

На правах рукописи

ЕРЕМЕЕВ Антон Валентинович

ИССЛЕДОВАНИЕ ЭВОЛЮЦИОННЫХ МЕТОДОВ
РЕШЕНИЯ ЗАДАЧ КОМБИНАТОРНОЙ ОПТИМИЗАЦИИ

05.13.17 — теоретические основы информатики

АВТОРЕФЕРАТ

диссертации на соискание ученой степени
доктора физико-математических наук

Москва – 2013

Работа выполнена в Омском филиале Федерального государственного
бюджетного учреждения науки Института математики им. С.Л. Соболева
Сибирского отделения Российской академии наук

Научный консультант: доктор физико-математических наук, профессор
Александр Александрович Колоколов

Официальные оппоненты: доктор физико-математических наук, профессор
Эдуард Хайрутдинович Гимади

доктор физико-математических наук, профессор
Владимир Константинович Леонтьев

доктор физико-математических наук, доцент
Михаил Юрьевич Хачай

Ведущая организация: Федеральное государственное бюджетное
учреждение науки Институт вычислительной
математики и математической геофизики
Сибирского отделения Российской академии наук

Защита состоится «___» _____ 2013 г. в _____ часов на заседании диссер-
тационного совета Д002.017.02 в Федеральном государственном бюджетном
учреждении науки Вычислительном центре им. А.А. Дородницына РАН по
адресу: 119333, г. Москва, ул. Вавилова, д. 40.

С диссертацией можно ознакомиться в библиотеке ВЦ РАН

Автореферат разослан “___” _____ 2013 г.

Ученый секретарь
диссертационного совета Д002.017.02
д.ф.-м.н., профессор

В.В. Рязанов

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы. Задачи комбинаторной оптимизации находят широкое применение в информатике, технике, экономике, планировании и других областях. В настоящее время в комбинаторной оптимизации интенсивно развивается подход, основанный на бионическом принципе моделирования эволюционных процессов адаптации в живой природе. Эволюционные методы успешно применяются в информационных технологиях проектирования, планирования, управления, распознавания образов и т.д. Этим методам посвящено большое число работ как отечественных авторов (Д.И.Батищев, И.Л.Букатова, А.Г.Ивахненко, Ю.А.Кочетов, В.М.Курейчик, Ю.И.Неймарк, И.П.Норенков, Л.А.Растригин, В.Г.Редько, Е.С.Семенкин и др.), так и зарубежных (E.Balas, D.Goldberg, J.Holland, I.Rechenberg, M.Schoenauer, P.Vitányi, M.Vose, I.Wegener, и др.).

Несмотря на большое количество результатов, полученных в области комбинаторной оптимизации, потребность в дальнейших исследованиях не уменьшается. Это связано как с постоянным притоком новых задач, так и со сложностью их решения. Многие задачи комбинаторной оптимизации являются NP-трудными, и построение оптимального решения требует значительных временных затрат даже при сравнительно низких размерностях исходных данных. В таких ситуациях возникает необходимость в более глубоком анализе задач с позиций теории вычислительной сложности, позволяющей оценить перспективы разработки алгоритмов с заданными характеристиками (время вычислений, требуемый объем памяти, точность и др.) и в итоге найти подход к решению задачи.

Важное место в комбинаторной оптимизации занимает проблематика целочисленного программирования, которая включает вопросы, связанные с теорией двойственности, полиэдральным подходом, методами отсечения, ветвей и границ, декомпозиции, множителей Лагранжа, выпуклым и невыпуклым программированием, устойчивостью решений и т.д. Исследованию данной проблематики посвящены работы В.Л.Береснева, В.П.Булатова, В.Т.Дементьева, Ю.Г.Евтушенко, В.А.Емеличева, И.И.Еремина, М.М.Ковалева, А.А.Колоколова, В.К.Леонтьева, Вл.Д.Мазурова, И.В.Сергиенко, А.С.Стрекаловского, В.Н.Шевченко, G.Dantzig, J.Edmonds, D.Fulkerson, R.Gomory, G.Nemhauser и многих других авторов как в России, так и за рубежом. В ряде известных методов решения задач целочисленного про-

