

# Анализ литературы на тему: "Современные реализации алгоритма дифференциальной эволюции и их применение"

Рублева Валерия БЭАД223

10 декабря 2023 г.

<https://www.overleaf.com/read/ypgtmwtdzgws#a7c6d6> - ссылка на overleaf

## 1 Вступление

Целью данного анализа является ознакомление с наиболее актуальными на сегодняшний день исследованиями, в которых задача оптимизации решается с помощью дифференциальной эволюции, с реализацией алгоритма и достигнутыми в этих исследованиях результатами.

Дифференциальной эволюцией называется алгоритм для многомерной оптимизации, основанный на принципах генетических алгоритмов с использованием случайных чисел. Он нужен для решения сложных оптимационных задач, в которых много переменных и ограничений, из-за чего получение точного ответа с использованием аналитических методов поиска оптимума либо невозможно, либо неэффективно.

Основные этапы дифференциальной эволюции:

1. Инициализируется определенное количество случайных индивидов - векторов, каждый из которых представляет собой набор параметров, для оптимизации. Эти параметры могут быть представлены вещественными числами.
2. Для каждого индивида в популяции создается мутантный индивид. Это делается путем комбинирования параметров из нескольких случайно выбранных индивидов с использованием оператора мутации. Обычно оператор мутации может выглядеть так:  $mutant = a + mutcoeff * (b - c)$
3. Мутантный индивид объединяется с исходным индивидом с использованием оператора кроссовера (рекомбинации). Этот оператор помогает определить, какие параметры оставить от исходного индивида и какие параметры заменить мутантными значениями.

4. С помощью целевой функции оценивается качество потомка, полученного после кроссовера. Как правило, эта функция перевод вектор в вещественное число. Чем меньше это число, тем лучше качество индивида.
5. С помощью функции селекции происходит оценка качества потомка и исходного индивида, лучший из них становится частью следующего поколения.
6. Повторение шагов 3-5 до достижения требуемой точности или количества итераций.

Алгоритм не гарантирует нахождение глобального оптимума, но при существенном количестве итераций предоставляет решение с высокой точностью, которое будет соответствовать требованиям. Основным недостатком данного алгоритма является его долгая работа, особенно при большом количестве итераций.

## 2 Анализ литературы

Одно из современных исследований на эту тему применяет модифицированный алгоритм дифференциальной эволюции (LSHADE-ORSP) [9] для решения проблемы экономической диспетчеризации, то есть для определения оптимальной мощности по производству электроэнергии с учетом ограничений и при минимально возможных затратах. Подобные задачи характеризуются большим набором переменных и нелинейными ограничениями, поэтому основой для оптимизации была выбрана именно дифференциальная эволюция.

В данном исследовании авторы предоставили улучшенную версию алгоритма LSADE, представленного в 2014 году. Для решения поставленной задачи авторы исследования применили метод SHA (Success History Adaptation). При его выборе авторы ссылаются на другие исследования, в которых доказана эффективность этого подхода. Механизм SHA используется для корректировки числовых значений коэффициентов кроссовера в процессе алгоритма. Если в результате мутации и оценки целевой функцией выявлено улучшение параметров вектора, значения коэффициентов кроссовера сохраняются как успешные и затем используются для подсчета новых значений в следующих итерациях. Также авторы изменили процесс мутации, добавив в него ранжирование по значениям целевой функции. После этого одним из векторов, участвующих в создании нового индивида, всегда был лучший представитель популяции. Для тестирования алгоритма было предложено десять задач экономической диспетчеризации, результаты сравнивались с другими методами поиска оптимума (GA-MPC Elsayed

2011, ENDML DE Mallipeddi 2011, DE-ACr Reynoso-Meza 2011, LSHADE Tanabe 2014). Представленный алгоритм показал результаты лучше, чем взятый за основу алгоритм LSADE. В сравнении с другими методами алгоритм предложил лучшее решение только для одной из десяти проблем. Однако, в других случаях большинство решений оказались близки к лучшему. Если рассматривать средние значения для каждого из методов, алгоритм оказался лучше в четырех случаях из десяти.

