



СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ ПРЕДСТАВЛЕНИЯ ХРОМОСОМЫ В ГЕНЕТИЧЕСКИХ АЛГОРИТМАХ ДЛЯ ЗАДАЧИ СОСТАВЛЕНИЯ УЧЕБНОГО РАСПИСАНИЯ

А. С. Смирнов, студент
E-mail: antonsmirnov2233@gmail.com
ФГБОУ ВО «Калининградский государственный
технический университет»

В статье проводится сравнительный анализ различных подходов к представлению хромосомы в генетических алгоритмах, применяемых для решения NP-трудной задачи составления учебного расписания. Актуальность исследования обусловлена тем, что эффективность генетического алгоритма в значительной степени зависит от способа кодирования решения. На основе анализа 15 российских и зарубежных научных источников систематизированы и рассмотрены три основных метода представления хромосомы: прямое (матричное), косвенное (перестановочное) и компонентное (агрегативное). Для каждого метода проанализированы его преимущества, недостатки и особенности применения в контексте специфических ограничений учебного процесса. Сделан вывод, что выбор оптимального метода представления не является универсальным и должен основываться на конкретных требованиях и ограничениях задачи, а наиболее перспективными для сложных реальных расписаний являются компонентные и гибридные подходы.

Ключевые слова: генетический алгоритм, составление учебного расписания, представление хромосомы, кодирование решения, комбинаторная оптимизация, эвристический алгоритм, фитнес-функция.

ВВЕДЕНИЕ

Задача составления учебного расписания является одной из ключевых и наиболее сложных организационных задач в любом высшем учебном заведении. От качества и оптимальности расписания напрямую зависит не только эффективность использования материальных ресурсов (аудиторного фонда, оборудования), но и качество самого образовательного процесса, а также уровень удовлетворенности всех его участников – студентов и преподавателей [11, 13]. Эффективное расписание минимизирует простоту, обеспечивает равномерную нагрузку и создает комфортные условия для обучения и работы.

Сложность этой задачи обусловлена необходимостью одновременного учета большого количества разнообразных и зачастую противоречивых ограничений. Их принято делить на две категории. Жесткие ограничения – это условия, нарушение которых недопустимо (например, один преподаватель не может вести два занятия одновременно; в одной аудитории в одно и то же время не может проходить более одного занятия; количество студентов в группе не должно превышать вместимость аудитории). Мягкие ограничения – это пожелания, выполнение которых желательно, но не обязательно (например, минимизация количества «окон» в расписании студентов и преподавателей, предпочтения преподавателей по времени работы, проведение лекций в первой половине дня) [2, 8]. С ростом числа курсов, групп, преподавателей и аудиторий пространство возможных решений растет экспоненциально, что относит задачу составления расписания к классу NP-трудных задач [12]. Это делает невозможным нахождение оптимального решения методами полного перебора за приемлемое время даже с использованием современных вычислительных мощностей.

В связи с этим для автоматизации процесса составления расписания широкое применение нашли эвристические и метаэвристические методы, среди которых особое место занимают генетические алгоритмы (ГА). Генетические алгоритмы, основанные на принципах естественной эволюции, таких как отбор, скрещивание (кроссинговер) и мутация, доказали свою высокую эффективность при решении сложных оптимизационных задач. Их преимущество заключается в способности исследовать обширное пространство решений, избегая застревания в локальных оптимумах, и находить решения, близкие к глобально оптимальным [1, 9, 10, 14].

Однако ключевым фактором, определяющим успешность применения ГА, является способ представления (кодирования) потенциального решения задачи в виде структуры данных, называемой хромосомой. От выбранной структуры хромосомы и логики ее интерпретации напрямую зависит эффективность генетических операторов, скорость сходимости алгоритма и, в конечном счете, качество финального решения [4, 7]. Неудачный выбор представления может приводить к частому порождению невалидных (некорректных) решений после операций кроссинговера и мутации, что требует введения сложных и ресурсоемких «ремонтных» процедур. С другой стороны, слишком сложное кодирование может неоправданно усложнить генетические операторы и замедлить работу алгоритма.

В научной литературе предложено множество подходов к кодированию расписания, от простых матричных представлений до сложных агрегированных структур. Несмотря на обилие исследований, отсутствует единый подход к выбору оптимального метода кодирования, а сравнительный анализ существующих методов часто носит фрагментарный характер. Цель данной статьи – провести систематический сравнительный анализ основных методов представления хромосомы в генетических алгоритмах для задачи составления учебного расписания. На основе анализа предоставленных научных источников будут классифицированы существующие подходы, выявлены их сильные и слабые стороны, а также определены области их наиболее эффективного применения в зависимости от специфики и ограничений конкретной задачи.