граммирования используется сведение исходной задачи к последовательности задач непрерывной оптимизации. На этом подходе основаны методы отсечения, декомпозиции, ветвей и границ, алгоритмы перебора L -классов и др. Актуальной проблемой при исследовании таких методов является построение нижних и верхних оценок числа итераций, в том числе оценок в среднем. Важные результаты в этом направлении получены В.П.Гришухиным, О.А.Заблоцкой, Л.А.Заозерской, А.А.Колоколовым, Н.Н.Кузюриным, Ю.Ю.Финкельштейном, R.Jeroslow, H.Lenstra, E.Venta.

При решении задач комбинаторной оптимизации широко используется метод динамического программирования, предложенный Р.Беллманом, и его обобщения. Актуальные вопросы распараллеливания, преодоления больших затрат памяти, разработки гибридных алгоритмов, анализа графовых интерпретаций динамического программирования и решения многокритериальных задач в рамках данного подхода рассматриваются в работах В.В.Быковой, Г.Г.Забудского, А.В.Кельманова, Д.И.Когана, Н.Н.Моисеева, В.В.Серваха, Н.З.Шора, О.А.Щербины, U.Bertele, F.Brioschi, Z.Galil и других авторов.

Среди первых методов решения задач комбинаторной оптимизации наряду с методами отсечений, динамического программирования и ветвей и границ возникли методы локального поиска и локальные алгоритмы. В нашей стране основоположниками этого направления стали Ю.И.Журавлев, который ввел в рассмотрение класс локальных алгоритмов и провел анализ их сложности, Л.А.Растрин, предложивший и исследовавший рандомизированные алгоритмы локального поиска, а также И.В.Сергиенко, развивший метод вектора спада и обосновавший его работоспособность. Значительные результаты в исследовании возможностей методов локального поиска получены А.Н.Антамошкиным, Ю.А.Кочетовым, О.Э.Семенкиной, В.Kernighan, S.Lin, C.Papadimitriou, C.Tovey, M.Yannakakis и другими.

Приближенные алгоритмы решения задач комбинаторной оптимизации оказываются незаменимыми в ситуациях, когда получение точного решения требует чрезмерных временных затрат. Значительный вклад в область разработки и анализа приближенных алгоритмов внесли А.А.Агеев, Э.Х.Гимади, Н.И.Глебов, А.А.Корбут, В.А.Перепелица, В.К.Попков, И.Х.Сигал, М.Ю.Хачай, S.Arora, G.Cornuéjols, В.Korte, L.Lovász, D.Hochbaum, P.Raghavan, C.Thompson и ряд других авторов. В настоящее время интенсивно исследуются вопросы существования

полиномиальных аппроксимационных схем и их трудоемкости. Существенные результаты в этом направлении получили Г.В.Генс, М.Я.Ковалев, Е.В.Левнер, С.В.Севастьянов, Я.М.Шафранский, О.Ibarra, С.Kim, W.Kubiak, G.Woeginger и другие.

Эволюционные алгоритмы (ЭА) берут начало в работах L.Fogel и J.Holland, где было предложено моделировать процесс биологической эволюции с целью синтеза эффективных в некотором смысле структур и создания систем искусственного интеллекта. В нашей стране А.Г.Ивахненко и Л.А.Растригиным независимо были предложены методы случайного поиска, где также использовались идеи эволюции. Характерной особенностью ЭА является имитация процесса эволюционной адаптации биологической популяции к условиям окружающей среды, при этом *особи* соответствуют пробным точкам в пространстве решений задачи оптимизации, а *приспособленность* особей определяется значениями целевой функции и штрафами за нарушение ограничений задачи, если такие имеются. В рамках данного подхода предложены эволюционные стратегии (I.Rechenberg), генетические алгоритмы (J.Holland), алгоритмы случайного поиска с адаптацией (Г.С.Лбов), эволюционного моделирования (И.Л.Букатова, L.Fogel), генетического программирования (J.Koza), многокритериальные ЭА (М.Peschel, С.Riedel). К классу ЭА также могут быть отнесены алгоритмы Метрополиса, имитации отжига, поиска с запретами и др. Указанные алгоритмы различаются способами моделирования эволюционного процесса и отражаемыми его аспектами, однако имеют много общих элементов. Области применения этих алгоритмов также несколько различаются.