Другое исследование [3] было направлено на создание новой стратегии мутации в дифференциальной эволюции. В этой статье предложена улучшенный алгоритм мутации, именуемая стратегией self-competitive mutation, для алгоритмов DE. Это модифицирует мутацию таким образом, чтобы движение всегда было в сторону членов с лучшими целевыми-показателями. Если в мутации вычитается два случайно выбранных индивида для формирования вектора разностей, то он умножается на управляющий параметр  $w$ . Параметр выбирается таким образом, чтобы гарантировать, что всегда более слабый член вычитается из выбранного. Таким образом обеспечивается то, что алгоритм всегда будет стремиться к поиску регионов с более высокой вероятностью улучшения решений.

Эта стратегия мутации затем применяется к нескольким основным и современным алгоритмам DE, которые широко использовались в последние годы, без изменения других параметров и структуры выбранных алгоритмов - jDE, EPSDE, SaDe, JADE. Усовершенствованные и оригинальные варианты алгоритмов DE были использованы для решения 25 реальных тестовых функций параметров, а также оптимальной настройки ПИД-регуляторов для системы автоматического регулирования напряжения. В представленных тестах заметна эффективность добавленной стратегии мутации в случае каждого из алгоритмов дифференциальной эволюции, о чем также автор говорит в заключении.

Развитие современных энергосистем требует рассмотрения вопроса о том, как согласовать компоненты в системе возобновляемой энергии с хранением. В системах хранения энергии существуют такие проблемы, как стоимость инвестиций, безопасность и срок службы. Алгоритмы DE имеют широкое применение в сфере энергетики. Это подтверждает еще одна проанализированная мной статья [4]. Данное исследование концентрирует свое внимание на применении алгоритмов искусственного интеллекта и алгоритма дифференциальной эволюции для оптимизации работы систем возобновляемой энергии, потому что авторы считают, что существует потребность в гибкой и надежной энергосистеме, способной управлять источниками энергии и подстраиваться под различные обстоятельства. В этом исследовании представлена практическая стохастическая многоцелевая оптимизация фотоэлектрической-аккумуляторной-гидроэнергетической си-

стемы с использованием стандартного DE и PSO (также является стохастическим алгоритмом для оптимизации), которые программируются с помощью программного обеспечения MATLAB. Оба алгоритма минимизировали значение LCOE - levilised cost of energy. Результаты тестирования показывают, что лучшим алгоритмом был DE с конвергенцией после 19 итераций и LCOE 0.06192\$/kWh, в то время как оптимальный LCOE, полученный из PSO, был 0.06358 \$/kWh. Так как алгоритм дифференциальной эволюции основан на случайных чисел. Проведение от 10 до 20 тестирований с одинаковыми и разными входными данными было бы более информативным в оценке эффективности рассматриваемых подходов. Дальнейший статистический анализ автора показал, что все методы оптимизации имеют удовлетворительную производительность и силу с преобладанием алгоритма PSO для надежности питания и DE для экономической эффективности. В исследовании рассматривается стандартный алгоритм дифференциальной эволюции, который может работать достаточно долго и не всегда выдавать точные результаты. В настоящий момент существуют доказанно более эффективные реализации DE, которыми можно воспользоваться для реализации подобной системы.

В алгоритме DE на большом количестве итераций происходит увеличение вычислительной стоимости, поскольку существующий оператор мутации не может обеспечить большее разнообразие.

Сравнение DE и PSO также происходит в исследовании, описывающем возможность применения алгоритма дифференциальной эволюции для создания системы общественных электрических автобусов с наименьшими затратами [6]. Условия эксплуатации, места остановки, траектория движения автобусов и их скорость рассматриваются как параметры для оценки мощности зарядных станций на основе фактических данных о маршруте. Целевой функцией являются затраты на станцию зарядки в сочетании с затратами на эксплуатацию аккумулятора в пределах заранее установленных ограничений. В качестве модели для анализа была выбрана существующая электроавтобусная система Университета Хон Каен. Поставленная задача решалась двумя методами - стандартным алгоритмом дифференциальной эволюции и методом PSO. По сравнению с решением метода PSO, метод DE смог достичь на 14,21% более низкой стоимости, чем метод PSO. Кроме того, метод DE может обеспечить решение с меньшим числом циклов движения автобусов, сохраняя хорошее обслуживание в течение срока службы. Однако эта работа представляет собой лишь имитацию системы электроавтобусов с упрощенными условиями эксплуатации. Можно также рассмотреть такие параметры, как периоды заторов дорожного движения и множественные маршруты.

