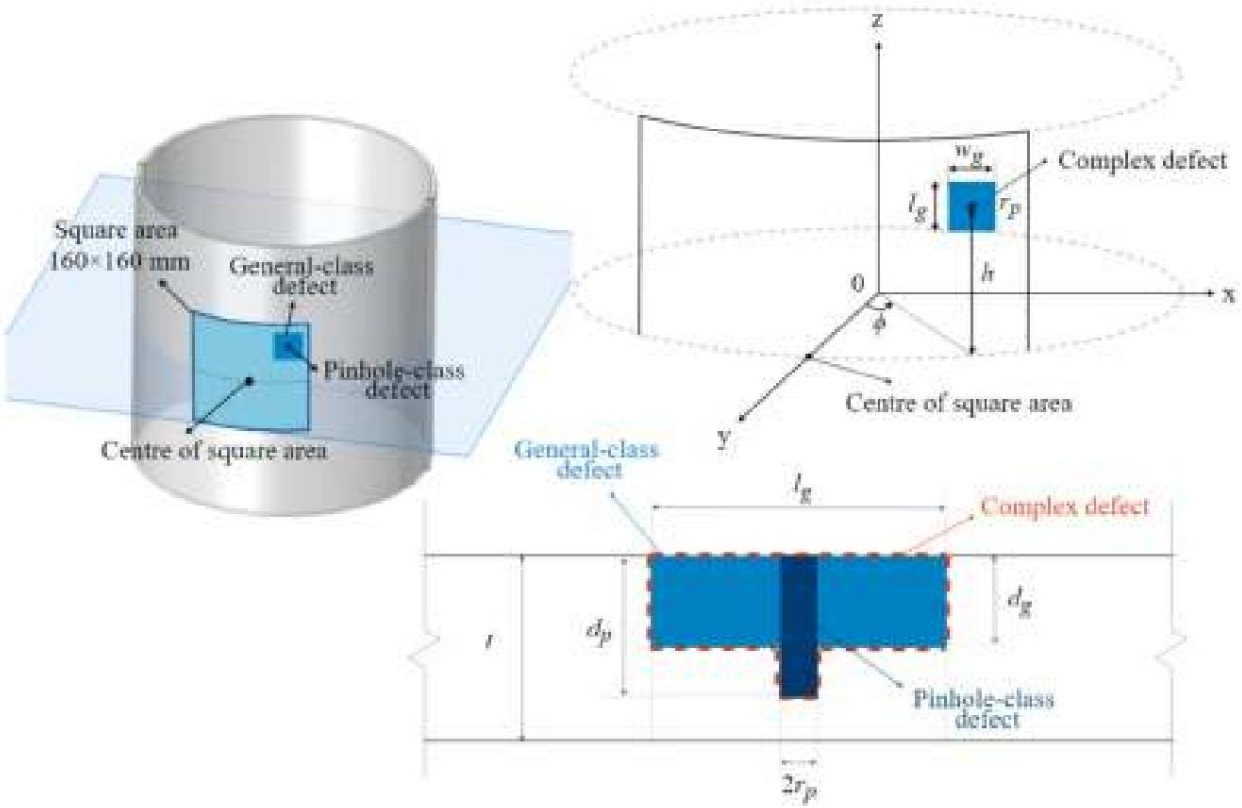


Рис. 5. Цилиндрическая система координат, определяющая параметры местоположения h и φ.

Machine Translated by Google

Случаи анализа, связанные с дефектами PIC, т. е. PICin и PICex, включают wg = lg = 50 мм и четыре значения rp, т. е. 1, 2, 3 и 4 мм. Что касается размеров общей и точечной коррозии, считается, что три группы охватывают более широкий диапазон относительных взаимосвязей между общей и точечной коррозией. Первый включает dg/t , равный 20 %, и dp/t, равный 30, 40, 50, 60, 70 или 80 %. Ко второй группе относятся dg/t , равный 30 %, и dp/t, равный 40, 50, 55, 60, 70 или 80 %. В последнюю группу входят dg/t , равный 40 %, и dp/t, равный 50, 60, 70 или 80 %. Следует отметить, что предполагается, что dp/t как минимум на 10% больше, чем dg/t в φ – всех трех группах. В результате всего получается 64 дефекта PIC разного размера. С точки зрения места коррозии рассматриваются пять значений h (-50, -25, 0, 25 и 50 мм) и пять значений φ (-10, -5, 0, 5 и 10 градусов). Перестановка 64 дефектов и 25 локаций приводит к 1600 случаям для каждого из PICin и PICex. Отсюда следует, что в этом исследовании создается набор данных, состоящий из 9600 случаев (1600 × 6 = 9600). Этот большой набор данных предназначен для облегчения обучения и проверки модели CNN, обеспечивая всесторонний анализ и надежную разработку модели.

Разработанная модель FEA строит сетки с использованием 4-узловых тетраэдрических элементов, где минимальный размер сетки составляет 4 мм, а максимальный размер сетки составляет 55 мм после исследования сходимости. В частности, сетки внутри области 160 × 160 мм четко разрешаются с интервалом 1 мм. В каждой точке сетки значение B рассчитывается посредством интерполяции между двумя узлами сетки. Система нелинейных уравнений решается методом Ньютона-Рафсона. В качестве иллюстрации на рисунке 6 изображена трехмерная информация о плотности магнитного потока (т. е. Bx, By и Bz) на участке размером 160 × 160 мм выбранной модели FE. Предложенная модель FEA была проверена путем сравнения результатов моделирования с экспериментальными данными инструмента CMFL, о котором сообщили Ирландия и Торрес [34] , и инструмента AMFL, о котором сообщили Азиззаде и Сафизаде [35]. Для получения подробной информации о проверке читатели отсылаются к Shen и Zhou [19] .

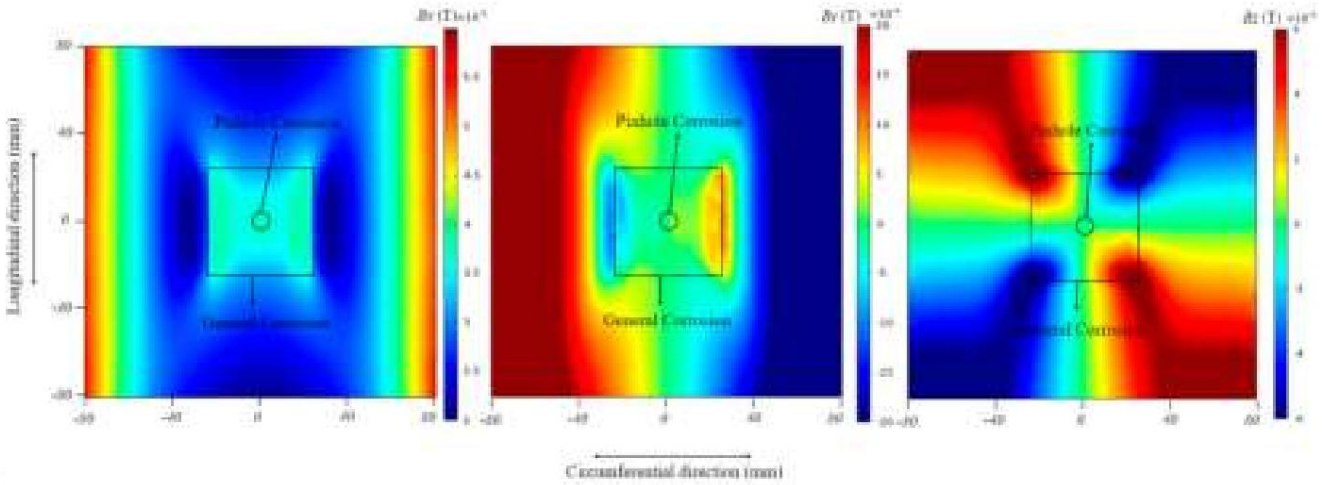


Рис. 6. Полученные методом ФЭА Bx, By и Bz , соответствующие дефекту ПОС на внутренней поверхности трубы (wg = 50 мм, lg = 50 мм, dg/t = 40 %, rp = 4 мм, dp/t = 80 %, h = 0 мм и φ = 0 градусов).

Сигналы MFL в практических приложениях неизбежно содержат шум, возникающий из-за таких факторов, как неравномерная толщина стенки трубы и различия в значениях отрыва датчика [36]. Эти шумы могут повлиять на точность обратной модели, что приводит к необходимости включения шумов в сигналы MFL, моделируемые методом FEA. Отношение сигнал/шум (SNR) служит важным параметром для количественной оценки шума в трехмерном сигнале MFL. Уравнение (3) определяет соотношение сигнал/шум в данном исследовании [37]:

$$\text{ОСШ} = 10\log 10 \frac{\sum_{n=1}^{160}\sum_{m=1}^{160} Bx(n,m)^2 + By(n,m)^2 + Bz(n,m)^2}{\sum_{n=1}^{160}\sum_{m=1}^{160} wx(n,m)^2 + wy(n,m)^2 + wz(n,m)^2} \quad (3)$$

Machine Translated by Google

где wx, wy и wz обозначают шумы вдоль трех направлений трубы, а именно окружного, радиального и продольного направлений соответственно, и (n, m) (где n, m = 1, 2, . . ., 160) представляет координату каждой сетки (с разрешением 1 мм) на площади 160 × 160 мм, всего 25 600 сеток. Принимая во внимание репрезентативное значение SNR, равное 20 [38], моделируются три матрицы гауссово распределенных белых шумов размером 160 × 160, которые считаются независимыми друг от друга, и затем применяются к Bx, By и Bz соответственно . .
Каждая матрица предполагает среднее значение, равное нулю, со стандартными отклонениями, корректируемыми методом проб и ошибок до достижения заранее определенного отношения сигнал/шум. Пусть σn обозначает стандартное отклонение wx; стандартные отклонения wy и wz принимаются равными 0,75σn и 0,25σn соответственно. Масштабирование σn основано на относительных величинах Bx, By и Bz , полученных в результате анализа методом конечных элементов: для данного случая анализа Bx постоянно имеет самую высокую величину, за которой следуют By и Bz соответственно. Окончательный зашумленный вход в модель CNN состоит из трех матриц Bx, By и Bz, каждая из которых объединена с соответствующими зашумленными матрицами wx, wy и wz, что приводит к зашумленным представлениям сигналов MFL.

Подчеркивается, что зашумленные сигнальные матрицы MFL вводятся в регрессионную модель CNN только как расширенная часть в разделе 4.2.4. Фан и др. [39] сообщили, что влияние гауссовского шума на точность модели классификации CNN незначительно. Ачарья и др. [40] указали, что удаление шума не является необходимым для классификации изображений с помощью алгоритмов глубокого обучения [41]. Поэтому в настоящем исследовании мы рассматриваем только влияние шумов на точность регрессионной модели.

4. Сверточная нейронная сеть 4.1.

Классификация CHC 4.1.1.

Входная информация

Входные данные для модели CNN обычно представляются как цветное изображение RGB, включающее три цветовых канала, соответствующие зеленому, красному и синему цветам соответственно. Этот вход совместим с трехмерными сигналами MFL. Для данной модели КЭ трехмерная информация о плотности магнитного потока (т. е. Bx, By и Bz) в пределах обозначенной квадратной области экспортируется из программного обеспечения COMSOL в сетках размером 160 × 160 мм с разрешением 1 мм.
Три сгенерированные матрицы безшумовых сигналов MFL размером 160 × 160 (по одной матрице для каждого измерения) обозначаются как входные данные предлагаемой модели CNN, а шесть типов коррозии (т. е. Gin, Gex, Pin, Pex, PICin, и PICex) являются выходными данными.

4.1.2. Предлагаемая

структура. Предлагаемая классификационная модель CNN в этом исследовании включает в общей сложности 20 слоев: уровни с 1 по 12 предназначены для извлечения признаков, а уровни с 13 по 20 ориентированы на классификацию. Слой 20 служит выходным слоем, состоящим из шести компонентов, которые представляют шесть типов коррозии. Подробная информация о каждом слое сведена в Т
Как показано в таблице 2, размер фильтра в сверточном слое выбран равным (5 × 5), что является типичным размером для больших фильтров [42]. Размер фильтра в слое maxpooling составляет (2 × 2), что соответствует размерам, обычно используемым в существующей литературе [43,44]. Чтобы избежать переобучения, в модель включены несколько слоев исключения, которые случайным образом исключают определенный процент нейронов из нейронной сети во время процесса обучения. Используемый коэффициент отсева составляет 0,2, т.е. 20% нейронов в нейронной сети случайно деа
На рисунке 7 изображена архитектура предлагаемой модели классификации с размерами каждого слоя, где входные слои, слои свертки и слои максимального пула изображены розовым, синим и зеленым цветом соответственно. Обратите внимание, что исключаемые слои не показаны на рисунке 7, поскольку они не влияют на размерность данных. Толстая белая линия представляет сплющенный слой, полностью связанные слои представлены оранжевым цветом, а выходные слои показаны желтым цветом на рисунке 7.

Machine Translated by Google

Таблица 2. Информация о слое модели классификации CNN.

Часть	Номер слоя	Имя слоя	Параметры
Извлечение признаков	1	Свертка	64 фильтра размером (5 × 5)
	2	Макспулинг	Размер пула равен (2 × 2)
	3	Выбывать	Ставка = 0,2
	4	Свертка	128 фильтров размером (5 × 5)
	5	Макспулинг	Размер пула равен (2 × 2)
	6	Выбывать	Ставка = 0,2
	7	Свертка	256 фильтров размером (5 × 5)
	8	Макспулинг	Размер пула равен (2 × 2)
	9	Выбывать	Ставка = 0,2
	10	Свертка	512 фильтров размером (5 × 5)
	11	Макспулинг	Размер пула равен (2 × 2)
	12	Выбывать	Ставка = 0,2
Классификация	13	Сгладить слой	Номер единицы равен 51 200
	14	Полностью связный слой	Номер единицы равен 128
	15	Выбывать	Ставка = 0,2
	16	Полностью связный слой	Номер единицы равен 64
	17	Выбывать	Ставка = 0,2
	18	Полностью связный слой	Номер единицы равен 32
	19	Выбывать	Ставка = 0,2
	20	Выход 1	Номер единицы равен 1
		Выход 2	Номер единицы равен 1
		Выход 3	Номер единицы равен 1
		Выход 4	Номер единицы равен 1
		Выход 5	Номер единицы равен 1
		Выход 6	Номер единицы равен 1

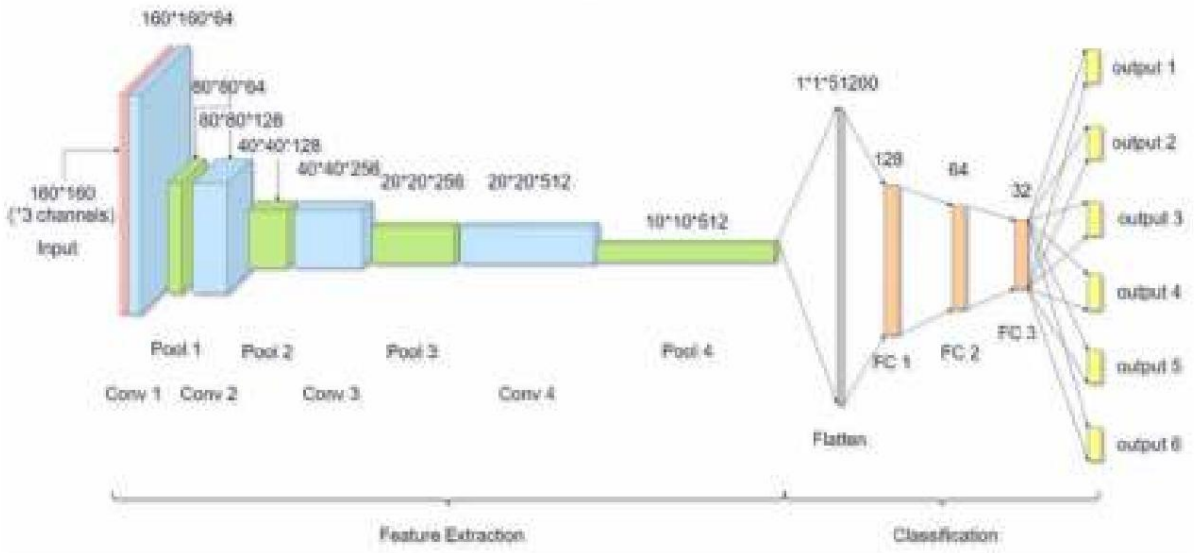


Рисунок 7. Предлагаемая структура модели классификации CNN.

Machine Translated by Google

4.1.3. Результаты

Если принять соотношение данных обучения и тестирования 80–20%, то получится 1280 учебных случаев и 320 тестовых случаев для каждого из шести типов коррозии. Результаты классификации, полученные из предложенная модель CNN обобщена в таблице 3. Таблица 3 показывает, что предлагаемая модель модель последовательно достигает точности прогнозирования выше 0,93 как при обучении, так и при обучении. и наборы тестовых данных, с самой низкой точностью 0,934, наблюдаемой в наборе обучающих данных для Гекс. Отмечается, что модель работает лучше в тестовом наборе данных, чем в обучающем наборе данных. со степенью точности каждого типа коррозии в наборе тестовых данных выше или равной ставки в наборе обучающих данных. Это может быть связано с методами регуляризации, такими как отсев, принятый в модели CNN. Хотя эти методы могут предотвратить переобучение, они также могут создавать шум или неопределенность во время тренировки, что может временно ухудшить производительность модели на обучающих данных, одновременно улучшая обобщение и производительность на тестовых данных [45].