Принципы наследственности, изменчивости и отбора в эволюционных алгоритмах реализуются при построении новых решений-потомков посредством рандомизированных процедур (операторов), модифицирующих полученные ранее пробные точки подобно процессам мутации и кроссинговера в живой природе. Отбор таких пробных точек производится с учетом значений функции приспособленности (особям, имеющим преимущество по приспособленности, даются большие шансы на отбор в качестве родительских решений). Эволюционные алгоритмы могут применяться как для задач с одним критерием оптимизации, так и для многокритериальных задач, поэтому допускается и многокритериальная вектор-функция приспособленности. Ввиду простоты адаптации вычислительных схем эволюционных алгоритмов эти методы

активно применяются для решения задач, возникающих в управлении, планировании, проектировании, распознавании образов и других областях.

О применимости ЭА к индивидуальной задаче комбинаторной оптимизации естественно судить по математическому ожиданию времени первого достижения оптимального решения или достаточно точного приближенного решения. Эти средние значения могут быть оценены снизу и сверху функциями от длины записи исходных данных задачи, что позволяет делать вывод об эффективности ЭА на задаче комбинаторной оптимизации в целом. Необходимо отметить, что существование ЭА, обнаруживающего оптимальное решение какой-либо NP-трудной задачи в среднем за время, полиномиально ограниченное сверху, представляется маловероятным (это противоречило бы предположению о неравенстве классов NP и RP, которое несколько десятилетий принимается в теории сложности в качестве рабочей гипотезы).

Теоретически наиболее детально исследованы эволюционные алгоритмы с достаточно простыми (как правило, линейными по трудоемкости) операторами мутации и кроссинговера. Как выяснилось, базовые схемы ЭА при подходящем способе представления решений и настройках параметров могут воспроизводить основные этапы работы известных алгоритмов дискретной оптимизации. Значимые результаты в данном направлении получены при анализе ЭА для таких классических задач дискретной оптимизации, как задача о кратчайшем пути в графе (J.Scharnow, K.Tinnefeld, I.Wegener), задача об остовном дереве минимального веса (F.Neumann, I.Wegener) и задача о разрезе минимального веса (F.Neumann, J.Reichel, M.Skutella). Оптимальные решения этих задач могут быть получены в среднем за полиномиальное время при помощи многокритериальных эволюционных алгоритмов, воспроизводящих работу алгоритмов Дейкстры и Краскала, или неявно учитывающих двойственность задач о минимальном разрезе и максимальном потоке. ЭА с аналогичными свойствами найдены и для некоторых других задач. Вместе с тем недостает обобщающих результатов об эффективности ЭА на крупных классах задач, а также теоретически обоснованных выводов о преимуществе одного ЭА над другим на некотором классе задач.

На практике попытки применения стандартных ЭА без учета специфики решаемых задач достаточно быстро показали их низкую эффективность в сравнении со специализированными алгоритмами. Тем не менее гибкость вычислительных схем ЭА и возможности их комбинирования с другими ме-

тодами позволили разработать конкурентоспособные *гибридные* алгоритмы. В таких алгоритмах отдельные операторы представляют собой уже известные алгоритмы, такие как локальный поиск, жадный алгоритм, метод ветвей и границ или перебор L -классов. Для некоторых гибридных ЭА удастся и теоретически гарантировать качество получаемого решения благодаря свойствам используемых в них классических методов. К сожалению, в большинстве случаев обоснование работоспособности гибридных ЭА ограничивается применением теоремы E.Aarts, A.Eiben и K.van Hee о сходимости к оптимуму «почти наверное», предъявляющей достаточно слабые требования к операторам ЭА, но не дающей оценок сверху на время достижения оптимума. Для выхода из создавшейся ситуации необходимы новые методы получения оценок времени отыскания решений требуемой точности и вероятности порождения достаточно точных решений на заданной итерации ЭА.