В ходе работы мною было прочитано несколько статей, в которых DE опти-

мизация применяется в сфере транспорта. Еще одно из таких исследований [10] рассматривает применение алгоритма дифференциальной эволюции для решения проблем с транспортировкой в момент их появления. Проблема транспортного планирования в данном исследовании - это проблема с множеством факторов, включая многочисленных производителей, источники сырья, распределительные центры и типы грузовиков. Глобальные транспортные компании концентрируют свое внимание и усилия на снижении транспортных издержек за счет оптимального планирования перевозок. Например, транспортное средство будет использоваться для сбора сырья, необходимого для производственного предприятия, у поставщика для доставки на производственный завод таким образом, чтобы снизить транспортные расходы для всей системы. Оптимизированная маршрутизация выполняется при обнаружении новой информации. Информация, которая обновляется, получается из веб-приложения и процесс оптимизации выполняется с помощью модифицированного алгоритма DE. В настоящее время, большие и сложные задачи трудно решить точно с помощью общего алгоритма, потому что процесс занимает много времени.

Первоначально для процесса выбора в дифференциальной эволюции алгоритм принимал только лучшее решение, но в этой статье представлены четыре новые формулы выбора, которые могут принять решение, которое хуже, чем текущее лучшее решение. Авторы данной статьи сконцентрировали свое внимание на проблеме остановки алгоритма на локальном оптимуме. Чтобы увеличить возможность выхода из локального оптимального решения и продолжить поиск глобального оптимума, вводится новый параметр  $P$  - вероятность выбора худшего индивида во время селекции. Этот параметр генерируется рандомно и остается неизменным в течении работы программы.

Данная модификация была протестирована на 16 рандомно сгенерированных данных о транспортировке. При этом тесты были разделены на три группы, в которых количество итераций алгоритма было 20,000, 40,000, и 100,000. Было предложено четыре метода, основанных на алгоритме дифференциальной эволюции (DE-AC1, DE-AC2, DE-AC3, DE-AC4), с использованием вероятности принятия текущего решения в качестве нового целевого вектора, когда он имеет более низкое качество, чем текущий целевой вектор. В исследовании представлены графики, где показано, как менялось целевое значение лучшего индивида в популяции на итерациях от 0 до 100.000. Это позволило заметить, что метод DE-AC3, начиная примерно с 5.000 до 50.000 итерации находился в локальном оптимуме, в то время как базовый алгоритм эволюции и три новых предложенных метода смогли получить значения лучше. Это говорит о том, что модификация алгоритма не гарантирует решение проблемы локального оптимума, а только снижа-

ет вероятность её возникновения. После анализа результатов тестирования был выбран лучший из четырех методов - DE-AC2. При получении новой информации решения о направлении транспортных средств, сделанные алгоритмом DE-AC2, выполнялись непрерывно и более эффективно.

Стоит заметить, что сравнение модифицированного алгоритма со стандартным алгоритмом DE может быть мало информативным, потому что на момент написания статьи уже были более эффективные реализации этого алгоритма. При сравнении с традиционными способами планирования пути издержки компаний увеличились на 8.47%, но поиск решения занял на 99.93% меньше времени.

Проблема остановки DE в локальном оптимуме оказалась очень актуальной на сегодняшний день. Ниже будет рассмотрено еще несколько статей, рассматривающих эту тему.