Таблица 3. Результаты классификации модели CNN.

№ категории	Категория	Всего поездов		Тест	Неправильный прогноз в поезде	Неправильный прогноз в тесте	Точность теста точности поезда	
0	Джин	1600	1280	320	7	0	0,978	1.000
1	Гекс	1600	1280	320	21	0	0,934	1.000
2	Приколоть	1600	1280	320	0	0	1.000	1.000
3	Пекс	1600	1280	320	2	0	0,994	1.000
4	ПИКин	1600	1280	320	1	0	0,997	1.000
5	PICex	1600	1280	320	1	1	0,997	0,997

Примечательно, что 32 случая в наборе обучающих данных и один случай в наборе тестовых данных были ошибочно классифицированы, в результате чего уровень ошибочных прогнозов составляет 0,34% (33/9600 = 0,34%), демонстрируя отличная точность разработанной модели. Рисунок 8 иллюстрирует распределение неправильные прогнозы в наборе обучающих данных по разным категориям. Например, 65,6% (т.е. 21) ошибочно классифицированных учебных случаев относятся к категории Gex . Среди них четыре случая ошибочно классифицированы как Pin, а 17 случаев ошибочно классифицированы как Pex. Кроме того, шесть из В 33 случаях неверно классифицированы местоположения (внутренняя/внешняя поверхность трубы), т.е. два Pex случаи, ошибочно классифицированные как Pin, и четыре случая Gex , ошибочно классифицированные как Pin.

Обратите внимание, что модель обеспечивает более высокую точность классификации для Pin, Pex, PICin и PICex. чем для Джина и Гекса. Значения dg/t , включенные во весь набор данных для всех шести типов коррозия составляет 20, 30, 40, 60 и 80%, а значения dp/t включены во весь набор данных составляют 20, 30, 40, 50, 55, 60, 70 и 80%. Более широкий диапазон глубины коррозии корпусов с участием точечных отверстий (т. е. Pin, Pex, PICin и PICex) по сравнению со случаями, связанными с общими отверстиями. коррозия (Gin и Gex) позволяет модели изучить более богатый набор функций и вариаций. связана с точечной коррозией, что делает модель более подходящей для различения различные типы и степень тяжести точечной коррозии. Все 28 неверно классифицированных случаев в Категории «Джин» и «Гекс» характеризуются относительно неглубокой коррозией, среди которых 26 случаев имели место. dg/t = 20%, а в двух случаях dg/t = 40%. Распределение 28 неверно классифицированных случаев по wg/lg показано на рисунке 9. Рисунок 9 показывает, что модель классификации CNN менее точен для обширной общей коррозии с wg/lg более 1,67. Также примечательно, что разработанная модель менее точна для случаев, содержащих внешние коррозии (т.е. Gex, Pex, и PICex) , чем в случаях внутренней коррозии (т. е. Gin, Pin и PICin). Потенциал Объяснение заключается в том, что входные данные модели (т. е. карта сигналов MFL) для внешних дефектов имеют меньше ярко выраженные особенности, поскольку изменение B из-за дефекта потерь металла на внешнем поверхность трубы меньше, чем вызванная внутренним дефектом потери металла, так как внешний дефект находится дальше от датчика, чем внутренний дефект.

Machine Translated by Google

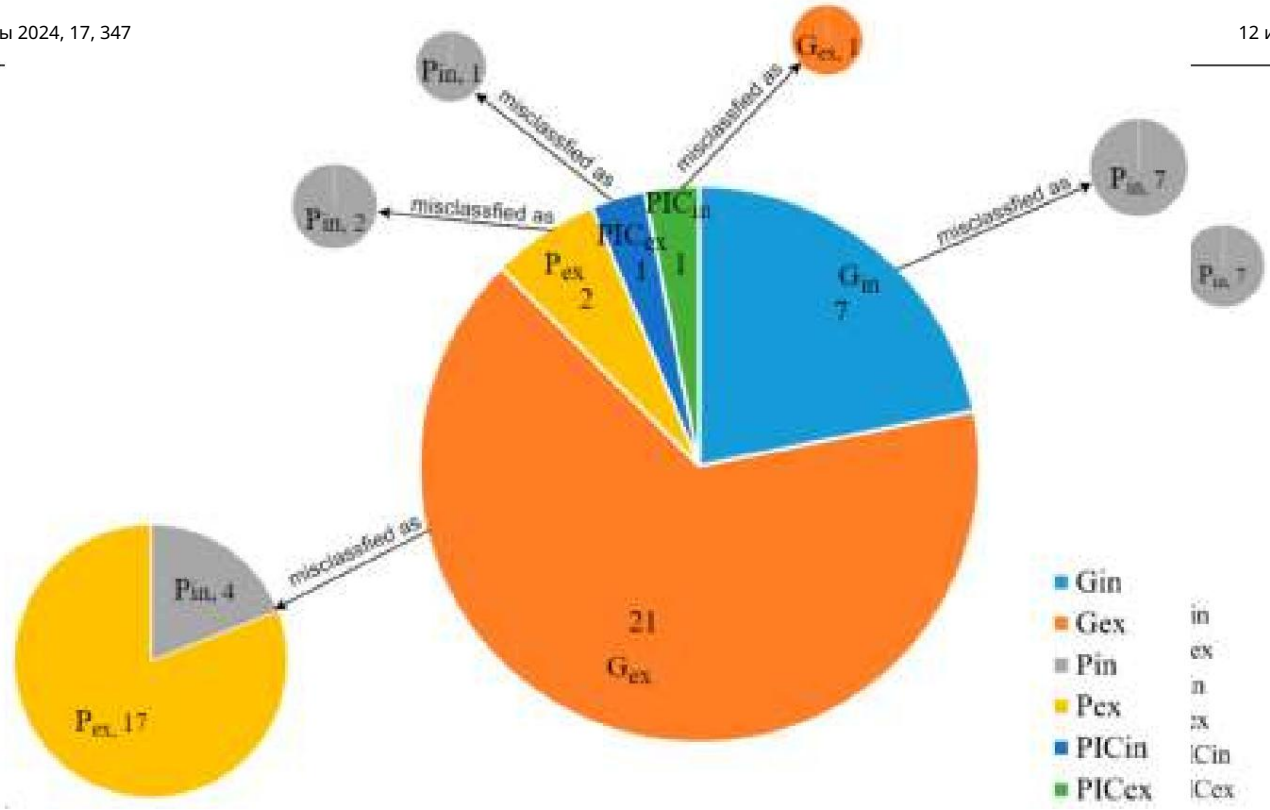


Рисунок 8. Распределение ошибок классификации в наборе обучающих данных.

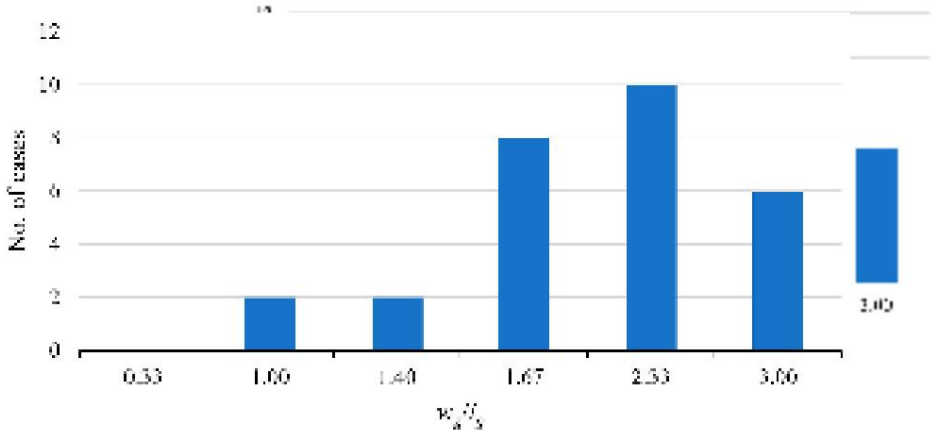


Рисунок 9. Распределение ошибочно классифицированных случаев в Джин и Гекс по wg/lg.

4.2. Регрессия CNN 4.2.1.

Входная информация.

Результаты модели CNN классификации используются для извлечения правильно классифицированных случаев для Pin, Pex, PICin и PICex, исключая четыре ошибки классификации в наборе обучающих данных φ и одну неправильную классификацию в наборе тестовых данных, что приводит к 1600, 1598, 1599, и 1598 входных вариантов для Pin, Pex, PICin и PICex соответственно для обучения и проверки регрессионной модели CNN. Три сгенерированные матрицы безшумовых сигналов MFL размером 160 × 160 для каждого из 6395 (= 1600 + 1598 + 1599 + 1598) случаев являются входными данными для разработанной регрессионной модели, а прогнозируемые значения rp, dp/t, h и Значения φ для каждого случая являются выходными данными. Правильно классифицированные данные используются для обучения регрессионной модели CNN, поскольку включение в обучение неправильно классифицированных данных (таких как общая коррозионная аномалия, ошибочно классифицированная как точечное отверстие) в обучении проблематично, например, определение функции ошибок, применимой ко всем входным данным, включая неправильно классифицированные данные. . В качестве альтернативы обучение регрессионной модели CNN можно полностью отделить от модели классификации; то есть все 6400 случаев параметрического анализа методом конечных элементов, включающих точечные отверстия, могут быть использованы для обучения регрессионной модели. В этом исследовании правильно классифицированный набор данных, который состоит из 6395 случаев и почти идентичен полному набору данных, используе

Machine Translated by Google

4.2.2. Предлагаемая структура

Предлагаемая регрессионная модель CNN в настоящем исследовании включает в общей сложности 23 слоя: слои с 1 по 16 предназначены для извлечения признаков, а слои с 17 по 23 ориентированы на регресс. Слой 23 — это выходной слой, который включает в себя четыре компонента, представляющие прогнозируемые значения rp, dp/t, h и φ соответственно. Подробная регрессионная модель CNN информация о слоях представлена в таблице 4. Отмечается, что помимо выпадающих слоев представленный в разделе 4.1.2, в него включено несколько слоев нормализации партии. типичные слои CNN для предотвращения переобучения [46–48]. Как показано в Таблице 4, настроенный гиперпараметры в процессе извлечения признаков регрессионной модели, такие как размер фильтра слоя свертки, размер пула и частота отсева такие же, как те, которые указаны в классификационной модели CNN, а коэффициент отсева в регрессии процесс регрессионной модели CNN равен 0,5. На рис . 10 изображена архитектура разработала регрессионную модель с размерами разных слоев. Обратите внимание, что отсев и слои пакетной нормализации исключены из диаграммы, поскольку они не влияют на размерность данных. Цветовая схема, используемая для различения различных слоев на рисунке 10 , следующая: соответствуют тем, которые используются на рисунке 7.

Таблица 4. Информация о слое регрессионной модели CNN.

Часть	Номер слоя	Имя слоя	Параметры
Извлечение признаков	1	Свертка	64 фильтра размером (5 × 5)
	2	Пакетная нормализация	
	3	Макспулинг	Размер пула равен (2 × 2)
	4	Выбывать	Ставка = 0,2
	5	Свертка	128 фильтров размером (5 × 5)
	6	Пакетная нормализация	
	7	Макспулинг	Размер пула равен (2 × 2)
	8	Выбывать	Ставка = 0,2
	9	Свертка	256 фильтров размером (5 × 5)
	10	Пакетная нормализация	
	11	Макспулинг	Размер пула равен (2 × 2)
	12	Выбывать	Ставка = 0,2
	13	Свертка	512 фильтров размером (5 × 5)
	14	Пакетная нормализация	
	15	Макспулинг	Размер пула равен (2 × 2)
	16	Выбывать	Ставка = 0,2
Регрессия	17	Сгладить слой	Номер единицы равен 51 200
	18	Выбывать	Ставка = 0,5
	19	Полностью связный слой	Номер единицы равен 64
	20	Полностью связный слой	Номер единицы равен 128
	21	Полностью связный слой	Номер единицы равен 256
	22	Полностью связный слой	Номер единицы равен 512
	23	Выход 1	Номер единицы равен 1
		Выход 2	Номер единицы равен 1
		Выход 3	Номер единицы равен 1
		Выход 4	Номер единицы равен 1

Machine Translated by Google

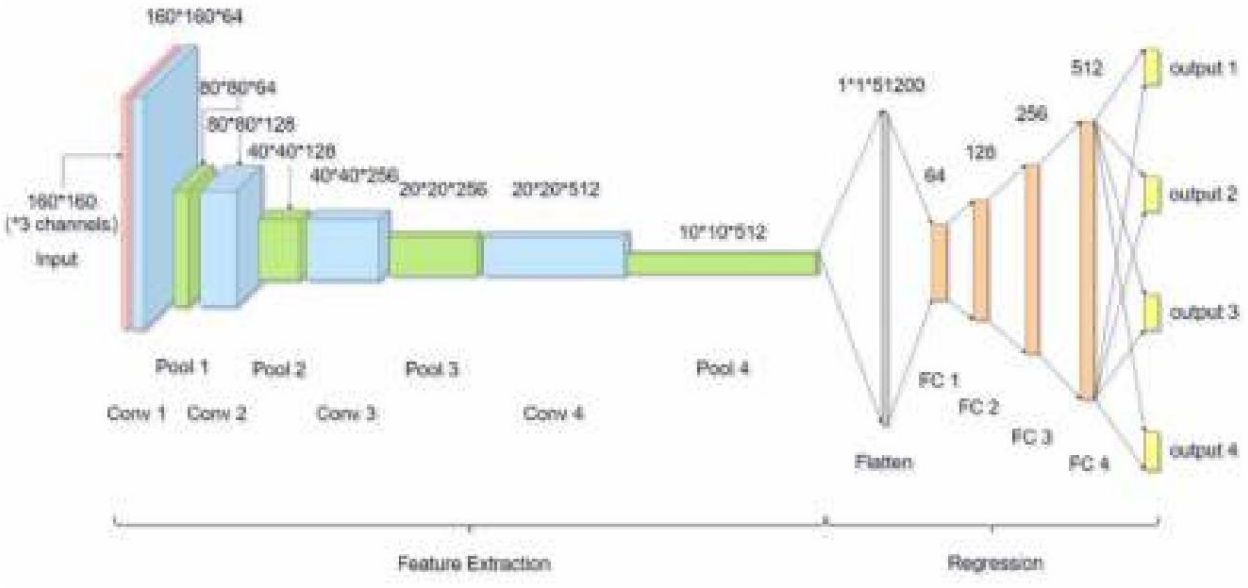


Рисунок 10. Предлагаемая структура регрессионной модели CNN.

4.2.3. Результаты

Коэффициент детерминации (т.е. R ffi 2) используется для измерения точности прогнозирования каждого параметра (т. е. rp, dp/t, h и φ), определяемого следующим уравнением.

$$p \triangleq 1 - \frac{\sum_{i=1}^{H} (Ytrue,i - Ypred,i)^2}{\sum_{i=1}^{H} (Ytrue,i - \overline{Ytrue})^2} \quad (4)$$

где Ytrue,i и Ypred,i представляют истинные и прогнозируемые значения соответственно каждого предсказанного параметра для i-й (i = 1, 2, … , N) точки в заданном наборе данных, содержащем N точек данных , и Ytrue обозначает истинное среднее значение N точек данных, связанных с прогнозируемым параметром. В таблице 5 суммированы значения четырех параметров в наборе тестовых данных также значения R, включенные в регрессионную модель. R ф 2 на рисунке 11. Истинные значения четырех параметров rp, dp/t, h и φ по сравнению с оценочными значениями CNN для коррозии в испытательном наборе изображены на рисунке 12. Это видно из таблицы 5 , что прогнозы модели CNN параметров места коррозии (т.е. h и φ) показывают сильное согласие с фактическими значениями, что отражено значениями R ф 2 , равными 1,0 для всех четырех категорий коррозии. С точки зрения прогнозирования размеров (т.е. rp и dp/t) модель более точна для внутренней коррозии (Pin и PICin), чем для внешней коррозии (Pex и PICex). Эти результаты также показывают, что производительность модели лучше для случаев, содержащих только точечные отверстия, чем для случаев, содержащих PIC. Это ожидаемо, поскольку PIC вносит дополнительную сложность, включая наличие общей коррозии, которая может повлиять на точность модели. В целом получена точность R при прогнозировании размерных параметров, при этом более высокая точность наблюдается для коррозии 2 значения демонстрируют модель CNN на внутренней поверхности и микроотверстий. Однако обратите внимание, что хотя прогнозы для rp и dp/t в целом точны, на рисунке 12 показаны относительно большие диапазоны ошибок как в прогнозах rp , так и в dp/t . Например, прогнозируемые значения rp варьируются от 1,14 мм до 3,45 мм для (rp)true, равного 2 мм. Большие диапазоны ошибок могут быть объяснены малыми размерами точечной коррозии и ограниченным пространственным разрешением модели. Кроме того, наличие элементов коррозии композита может внести дополнительные сложности и неопределенности, что приводит к увеличению диапазона ошибок в прогнозах. Поэтому необходимы дальнейшие исследования для изучения факторов, способствующих уменьшению ошибок регрессии для точечной коррозии.