Работоспособность генетического алгоритма (ГА) существенно зависит от выбора оператора кроссинговера, где комбинируются элементы родительских решений. Новым направлением исследований является анализ сложности и разработка эффективных алгоритмов решения *задачи оптимальной рекомбинации* (ЗОР), состоящей в отыскании наилучшего возможного результата кроссинговера при заданных родительских генотипах. Результаты C.Aggarwal, E.Balas, W.Cook, J.Orlin, P.Seymour и др. дают экспериментальное подтверждение целесообразности решения ЗОР (точного или приближенного) в операторах кроссинговера. Тем не менее до сих пор не было проведено систематического анализа вычислительной сложности ЗОР и эффекта от ее решения в процессе работы ЭА.

Цель диссертации – развитие и исследование эволюционных методов решения задач комбинаторной оптимизации. Поставленная цель определила следующие основные задачи исследования:

- 1) разработка эволюционных алгоритмов и их гибридных вариантов с использованием классических методов комбинаторной оптимизации;
- 2) оценка точности решений, получаемых эволюционными алгоритмами, построение оценок трудоемкости этих алгоритмов и их основных операторов, сопоставление полученных оценок с известными аналогами;
- 3) анализ сложности задач комбинаторной оптимизации, и в частности, задач оптимальной рекомбинации, оценивание параметров задач, влияющих

на работоспособность ЭА.

Методы исследования. При выполнении работы использовались методы математического программирования, теории вероятностей и математической статистики, теории вычислительной сложности, бионические методы и модели, а также современная методология экспериментальных исследований с применением вычислительной техники.

Научная новизна. Введена в рассмотрение общая постановка задачи оптимальной рекомбинации в терминах задач комбинаторной оптимизации; предложен метод построения сводимостей ЗОР, подобных известным сводимостям задач комбинаторной оптимизации, позволивший доказать полиномиальную разрешимость или NP-трудность ЗОР для различных задач. Разработана методика использования в ГА операторов кроссинговера, основанных на решении ЗОР, что дало возможность улучшить известные ранее экспериментальные результаты для ряда задач.

Впервые введено отношение доминирования для операторов мутации и кроссинговера, позволившее при некоторых условиях выделить наилучшего представителя в классе ЭА.

Новым является предложенный подход к получению верхних оценок среднего числа итераций ЭА до нахождения оптимального или приближенного решения через сопоставление этапов развития популяции ЭА с этапами работы алгоритмов локального поиска и динамического программирования.

Благодаря группировке состояний в модели ГА впервые получены оценки числа особей заданного качества в популяции ГА с турнирной селекцией, применимые для ряда задач комбинаторной оптимизации.

Впервые предложены статистические методы построения нижних оценок числа локальных оптимумов задачи комбинаторной оптимизации.

На защиту выносятся следующие основные результаты.

1. Проведено комплексное исследование эволюционных методов с позиций локальной оптимизации: найдены необходимые и достаточные условия, при которых рандомизированный вариант локального поиска является наилучшим методом по вероятности отыскания оптимума в классе эволюционных алгоритмов с заданным оператором мутации; найдены достаточные условия, при которых генетический алгоритм с турнирной селекцией достигает локальный оптимум в среднем за полиномиально ограниченное время, и показано выполнение этих условий на классе задач с гарантированными локаль-

ными оптимумами; предложены статистические методы нахождения нижних оценок числа локальных оптимумов задачи.

2. Исследована сложность задач оптимальной рекомбинации: установлена полиномиальная разрешимость задач оптимальной рекомбинации в генетических алгоритмах для ряда задач булевого линейного программирования, в частности задач упаковки и разбиения множества и простейшей задачи размещения производства; установлена NP-трудность оптимальной рекомбинации для задачи коммивояжера, задач о рюкзаке, кратчайшем гамильтоновом пути, упаковке в контейнеры и некоторых других классических задач комбинаторной оптимизации.

3. Построены эволюционные алгоритмы, аналогичные по свойствам алгоритмам динамического программирования: получены полиномиальные оценки средней трудоемкости многокритериального эволюционного алгоритма через трудоемкость алгоритма динамического программирования; для класса задач, удовлетворяющих условиям существования вполне полиномиальной аппроксимационной схемы Г. Воегингера, предложена вполне полиномиальная рандомизированная аппроксимационная схема, основанная на эволюционном алгоритме.