В этом исследовании [8], предложен новый вариант DE, включающий оператор мутации на основе адаптации гомеостаза - биологического процесса поддержания постоянства внутренней среды организма. (HABDE), который поддерживает разнообразие, когда алгоритм находится в локальном оптимальном решении. Основной целью авторов статьи является поддержание разнообразия популяции даже на поздних итерациях алгоритма DE. Ответственность за поддержание разнообразия в DE лежит на операторе мутаций, которому уделяется внимание в рамках предлагаемого подхода. Для решения поставленной задачи вводится новый параметр -  $HA_j$ , который принимает значения от 0.1 до 1. После этого каждый вектор-индивидуум умножается на этот параметр и считаются homeostasis adaption based vectors (HAVs). Подобная стратегия может также обеспечить новых индивидов для дальнейшей эволюции. В отличии от метода, предложенного в статье выше, здесь два из трех векторов, участвующих в операторе мутации, подвергаются изменениям. Оператор мутации выглядит так:

$$mutant = x_{rand} + \delta(HAV_1 - HAV_2)$$

где  $x_{rand}$  обозначает случайный вектор текущей популяции.

Этот способ кажется более надежным в борьбе с проблемой локального оптимума, чем решение, описанное в статье выше. Но оно будет работать дольше, потому что приходится взаимодействовать с большим количеством индивидов. Модифицированный алгоритм применяется для оценки затрат при разработке программного обеспечения. При тестировании результат алгоритма HABDE сравнивается с результатами других реализаций дифференциальной эволюции - JADEcr, ToPDE, TSDE, MPEDE, CPI-DE. Здесь, в отличии от исследования выше при тестировании сравниваются разные реализации дифференциальной эволюции. Для более информативного анализа рассматриваются значения целевой функции на разных

итерациях алгоритмов. Можно заметить, что предложенный алгоритм действительно показывает более эффективные решения.

DEHeO [7] - другой алгоритм борьбы с проблемой локального оптимума, также вдохновленный процессом гомеостаза представлен в статье. Авторы этой статьи уделяют больше внимания сходству этого метода с биологическим процессом. Большая часть их исследования уделена описанию биологического процесса гомеостаза с целью доказать его эффективность для дифференциальной эволюции. В этом исследовании предлагают модифицировать алгоритм мутации таким образом:

$$\text{mutant} = x_{best} + w(x_{HeV_1} + x_{HeV_2})$$

где  $x_{HeV_1}$  и  $x_{HeV_2}$  - два вектора, выбранных рандомно из отсортированной популяции, состоящей только из 50% лучших индивидов.  $x_{best}$  обозначает лучшего индивида в данной группе.

Получается, вся популяция делится на Revision pool и Reminder pool. Предполагается, что индивиды, попавшие в Revision pool имеют значения целевой функции лучше, чем те, кто попал в Reminder pool. Обе группы проходят эволюцию отдельно, чтобы произвести потомка 1 и потомка 2. То есть описанные выше действия мутации повторяются так же и в группе Reminder pool. В итоге получаются два мутанта из разных групп и создается два потомка, далее алгоритм использует стандартную DE. Можно заметить, что здесь модифицируется именно лучший индивид, в то время как в алгоритме HABDE - случайный.

В результате тестирования DEHeO и сравнения скорости сходимости с такими реализациями DE, как JADE, DEAE, CPI-DE, MVDE, TSDE, Deb и стандартной DE алгоритм показал один из самых высоких результатов. Было доказано, что предложенный подход мутантного вектора гомеостаза повышает скорость сходимости при исследовании поискового пространства.

Модификации мутации, предложенные в рассмотренных статьях, действительно будут лучше справляться с этой проблемой попадания в локальный оптимум. Потому что добавляются параметры, обеспечивающие большую случайность и, как следствие, большее разнообразие в популяции. Для получения лучшего результата можно было сравнить разработанные алгоритмы с другими возможными решениями этой проблемы, например, сравнить подходы DEHeO [7], HABDE [8] и DE-AC2[10] между собой. А также обратить внимание на время работы каждого из алгоритмов.