Machine Translated by Google

Таблица 5. Результаты регрессионной модели CNN без шума.

Категория №.	Категория	Случаи			Тест R^2			
		Общий	Тренироваться	Тест	рп	дп/т	час	φ
0	Прикоснуться	1600	1280	320	0,91	0,95	1.00	1.00
1	Пекс	1598 г.	1277	321	0,88	0,92	1.00	1.00
2	ПИКин	1599 г.	1280	319	0,89	0,85	1.00	1.00
3	PICex	1598 г.	1279	319	0,86	0,77	1.00	1.00
Общий		6395	5116	1279	0,89	0,91	1.00	1.00

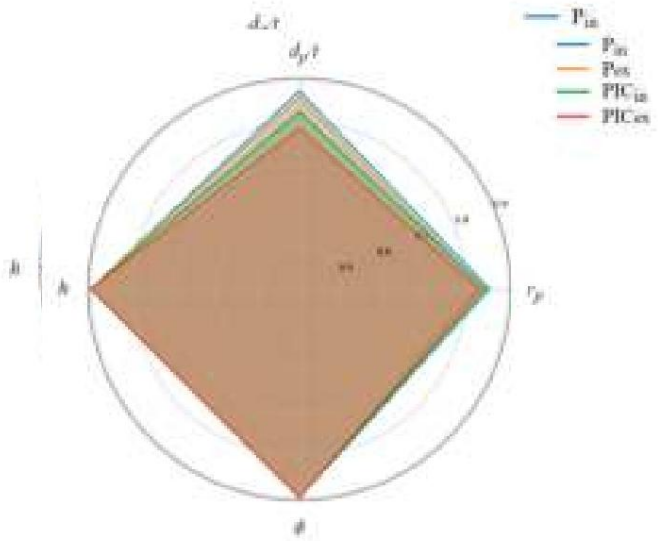


Рисунок 11. Радарная диаграмма для R^2 показателей (rp, dp/t, hφ и φ) для коррозии Pin, Pex, PICin и PICex в тестовый набор данных.

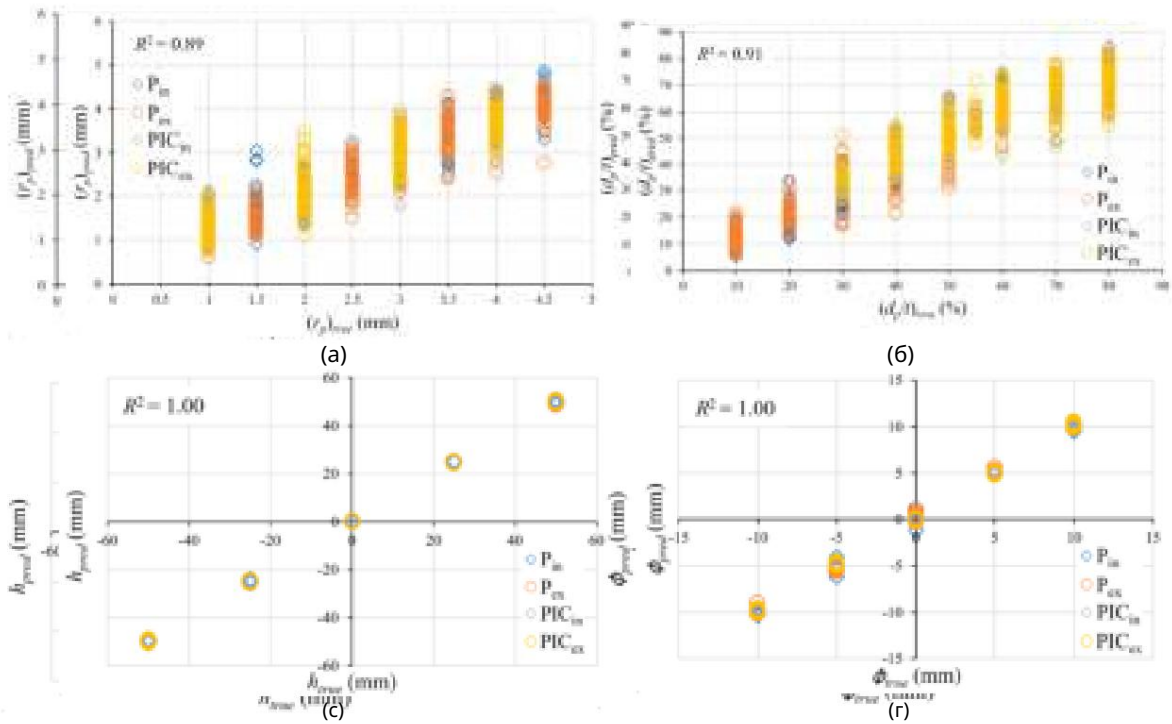


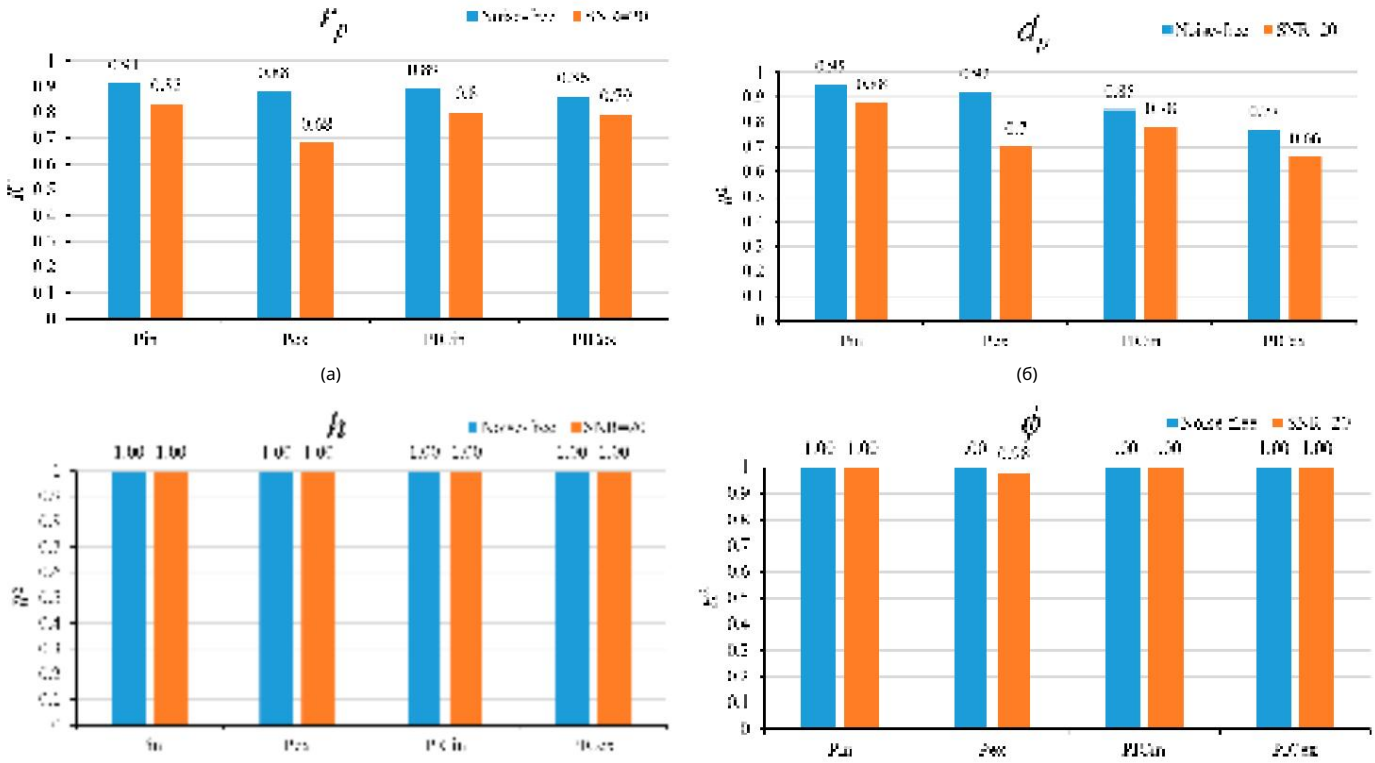
Рис. 12. Сравнение истинных и прогнозируемых значений rp, dp/t, h и φ для аномалий в набор данных регрессионного теста без шума.

Machine Translated by Google

4.2.4. Влияние шума

На рисунке 13 сравниваются результаты регрессионной модели CNN для зашумленного входного сигнала (SNR = 20) с результатами сценария без шума, описанного в разделе 4.2.3. При наличии шума 2 значения R для прогнозирования rp и dp/t несколько уменьшаются по сравнению со значениями без шума. ff более заметен для случаев Pex и PICex , что указывает на то, что модель больше подвержена влиянию шума при прогнозировании сценарий. Уменьшение R^2 размеров внешняя коррозия.

Несмотря на снижение точности прогнозирования размеров коррозии, модель сохраняет 2 R = 1 для параметров местоположения (h и φ) для всех типов коррозии, кроме Pex, что указывает на ее устойчивость при сборе пространственной информации даже при наличии шума. .



(а) (б)

Рисунок 13. Сравнение R^2 между сценариями без шума и SNR = 20 для разных типов коррозии.

Наконец, важно отметить, что модели классификации и регрессии CNN , разработанные в этом исследовании, специально настроены на определенные атрибуты трубопровода и инструмента MFL, такие как толщина стенки трубы, магнитные свойства, значения отрыва и формы дефектов. Таким образом , необходимы дальнейшие исследования, чтобы определить, насколько хорошо модель CNN работает с различной толщиной стенок трубы и различными параметрами инструмента MFL. Также необходимы дальнейшие исследования , чтобы учесть перемещение инструмента MFL в модели прогнозирован

5. Выводы

В этой статье сообщается о новом исследовании, посвященном классификации и определению размеров точечной коррозии на трубопроводах с использованием алгоритмов глубокого обучения. Мы предлагаем классификационную модель CNN для классификации шести различных типов коррозии (т.е. Gin, Gex, Pin, Pex, PICin и PICex) на трубопроводах, а также регрессионную модель CNN для оценки размеров и расположения точечных дефектов на основе сигналов MFL. генерируется с помощью магнитостатического МКЭ. Для моделирования сигналов 3D MFL путем изменения глубины дефекта, длины, ширины, а также продольного и окружного расположения используются обширные трехмерные параметрические случаи FEA, включающие коробчатую общую коррозию и точечные отверстия цилиндрической формы . Затем модели классификации и регрессии CNN обучаются и проверяются с использованием смоделированных сигналов MFL.

Показано, что предложенная модель классификации имеет превосходную точность при классификации шести типов коррозионных дефектов с уровнем ошибочного прогнозирования 0,34% (33/9600 = 0,34%).

Machine Translated by Google

Предлагаемая регрессионная модель очень точна в прогнозировании местоположения точечного отверстия и обеспечивает хорошую точность прогнозирования глубины и диаметра точечного отверстия. Путем сравнительного анализа модели регрессии CNN с бесшумными и зашумленными сигналами (SNR = 20) в качестве входных данных видно, что влияние шума на точность прогнозирования модели регрессии является умеренным. Это исследование демонстрирует применение алгоритмов глубокого обучения для облегчения управления целостностью трубопроводов, содержащих коррозионные дефекты сложной формы.

Вклад автора: YS: Исследование, визуализация, формальный анализ, методология, написание — оригинальный проект. WZ: Привлечение финансирования, надзор, концептуализация, методология, написание — обзор и редактирование. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: Авторы признают финансовую поддержку, полученную от Совета естественных наук и инженерных исследований Канады (NSERC) (грант № RGPIN-2019–05160) и инженерного факультета Университета Западного Онтарио.

Заявление о доступности данных: данные доступны по запросу у авторов.

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Ссылки

1. Чжоу, К.; Ву, В.; Лю, Д.; Ли, К.; Цяо, К. Оценка вероятности коррозионного разрушения нефтегазопровода на основе подхода нечеткой логики. англ. Неудача. Анальный. 2016, 70, 48–55. [Перекрестная ссылка]
2. Мерфи, Дж. Ф. Кошмарные сбои в конвейере, фантастическое планирование, черные лебеди и управление целостностью — обзор. Процесс Саф. Прог. 2015, 34, 207. [CrossRef]
3. Шен Ю.; Чжоу, В. Сравнение статистики происшествий на береговых нефте- и газопроводах в Канаде и США. Межд. Дж. Крит. Инфраструктура. Прот. 2024, 45, 100679. [CrossRef]
4. Ванаи, HR; Эслами, А.; Эгбеванде, А. Обзор моделей коррозии трубопроводов, внутритрубного контроля (ILI) и скорости роста коррозии . Межд. Дж. Пресс. Сосуды Пип. 2017, 149, 43–54. [Перекрестная ссылка]
5. Сонг, Х.; Ян, Л.; Лю, Г.; Тиан, Г.; Она, Д.И.; Сонг, Ю.; Ли, С. Сравнительный анализ оборудования для линейного контроля и технологии. Конференция ИОП. Сер. Матер. наук. англ. 2018, 382, 032021. [CrossRef]
6. Форум операторов трубопроводов. Технические условия и требования к внутритрубному контролю трубопроводов. Версия 2021. Доступно онлайн: https://pipelineoperators.org/documents (по состоянию на 18 мая 2022 г.).
7. Сазерленд, Дж.; Блюк, М.; Пирс, Дж.; Квик, Э. Валидация технологии линейного контроля MFL последнего поколения приводит к улучшению обнаружения и определения размеров точечных отверстий, точечной коррозии, осевых канавок и осевых прорезей. В материалах 8-й Международной конференции по трубопроводам ASME 2010 , Калгари, АВ, Канада, 27 сентября – 1 октября 2010 г.
8. Пэн, Х.; Аньяоха, У.; Лю, З.; Цукада, К. Анализ данных утечки магнитного потока (MFL) для оценки коррозии трубопроводов. IEEE Транс. Магн. 2020, 56, 1–15. [Перекрестная ссылка]
9. Хан, В.; Ян, П.; Ся, Ф.; Сюэ, Ю. Инверсия сигнала утечки магнитного потока коррозионных дефектов на основе модифицированного генетического локального поиска алгоритм. J. Шанхайский университет Цзяотун. наук. 2009, 14, 168–172. [Перекрестная ссылка]
10. Привальд, Р.Х.; Магеле, К.; Леджер, PD; Пирсон, Северная Каролина; Мейсон, Дж.С.Д. Быстрая инверсия сигнала рассеяния магнитного потока для восстановления произвольных профилей дефектов в стали с использованием конечных элементов. IEEE Транс. Магн. 2012, 49, 506–516. [Перекрестная ссылка]
11. Хван, К.; Мандаям, С.; Удпа, С.С. Характеристика сигналов контроля газопровода с использованием нейронной вейвлет-функции сети. НК Е Int. 2000, 33, 531–545. [Перекрестная ссылка]
12. Рамухалли, П.; Удпа, Л.; Удпа, С.С. Алгоритмы инверсии на основе нейронных сетей для неразрушающей оценки рассеяния магнитного потока. Дж. Прил. Физ. 2003, 93, 82748276. [CrossRef]
13. Хан, WH; Que, PW 2-D реконструкция дефектов по сигналам MFL на основе алгоритма генетической оптимизации. В материалах Международная конференция IEEE 2005 по промышленным технологиям, Гонконг, Китай, 14–17 декабря 2005 г.
14. Кандрооди, MR; Арааби, Б.Н.; Бассири, ММ; Ахмадабади, М.Н. Оценка глубины и длины дефектов на основе измерений рассеяния магнитного потока: проверка с помощью моделирования, экспериментов и данных скребков. IEEE Транс. Магн. 2016, 53, 1–10. [Перекрестная ссылка]
15. Фэн, Дж.; Ли, Ф.; Лу, С.; Лю, Дж. Быстрая реконструкция профилей дефектов на основе измерений рассеяния магнитного потока с использованием методологии корректировки ошибок на основе RBFNN. ИЭПП Науч. Измер. Технол. 2017, 11, 262–269. [Перекрестная ссылка]
16. Яо, Г.; Лей, Т.; Чжун, Дж. Обзор распознавания действий на основе сверточных нейронных сетей. Распознавание образов. Летт. 2019, 118, 14–22. [Перекрестная ссылка]
17. Тайе М.М. Теоретическое понимание сверточной нейронной сети: Концепции, архитектуры, приложения, будущие направления. Расчет 2023, 11, 52. [CrossRef]
18. Лу, С.; Фэн, Дж.; Чжан, Х.; Лю, Дж.; Ву, З. Метод оценки размера дефекта по изображению MFL с использованием визуального преобразования. сверточная нейронная сеть. IEEE Транс. Инди. Информ. 2018, 15, 213–224. [Перекрестная ссылка]

Machine Translated by Google

19. Шен Ю.; Чжоу, В. Разработка модели сверточной нейронной сети для прогнозирования размера и местоположения коррозионных дефектов на трубопроводов на основе сигналов рассеяния магнитного потока. Межд. Дж. Пресс. Сосуды Пип. 2023, 207, 105–123. [Перекрестная ссылка]

20. Ван, ХА; Чен, Г. Метод оценки размера дефекта для сигналов утечки магнитного потока с использованием сверточных нейронных сетей. Инсайт 2020, 62, 86–91. [Перекрестная ссылка]

21. ЭГИГ. 10-й отчет Европейской группы по сбору данных о происшествиях на газопроводах (период 1970–2016 гг.); EGIG: Гронинген, Нидерланды, 2018 г.; Док. номер ВА 17.P.0395.