ОБЪЕКТ ИССЛЕДОВАНИЯ

Объектом исследования является процесс применения генетических алгоритмов для решения задачи автоматизированного составления учебного расписания в высших учебных заведениях. В частности, в качестве непосредственного объекта анализа выступают различные методы представления (кодирования) решения в виде хромосомы. Исследование охватывает не только сами структуры данных (матричные, перестановочные, агрегативные), но и их влияние на ключевые компоненты генетического алгоритма: разработку и эффективность генетических операторов (кроссинговера и мутации), а также на сложность и адекватность функции приспособленности (фитнес-функции). Все эти аспекты рассматриваются в контексте специфических ограничений и требований, присущих задаче составления учебного расписания.

ЦЕЛЬ И ЗАДАЧИ ИССЛЕДОВАНИЯ

Цель исследования заключается в проведении комплексного сравнительного анализа и систематизации методов представления хромосомы в генетических алгоритмах, применяемых для решения задачи автоматизированного составления учебного расписания, с целью выработки рекомендаций по выбору наиболее эффективного подхода в зависимости от специфики задачи.

Для достижения поставленной цели были сформулированы следующие задачи:

- Провести углубленный анализ научной литературы, посвященной применению генетических алгоритмов для автоматизации составления расписаний, чтобы определить текущее состояние исследований в данной области и выявить ключевые проблемы, связанные с кодированием решений.

- Идентифицировать, классифицировать и детально описать основные подходы к кодированию учебного расписания в виде хромосомы, сгруппировав их по фундаментальным принципам (прямое, косвенное, компонентное представление).
- Проанализировать преимущества и недостатки каждого из выявленных методов представления с точки зрения их влияния на ключевые аспекты генетического алгоритма: сложность реализации, эффективность генетических операторов (кроссинговера и мутации), вероятность генерации невалидных решений, а также гибкость в учете разнообразных жестких и мягких ограничений задачи.
- Сформулировать обоснованные выводы и практические рекомендации о целесообразности применения различных методов кодирования в зависимости от характеристик конкретной задачи составления расписания (масштаб учебного заведения, количество и тип ограничений, требуемая скорость поиска решения).

МЕТОДЫ ИССЛЕДОВАНИЯ

Основными методами, использованными в данном исследовании, являются теоретический анализ, систематизация, сравнительный анализ и обобщение информации, представленной в релевантных научных публикациях. Работа носит теоретический характер и не предполагает проведения собственных вычислительных экспериментов, а основывается на глубоком изучении и сопоставлении уже существующих подходов.

Исследовательская база сформирована на основе детального анализа 15 научных статей, диссертаций и материалов конференций, посвященных разработке и применению генетических и гибридных алгоритмов для решения задач планирования и составления расписаний. В ходе исследования были проанализированы предложенные авторами математические модели, структуры данных для представления хромосом, реализации генетических операторов и функции приспособленности. Сравнительный анализ подходов проводился на основе таких критериев, как сложность имплементации, вычислительная эффективность, способность к сохранению и передаче полезных признаков (строительных блоков решения), а также адекватность представления специфических ограничений задачи составления учебного расписания.

РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ

В ходе проведенного анализа научной литературы, посвященной применению генетических алгоритмов для решения задачи составления учебного расписания, были систематизированы и детально изучены основные подходы к представлению решения в виде хромосомы. Результаты исследования показывают, что выбор метода кодирования является фундаментальным этапом проектирования алгоритма, который оказывает прямое влияние на его эффективность, скорость сходимости и качество получаемых решений. Все многообразие подходов, описанных в работах [1–15], можно условно классифицировать по трем основным группам. В частности, прямое (матричное) представление подробно рассматривается в работах [4, 8, 13, 15], косвенное (перестановочное) – в [2, 7], а наиболее современные компонентные (агрегативные) подходы описаны в источниках [5, 6].

Для наглядного сопоставления этих подходов и облегчения их восприятия систематизируем ключевые характеристики в виде сравнительной таблицы.

Как видно из таблицы, выбор метода представления является компромиссом между простотой реализации, вычислительной эффективностью и мощностью самого представления. Далее рассмотрим каждый подход подробнее.