Другой подход к решению проблемы локального максимума и разнообразия популяции называется CODE [1]. В отличии от других методов CODE направлен на модификацию именно инициализации популяции. Модифицированная схема инициализации использует преимущества как хаотиче-

ских карт, так и стратегии OBL. В частности, хаотическая карта используется для решения проблем преждевременной конвергенции путем выработки первоначальных решений с более высокими уровнями разнообразия. Стратегия OBL направлена на ускорение скорости конвергенции алгоритма путем исследования более широких областей пространства решений в процессе инициализации. В CODE включены в общей сложности семь хаотических карт, чтобы оценить их пригодность для решения различных типов задач оптимизации и вариант CODE, интегрированный с круговой картой: Chaotic Sine-Oppositional DE (CSODE), Chaotic Chebyshev-Oppositional DE (CCBODE), Chaotic Iterative-Oppositional DE (CIODE), Chaotic Logistic-Oppositional DE (CLODE), Chaotic Sinusoidal-Oppositional DE (CSSODE), Chaotic Tent-Oppositional DE (CTODE) and Chaotic Circle-Oppositional DE (CCODE). После тестирования был сделан вывод, что вариант с круговой картой показал себя лучше всего. Затем в исследовании сравнили решения алгоритма CCODE с другими реализациями дифференциальной эволюции - BDE, ODE, OMLDE, CJADE, DEGLS, CCODE с точки зрения точности решения, его эффективности и времени расчета. Тестирование показало, что CCODE может значительно опережать другие алгоритмы, из чего был сделан вывод об эффективности модифицированной схемы инициализации.

В конце статьи авторы также рассказывают о возможных темах для дальнейших исследований. Например, возможность использования других видов хаотических карт.

Другой важной сферой, в которых стали использовать алгоритм DE является компьютерная безопасность.

В данной статье [5] рассматривается проблема безопасности в сети и обнаружении угроз для компьютера. Традиционные методы обнаружения, по мнению авторов, могут иметь низкую скорость обнаружения и низкую точность. Основываясь на результатах других исследований, авторы делают вывод о том, что алгоритм MODE (Multi-objective DE) может быть применен в этой сфере. По сравнению с традиционными алгоритмами дифференциальной эволюции MODE алгоритмы могут не только оптимизировать значение целевой функции, но и генерировать набор нескольких решений. Основная цель алгоритма - найти набор решений, который максимально охватывает глобальное оптимальное решение. Таким образом, алгоритм Multi-objective DE должен решать две основные проблемы: как оценить и сортировать эти наборы решений, и как генерировать детерминированные и разнообразные наборы решения, не уступающие друг другу. Каждый индивид в данном алгоритме имеет сразу несколько значений по нескольким целевым функциям.

Алгоритм MODE в исследовании используется в системе обнаружения вторжений IDS для мониторинга сетевого трафика и обнаружения попыток вторжения. IDS - это технология, используемая для обнаружения и предотвращения сетевых атак. В области сетевой безопасности, IDS является важной технологией, которая играет решающую роль в защите компьютерной сетевой безопасности. В ней MODE может быть использована для анализа состояния трафика и поиска вторжений.

Для проверки эффекта применения алгоритма были отобраны набор данных CNS и набор данных KDD CUP 1999. В которые входят данные по четырем аспектам: ноутбуки, сетевые устройства, серверы и конечные пользователи. Перед тем как проводить тестирование, данные были обработаны. Эффективность была протестирована в сравнении с другими алгоритмами для обнаружения атак - NSGA-II и SPEA2. Анализ показал, что модель обнаружения вторжений, разработанная с использованием алгоритма дифференциальной эволюции, может достичь более высокой скорости обнаружения, что делает его применение в системах обнаружения атак компьютерных сетей оправданным.

Другой алгоритм, способный решать проблемы локального оптимума, также примененный в IDS называется Differential Evolution with Maturity Extension [Article10] (DE-ME). Целью авторов этого исследования также было применения алгоритмов дифференциальной эволюции в системе IDS, они предлагают использовать DE-ME с целью выбора оптимального количества релевантных признаков из массивных наборов данных IDS. В ходе работы авторы освещают проблему DE сходимости к локальному оптимуму и предлагают свой вариант модификации оператора мутации для ее решения и общего увеличения точности алгоритма.

$$mutant_i = sign(x_{best}) \cdot \left| \frac{x_{best} - x_{current}}{x_{best} + x_{current}} \right| + w(x_{random_1} - x_{random_2})$$

где  $sign(x) = 1 \text{ if } x > 0, -1 \text{ if } x < 0, 0 \text{ if } x = 0$  Эта модификация, в отличии от описанных выше, не добавляет новые случайные параметры в оператор мутации.