22. Чжан Х.; Ша, С.; Уиллис, К.; Циншань, Ф.; Чен, П. Технико-экономическое обоснование контроля точечных отверстий с помощью утечки магнитного потока и гидростатические испытания нефте- и газопроводов. Конференция ИОП. Сер. Матер. наук. англ. 2021, 1043, 022053.

23. Фэн, К.; Ян, Б.; Чен, П.; Ширази С.А. Анализ отказов и имитационная модель точечной коррозии нефтепровода нефтепровода. англ. Неудача. Анальный. 2019, 106, 1–28. [Перекрестная ссылка]

24. Субраманиан, К. Локализованная питтинговая коррозия труб класса А API 5L, используемых в промышленных трубопроводах противопожарной воды. англ. Неудача. Анальный. 2018, 92, 405–417. [Перекрестная ссылка]

25. Аскари, М.; Алиофхазраи, М.; Афрухтех, С. Комплексный обзор внутренней коррозии и растрескивания нефте- и газопроводов. Дж. Нэт. Газовая наука. англ. 2019, 71, 102971. [CrossRef]

26. Кадхим, КН; Аль-Руфайе, АНР Влияние однородного поперечного магнитного поля на локальный профиль потока и скорости. Межд. Дж. Сив. англ. Технол. 2016, 7, 140–151.

27. Джи, Ф.; Ван, К.; Солнце, С.; Ван, В. Применение 3D FEM при моделировании сигналов MFL. Инсайт 2009, 51, 32–35. [Перекрестная ссылка]

28. Бубеник Т. Электромагнитные методы обнаружения коррозии в подземных трубопроводах: утечка магнитного потока (MFL); Метро Коррозия трубопроводов; Woodhead Publishing: Кембриджшир, Великобритания, 2014 г.; стр. 215–226.

29. Ши, Ю.; Чжан, К.; Ли, Р.; Кай, М.; Цзя, Г. Теория и применение обнаружения утечки магнитного потока в трубопроводе. Датчики 2015, 15, 31036–31055. [Перекрестная ссылка]

30. Чжан Ю.; Йе, З.; Ван, К. Быстрый метод реконструкции размеров прямоугольных трещин при испытаниях на утечку магнитного потока. НК Е Int. 2009, 42, 369–375. [Перекрестная ссылка]

31. Уокер Дж. Внутритрубная инспекция трубопроводов: передовые технологии экономичной и безопасной эксплуатации нефтегазопроводов; Verlag Moderne Industrie: Ландсберг-ам-Лех, Германия, 2010 г.

32. Лю, Ю.; Гао, Х.; Ван, Ю.; Ян, Х. Оптимизация чувствительных параметров механизма поддержки постоянных магнитов. Дж. Мех. наук. Технол. 2014, 28, 2707–2714. [Перекрестная ссылка]

33. Ян, Л.; Чжан, Г.; Лю, Г.; Гао С. Влияние отрыва на проверку утечки магнитного потока в трубопроводе. В материалах 17-го заседания 2008 г. Всемирная конференция по неразрушающему контролю, Шанхай, Китай, 25–28 октября 2008 г.

34. Ирландия, Республика Колорадо; Торрес, К.Р. Конечно-элементное моделирование окружного намагничивателя. Сенсорные приводы А Phys. 2006, 129, 197–202. [Перекрестная ссылка]

35. Азиззаде, Т.; Сафизаде, М.С. Трехмерный метод конечных элементов и экспериментальное моделирование неразрушающего контроля типа рассеяния магнитного потока для обнаружения питтинговой коррозии. В материалах 4-й Иранской международной конференции по неразрушающему контролю (IRNDT) 2017 г., Тегеран, Иран, 21–22 февраля 2017 г.

36. Чен, Л.; Ли, Х.; Цинь, Г.; Лу, К. Обработка сигналов контроля поверхностных дефектов рассеяния магнитного потока в стали трубопроводов. Расс. Дж. Неразрушать. Тест. 2008, 44, 859–867. [Перекрестная ссылка]

37. Пяо, Г.; Го, Дж.; Хижина.; Люнг, Х. Влияние вихревых токов, индуцированных движением, на высокоскоростную утечку магнитного потока (MFL) осмотр толстостенных стальных труб. Рез. Неразрушать. Эваль. 2020, 31, 48–67. [Перекрестная ссылка]

38. Ли, Ф.; Фэн, Дж.; Чжан, Х.; Лю, Дж.; Лу, С.; Ма, Д. Быстрая реконструкция произвольных профилей дефектов трубопровода по измерениям MFL с использованием модифицированного алгоритма поиска гармонии. IEEE Транс. Инструмент. Измер. 2018, 67, 9. [CrossRef]

39. Фан, Х.; Дай, М.; Лю, К.; Ву, Ф.; Ян, Х.; Фэн, Ю.; Фэн, Ю.; Су, Б. Влияние шума изображения на классификацию поражений кожи с помощью глубокие сверточные нейронные сети. Цинхуа наук. Технол. 2019, 25, 425–434. [Перекрестная ссылка]

40. Ачарья, УР; О, СЛ; Хагивара, Ю.; Тан, Дж. Х.; Адам, М.; Гертыч А.; Сан Тан, Р. Модель глубокой сверточной нейронной сети для классификации сердцебиений. Вычислить. Варить. Мед. 2017, 89, 389–396. [Перекрестная ссылка] [ПабМед]

41. Тесфаи, Х.; Салех, Х.; Аль-Кутайри, М.; Мохаммед, МБ; Текесте, Т.; Хандокер, А.; Мохаммад, Б. Облегченный cnn на основе шаффлнета для классификации аритмии. Доступ IEEE. 2022, 10, 111842–111854. [Перекрестная ссылка]

42. Запад, Северная Каролина; О'ши, Т. Глубокие архитектуры для распознавания модуляции. В материалах Международного симпозиума IEEE 2017 г. о сетях динамического доступа к спектру (DySPAN), Балтимор, Мэриленд, США, 6–9 марта 2017 г.

43. Баранвал, С.; Хандельвал, С.; Арора, А. Сверточная нейронная сеть глубокого обучения для обнаружения болезней листьев яблони. В материалах Международной конференции по устойчивым вычислениям в науке, технологиях и менеджменте 2019 года (SUSCOM), Университет Эмити, Раджастхан, Джайпур, Индия, 26–28 февраля 2019 г.

44. Вирупакшаппа, К.; Марино, М.; Оруклу, Э. Архитектура сверточной нейронной сети с несколькими разрешениями для ультразвуковой дефектоскопии . В материалах Международного симпозиума по ультразвуковой технике (IUS) IEEE 2018 г., Кобе, Япония, 22–25 октября 2018 г.

45. Билмес Дж. Недостаточное и переобучение в машинном обучении. Конспекты курса UW ECE. 2020. Доступно онлайн: https://people.ece. uw.edu/bilmes/classes/ee511/ee511_spring_2020/overfitting_underfitting.pdf (по состоянию на 20 марта 2023 г.).

46. Иоффе С.; Сегеди, К. Пакетная нормализация: ускорение глубокого обучения сети за счет уменьшения внутреннего сдвига ковариат. В материалах Международной конференции по машинному обучению 2015 г., Лилль, Франция, 6–11 июля 2015 г.

Machine Translated by Google

47. Таккар, В.; Тьюари, С.; Чакраборти, К. Пакетная нормализация в сверточных нейронных сетях — сравнительное исследование с данными CIFAR-10. В материалах Пятой международной конференции IEEE 2018 по новым приложениям информационных технологий (EAIT), IIEST Шибпур, Западная Бенгалия, Индия, 12–13 января 2018 г.
48. Бьорк, Н.; Гомес, КП; Селман, Б.; Вайнбергер, К.К. Понимание нормализации партий. Адв. Нейронная инф. Процесс. Сист. 2018, 31, 1–12.

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.

Machine Translated by Google

алгоритмы

MDPI

Статья

Задача планирования работы параллельных машин с разными скоростями и временем выпуска при транспортировке руды

Луис Таразона-Торрес, Чиро Амайя *, Альваро Пайпилья, Камило Гомес и Давид Альварес-Мартинес

Факультет промышленной инженерии Университета Лос-Андес, Cra 1 N18A12, Богота 111711, Колумбия; le.tarazona@uniandes.edu.co (LT-T.); a.paipilla@uniandes.edu.co (AP); gomez.ch@uniandes.edu.co (CG); d.alvarezm@uniandes.edu.co (DA-M.)
* Переписка: ca.amaya@uniandes.edu.co.

Аннотация: Операции по доставке руды имеют решающее значение в горнодобывающей промышленности, поскольку они поставляют необходимые полезные ископаемые на производственные предприятия. Эти операции, выполняемые с использованием сложного и дорогостоящего оборудования, требуют тщательного планирования, чтобы гарантировать достижение производственных целей при оптимизации использования оборудования. В этом исследовании мы представляем алгоритм для определения минимального количества транспортировочного оборудования, необходимого для достижения цели по транспортировке руды. Для достижения этой цели была разработана математическая модель , рассматривающая ее как задачу планирования параллельных машин с разными скоростями и временем выпуска, с упором на минимизацию как времени завершения, так и затрат, связанных с использованием оборудования. Кроме того, был разработан еще один алгоритм, позволяющий тактическую оценку этих двух переменных. Эти процедуры и модель вносят значительный вклад в работу лиц, принимающих решения, обеспечивая систематический подход к распределению ресурсов, гарантируя, что погрузочно-транспортное оборудование используется в полной мере, соблюдая при этом бюджетные ограничения и графики работы. Такой подход оптимизирует использование ресурсов и повышает операционную эффективность, способствуя постоянному совершенствованию горнодобывающих операций.

Ключевые слова: задача планирования параллельных машин; оборудование для перевозки руды; математическая модель; горнодобывающая промышленность

check for updates

Образец цитирования: Таразона-Торрес, Л.; Амайя, К.; Пайпилла, А.; Гомес, К.; Альварес-Мартинес, Д. Проблема планирования работы параллельных машин с разными скоростями и временем выпуска при транспортировке руды. Алгоритмы 2024, 17, 348. https://doi.org/10.3390/a17080348

Академические редакторы: Александр Долги, Давид Лемуан, Мария И. Рестрепо и Фрэнк Вернер

Поступила: 2 мая 2024 г.
Пересмотрено: 24 июля 2024 г.
Принято: 30 июля 2024 г.
Опубликовано: 8 августа 2024 г.

Копирайт: © 2024 авторов. Лицензиат MDPI, Базель, Швейцария. Эта статья находится в открытом доступе. распространяется на условиях и условия Creative Commons Лицензия с указанием авторства (CC BY) (https://creativecommons.org/licenses/by/4.0/).

1. Введение

Одним из наиболее важных видов деятельности в горнодобывающей промышленности является транспортировка руды; Этот процесс предполагает перемещение обломков горной породы с места взрывных работ с помощью погрузочно-транспортной техники [1]. Затем рудные породы транспортируются на процесс дробления для уменьшения их размера перед отправкой на обогатительную фабрику, где, среди прочего, извлекаются ценные металлы, такие как медь, золото, молибден и алюминий [2–4]. На рис . 1 представлена схема описываемого процесса. Плохое планирование операций по транспортировке руды может снизить уровень производительности, повлиять на достижение производственных целей в установленные сроки и, как следствие, повлиять на эксплуатационные расходы. Для эффективного планирования операций по транспортировке руды необходимо учитывать несколько факторов , таких как количество доступного погрузочно- транспортного оборудования и его плановое обслуживание [5]. Также важен учет установленных производственных заданий, рабочего времени и топографических уровней. Другие важные аспекты включают содержание руды, количество доступных погрузочных эстакад для погрузочных операций и воздействие на окружающую среду [6].

Горнодобывающие компании имеют определенное количество погрузочно-транспортной техники, выделяемое исходя из производственных задач, установленных на конкретный период. Крайне важно определить необходимое количество этого оборудования для эффективной транспортировки рудной породы для контроля сопутствующих затрат, которые могут быть существенными в зависимости от времени, необходимого для достижения этих целей [7].

Знание минимального количества необходимого оборудования оптимизирует операции и облегчает тактический анализ. Этот анализ позволяет лучше принимать решения в случае непредвиденных обстоятельств, обеспечивая достижение производственных целей с наименьшими возможными затратами.

Machine Translated by Google



Рисунок 1. Схема перевозки руды.

Как показано на рисунке 1, процесс перевозки руды включает в себя назначение транспортировочного оборудования погрузочному оборудованию. Они не обязательно работают с одинаковой скоростью загрузки, поскольку некоторое оборудование имеет разную грузоподъемность. Загрузка осуществляется одновременно и описывается как система параллельных машин с разной скоростью. Поскольку количество транспортных средств ограничено, они используются столько раз, сколько необходимо для достижения производственного задания. Однако они недоступны во время транспортировки руды на процесс дробления, поскольку у них есть «время освобождения», прежде чем они снова станут доступны для перегрузки. Таким образом, этот сценарий можно адаптировать к проблеме параллельных машин с разной скоростью и временем выпуска.

Новизна данной работы заключается во включении различных скоростей и времени выпуска в математическую модель параллельного планирования машин при перевозке полезных ископаемых (PMSPOHO). Эти функции позволяют более реалистично и точно отображать рабочие условия в горнодобывающей среде. В этой модели решающую роль играет скорость погрузочной техники . Изменения скорости погрузки могут существенно повлиять на эффективность перевозки руды. Учитывая различные скорости загрузки, модель может более точно распределять ресурсы в соответствии с динамическим характером операций по добыче полезных ископаемых.

Это гарантирует, что оборудование с более быстрой загрузкой будет использоваться с максимальным потенциалом, в то время как более медленному оборудованию будут поручены задачи, которые лучше соответствуют его возможностям, оптимизируя общий рабочий процесс.

Включение скорости погрузочного оборудования в алгоритм планирования максимизирует использование оборудования и минимизирует время простоя транспортного оборудования. Это повышает операционную эффективность за счет сокращения задержек и обеспечения непрерывного потока материалов.

Разработанная математическая модель учитывает эти критические переменные, оптимизируя размещение транспортного оборудования, максимизируя его использование и сокращая время простоя. Такой подход обеспечивает большую эффективность работы за счет согласования возможностей оборудования с требованиями процесса добычи полезных ископаемых. Стратегическое распределение ресурсов, основанное на скорости и доступности каждой погрузочной машины, способствует более упорядоченной и экономически эффективной работе, подчеркивая важность этих переменных для достижения оптимальной производительности.

Математическая модель вносит значительный вклад в несколько жизненно важных аспектов. Это способствует более эффективному планированию и составлению графиков транспортных операций, обеспечивая возможность динамической корректировки в зависимости от меняющихся условий. Кроме того, это позволяет проводить сравнительный анализ затрат и сроков завершения, оптимизируя экономические решения и решения по производительности.

Сокращая время простоев и улучшая использование ресурсов, модель эффективно решает логистические и финансовые проблемы горнодобывающей деятельности и адаптируется к различным сценариям эксплуатации, повышая ее применимость и актуальность в отрасли. Оставшаяся часть статьи структурирована следующим образом: В разделе 2 представлены соответствующие работы. В разделе 3 подробно описана методология. Вычислительные эксперименты и анализ результатов обсуждаются в разделе 4.

Наконец, выводы, сделанные по результатам исследования, представлены в разделе 5.

2. Сопутствующие работы

В таблице 1 представлен обзор смежных работ, посвященных этой проблеме. В нем подробно описаны характеристики, рассматриваемые в нашей области интересов, такие как время выпуска (ri) и различные скорости (Qm). Также анализируются используемые целевые функции, такие как время завершения (Cmax) и затраты (k) и другие. Кроме того, используемые методы решения исследуются и классифицируются как точные или приблизительные, а также количество используемых маши

Machine Translated by Google

подходы, принятые в каждом случае. Такое сравнение позволяет визуализировать различные методологии и приемы, реализованные в литературе, подчеркивая их различия и сходства, касающиеся текущего анализа.