4. Разработан подход к построению гибридных генетических алгоритмов, использующих оптимальную рекомбинацию и жадные алгоритмы для решения NP-трудных задач комбинаторной оптимизации: предложены новые генетические алгоритмы для задач покрытия множества, управления поставками продукции и балансировки производственной линии, экспериментально установлены классы задач, на которых такие алгоритмы имеют преимущество по сравнению с известными алгоритмами; выполнен теоретический анализ сложности приближенного решения задачи управления поставками продукции.

5. Проведено исследование генетического алгоритма с турнирной селекцией и мутацией: предложена математическая модель эволюционного процесса в популяции генетического алгоритма; с использованием данной модели построены оценки динамики численности особей с высокой приспособленностью и изучены частные случаи достижимости оценок.

Таким образом, в данной работе проведено комплексное исследование эволюционных методов решения задач комбинаторной оптимизации: выявлены широкие классы задач, для которых существуют эволюционные алгоритмы,

аналогичные по свойствам методам локального поиска и динамического программирования; проведен теоретический анализ сложности задач оптимальной рекомбинации и экспериментально показана целесообразность применения оптимальной рекомбинации в гибридных эволюционных алгоритмах; построены оценки вероятностей получения решений заданного качества.

Практическая и теоретическая ценность. Работа имеет теоретический и экспериментальный характер. Полученные в ней результаты позволяют оценить возможности эволюционных методов и области их применимости. Разработанные алгоритмы могут использоваться в информационных технологиях проектирования и управления производственными и транспортными системами, а также в научных исследованиях для оценки параметров комбинаторных объектов. Материалы диссертации применяются при подготовке студентов и аспирантов.

Апробация работы. Основные результаты и положения работы докладывались автором на следующих научных конференциях:

Всероссийской конференции «Проблемы оптимизации и экономические приложения» (Омск, 1997, 2003, 2006, 2009, 2012);

Всероссийской конференции «Математическое программирование и приложения» (Екатеринбург, 1997, 1999, 2007, 2011);

Российской конференции «Дискретный анализ и исследование операций» (Новосибирск, 1998, 2000, 2002);

Российской конференции «Дискретная оптимизация и исследование операций» (Владивосток, 2007; Алтай, 2010; Новосибирск, 2013);

International Conference on Operations Research (Цюрих, Швейцария, 1998);

Байкальской международной школе-семинаре «Методы оптимизации и их приложения» (Иркутск, 1998, 2001; Северобайкальск, 2008);

Conference Evolution Artificielle (Дункерк, Франция, 1999);

International Workshop «Discrete Optimization Methods» (Минск, 2000; Омск - Иркутск, 2004);

European Workshop on Evolutionary Computation in Combinatorial Optimization (Милан, Италия, 2001; Турин, Италия, 2011);

International Seminar «Theory of Evolutionary Algorithms» (Дагштул, Германия, 2002, 2004, 2006, 2008, 2010, 2013);

Workshop on the Foundations of Genetic Algorithms (Торремолинос, Испания, 2003);

Conference of the European Chapter on Combinatorial Optimization (Минск, 2005);

Азиатской школе-семинаре «Оптимизация сложных систем» (Новосибирск, 2006; Усть-Каменогорск, 2010);

Международной научной конференции «Дискретная математика, алгебра и их приложения» (Минск, 2009);

Международной конференции «Интеллектуализация обработки информации» (Будва, Черногория, 2012);

Euro Mini Conference on Variable Neighbourhood Search (Херцег Нови, Черногория, 2012);

а также на семинарах в Институте математики им. С.Л. Соболева СО РАН, его Омском филиале и Вычислительном центре им. А.А. Дородницына РАН. Материалы диссертации легли в основу курса «Эволюционные алгоритмы», читаемого магистрантам Института математики и информационных технологий Омского государственного университета им. Ф.М. Достоевского.

Публикации. По теме диссертации автором опубликовано 38 статей и глав монографий, в том числе 20 статей в журналах из списка ВАК.