При тестировании DE-ME оценивался уровень точности поиска релевантных признаков, результаты сравнивались с такими алгоритмами, как DT, ET, NB и SVM-G. Анализ показал, что точность предложенного метода составила 99,66 %, по сравнению со точностью DE-ELM 80,15 %, точностью LNNLS-KH 96,12 %, точностью GWO-GA 99,44 %, точностью FF-WMOGWO - 98 % и ANN-бинарным классом - 81,2 %. Для того, чтобы использовать этот алгоритм для работы в качестве выбора признаков для IDS, авторы предлагают идею использования обычного алгоритма DE с синусо-

иальным коэффициентом кроссовера  $\beta$  и синусоидального кроссовера- $\beta$  с DE-ME

Применение алгоритма дифференциальной эволюции в этой сфере действительно имеет большой потенциал. Здесь предоставляется значительная основа для будущих исследований, потому что DE является достаточно гибким алгоритмом, который можно совершенствовать для решения все более сложных проблем с сетевой безопасностью.

Некоторые современные исследования направлены именно на улучшение качества работы алгоритма без рассмотрения его практического применения. В одном из подобных исследований применение новой стратегии мутации и кроссовера [2]. Ранее уже были представлены исследования, в которых рассматривались некоторые стратегии мутации. Различные мутационные стратегии способствуют достижению большего разнообразия решений. В статье представлена новая мутационная стратегия, которая использует позицию наилучшего найденного решения, позицию текущего решения и другое случайно выбранное решение из популяции. Эта стратегия называется DE/current-to-best/2. В предложенной мутационной стратегии новая позиция вычисляется пропорционально наилучшему решению, найденному путем умножения случайного веса на разницу между текущим решением и лучшим и другого случайного веса на разницу между случайно выбранным решением и лучшим:

$$\text{mutant}_i = x_{best} + w(x_{best} - x_i) + w(x_{best} - x_{random})$$

Параметр  $w$  выбирается как нормально распределенное случайное число, включающее в себя как отрицательные, так и положительные значения, что способствует увеличению разнообразия компонентов мутированного вектора.

Другая модификация алгоритма DE в данной работе вводит новую стратегию самоадаптации, основанную на процедуре кроссовера алгоритма. Эта процедура кроссовера включает в себя либо генерацию нового разнообразного решения с использованием более высокой вероятности кроссовера, что позволяет получить больше характеристик от мутированного решения, найденного с помощью DE/current-to-best/2, либо генерацию нового решения, которое включает в себя больше характеристик из текущего решения.

Для тестирования предложенных модификаций было отобрано 11 оптимизационных задач. Кроме того, в анализ результатов производился с помощью таких статистических методов, как дисперсионный анализ ANOVA и тест Тьюки. В сравнительном анализе приведены результаты реализации предложенного алгоритма DE с его оптимизированными параметрами на

11 функциях тестовой оптимизации. Сравнение проводится с тремя различными версиями классического алгоритма DE. Модифицированный алгоритм показал лучшие результаты на 8 функциях из 11, из чего авторы делают вывод об эффективности данных модификаций и их возможных применениях в будущем. Для более корректных выводах об эффективности предложенных изменений можно сравнить результаты этого алгоритма с другими современными модификациями, которые также получают решения точнее стандартного алгоритма дифференциальной эволюции.

Метод FastRankDE [11] был предложен исследователями для использования алгоритма дифференциальной эволюции в генерации рекомендаций пользователю. Авторы сосредоточили свое внимание на ускорении работы алгоритма и его интеграцию в создание списка рекомендаций. Предлагаемый метод FastRankDE использует операцию умножения матриц для расчета рейтинга элементов. Для этого нужно хранить популяцию и остальные переменные в виде матриц. Это позволит умножить матрицу индивидуумов и матрицу элементов. Такие операции могут быть проще распараллелены с помощью различных библиотек, что существенно может ускорить программу. Предлагаемые изменения применяются к стандартному алгоритму дифференциальной эволюции, далее проводятся тесты. В большей степени исследовалось именно время работы алгоритма. Был сделан вывод, что по мере увеличения числа итераций время вычисления также увеличивается. Тем не менее, для алгоритма FastRankDE расчеты выполнялись в среднем на 36 % быстрее, чем для алгоритма без модификации. Кроме того, отмечено, что качество рекомендации для обоих алгоритмов практически идентичны.