Характеристика ri еще недостаточно изучена, особенно в исследованиях горного дела. где наличие транспортной техники имеет решающее значение, как показано в [8,9]. В отличие, изучение Qm привлекло больше академического внимания, хотя основное внимание по-прежнему уделяется одинаковые машины. Что касается целевой функции, исследования, как правило, сосредоточены на минимизация переменных, таких как k и Cmax, по отдельности. Меза [2] иллюстрирует более сложную интеграция, которая включает анализ времени и затрат на погрузочно-транспортное оборудование посредством имитационной модели. Хонг и Лин [10] сосредоточились на минимизации максимальной задержки. (Lmax), тогда как Корягин и Воронов [11] и Олискевич с соавт. [12] рассмотрел встречу P как важное условие поддержания непрерывности работы производственных предприятий.

Для сравнения, наша работа значительно расширяет уровень анализа, включая как Qm и ri , позволяющие нам более реалистично моделировать производственную среду, особенно при добыче руды. Это особенно актуально в конфигурациях, где число машин превышает пять — сценарий, при котором многие исследования отдают предпочтение приближенным методам из-за их вычислительной эффективности. Кроме того, наш подход включает в себя итеративное исследование, которое позволяет проводить анализ тактического уровня для определения оптимального использования оборудования, дополняется за счет надежной аналитической поддержки принятия решений для максимизации операционной эффективности. Наш подход также включает анализ сложности, который еще предстоит изучить [10,13,14].

Таким образом, наша работа устраняет значительный пробел в литературе и дает критические выводы, которые облегчить стратегические решения и повысить операционную эффективность в горнодобывающей промышленности.

Таблица 1. Сопутствующие работы по проблеме планирования параллельных машин.

	Характеристики		Целевая функция			Методы		Число. Машины				
Ссылка	Qm	ri	Cmax	k	Другое	Точное	Приблизительное	m	5 m	6	Сложность Анализ	Подходы
[2]	x		x	x			x			x		Имитационная модель ФИФО
[15]			x			x			x			Алгоритм ветви и цены с одиночный сервер
[16]			x			x	x		x	x		Состав MIP и гибрид эвристический алгоритм с одиночный сервер
[17]			x			x	x		x			Формулировка MIP и Табу-поиск с одним сервером
[18]			x			x			x			Формулировка MIP с неидентичными размеры рабочих мест
[19]	x			x		x			x			формулировка MIP
[20]	x			x		x			x			Формулировка MILP
[21]	x			x		x	x		x	x		Формулировка и введение MILP Эвристика по тарифам по времени использования
[22]	x			x		x			x			Формулировка MILP под тарифы по времени использования
[8]		x	x			x	x		x	x		Формулировка MILP и Алгоритм ГА
[11]	x				П		x		x			Эвристический алгоритм
[23]			x			x			x			Формулировка MIP со временем установки и один сервер
[10]			x		Lмакс.		x		x		x	Динамические конструктивные алгоритмы
[24]			x			x	x					Модель программирования ограничений в условиях ограничений недоступности и изменить LPT
[13]			x			x	x		x		x	Формулировка MIP и Табу-поиск

Machine Translated by Google

Таблица 1. Продолжение.

Ссылка	Характеристики Qm	ri	Целевая функция Cmax	k	Другое	Методы Точное	Приблизительное	Число. Машины m	5 m	6	Сложность Анализ	Подходы
[25]				x		x				x		формулировка MIP
[14]			x			x		x	x	x		Состав MIP с независимые рабочие места
[26]	x		x			x		x				Состав MIP с функция ограничения
[12]	x				П	x		x				формулировка MIP
[27]				x			x	x				Имитационная модель Монте-Карло
[9]		x	x				x	x				Алгоритмы VNS и ACO
[28]	x		x			x	x			x		Алгоритм MIP и TSP
[29]			x			x		x				Модель программирования ограничений
[30]				x		x	x	x				формулирование MIP и Алгоритм МОБСО
[31]	x			x		x	x	x				Рецептура MILP с указанием времени использования и SPT, MDPC, MDEC
[32]	x		x			x	x			x		Формулировка MIP и GA–GWO гибридный алгоритм
Наша статья	x	x	x	x		x		x	x	x		Итеративная формулировка MIP

Учитывая сопутствующие исследования по оптимизации операции по транспортировке руды, было обнаружили, что Меза [2] разработал модель для применения концепций моделирования система погрузки и транспортировки при открытых горных работах с учетом типа погрузки и транспортно-транспортное оборудование, время прибытия, время ожидания, время позиционирования, время погрузки, движение время, время установки, время разгрузки и время возврата, а также количество поездок в качестве производственный план и рассмотрение системы ФИФО для погрузки и разгрузки. С другой стороны рука, Васкес и др. [3] реализовали модель целочисленного программирования, чтобы минимизировать общее количество время работы парка низкопрофильных погрузчиков в подземном руднике, а также разработка полиномиального оптимального по времени алгоритма, интегрированного в процесс принятия решений, получение результатов, достаточно близких к оптимальному общему времени. Эйвази и Аскари-Насаб [33] разработали модель смешанного целочисленного линейного программирования (MILP) с несколькими пунктами назначения для минимизации операционные затраты при открытой добыче полезных ископаемых; в модели они учитывали запасы и штабели-смесители в дополнение к системам загрузки в горизонтальном направлении и принятию решений по пандусам. Анджомшоа и др. [34] разработали модель моделирования и смешанного целочисленного программирования. (MIP) для оптимизации площадок для обгона, определения размеров и планирования парка транспортной техники в подземная добыча.

Более того, Глигорич [35] разработал исследование об обгонных площадках для буксировки оборудование в подземных горных работах, при котором стремились оптимизировать необходимое количество количество отсеков и оптимальное расположение для минимизации времени ожидания. Гонен и др. [36] провели исследование подземной добычи полезных ископаемых, в котором оценивались годовые производственные мощности с различные системы транспортировки с учетом удельных затрат на транспортировку и глубины шахты. Эйвази и Аскари-Насаб [37] представил методологию, основанную на иерархиях планирования производства. для открытой добычи полезных ископаемых в среднесрочной перспективе разработка MILP для минимизации эксплуатационных расходов; планирование включало склады, перерабатывающие заводы и свалки, а также выборку маршрутов и пандусов. Том-Сокаррас [38] разработал инструмент управления затратами, основанный на Дерево решений для транспортировки материалов при открытой добыче полезных ископаемых. Урибе [39] разработал исследование который проанализировал функциональность и производительность полуавтоматического режима загрузки и транспортное оборудование, используемое при подземных горных работах. Вичечек [40] предложил подходящую способ управления погрузкой и транспортировкой руды в шахте со сваями и

Machine Translated by Google

Система камер. Вист [41] разработал модель планирования для открытых горных работ, уделяя особое внимание снижению затрат.

Соломон [42] смоделировал системы погрузки и транспортировки руды при открытой добыче, разработав модель, учитывающую забой, маршруты, оборудование и пункты назначения руды. Гаспар и Жасмин [43] проанализировали показатели использования и доступности для оптимизации затрат на транспортировку руды. Ли и др. [44] в своих исследованиях подземных горных работ разработали алгоритм пчелиного роя для максимизации общего дохода в течение планового периода; кроме того, они создали генетический алгоритм планирования диспетчеризации транспортной техники, чтобы минимизировать время ожидания этой техники. Элайджа [45] разработал математическую модель для расчета прибытия и скорости обслуживания различного количества транспортного оборудования, участвующего в транспортировке материалов. Уиса [46] усовершенствовал использование погрузочно-транспортного оборудования, чтобы снизить стоимость этих операций.

Чой и др. [47] разработали интеллектуальную неконтролируемую систему для прогнозирования производительности системы подъемного оборудования при открытых горных работах с использованием комбинации различных моделей оптимизации. Ли и др. [48] сосредоточились на математической модели перепланирования операций с учетом требований производственных помещений, условий эксплуатации и износа производственного оборудования, стремясь при этом получить максимальную скорость завершения планирования и наименьшие колебания содержания руды. Аболгасемян и др. [49] представили многоцелевую оптимизацию системы открытых горных работ, направленную на получение максимального объема добычи и минимизацию времени транспортировки с учетом емкости хранилища, транспортного оборудования и бюджета. Шамси и др. [50] выполнили ИП с эксплуатационными ограничениями для планирования транспортной системы при открытой добыче полезных ископаемых, рассматривая в качестве цели наилучшую чистую приведенную стоимость. Паричехре и др. [51] рассмотрели проблему энергосберегающего несвязанного параллельного планирования работы машин, включив в себя ухудшение работы и эффект обучения операторов, предоставив ценную информацию о многоцелевой оптимизации для сложных производственных сред.

Наше исследование представляет новый и важный подход к оптимизации операций по погрузке и транспортировке руды путем моделирования тактической задачи с использованием модели планирования, обычно используемой для принятия оперативных решений. Эта модель учитывает параллельные машины с разной скоростью, что добавляет большей реалистичности. В частности, характер проблемы отражается в динамической доступности погрузочного оборудования, представленной динамическим временем выпуска, когда задания не известны заранее, но обнаруживаются по мере их выполнения. Этот подход позволяет проводить более глубокий и детальный анализ, который лучше подходит для сложных условий горнодобывающего производства. Внедрение этой модели можно интегрировать в тактические и оперативные стратегии, чтобы сделать их более эффективными и результативными, поддерживая принятие решений в горнодобывающих компаниях за счет значительного повышения операционной эффективности, снижения затрат и повышения производительности. Кроме того, это исследование будет способствовать как теоретическому развитию, так и практическому применению в горном машиностроении, улучшая основные операции в этом секторе.

3. Методология В

этом разделе описывается методология, использованная в данном исследовании. Он структурирован следующим образом: Раздел 3.1 определяет проблему и задает контекст исследования. В разделе 3.2 подробно описаны допущения моделирования и представлена математическая формулировка проблемы, включая целевые функции и ограничения. В разделе 3.3 описаны алгоритмы, используемые для определения и оценки оптимального количества рудотранспортной техники. Сюда входит процедура определения минимального количества транспортировочного оборудования и оценка соотношения затрат и времени. Следующие подразделы систематически рассматривают цели исследования и обеспечивают всестороннее понимание

3.1. Постановка проблемы

В связи с критической важностью транспортировки руды для производства, данное исследование направлено на определение минимального количества транспортировочного оборудования, необходимого для минимизации как затрат, связанных с использованием оборудования, так и времени завершения производственного задания.

Machine Translated by Google

в пределах определенного временного горизонта. Операции по доставке руды имеют решающее значение для непрерывной поставки необходимых полезных ископаемых на производственные предприятия. Однако технические решения относительно необходимого оборудования часто принимаются на основе опыта и в условиях ограниченности времени. Хотя эти решения могут быть быстрыми, они лишь иногда могут быть оптимальными, что потенциально может привести к увеличению затрат или задержкам, которые влияют на производство растений.

Это исследование посвящено неэффективности и потенциальной неоптимальности традиционных процессов принятия решений при определении минимального количества транспортного оборудования, необходимого для реализации производственной цели. Это может привести как к переоценке, так и к занижению стоимости транспортировочного оборудования, что приведет к ненужным затратам или узким местам при транспортировке руды. Разрабатывая математическую модель, которая рассматривает проблему планирования параллельных машин с разными скоростями и временем выпуска, мы стремимся обеспечить систематический и основанный на данных подход к оптимизации распределения оборудования.

Детальный анализ того, как затраты связаны со временем завершения, может выявить ценные сценарии , позволяющие принимать более обоснованные и правильные решения в случае каких-либо непредвиденных обстоятельств в транспортной операции. Эта модель позволит лицам, принимающим решения, найти баланс между стоимостью оборудования и операционной эффективностью, гарантируя, что необходимое количество оборудования будет задействовано для достижения производственных целей без ненужных р

Эту проблему можно представить как задачу планирования параллельных машин с разными скоростями и временем выпуска. Хотя основное внимание уделяется тактическому решению для определения минимального количества транспортного оборудования и анализа затрат, связанных со временем завершения, выполняется адаптация к этой проблеме планирования в рамках модели целочисленного программирова Согласно обозначениям, установленным Грэмом [52], задача планирования операций определяется тремя полями (α |β |γ): α представляет собой конфигурацию системы, β – ограничения и характеристики системы, а γ – целевую функцию. Используя эту структуру и следуя номенклатуре, описанной Пинедо и Хадави [53], конфигурация системы соответствует параллельным машинам с разными скоростями (Qm), где эти машины представляют собой погрузочное оборудование с различным временем обработки. Важной характеристикой системы является время выпуска (ri), включенное во время транспортировки. Целевая функция направлена на минимизацию как затрат, так и времени завершения. Рисунок 2 иллюстрирует график операции по транспортировке руды, концептуализированной как проблема, упомянутая ранее.

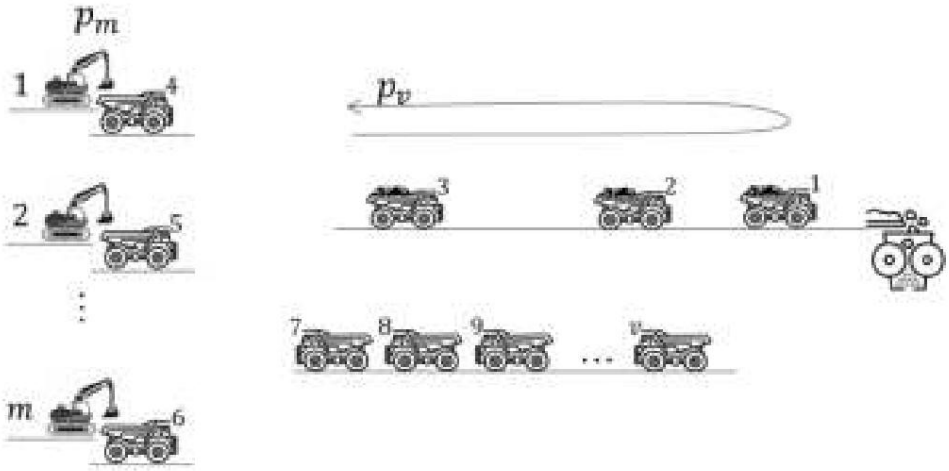


Рисунок 2. График с учетом операции по вывозке руды.

3.2. Моделирование

Модельные предположения. Допущения модели основаны на работе, представленной Меза [2]:

• Рассматривалась только одна точка разгрузки руды. • Время заправки незначительно. • Погрузочно-транспортное оборудование доступно в любое время. • Время ожидания выгрузки руды незначительно.

Machine Translated by Google

Обозначения. В табл. 2 приведены обозначения предложенной модели с указанием типа и определение соответственно.

Таблица 2. Обозначения предлагаемой модели.