Прямое представление, как наиболее интуитивно понятный метод, предполагает создание структуры данных, напрямую отражающей сетку расписания. В работах А. С. Маслова и Ю. С. Белова [8], К. Н. Subang и соавторов [15], а также в исследовании О. Аки [13] расписание кодируется в виде матрицы или набора векторов, где каждый ген напрямую соответствует определенному временному слоту в конкретной аудитории, а его значением является идентификатор проводимого занятия. Аналогичная матричная структура

используется и для расписания экзаменов в работе М. Ю. Жуковой и В. М. Аль-Габри [4]. Ключевым преимуществом данного подхода является его простота и наглядность, что значительно облегчает процесс декодирования хромосомы в готовое расписание и его визуальную интерпретацию. Однако этот метод обладает существенным недостатком: стандартные генетические операторы кроссинговера и мутации с высокой вероятностью порождают невалидные («нежизнеспособные») решения, нарушающие жесткие ограничения (например, два занятия в одной аудитории или у одного преподавателя одновременно). Это приводит к необходимости введения сложных и вычислительно затратных «ремонтных» процедур для исправления таких хромосом, что в конечном итоге нивелирует простоту самого представления и снижает общую производительность алгоритма.

Таблица – Сравнительный анализ методов представления хромосомы

Критерий	Прямое (матричное) представление	Косвенное (перестановочное) представление	Компонентное (агрегативное) представление
Суть метода	Хромосома напрямую кодирует сетку расписания (время, аудитория, занятие)	Хромосома кодирует последовательность (приоритет) занятий, которые затем размещаются декодером	Хромосома оперирует семантическими блоками (преподаватель, группа, дисциплина)
Простота реализации	Высокая. Интуитивно понятная структура, легко декодируется в готовое расписание	Средняя. Основная сложность переносится на разработку алгоритма-декодера	Низкая. Требует глубокого анализа предметной области для формирования «блоков занятий»
Генерация невалидных решений	Высокая вероятность. Стандартные операторы кроссинговера и мутации часто нарушают жесткие ограничения	Низкая. Генерируются синтаксически корректные перестановки, валидность обеспечивается декодером	Низкая. Операторы работают с осмысленными блоками, что снижает вероятность коллизий
Сохранение «строительных блоков»	Низкая. Стандартные операторы могут разрушать удачные комбинации (например, связку лекции и практики)	Низкая. Алгоритм теряет информацию о структурных взаимосвязях в итоговом расписании	Высокая. Удачные структурные комбинации с большей вероятностью сохраняются и передаются потомкам
Масштабируемость и применимость	Задачи с небольшим числом жестких ограничений, где допустимо использование «ремонтных» процедур	Задачи, для которых можно разработать мощный эвристический декодер, учитывающий все ограничения	Крупномасштабные, реальные задачи с большим количеством взаимосвязанных ограничений

Косвенное представление предлагает иной подход к решению проблемы. В этом случае хромосома кодирует не само расписание, а упорядоченную последовательность (перестановку) событий, которые необходимо запланировать. Фактическое расписание генерируется на отдельном шаге с помощью специального алгоритма-декодера (Schedule Generation Scheme), который последовательно размещает занятия из хромосомы в сетке расписания согласно заданным эвристическим правилам. Такой подход используется в работах П. В. Афонина и О. В. Кокшагиной [2], где хромосома является «последовательностью работ». Более сложная его вариация представлена у В. В. Короткова [7], который использует булевы матрицы предшествования для кодирования относительного порядка работ. Главное достоинство этого метода заключается в том, что классические операторы для перестановок всегда генерируют синтаксически корректные хромосомы, устранивая проблему невалидности на уровне кодирования. Однако вся сложность задачи переносится на алгоритм-декодер, от качества которого теперь полностью зависит итоговое решение. Кроме того, генетические операторы, работая с абстрактной последовательностью, теряют информацию о структурных

взаимосвязях в итоговом расписании, что может затруднять сохранение и передачу удачных комбинаций (строительных блоков).

Наиболее сложным и перспективным для комплексных задач является **компонентное (агрегативное) представление**. Этот метод является развитием прямого подхода, но оперирует не элементарными событиями, а семантически нагруженными, агрегированными объектами. Яркими примерами такого подхода являются «агрегативный» ГА, предложенный А. А. Коробкиным [6], и «композиционный» ГА, разработанный Ю. С. Кабальновым и соавторами [5]. В их работах ген или группа генов представляют собой «блок занятий», который уже инкапсулирует в себе информацию о преподавателе, дисциплине, группе и других атрибутах. Особь в таких алгоритмах часто состоит из нескольких хромосом, каждая из которых отвечает за свой аспект расписания (например, одна за аудитории, другая за время), но все они привязаны к единому списку агрегированных блоков. Этот подход позволяет значительно сократить размерность пространства поиска и оперировать более осмысленными единицами. Генетические операторы, работая с такими блоками, с большей вероятностью сохраняют и передают потомкам удачные структурные комбинации. Основным недостатком является повышенная сложность разработки и необходимость глубокого предварительного анализа предметной области для корректного формирования агрегированных объектов.