Для более точных исследований в будущем можно сравнить этот алгоритм с более сложными его аналогами и другими алгоритмами подбора рекомендаций.

### 3 Заключение

В данном обзоре литературы мною было проанализировано 12 статей на тему применения дифференциальной эволюции в разных сферах деятельности, также были рассмотрены модификации с целью улучшения работы алгоритма.

DE алгоритмы действительно являются хорошим способом поиска оптимума в сложных задачах. В современных работах они часто применяются для снижения затрат в сфере энергетики и транспорта, также применение DE развивается в сфере компьютерной безопасности, рассматривается вариант его использования для систем обнаружения вторжений (IDS).

Однако, у DE алгоритмов есть несколько существенных проблем, таких как остановка в локальном максимуме, точность работы или сложность и скорость работы алгоритма. В связи с этим алгоритм постоянно подвергается модификациям. В большей степени исследования направлены именно на решение проблемы локального оптимума, и как следствие, улучшение точности алгоритма. Часто изменения направлены на сохранение разнообразия внутри популяции на всех итерациях работы алгоритма, поэтому предлагаются новые варианты оператора мутации. Модификациям также могут подвергнуться операторы инициализации или кроссовера, но они в меньшей степени влияют на ход работы алгоритма, поэтому эти идеи менее популярны. Современные работы направлены на уменьшение количества постоянных (констант) в алгоритме и добавление случайности или переменных коэффициентов.

Скорость также является важной составляющей алгоритма из-за большого количества итераций и его сложности. На эту тему написано меньше статей. По моему мнению, для развития и применения алгоритма в будущем следует обратить больше внимания на возможные варианты ускорения его работы. Один из таких примеров рассмотрен мной в последней статье.

## Список литературы

- [1] Mohamad Faiz Ahmad, Nor Ashidi Mat Isa и Koon Meng Ang Wei Hong Lim. “Differential evolution with modified initialization scheme using chaotic oppositional based learning strategy”. B: (2022).
- [2] Sadeer Fadhil и др. “A Modified Differential Evolution Algorithm Based on Improving A New Mutation Strategy and Self-Adaptation Crossover”. B: (2023).
- [3] Mojtaba Ghasemi, Abolfazl Rahimnejad и Ebrahim Akbariand S. Andrew Gadsden Milad Gil. “A self-competitive mutation strategy for Differential Evolution algorithms with applications to Proportional–Integral–Derivative controllers and Automatic Voltage Regulator systems”. B: (2023).
- [4] Chu Donatus Iweh и Ebunle Rene Akupana. “Control and optimization of a hybrid solar PV – Hydro power system for off-grid applications using particle swarm optimization (PSO) and differential evolution (DE)”. B: (2023).
- [5] Zhenghong Jiang и Chunrong Zhou. “Application of Multi-objective Differential Evolution Algorithm in Computer Network Intrusion Detection System”. B: (2022).
- [6] Kittiphan Nawakiatikorn, Niphon Kaewdornhan и Warayut Kampeerawat. “An optimal design for public electric bus system based on Differential Evolution Algorithm”. B: (2023).
- [7] Shashi Prabha и Raghav Yadav. “Differential evolution with biological-based mutation operator”. B: (2019).
- [8] Vibhav Prakash Singh и Ashok Kumar Mehta. “Differential evolution using homeostasis adaption based mutation operator and its application for software cost estimation”. B: (2018).
- [9] Vladimir Stanovov, Shahnaz Akhmedova и Eugene Semenkin. “Differential Evolution with Selective Pressure to Economic Dispatch Optimization Problems”. B: (2019).
- [10] Naratip Supattananon и Raknol Akararungruangkul. “Modified Differential Evolution Algorithm for a Transportation Software Application”. B: (2019).
- [11] Michał Balchanowski Urszula Boryczka. “Speed up Differential Evolution for ranking of items in recommendation systems”. B: (2021).