Вар	Тип	Определение
M	Набор	Комплект погрузочного оборудования M = {1, 2, . . . , м} .
V	Набор	Комплект подъемно-транспортного оборудования V = {1, 2, . . . , в}.
T	Набор	Набор единиц времени: T = {1, 2, . . . , ч}.
$п^{л}_{дн}$	Параметр	Время погрузки погрузочного оборудования j.
$п^{час}_{я}$	Параметр	Время перевозки оборудования i.
П	Параметр	Производственная цель – общее количество поездок, которые необходимо совершить.
	Параметр	Временной горизонт для достижения производственной цели.
$хк^{л}_{дн}$	Параметр	Затраты, связанные с использованием погрузочной техники.
$к^{ч}_{я}$	Параметр	Затраты, связанные с использованием транспортной техники.
x_{ijt}	Переменная	Основная переменная решения принимает значение 1, если транспортное оборудование i использует погрузочное оборудование j в момент времени t, в противном случае — 0.
y_{jt}	Переменная	Вспомогательная переменная решения принимает значение 1, если погрузочное оборудование j используется в момент времени t, 0, в противном случае.
$_{принцип}$	Переменная	Вспомогательная переменная решения принимает значение 1, если транспортное оборудование i используется в момент времени t, 0, в противном случае.
Смакс	Переменная	Срок выполнения производственного задания
к	Переменная	Затраты, связанные с использованием оборудования

Целевая функция. Первая целевая функция (уравнение (1)) направлена на минимизацию k связанные с использованием оборудования. Вторая целевая функция (уравнение (2)) направлена на минимизацию Cmax всех поездок.

$$\text{мин } к \tag{1}$$

$$\text{мин } Cmax \tag{2}$$

Ограничения. Что касается ограничений, набор ограничений в уравнении (3) обеспечивает назначение только одного транспортного средства i одному погрузочному транспортному средству j в момент времени t. Набор ограничения в уравнении (4) обеспечивают наличие только одного погрузочного оборудования j на одно транспортировочное оборудование i во время т.

$$\sum_{i \varepsilon V} x_{ijt} \quad 1 \quad j \quad M, t \quad T \tag{3}$$

$$\sum_{j \varepsilon M} x_{ijt} \quad 1 \quad i \quad V, t \quad T \tag{4}$$

Набор ограничений в уравнениях (5) и (6) обеспечивает, чтобы погрузка и транспортировка оборудование используется в течение необходимого периода времени при его назначении.

$$\sum_{\tau=t+1}^{т+п^{л}_{дн}} y_{j\tau} \quad (п^{л}_{дн} \quad 1) \cdot x_{ijt} \quad i \quad V, j \quad M, t \quad T \setminus (t \quad h \quad п^{л}_{дн}) \tag{5}$$

$$\sum_{\tau=t+1}^{т+п^{л}_{j}+п^{час}_{я}} z_{i\tau} \quad (п^{час}_{я} + п^{л}_{дн} \quad 1) \cdot x_{ijt} \quad i \quad V, j \quad M, t \quad T \setminus (t \quad h \quad п^{час}_{я} \quad п^{л}_{дн}) \tag{6}$$

Набор ограничений в уравнениях (7) и (8) гарантирует, что погрузка и транспортировка оборудование не резервируется дважды и используется только при его наличии.

$$x_{ijt} \quad 1 \quad y_{jt} \quad i \quad V, j \quad M, t \quad T \tag{7}$$

$$x_{ijt} \quad 1 \quad z_{it} \quad i \quad V, j \quad M, t \quad T \tag{8}$$

Machine Translated by Google

Набор ограничений в уравнении (9) обеспечивает Cmax.

$$Cmax \quad t \cdot xijt + p_{я}^{час} + п_{дж}^{л} \qquad i \quad V, j \quad M, t \quad T \tag{9}$$

Набор ограничений в уравнении (10) гарантирует соблюдение P.

$$\sum_{я}^{В} \sum_{дж}^{M} \sum_{т}^{T} xijt \quad P \quad i \quad V, j \quad M, t \quad T \tag{10}$$

Уравнение (11) рассчитывает k.

$$K = \sum_{я}^{В} \sum_{т}^{T} чк \cdot зит + я \sum_{дж}^{M} \sum_{т}^{T} л_{дж}^{к} \cdot yjt \quad i \quad V, j \quad M, t \quad T \tag{11}$$

Набор ограничений в уравнениях (12) и (13) обеспечивает природу переменных.

$$xijt, yjt, zit \in \{0, 1\} \quad i \in V, j \in M, t \in T \tag{12}$$

$$Cmax, k \quad 0 \tag{13}$$

Предложенная модель может быть использована в процессах принятия тактических и оперативных решений, поскольку она точно определяет оба решения. Однако сложность этой модели смешанно-целочисленного линейного программирования с точки зрения количества целочисленных переменных составляет i·j·t + j·t + i·j, и в ней есть еще две непрерывные переменные. Если мы проанализируем количество ограничений, мы найдем в общей сложности 7 ·i·j·t + j·t + i·j. Тот факт, что модель опирается на бинарные векторы (для указания времени запуска грузовика и периодов активности), по своей сути увеличивает сложность, но также открывает возможность (i) включения допустимых неравенств для отбрасывания решений, которые не соответствуют структуре задачи или (ii) методы декомпозиции, такие как использование комбинаторных разрезов Бендерса или использование концепции катящегося горизонта [54], чтобы использовать стратегии «разделяй и властвуй»; оба они являются частью продолжающихся исследований. Можно было бы рассмотреть возможность сокращения периодов, но это помешало бы использовать эти модели в схемах управления, таких как цифровые двойники — метод, широко распространенный в управлении горнодо

3.3. Алгоритмы определения и оценки оптимального количества оборудования для перевозки руды. Чтобы учесть двойную цель модели: минимизировать время завершения и стоимость, был использован итерационный подход, при котором модели создаются с использованием только одной целевой функции. Как показано в алгоритмах 1 и 2, итерационный процесс позволяет найти компромисс между временем завершения и затратами, предоставляя набор решений , которые уравновешивают обе цели.

При рассмотрении анализа тактического уровня была предложена следующая процедура для определения оптимального количества транспортировочного оборудования, подробно описанная в алгоритме 1. Алгоритм hl начинается с определения входных данных, которые включают M, V, T , pi , п j , h и P. Затем определяются выходные данные, которые включают минимальное количество транспортного средства (vopt), и назначение транспортного средства погрузочному оборудованию (График), включая Cmax.

Затем присвойте v величине V, общему количеству доступного транспортного оборудования (строка 1). Переменной Continue присваивается значение true, чтобы инициировать цикл (строка 2). Пока это остается верным, создается модель (с учетом уравнений (2)–(13) модели PMSPOHO, входных данных и v) и получается результат , который минимизирует Cmax (строки 3–5). Проводится технико-экономическое обоснование, где, если Результат достижим, v уменьшается на одну единицу (попытка решить проблему меньшим количеством транспортировочной техники) (строки 6–8). Если это невозможно, vopt увеличивается на одну единицу, и цикл завершается (строки 9–13). Этот алгоритм возвращает vopt и Schedule (строка 14).

Machine Translated by Google

Алгоритм 1. Определитель оптимального транспортного средства.

Ввод: M, B, T , p $^{час}_{я}$, п $^{л}_{дж}$, ч, П;

Выход: vopt: минимум буксирной техники, График: назначение буксировочной техники.
к погрузочному оборудованию, включая Cmax.

```
1: v  |V|
2: Продолжить  true
3: пока продолжаем делать
4:     модель  Создать модель (Уравнения (2)–(13), Входные данные, v)
5:     Результат  Решить(модель, Минимизировать)
6:     если IsFeasible(Результат), то
7:         v  v - 1
8:         Результат, Расписание
9:     еще
10:        вопт  v + 1
11:        Продолжить  false
12: конец, если
13: закончить пока
14: возврат вопт, расписание
```

Для анализа связи между k
и Cmax, подробно описанные в алгоритме 2. Этот псевдокод позволяет наблюдать за поведением k по мере увеличения Cmax . Алгоритм начинается с определения входных данных, которые включают в себя те же факторы, что и в алгоритме 1: k h $_{я,}$ $^{к\,л}_{дж.}$ и v. Затем определение выходных данных включает в себя список затрат и список сроков завершения (Cmax). Определите v (строка 1). МодельIb создана (с учетом уравнений (2)–(13) модели PMSPOHO и входных данных) (строка 2) и ResultIb из минимизации Cmaxlb получается (строка 3), извлекая этот результат (строка 4). модельный клуб создается (с учетом уравнений (1), (3)–(13) модели PMSPOHO и входных данных (строка 5), и получается Resultub в результате минимизации k (строка 6), извлекая для этого Cmaxub результат (строка 7).

Алгоритм 2. Прогрессивный оценщик затрат и времени.

Ввод: M, B, T , p $^{час}_{я}$, п $^{л}_{дж.}$ $^{к\,ч}_{я,}$ $^{к\,л}_{j}$, h, P и v;

Вывод: ListPair(Cmax, k): список затрат относительно Cmax.

```
1: Определить v
2: modellb  Createmodel(Уравнения(2)–(13), Входные данные)
3: Resultlb  Решить(modellb, Минимизировать)
4: Cmaxlb  ExtractCmax(Resultlb)
5: modelub  Создать модель (уравнения (1), (3)–(13), входные данные)
6: Resultub  Solve(modelub, Minimize)
7: Cmaxub  ExtractCmax(Resultub)
8: для Cmaxre ж из Cmaxlb в Cmaxub сделать
9:     модель f  Создать модель (уравнения (1), (3)–(8), (10)–(13), входные данные)
10:    Результат f Решить(модель f , Минимизировать)
11:    Стоимость  ExtractCost(Resultre f)
12:    ListPair(Cmax, k)  AddToList(Cmaxre ж, Расходы)
13: конец для
14: возврат Cmax, k
```

Проводится прогрессивная оценка, повторяющая Cmaxre ж на 1 от Cmaxlb до
Cmaxub (строка 8). Для каждого значения создается модель f (с учетом уравнений (1), (3)–(8),
и (10)–(13) модели PMSPOHO и входных данных) (строка 9). Результат f получается путем минимизации k для каждой итерации (строка 10); k извлекается и добавляется в список (Cost) для каждого соответствующего Cmaxre , k) (строки 14),
ж (строки 11–12). Этот алгоритм возвращает список пар (Cmaxre ж

Machine Translated by Google

предоставление подробной оценки того, как k изменяется в зависимости от изменений ϕ. В алгоритме Cmaxre, Cmaxre служит промежуточным эталонным значением для итеративной оценки компромисса между минимизацией Cmax и k. Эта итеративная оценка дает представление о соотношении затрат и времени. компромиссы в модели.

4. Вычислительные эксперименты и анализ результатов.

В этом разделе представлены вычислительные эксперименты по оценке производительности предлагаемая модель и алгоритмы PMSPOHO. Все эксперименты проводились на компьютер с операционной системой Windows 11, Intel(R) Core(TM) i5-12450H 12-го поколения, Процессор 2,50 ГГц и 8 ГБ оперативной памяти. Для этого использовался язык программирования Python 3.11. развитие. Коммерческий оптимизатор Gurobi 10.0.1 и бесплатный оптимизатор CBC были используется с ограничением времени поиска 3600 с и GAP 0,01.

На основе фактических данных, предоставленных в исследованиях Мезы [2], Васкеса [3] и Анчирайко. и Рохас [4], включающие количество погрузочно-транспортной техники, погрузочно-разгрузочных работ время, время перевозки, временные горизонты и производственные цели, было 34 экспериментальных экземпляра. генерируется. Эти экземпляры предназначены для моделирования реальных сценариев, обеспечивая надежную оценка предлагаемых нами методик в различных условиях. Такой подход обеспечивает что наша экспериментальная установка основана на реалистичных рабочих параметрах, что повышает применимость и актуальность наших выводов. Важно отметить, что прямое сравнение с приведенными выше исследованиями авторов не проводилось, так как их методологии исследования различаются значительно отличается от предложенного нами подхода.

4.1. Настройка значений параметров оптимизатора

Gurobi Optimizer версии 10.0.1 имеет обширный набор параметров, разработанных для тонкой настройки производительности оптимизатора. Учитывая огромное количество вариантов среди возможных конфигурации, определяя оптимальное сочетание параметров, повышающих общую эффективность Производительность модели представляет собой серьезную проблему. TuningAPI от Gurobi предлагает ценный инструмент для изучения различных настроек параметров, которые могут улучшить производительность модели. в конкретных случаях [56].

Учитывая процедуру, указанную Cuellar-Usaquén et al. [57], выборка из семи экземпляры были созданы для определения соответствующих настроек параметров для нашей модели с помощью API настройки. В этот образец включены те случаи, когда время вычислений, полученное с помощью модель PMSPOHO с использованием параметров по умолчанию (PMSPOHO-D) превышает среднее время соответствующего класса на два стандартных отклонения.

В таблице 3 показаны результаты настройки параметра Гуроби для семи различных экземпляров проблемы . Каждая строка таблицы представляет экземпляр, а столбцы указывают конкретный Гуроби. параметры, скорректированные для этого экземпляра. Параметры включают Symmetry, DegenMove, Эвристика, GomoryPasses, PreDepRow, MIPFOcus, Method, CutPasses и BranchDir.

Таблица 3. Конфигурации параметров, найденные для математической модели.

Инст	Краткое содержание вм	П	Симметрия		Параметры значений ДегенMoves	Эвристика	ГоморыПроходит	Метод	PreDepRow	MIPFOcus	CutPasses	Директор филиала	ТС-Д	Тс-Т	Вариация
1	3	2	20	2									0,8	0,79	1,25%
2	6	2	20		2								1,22	1,15	5,74%
3	8	4	81			0	1	1					69,59	55,03	20,92%
4	12	4	81						2				46,54	44,4	4,60%
5	15	4	81							0			32,53	31,64	2,74%
6	8	4	90				1					1	139,22	120,68	13,32%
7	20	4	100								1		81,62	79,9	2,11%

Третий экземпляр значительно выделяется, поскольку демонстрирует наибольшую вариативность. (20,92%) между временем расчета по умолчанию (Tc-D) и настроенным вычислительным временем. время (Tc-T), что указывает на значительное улучшение производительности при скорректированных параметрах. В этом случае параметры Heuristics, GomoryPasses и PreDepRow были скорректированы, что привело к существенному сокращению времени вычислений с 69,59 до 55,03 единицы времени. Эта корректировка сократила общее время расчета и оптимизировала использование вычислительных ресурсов, что делает процесс более эффективным и действенным. Кон-

Machine Translated by Google

Значительная разница между Tc-D и Tc-T подчеркивает важность настройки и точной настройки параметров модели для достижения максимальной эффективности при решении сложных математических задач.

В этом случае скорректированный параметр «Эвристика» становится ключевым игроком в ускорении процесса решения. За счет увеличения использования эвристических методов, которые представляют собой стратегии решения, использующие простые и быстрые правила для быстрого создания удовлетворительных решений, модель более эффективно исследовала и оценивала потенциальные решения. Такой подход значительно сократил время, необходимое для достижения осуществимого решения, продемонстрировав его практичность и эффективность. Применение эвристики является ценным инструментом, особенно в сценариях, где скорость имеет решающее значение и некоторой точностью можно пожертвовать ради повышения скорости разреше

GomoryPasses, еще один измененный параметр, контролирует количество проходов резки Gomory. Эти проходы представляют собой методы оптимизации, которые усиливают линейную релаксацию модели, что приводит к более быстрой сходимости. Регулировка количества таких проходов может существенно повлиять на эффективность решения, уравновешивая необходимость улучшения решения с дополнительным временем, необходимым для каждого разреза. PreDepRow, который осуществляет предварительную обработку зависимых строк, также существенно повысил эффективность за счет уменьшения размера задачи и повышения числовой стабильности. Эффективная предварительная обработка может упростить модель, устранив избыточность и уменьшив общую сложность. В третьем случае настройка PreDepRow значительно сократила время вычислений, отметив важность правильного управления моделью. Эти корректировки подчеркивают важность настройки параметров для повышения эффективности решателя, что позволяет получать более быстрые и точные результаты.

4.2. Результаты сравнительного анализа для оптимизаторов

В таблице 4 представлены результаты сравнительного анализа оптимизаторов для 34 экземпляров. Экземпляры были организованы с учетом производственной цели, поскольку по мере увеличения этой цели отмечалось увеличение времени вычислений (Tc). Результаты сравнивались между моделью по умолчанию (PMSPOHO-D), настроенной моделью (PMSPOHO-T) и моделью со свободным оптимизатором (PMSPOHO-CBC). Критическими параметрами для сравнения были Tc и GAP, которые измеряют сложность модели и эффективность решателя. В этом исследовании значение GAP предоставляется непосредственно оптимизатором Gurobi как часть выходных данных его решения.

Настроенная модель (PMSPOHO-T) постоянно требует меньшего времени вычислений, чем другие модели. Это указывает на то, что настройка параметров значительно повысила эффективность решателя. GAP, который представляет собой разницу между самым известным решением и лучшей нижней границей, найденной решателем, составляет 0,00% как для PMSPOHO-D, так и для PMSPOHO-T, что указывает на то, что найденные решения являются оптимальными. Однако PMSPOHO-CBC в нескольких случаях демонстрирует более высокий GAP, что отражает более низкую точность и подчеркивает ограничения коммерческого оптимизатора.

PMSPOHO-T отличается лучшими результатами по Tc, быстрее решая задачи, сохраняя при этом точность решения. Это резко контрастирует с моделью со свободным оптимизатором (PMSPOHO-CBC), которая во многих случаях показывает более высокий GAP, тем самым подчеркивая ее ограничения. Это также демонстрирует, что сложность проблемы составляет NP-Hard, а это означает, что по мере увеличения количества экземпляров оптимальные результаты могут быть получены через значительно большее время, что делает бесплатный оптимизатор неэффективным вариантом для полевых приложений.

На рисунке 3 сравниваются время вычислений (Tc) и Cmax для оптимизаторов PMSPOHO-D, PMSPOHO- T и PMSPOHO-CBC, например 34. PMSPOHO-D и PMSPOHO-T достигают одинакового Cmax , равного 183, но PMSPOHO-T больше. эффективен с точки зрения времени вычислений: 671,8 по сравнению с 798,8 для PMSPOHO-D. С другой стороны, PMSPOHO-CBC показывает худшую производительность с Cmax 229 и временем вычислений 3600. Это указывает на то, что PMSPOHO-T является наиболее эффективным среди трех оцениваемых оптимизаторов, достигая хорошего баланса между временем вычислений и Cmax. .

Machine Translated by Google

Таблица 4. Результаты сравнительного тестирования оптимизаторов по экземплярам.