Таким образом, сравнительный анализ показывает, что выбор метода представления хромосомы – это поиск компромисса между простотой реализации, вычислительной эффективностью и мощностью представления. Прямые методы подходят для задач с относительно небольшим числом жестких ограничений. Косвенные методы эффективны, когда можно разработать мощный эвристический декодер. Компонентные же подходы являются наиболее мощным инструментом для решения крупномасштабных, реальных задач составления расписания в вузах, где присутствует сложная иерархия и множество взаимосвязанных ограничений.

ЗАКЛЮЧЕНИЕ

В настоящей работе был проведен комплексный теоретический анализ и систематизация методов представления хромосомы в генетических алгоритмах, применяемых для решения NP-трудной задачи составления учебного расписания. Было подтверждено, что выбор способа кодирования является фундаментальным решением при проектировании генетического алгоритма, напрямую влияющим на его производительность, скорость сходимости и качество итогового решения.

В результате анализа научной литературы были выделены и подробно рассмотрены три основных класса методов представления: прямое (матричное), косвенное (перестановочное) и компонентное (агрегативное). Проведенный сравнительный анализ показал, что каждый из подходов обладает своими преимуществами и недостатками, что делает выбор метода кодирования компромиссным решением. Прямое представление отличается простотой и наглядностью, но страдает от проблемы генерации невалидных решений. Косвенное представление эффективно решает эту проблему, перенося сложность на алгоритм-декодер, однако при этом теряется контекст структуры расписания на уровне генетических операторов.

Наиболее перспективным для решения сложных, крупномасштабных задач составления расписания в реальных условиях высших учебных заведений был признан компонентный (агрегативный) подход. Оперирование семантически осмысленными «блоками занятий» позволяет не только значительно сократить пространство поиска, но и более эффективно сохранять и передавать потомкам удачные структурные комбинации, что способствует более быстрой сходимости к качественным решениям.

Основываясь на проведенном анализе, можно сформулировать следующие практические рекомендации по выбору метода кодирования:

- Для небольших учебных заведений или отдельных факультетов с относительно простым набором ограничений, а также для задач, где скорость разработки прототипа важнее оптимальности, целесообразно использовать прямое (матричное) представление. Его просто-

та позволит быстро получить работающее решение, а проблемы с невалидными хромосомами можно решать с помощью «ремонтных» процедур.

- В задачах, где можно разработать эффективный эвристический декодер, способный самостоятельно разрешать большинство конфликтов при последовательном размещении занятий (например, при наличии четких правил приоритетов), предпочтение следует отдать косвенному (перестановочному) представлению. Этот подход гарантирует генерацию синтаксически корректных хромосом и переносит основную «интеллектуальную» нагрузку на детерминированный алгоритм, что делает эволюционный процесс более стабильным.
- Для крупных, многопрофильных университетов со сложной иерархией жестких и мягких ограничений (пересекающиеся потоки, предпочтения преподавателей, требования к аудиториям) наиболее перспективными являются компонентные (агрегативные) и гибридные подходы. Несмотря на сложность реализации, они наиболее адекватно отражают семантику задачи, значительно сокращают пространство поиска и обеспечивают наилучшую сходимость к качественным, практически применимым решениям.

Итоговый вывод исследования заключается в том, что не существует единственного универсального метода кодирования, оптимального для всех случаев. Выбор должен быть ситуативным и основываться на тщательном анализе конкретной задачи: ее масштаба, количества и типа жестких и мягких ограничений. Данная работа предоставляет исследователям и разработчикам систем автоматизации расписаний структурированную основу для совершенствования этого выбора, систематизируя накопленный в научной литературе опыт и предлагая четкую классификацию существующих подходов. Дальнейшие исследования могут быть направлены на разработку новых гибридных методов кодирования, а также на проведение масштабных эмпирических сравнений эффективности различных представлений на стандартизованных наборах тестовых данных.