Личный вклад. Диссертационная работа представляет собой единый цикл исследований автора, объединенных не только постановками задач и схемами алгоритмов, но и методами исследований. В совместных работах соискателю принадлежат основные идеи построения новых эволюционных алгоритмов, методов обоснования оценок точности и трудоемкости алгоритмов, а также исследования вычислительной сложности рассматриваемых задач. Отдельные элементы доказательств утверждений и теорем выполнены в соавторстве при непосредственном участии соискателя. Конфликт интересов с соавторами отсутствует.

Объем и структура диссертации. Диссертация состоит из введения, пяти глав, заключения и списка литературы (296 наименований). Объем диссертации 300 страниц.

Содержание работы

Глава 1 содержит постановки задач комбинаторной оптимизации и общие схемы эволюционных алгоритмов, которые многократно используются далее в различных разделах диссертации.

В разделе 1.1 приводится общая постановка задачи комбинаторной оптимизации. Пусть $\mathbb{R}$ и $\mathbb{N}$ – множества вещественных и натуральных чисел

соответственно; $\{0, 1\}^*$ – множество всевозможных строк из нулей и единиц. Для строки $S \in \{0, 1\}^*$ символом $|S|$ обозначается ее длина.

Определение 1.1 *Задача комбинаторной оптимизации – это тройка $\Pi = (\text{Inst}, \text{Sol}, f_I)$, где*

- 1) $\text{Inst} \subseteq \{0, 1\}^*$ – множество индивидуальных задач из Π ;
- 2) $\text{Sol}(I) \subseteq \{0, 1\}^{n(I)}$ – множество допустимых решений индивидуальной задачи $I \in \text{Inst}$, где $n(I)$ – размерность пространства решений;
- 3) для каждой $I \in \text{Inst}$ определена функция $f_I : \text{Sol}(I) \rightarrow \mathbb{R}$, которую требуется максимизировать (если Π – задача максимизации) или минимизировать (если Π – задача минимизации).

Далее через f_I^* обозначается оптимальное значение целевой функции в индивидуальной задаче I . По умолчанию под *полиномиально ограниченной* величиной подразумевается величина, ограниченная сверху полиномом с положительными коэффициентами относительно $|I|$. Задача комбинаторной оптимизации называется полиномиально ограниченной, если значения $f_I(\mathbf{x})$ полиномиально ограничены.

Наибольший интерес представляют задачи комбинаторной оптимизации из класса NPO, где вводятся следующие «технические» предположения.

Определение 1.2 *Задача комбинаторной оптимизации принадлежит классу NPO, если отношения $I \in \text{Inst}$ и $\mathbf{x} \in \text{Sol}(I)$ могут быть проверены за полиномиально ограниченное время, размерность $n(I)$ полиномиально ограничена, а функция $f_I : \text{Sol}(I) \rightarrow \mathbb{N}$ вычислима за полиномиально ограниченное время для любой $I \in \text{Inst}$.*

Из контекста, как правило, ясно, о какой индивидуальной задаче I идет речь. Для краткости символ I в таких случаях будет опускаться.

Далее в разделе 1.1 рассматривается несколько классических задач комбинаторной оптимизации в качестве примеров: задача об одномерном булевом рюкзаке, задача коммивояжера, задача о наименьшем покрытии, задача о вершинном покрытии и простейшая задача размещения.

Здесь также вводятся стандартные определения, формализующие понятие приближенного решения задачи и вполне полиномиальной аппроксимационной схемы. В завершение раздела излагается общая схема алгоритма локального поиска и связанные с ней определения.