Пример	P vm Cmax	ПМСПОХО-Д			ПМСПОХО-Т			ПМСПОХО-СВС		
			Тс	ГАП Cmax		Тс	ГАП Cmax		ТЦ ГАП	
	20 20 2 20 16 2 20	29	14,52	0,00% 9,51	29	9,73	0,00% 5,92	30	3600	4,58%
1	10 2 20 8 2 20 6 2	29	0,00%	3,82 0,00%	29	0,00%	2,91 0,00%	30	3600	4,64%
2	20 5 2 20 4 2 20 3	29	2,51	0,00% 1,6	29	1,85	0,00% 1,58	31	3600	5,21%
3	2 81 20 4 81 16 4	29	0,00%	1,41 0,00%	29	0,00%	1,03 0,00%	30	3600	4,82%
4	81 10 4 81 8 4 81	31	1,13	0,00% 0,9	31	1,01	0,00% 1,82	32	3600	3,77%
5	6 4 81 4 4 81 20 6	34	0,00%	28,25	34	0,00%	26,74	35	3600	2,36%
6	81 16 81 10 6 81 8	40	0,00%	18,34	40	0,00%	18,61	41	3600	2,56%
7	6 81 6 6 90 5 3 90	51	0,00%	49,23	51	0,00%	46,08 %	51	1846	0,00%
8	4 3 90 3 3 200 19	55	69,59	0,00% 66,54	55	50,86	0,00% 63,01	60	3600	8,52%
9	4 200 18 4 200 17	55	0,00%	76,29 0,00%	55	0,00%	59,78 0,00%	60	3600	8,20%
10	4 200 16 4 200 15	64	42,12	0,00% 318,67	64	42,26	0,00% 130,11	68	3600	6,45%
11	4 200 14 4 200 13	78	0,00%	94,3 0,00 %	78	0,00%	76,82 0,00	82	3600	5,50%
12	4 200 12 4 200 11	100	100,46	0,00%	100	% 100,13	0,00%	106	3600	5,72%
13	4 200 10 4 200 9 4	147	145,77	0,00%	147	90,32	0,00% 26,41	152	3600	3,27%
14	200 8 4	38	42,98	0,00% 83,77	38	0,00%	40,36 0,00%	42	3600	10,25%
15		42	0,00%	44,52 0,00%	42	30,78	0,00%	46	3600	9,86%
16		63	1784,21	0,00%	63	1627,21	0,00%	70	3600	10,52%
17		77	1693,64	0,00%	77	1437,64	0,00%	85	3600	10,73%
18		98	1634,63	0,00%	98	1463,63	0,00%	110	3600	12,72%
19		130	1640,24	0,00% 158	130	1400,24	0,00%	163	3600	25,32%
20		161	4,04	0,00% 1554,43	161	1437	0,04 0,00%	202	3600	25,47%
21		212	0,00%	1532,23	212	1291,43	0,00%	259	3600	22,23%
22		125	0,00%	1524,64	125	1349,23	0,00%	161	3600	29,03%
23		125	0,00%	1279,45	125	1372,64	0,00%	161	3600	28,53%
24		125	0,00%	1017,6 0,00%	125	1019,45	0,00%	161	3600	28,93%
25		125	865,6	0,00% 798,8	125	746,6	0,00% 615,6	161	3600	28,73%
26 27		125	0,00%		125	0,00%	671,8 0,00%	164	3600	31,45%
28		125			125			159	3600	27,32%
29		125			125			161	3600	29,05%
30		151			151			194	3600	28,63%
31		151			151			189	3600	25,43%
32		151			151			186	3600	23,23%
33		166			166			203	3600	22,01%
34		183			183			229	3600	24,93%

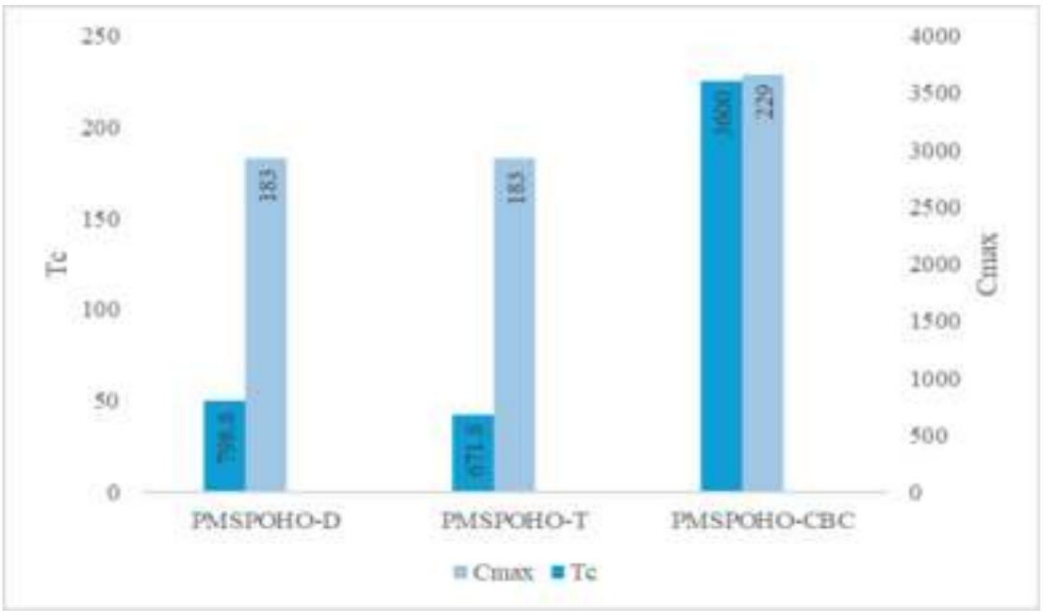


Рисунок 3. Сравнение времени вычислений и Cmax для примера 34.

4.3. Тактический подход

В таблице 5 показана эффективность примененных алгоритмов. Алгоритм 1 вычисляет минимальное количество буксирного оборудования, в результате чего получается восемь единиц техники. С другой стороны С другой стороны, алгоритм 2 выполняет оценку k-Cmax , что позволяет провести более подходящий анализ. между этими двумя переменными. Это улучшает процесс принятия тактических решений, показывая, как изменение Cmax влияет на k и наоборот. Минимизация Cmax приводит к значительному снижению значения (между 33 и 49), хотя увеличение k позволяет планировщикам сбалансировать временные эффективность и эксплуатационные расходы. Алгоритм 2 обеспечивает более полную и сбалансированную оценку.

Machine Translated by Google

перспектива для принятия тактических решений и поддерживается Алгоритмом 1, который определяет количество ресурсов, необходимых для достижения производственного задания.

Таблица 5. Результаты оптимизации предложенных алгоритмов.

			Минимизировать k		Минимизировать Cmax		
в	м	П	Cmaxub к	Тс	Cmaxlb	к	Тс
19	4	200	51	2350 151,46 2350	33	6998	54,97
18	4	200	51	219,68 2350	33	6648	46.31
17	4	200	51	117,08 2350	33	6298	38,57
16	4	200	51	99,44 2350	33	5948	33,81
15	4	200	51	125,16 2350	33	5598	31.16
14	4	200	51	162,80 2350 75,34	33	5248	26,78
13	4	200	51		34	4898	24.63
12	4	200	51	2350 58,55	35	4548	20.73
11	4	200	51	2350 111,22	37	4198	21.40
10	4	200	51	2350 103,30	40	3848	32,76
9	4	200	50	2350 44,62	44	3498	19.40
8	4	200	51	2350 76,94	49	3148	15.93

По мере сокращения сроков завершения работ непропорционально возрастают затраты, что указывает на компромисс между временем и экономической эффективностью операций по транспортировке руды. Более крутые кривые на нижних время завершения влечет за собой значительное увеличение затрат для более быстрых операций, в то время как более плоские наклоны в более высокие времена предполагают минимальный порог стоимости. Более широкий диапазон сроков завершения строительства с использованием большего количества транспортировочного оборудования означает, что, хотя дополнительное оборудование может ускорить операции, это также влечет за собой более высокие затраты, что подчеркивает важность балансировки бюджетные ограничения при оперативной срочности планирования горнодобывающих работ.

На рисунке 4 показана взаимосвязь между затратами и временем завершения для различных проектов. количество транспортной техники. Заметное снижение затрат при незначительном увеличении во время завершения. Тактический подход к определению минимального объема перевозки оборудования, необходимого для достижения производственных целей, следует учитывать не только стоимость и завершенность время, но и наклон каждой линии, что поддерживает применение алгоритмов предложено в этой статье. Например, переход с девяти на восемь транспортных единиц незначительно увеличивает время завершения, значительно снижая затраты. Это говорит о том, что операционная с восемью буксирными единицами может быть эффективным тактическим решением, если дополнительное время не ставить под угрозу другие приоритеты перевозки руды.

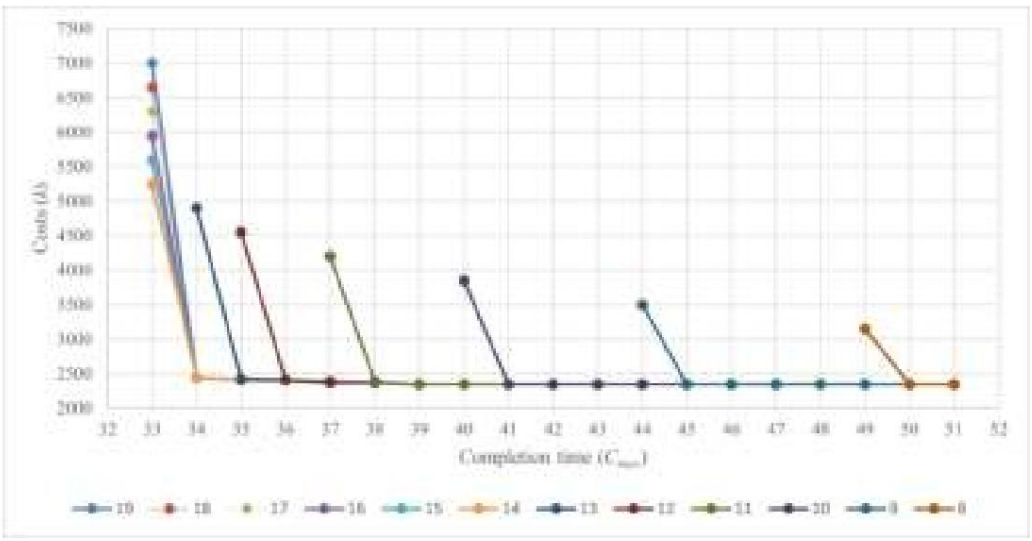


Рисунок 4. Затраты k в зависимости от времени завершения Cmax для случаев, предложенных в таблице 5.

Machine Translated by Google

Лица, принимающие решения, должны сопоставить увеличение затрат, связанное с использованием большего количества оборудования, с сокращением времени перевозки. После определенного момента добавление большего количества транспортных средств сводит на нет выгоду от сокращения времени завершения, но влечет за собой существенно более высокие затраты. Определение момента, когда дополнительные затраты больше не оправдывают сокращение времени перевозки, имеет решающее значение для оптимизации операций и эффективного управления ресурсами.

4.4. Аналитика оперативного подхода

На рисунке 5 показан график, например 34, в котором учтено восемь транспортных и четыре погрузочных оборудования. Это визуализирует программу загрузки руды для горнодобывающих работ во время первых 50 рейсов до производственного объекта. Интересно посмотреть, как организовано распределение так, чтобы минимально необходимое количество транспортной техники находилось в постоянном движении, что указывает на тщательную оценку времени погрузки и интервалов между задачами. Эта стратегия гарантирует, что ни одно погрузочное оборудование не останется бездействующим, а транспортное оборудование будет эффективно распределено для достижения производственных целей.

Минимизируя количество транспортировочного оборудования, операция может стать более рентабельной и улучшить логистику за счет уменьшения заторов и потенциальных узких мест при транспортировке руды. Этот сбалансированный подход между эффективностью и результативностью планирования имеет решающее значение для успешной работы, которая адаптируется к динамичным и часто непредсказуемым требованиям горнодобывающей промышленности.

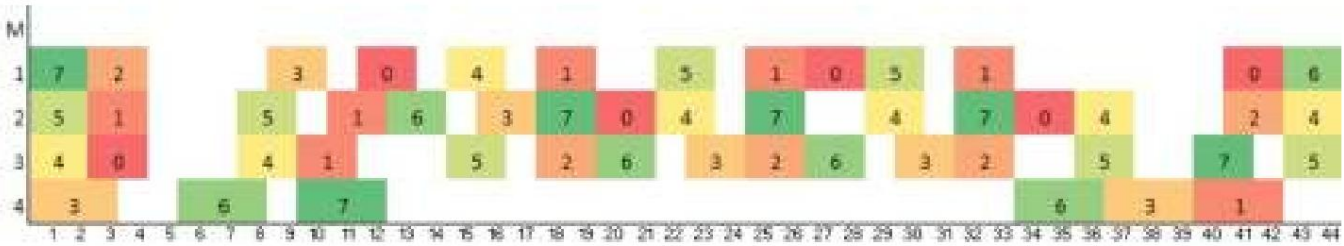


Рисунок 5. График для примера 34 с учетом минимизации времени завершения Cmax.

В Таблице 6 представлено подробное распределение использования погрузочно-транспортного оборудования при добыче полезных ископаемых, подчеркнуто распределение и использование этих ресурсов в производстве . Замечено, что погрузочное оборудование №2 и №3 используется наиболее часто, при этом процент использования составляет 32% и 30% соответственно, что указывает на высокий спрос и частое использование. Напротив, четвертая погрузочная машина используется меньше всего, ее загрузка составляет во Это говорит о том, что ресурсы не полностью оптимизируются, особенно с учетом того, что четвертая погрузочная машина может быть самой дорогостоящей в эксплуатации, поскольку менее часто используемое оборудование обычно требует регулярного обслуживания и занимает ценное пространство, которое можно было бы использовать более эффективно.

Недостаточное использование дорогостоящего погрузочного оборудования приводит к серьезным проблемам управления затратами и операционной эффективности. Высокая стоимость этого оборудования может быть обусловлена специальными функциями, которые требуют более сложного обслуживания или более высоких первоначальных инвестиций. Однако его неоптимальное использование подразумевает, что эти инвестиции должны приносить ожидаемую прибыль, что имеет решающее значение в операциях, направленных на максимизацию эффективности и прибыльности. Эта ситуация подчеркивает острую необходимость пересмотреть распределение задач и ресурсов, чтобы гарантировать, что все погрузочное оборудование используется таким образом, чтобы оправдать его стоимость, тем самым оптимизируя всю операцию и сокращая потенциальную потерю ресурсов.

Результаты показывают, что предложенная модель эффективна с точки зрения вычислительного времени и что предложенные алгоритмы являются полезными тактическими инструментами для более эффективного принятия решений относительно распределения транспортировочного оборудования, необходимого для достижения целей, поставленных в операциях по транспортировке руды. Однако важно признать, что анализ более значимых случаев в виде математической модели может усложнить время, необходимое для получения результатов, даже при точной настройке параметров оптимизатора. Поэтому будет необходимо увеличить сроки поиска решения и разработать приближенные методы, позволяющие сократить это время и получить практические и выполнимые решения.

Machine Translated by Google

Таблица 6. Загрузка погрузочно-транспортной техники.

Транспортное оборудование	Погрузочное оборудование				Общий	Использование
	1	2	3	4		
0	3	2	1	0	6	12%
1	3	2	1	1	7	14%
2	1	1	3	0	5	10%
3	1	1	2	2	6	12%
4	1	4	2	0	7	14%
5	2	2	3	0	7	14%
6	1	1	2	2	6	12%
7	1	3	1	1	6	12%
Общий	13	16	15	6	50	
Использование	26%	32%	30%	12%		

5. Выводы

Предложенная модель демонстрирует значительную эффективность с точки зрения вычислительных затрат. время, эффективно балансируя затраты и время завершения операций по транспортировке руды. Применение эти два алгоритма предоставили тактические инструменты, которые улучшают процесс принятия решений относительно выделение транспортной техники. Эффективность модели очевидна в вычислительных результаты, показывающие, что настройка параметров может значительно улучшить производительность решателя. Настроенная модель последовательно обрабатывала меньшее время вычислений, чем настройки по умолчанию. подчеркивая важность тщательной настройки параметров. Кроме того, включение динамическое время выпуска и различные скорости буксировки оборудования в модели добавляют новый уровень реализма, гарантируя, что планирование отражает фактические условия эксплуатации. Этот аспект дополнительно оптимизирует процесс распределения, делая модель вычислительно надежной.

Несмотря на результаты, модель имеет ограничения. Основным ограничением является его масштабируемость. По мере увеличения размера экземпляров время, необходимое для получения результатов, может стать непомерно, даже при оптимизированных параметрах. Эта проблема присуща математическим моделям. работа со сложными NP-трудными задачами. Кроме того, зависимость модели от точных методы оптимизации могут лишь иногда быть практичными для принятия решений в реальном времени в динамичные среды, где важно быстрое реагирование.

Будущая работа должна быть сосредоточена на продлении периодов поиска решения и разработке приближенных методов устранения этих ограничений. Эти методы, включая эвристические или метаэвристические подходы могут сократить время вычислений и предоставить почти оптимальные практические и осуществимые решения в реальных сценариях. Кроме того, интегрирующая машина методы обучения прогнозированию и динамической корректировке параметров могут улучшить производительность и адаптивность модели [58].

Вклад авторов: концептуализация, LT-T., CA и DA-M.; методология, ЛТ-Т. и Калифорния; программное обеспечение, ЛТ-Т. и компьютерная графика; формальный анализ, LT-T. и АП; расследование, ЛТ-Т. и АП; данные курирование, LT-T.; письмо — подготовка первоначального черновика, ЛТ-Т.; написание-обзор и редактирование, LT-T., CA и Д.А.-М.; визуализация, ЛТ-Т.; надзор, CA и Д.А.-М.; администрация проекта, ЛТ-Т. Все авторы прочитали и согласились с опубликованной версией рукописи.