СПИСОК ЛИТЕРАТУРЫ

1. Астахова, И. Ф. Составление расписания учебных занятий на основе генетического алгоритма / И. Ф. Астахова, А. М. Фирас // Вестник ВГУ. Серия: Системный анализ и информационные технологии. – 2013. – № 2. – С. 93–99.
2. Афонин, П. В. Гибридные генетические алгоритмы для задачи составления расписания проекта / П. В. Афонин, О. В. Кокшагина // Известия ЮФУ. Технические науки. – 2009. – № 4 (93). – С. 46–51.
3. Безуглый, М. А. Модифицированный генетический алгоритм составления расписания в условиях учебного заведения / М. А. Безуглый, А. И. Секирин // Информатика, управляющие системы, математическое и компьютерное моделирование. – 2021. – № 6 (36). – С. 365–370.
4. Жукова, М. Ю. Автоматизация построения расписания экзаменов ВУЗа с использованием генетического алгоритма / М. Ю. Жукова, В. М. Аль-Габри // Инженерный вестник Дона. – 2017. – № 3 (46). – С. 1–11.
5. Композиционный генетический алгоритм составления расписания учебных занятий / Ю. С. Кабальнов, Л. И. Шехтман, Г. Ф. Низамова, Н. А. Земченкова // Вестник Уфимского государственного авиационного технического университета. – 2006. – Т. 7, № 2 (15). – С. 99–109.
6. Коробкин, А. А. Использование агрегативного генетического алгоритма для составления расписания / А. А. Коробкин // Вестник Воронежского государственного технического университета. – 2009. – Т. 5, № 11. – С. 184–186.
7. Коротков, В. В. Модифицированный генетический алгоритм построения расписания проекта / В. В. Коротков // Моделирование, оптимизация и информационные технологии. – 2022. – Т. 10, № 3. – С. 28–40.
8. Маслов, А. С. Генерация расписания для вуза с использованием генетического алгоритма / А. С. Маслов, Ю. С. Белов // Международный студенческий научный вестник. – 2018. – № 6. – URL: <https://s.eduherald.ru/pdf/2018/6/19255.pdf> (дата обращения: 15.10.2025).

9. Масляев, Д. А. Современное состояние задачи автоматизации составления оптимального учебного расписания в вузе / Д. А. Масляев // Вестник Сыктывкарского университета. Серия 1: Математика. Механика. Информатика. – 2022. – № 1 (42). – С. 23–40.
10. Скачкова, А. Н. Применение генетического алгоритма для решения задачи составления расписания / А. Н. Скачкова, Л. Т. Ткачева // Сборник научных статей по итогам международной научно-практической конференции. – Минск, 2018. – С. 1–4.
11. Стальмаков, В. А. Применение параллельных генетических алгоритмов для составления расписания прохождения судов через шлюзовые системы / В. А. Стальмаков // Вестник государственного университета морского и речного флота имени адмирала С. О. Макарова. – 2015. – № 1 (29). – С. 16–25.
12. Устинов, И. А. Анализ существующих алгоритмов для составления расписания занятий / И. А. Устинов, И. Н. Набродова // Известия Тульского государственного университета. Технические науки. – 2022. – № 9. – С. 109–113.
13. Aki, O. University Course Timetabling Using Genetic Algorithms / O. Aki // International Scientific Conference «UNITECH 2020». – Gabrovo, 2020. – P. I-390–I-394.
14. Herath, A. K. Genetic Algorithm For University Course Timetabling Problem: A Thesis / A. K. Herath. – The University of Mississippi, 2017. – 47 p.
15. Subang, K. N. Optimizing Course Scheduling with Genetic Algorithms: A Dynamic Approach / K. N. Subang, E. I. Balaba, J. C. Agoyo Jr // SAR Journal. – 2024. – Vol. 7, № 4. – P. 296–302. – URL: https://www.sarjournal.com/content/74/SARJournalDecember2024_296_302.pdf (дата обращения: 15.10.2025).

COMPARATIVE ANALYSIS OF CHROMOSOME REPRESENTATION METHODS IN GENETIC ALGORITHMS FOR THE UNIVERSITY TIMETABLING PROBLEM

A. S. Smirnov, student
 E-mail: antonsmirnov2233@gmail.com
 Kaliningrad State Technical University

The article provides a comparative analysis of various approaches to chromosome representation in genetic algorithms applied to solving the NP-hard problem of academic scheduling. The relevance of the study is due to the fact that the effectiveness of a genetic algorithm significantly depends on the solution encoding method. Based on the analysis of 15 Russian and foreign scientific sources, three main methods of chromosome representation have been systematized and reviewed: direct (matrix-based), indirect (permutation-based), and component-based (aggregative). For each method, its advantages, disadvantages, and application specifics have been analyzed within the context of the particular constraints of the academic process. The conclusion is drawn that the choice of an optimal representation method is not universal and must be based on the specific requirements and constraints of the task. It has also been concluded that component-based and hybrid approaches are the most promising for complex, real-world scheduling problems.

Keywords: *genetic algorithm, university timetabling, chromosome representation, solution encoding, combinatorial optimization, heuristic algorithm, fitness function.*