Финансирование: APC финансировался Университетом Лос-Андеса.

Заявление Институционального наблюдательного совета: Неприменимо.

Заявление об информированном согласии: Не применимо.

Заявление о доступности данных: данные, представленные в этом исследовании, доступны по адресу: https://github.com/.ЛуисТаразонаТоррес/Проблема планирования параллельного оборудования --- Hualing-Equipment.git (доступ на 29 июля 2024 г.).

Благодарности: Авторы выражают признательность всем участникам, внесшим вклад в проведение данного исследования.

Machine Translated by Google

Конфликты интересов: Авторы заявляют об отсутствии конфликта интересов.

Ссылки

1. Раут, М.; Парида, К.К. Оптимизация параметров взрывных работ в карьерах. доктор философии Диссертация, Национальный технологический институт Руркела, Одиша, Индия, 2007 г.
2. Меса Кастро, Дж. Э. Разработка модели для применения моделирования в системе погрузки и выгрузки в горнодобывающей промышленности в Тахо Аберто. Бакалаврская диссертация, Папский католический университет Перу, Сан-Мигель, Перу, 2011 г.
3. Васкес, О.К.; Сепульведа, Дж. М.; Кордова Ф. Моделирование и оптимизация работы транспортных средств при подземной добыче меди. Рем Преподобный Esc. Де Минас 2011, 64, 365–371. [Перекрестная ссылка]
4. Анчирайко, А.; Рохас, К. Оптимизация системы акаррео и транспортировки в подготовительных работах зон углубленного изучения методологии «Шесть сигм» для ECM Zicsa и Unidad Minera Inmaculada. Бакалаврская диссертация, Перуанский университет прикладных наук, Сантьяго-де-Сурко, Перу, 2023.
5. Суфастаи, А.; Каримпур, Э.; Найтс, П.; Кизил М. Энергоэффективные погрузочно-транспортные работы; Спрингер: Чам, Швейцария, 2018 г.; стр. 121–146. [Перекрестная ссылка]
6. Цой Ю.; Бэк, Дж.; Парк, С. Обзор приложений на основе ГИС для горнодобывающей промышленности: планирование, эксплуатация и управление окружающей средой. Прил. наук. 2020, 10, 2266. [CrossRef]
7. Оволаби А. Выбор погрузочно-транспортного оборудования для оптимального производства в гранитном карьере. Межд. Дж. Мин. наук. 2019, 5, 35–40.
8. Марте Кольядо, Дж. М. Программа производства параллельных машин, моделей, отступов и ограниченного использования. Магистерская диссертация, Политехнический университет Валенсии, Валенсия, Испания, 2017.
9. Салазар-Хорниг Э.; Сото-Гавилан, Г. Минимизация времени простоя для устранения проблем с параллельными машинами без связи с настройками, зависящими от безопасности при работе гибридного алгоритма VNS/ACO. Преподобный инж. унив. Де Медельин, 2021 г., 20, 171–184. [Перекрестная ссылка]
10. Хонг, ХК; Лин, Б.М. Параллельное планирование выделенных машин с графами конфликтов. Вычислить. Индийский англ. 2018, 124, 316–321. [Перекрестная ссылка]
11. Корягин, М.; Воронов А. Совершенствование организации системы «экскаватор-автомобиль» на угольных разрезах. Трансп. Пробл. 2017, 12, 113–122. [Перекрестная ссылка]
12. Олискевич М.; Ковалышин С.; Магац, М.; Шевчук В.; Сукач О. Оптимизация графика работы автопарка с учетом их взаимодействия и ограничений Европейского соглашения о работе бригад. Трансп. Пробл. 2020, 15, 157–170. [Перекрестная ссылка]
13. Ву, Л.; Ван, С. Точные и эвристические методы решения проблемы планирования параллельных машин с многопроцессорными задачами. Межд. Дж. Прод. Экон. 2018, 201, 26–40. [Перекрестная ссылка]
14. Элидрисси А.; Бенмансур, Р.; Бенбрагим, М.; Дювивье, Д. Математические формулировки планирования параллельных машин. проблема с одним сервером. Межд. Дж. Прод. Рез. 2021, 59, 6166–6184. [Перекрестная ссылка]
15. Ган, Х.С.; Вирт, А.; Абдеходаи, А. Алгоритм ветвей и цен для общего случая планирования параллельных машин с одиночный сервер. Вычислить. Опер. Рез. 2012, 39, 2242–2247. [Перекрестная ссылка]
16. Ким, МЮ; Ли, Ю. Х. Модели MIP и гибридный алгоритм для минимизации времени выполнения задачи планирования параллельных машин с одним сервером. Вычислить. Опер. Рез. 2012, 39, 2457–2468. [Перекрестная ссылка]
17. Хасани, К.; Кравченко С.А.; Вернер, Ф. Минимизация помех при планировании двух параллельных компьютеров на одном сервере. Межд. Дж. Прод. Рез. 2014, 52, 7148–7158. [Перекрестная ссылка]
18. Кош, С.; Бек, Дж. К. Новая модель MIP для параллельного пакетного планирования с неидентичными размерами заданий. В материалах по интеграции методов искусственного интеллекта и ИЛИ в программировании с ограничениями: 11-я Международная конференция (CPAIOR 2014), Корк, Ирландия, 19–23 мая 2014 г.; Известия 11; Springer: Берлин/Гейдельберг, Германия, 2014 г.; стр. 55–70.
19. Шарма, А.; Чжао, Ф.; Сазерленд, Дж. В. Экономическое планирование производственного предприятия, работающего по тарифу на электроэнергию по времени использования. Дж. Чистый. Прод. 2015, 108, 256–270. [Перекрестная ссылка]
20. Баяны, Д.; Ся, Х.; Чжан, Л. Модель MILP для планирования работы экскаваторов с целью минимизации расхода топлива. Энергетический процесс 2017, 105, 2739–2745. [Перекрестная ссылка]
21. Че, А.; Чжан, С.; Ву, Х. Энергосберегающее несвязанное планирование параллельных машин в соответствии с тарифами на электроэнергию по времени использования. Дж. Чистый. Прод. 2017, 156, 688–697. [Перекрестная ссылка]
22. Ченг Дж.; Чу, Ф.; Чжоу, М. Улучшенная модель планирования параллельных машин с учетом цены на электроэнергию по времени использования. IEEE Транс. Автомат. наук. англ. 2017, 15, 896–899. [Перекрестная ссылка]
23. Элидрисси, А.; Бенмансур, Р.; Бенбрагим, М.; Дювивье, Д. Формулировки MIP для идентичной задачи планирования параллельных машин с одним сервером. В материалах 4-й Международной конференции по оптимизации и приложениям (ICOA) 2018 г., Мохаммедия, Марокко, 26–27 апреля 2018 г.; IEEE: Пискатауэй, Нью-Джерси, США, 2018 г.; стр. 1–6.
24. Кааби Дж.; Харрат, Ю. Планирование на однородных параллельных машинах с периодическими ограничениями недоступности. Межд. Дж. Прод. Рез. 2019, 57, 216–227. [Перекрестная ссылка]
25. Кирос Сильва, ЕЕ Programación por Objetivos para el Dimensionamiento и la Asignación de Una Flota de Camiones en una Эмпреса Минера. Интерфазы 2021, 87–112. [Перекрестная ссылка]
26. Сюй, CL; Ляо, Дж. Р. Две задачи планирования параллельных машин с функциональными ограничениями. Дискретный. Дин. Нат. Соц. 2020, 2020, 1–6. [Перекрестная ссылка]

Machine Translated by Google

27. Тапиа, Э.; Салазар Арайя, А.; Сааведра, Н.; Неринг, М.; Мора, Дж. Анализ стратегии использования полных грузовиков в сравнении с полными ковшами при погрузочно-транспортных операциях на открытых горных работах. Межд. Дж. Мин. Восстановить. Окружающая среда. 2021, 35, 1–11. [Перекрестная ссылка]
28. Тигрерос Карденас, ID Propuesta Para la Programación de Producción en Maquinas Paralelas con Tiempos de Alistamiento en una Empresa del Sector Agroindustrial. 2021. Доступно онлайн: https://repositorio.uceva.edu.co/bitstream/handle/20.500.129. 93/2744/T00031983.pdf?sequence=1&isAllowed=y (по состоянию на 29 июля 2024 г.).
29. Юнусоглу, П.; Топалоглу Йылдыз, С. Подход к программированию с ограничениями для задачи планирования несвязанных параллельных машин с несколькими ограничениями ресурсов и временем настройки, зависящим от последовательности. Межд. Дж. Прод. Рез. 2022, 60, 2212–2229. [Перекрестная ссылка]
30. Ма, Х.; Фу, Ю.; Гао, К.; Чжу, Л.; Садолла, А. Многоцелевая задача планирования и маршрутизации для служб здравоохранения на дому посредством оптимизации мозгового штурма. Комплексная система. Модель. Симул. 2023, 3, 32–46. [Перекрестная ссылка]
31. Фэн, Л.; Чен, Г.; Чжоу, С.; Чжоу, Х.; Джин, М. Энергоэффективная несвязанная проблема планирования параллельных машин с пакетной обработкой и ценами на электроэнергию по времени использования. Математика 2024, 12, 376. [CrossRef]
32. Ху, Б.; Сюн, З.; Солнце, А.; Юань, Ю. Планирование движения контейнерных транспортных средств в угольных шахтах на основе гибридного алгоритма GA – GWO. Прил. наук. 2024, 14, 3986. [CrossRef]
33. Эйвази, Х.; Аскари-Насаб, Х. Модель смешанного целочисленного линейного программирования для краткосрочного планирования добычи на открытых рудниках. Мин. Технол. 2012, 121, 97–108. [Перекрестная ссылка]
34. Аджомшоа, Х.; Альбрехт, А.; Ли, Д.; Падни, П. Эффективная транспортировка автотранспортом в подземных шахтах. Мин. Технол. 2012, 121, 83–90. [Перекрестная ссылка]
35. Глигорич З. Оптимизация системы транспортировки в подземной небольшой шахте с использованием нечетких множеств. Системный дизайн. п. 63. Доступно онлайн: https://www.researchgate.net/publication/287397565_Optimization_of_haulage_system_in_an_underground_ . small_scale_mine_using_fuzzy_sets (по состоянию на 29 июля 2024 г.).
36. Гонен А.; Малли, Т.; Козе, Х. Выбор системы транспортировки руды для подземного металлоносного рудника/Dobór systemu Transportu руд с копальни подземней руд металу. Арх. Мин. наук. 2012, 57, 779–785. [Перекрестная ссылка]
37. Эйвази, Х.; Аскари-Насаб, Х. Иерархическая модель оптимизации планирования добычи на открытых рудниках. Межд. Дж. Мин. Шахтер. англ. 2012, 4, 89–115. [Перекрестная ссылка]
38. Том Сокаррас, Ж. Н. Арбол по принятию решений по действиям и контролю внутренних расходов на транспортировку. Бакалаврская диссертация, Университет Чили, Сантьяго, Чили, 2014 г.
39. Уribe Нейра, П.А. Анализ функционирования и разработки полуавтономной эксплуатации грузового оборудования; Acarreo y Descarga en Minería Subterránea Погрузочно-разгрузочная машина (LHD). Бакалаврская диссертация, Чилийский университет, Сантьяго, Чили, 2014 г.
40. Вичек, Д.; Бурдук, А.; Курич И. Использование ИНС в повышении эффективности и обеспечении стабильности добычи медной руды процесс. Акта Монтан. Словака 2019, 24, 1–14.
41. Уист Гойенече, Дж. Р. Модель программных операций в Мине и Сьело-Абиерто: Применение в организации Корона. Магистерская диссертация, Университет Ла Сабаны, Чиа, Колумбия, 2019.
42. Соломон Л.; Ортис А. Концептуальная модель систем погрузки и транспортировки руды открытым способом. унив. наук. Ю Технол. 2020, 24, 41–50.
43. Гаспар С.; Жасмин, К. Большая производительность оборудования для добычи полезных ископаемых и транспортировки полезных ископаемых и монтирования на Вете Гавиа – Нивель 100, Unidad Minera Huarón. Бакалаврская диссертация, Континентальный университет, Лос-Оливос, Перу, 2020.
44. Ли, Н.; Фэн, С.; Йе, Х.; Ван, К.; Цзя, М.; Ван, Л.; Чжао, С.; Чен, Д. Модель оптимизации диспетчеризации транспортного оборудования между забоями на основе краткосрочного планирования ресурсов рудника. Металлы 2021, 11, 1848. [CrossRef]
45. Илия, К.; Джозеф, Г.; Сэмюэл, М.; Маути, Д. Оптимизация системы транспортировки экскаватор-самосвал на открытом карьере с использованием подхода к организации массового обслуживания. Араб. Дж. Геоски. 2021, 14, 973. [CrossRef]
46. Уиса Суфо, А. Оптимизация грузового и грузового оборудования на трамо-ботадерос-Тритурадора Тиссен Крупп, для увеличения производства на цементном заводе. Бакалаврская диссертация Universidad Tecnológica Del Perú, Лима, Перу, 2021.
47. Цой, Ю.; Нгуен, Х.; Буй, XN; Нгуен-Той, Т. Оптимизация производительности системы самосвалов для добычи руды на открытых рудниках с использованием больших данных и методов машинного обучения. Ресурс. Политика 2022, 75, 102522. [CrossRef]
48. Ли, Н.; Фэн, С.; Лей, Т.; Йе, Х.; Ван, К.; Ван, Л.; Цзя, М. Оптимизация плана перепланирования подземного горнотранспортного оборудования на основе моделирования случайных поломок. Устойчивое развитие 2022, 14, 3448. [CrossRef]
49. Аболгасемян, М.; Канафи, АГ; Данешманд-Мехр, М. Многокритериальная оптимизация системы транспортировки карьеров на основе моделирования : модифицированный метод NBI и подход метамоделирования. Сложность 2022, 2022, 3540736. [CrossRef]
50. Шамси, М.; Пуррахимян, Ю.; Рахманпур, М. Оптимизация планирования добычи на открытых рудниках с учетом оптимальной системы транспортировки между автомобильными перевозками и полумобильным дроблением и транспортировкой в карьере. Межд. Дж. Мин. Восстановить. Окружающая среда. 2022, 36, 142–158. [Перекрестная ссылка]
51. Паричехре, М.; Голизаде, Х.; Фатоллахи-Фард, А.; Вонг, К. Энергоэффективная несвязанная проблема планирования параллельных машин с эффектом обучения операторов и ухудшением рабочих мест. Межд. Дж. Энвайрон. наук. Технол. 2024, 1–26. [Перекрестная ссылка]
52. Грэм, Р.Л.; Лоулер, Эл.; Ленстра, Дж. К.; Кан, А.Р. Оптимизация и аппроксимация в детерминированном секвенировании и планировании: обзор. В Анналах дискретной математики; Elsevier: Амстердам, Нидерланды, 1979 г.; Том 5, стр. 287–326.
53. Пинедо, М.; Хадави, К. Планирование: теория, разработка алгоритмов и систем. В Proceedings of the Operations Research Proceedings 1991: Материалы 20-го ежегодного собрания/Vorträge der 20. Jahretagung; Springer: Берлин/Гейдельберг, Германия, 1992 г.; стр. 35–42.
54. Ю, К.; Ян, Дж. Модель MILP и алгоритм скользящего горизонта для планирования работы кранов в гибридном контейнерном терминале. Математика. Пробл. англ. 2019, 2019, 4739376. [CrossRef]

Machine Translated by Google

55. Таразона-Торрес, Ю.Л.; Амайя-Гио, Калифорния; Альварес-Мартинес, Д. Асинхронная группа для решения задачи планирования цеха потока: практический пример . В материалах 4-й Южноамериканской международной конференции по промышленному проектированию и управлению операциями, Лима, Перу, 9–11 мая 2023 г.
56. Гуроби Оптимизация, ООО. Параметры оптимизатора Gurobi версии 10.0.2; Gurobi Optimization, LLC.: Бивертон, Орегон, США, 2024 г.
57. Куэльяр-Усакен, Д.; Паласио, А.; Оспина, Э.; Ботеро, М.; Альварес-Мартинес, Д. Моделирование и решение проблемы обрезки конечных точек . Межд. Пер. Опер. Рез. 2023, 30, 800–830. [Перекрестная ссылка]
58. Хоу, Ю.; Ван, Х.; Хуанг, Х. Многоцелевой эволюционный алгоритм на основе Q-обучения для интегрированного экологически чистого производства и проблемы с планированием распределения. англ. Прил. Артиф. Интел. 2024, 127, 107434. [CrossRef]

Отказ от ответственности/Примечание издателя: Заявления, мнения и данные, содержащиеся во всех публикациях, принадлежат исключительно отдельному автору(ам) и соавторам(ам), а не MDPI и/или редактору(ам). MDPI и/или редактор(ы) не несут ответственности за любой вред людям или имуществу, возникший в результате любых идей, методов, инструкций или продуктов, упомянутых в контенте.