Пусть для всякого $\mathbf{y} \in \text{Sol}(I)$ определена окрестность $\mathcal{N}_I(\mathbf{y}) \subseteq \text{Sol}(I)$. Совокупность $\{\mathcal{N}_I(\mathbf{y}) : \mathbf{y} \in \text{Sol}(I)\}$ называется *системой окрестностей*. Если для $\mathbf{x} \in \text{Sol}(I)$ при всяком $\mathbf{y} \in \mathcal{N}_I(\mathbf{x})$ выполняется неравенство $f_I(\mathbf{y}) \leq f_I(\mathbf{x})$ в случае задачи максимизации, или $f_I(\mathbf{y}) \geq f_I(\mathbf{x})$ в случае задачи минимизации, то $\mathbf{x}$ называется локальным оптимумом. Система окрестностей $\{\mathcal{N}(\mathbf{x}) \mid \mathbf{x} \in \text{Sol}(I)\}$ называется *k-ограниченной*, если для любых $\mathbf{x} \in \text{Sol}$ и $\mathbf{y} \in \mathcal{N}(\mathbf{x})$ выполнено $\delta(\mathbf{x}, \mathbf{y}) \leq k$, где $\delta(\cdot, \cdot)$ – метрика Хэмминга.

В разделе 1.2 описываются класс эволюционных алгоритмов и наиболее известные его представители.

Первым излагается классический генетический алгоритм (КГА), предложенный Дж. Холландом, как алгоритм, имитирующий процесс адаптации популяции к окружающей среде, взаимодействие с которой задается *функцией приспособленности* особей. На каждой итерации КГА с помощью рандомизированных операторов *мутации* и *кроссинговера* строится новая популяция (поколение). Численность популяции N фиксирована от начала работы алгоритма до конца. В КГА каждая особь текущей популяции выбирается в качестве родительской для построения новой особи-потомка с вероятностью, пропорциональной приспособленности родительской особи с помощью так называемого *оператора пропорциональной селекции*.

Процесс работы ГА представляет собой последовательную смену популяций, состоящих из фиксированного числа особей, соответствующих пробным точкам пространства решений. Особи, отвечающие решениям с большим качеством (более приспособленные), получают преимущество в порождении потомков в следующей популяции.

С 80-х годов XX века генетические алгоритмы стали активно использоваться как метод решения задач оптимизации, при этом особи представляют пробные решения задачи, а функция приспособленности в некотором смысле характеризует их «качество». Синонимом термина «особь» является «генотип». Как правило, первый термин используется, когда речь идет об особи в составе популяции, а последний – когда популяция не учитывается.

Генотип ξ представляет собой строку $(\xi_1, \xi_2, \dots, \xi_l)$ фиксированной длины l . Элементы этой строки – символы некоторого конечного алфавита, их принято называть *генами* по аналогии с генами живых организмов, которые представляют собой участки молекулы ДНК.

Обозначим через $\mathcal{B}$ множество всевозможных генотипов длины l с сим-

волами из заданного набора алфавитов $A_1, \dots, A_l$, т.е. $\mathcal{B} = A_1 \times \dots \times A_l$. Для применения ГА к задаче комбинаторной оптимизации необходимо определить *функцию представления решений* $\Psi : \mathcal{B} \rightarrow \{0, 1\}^n$, задающую способ кодирования элементов пространства решений с помощью генотипов.

Образ $\mathbf{x} = \Psi(\xi)$ принято называть *фенотипом*, соответствующим генотипу ξ . Во многих случаях в ГА используется тот же способ кодировки решений, что и в постановке задачи комбинаторной оптимизации. В таком случае термины «генотип», «фенотип» и «пробное решение» оказываются эквивалентными, а отображение $\Psi(\cdot)$ становится тождественным.

Популяция представляет собой вектор $X = (\xi^1, \xi^2, \dots, \xi^N) \in \mathcal{B}^N$, элементами которого являются генотипы. Способ нумерации особей в популяции не имеет значения. Популяция поколения t , $t = 0, 1, \dots$, обозначается через $X^t = (\xi^{1t}, \xi^{2t}, \dots, \xi^{Nt})$. Итерацией ГА является переход от текущей популяции X^t к следующей популяции X^{t+1} .

В соответствии с общепринятым подходом будем предполагать, что при $\Psi(\xi) \in \text{Sol}$ функция приспособленности имеет вид $\Phi(\xi) = \phi_{\text{fit}}(f(\Psi(\xi)))$, где $\phi_{\text{fit}} : \mathbb{R} \rightarrow \mathbb{R}$ – некоторая строго возрастающая функция, если решается задача на максимум, или строго убывающая функция в случае задачи минимизации. Если же $\Psi(\xi) \notin \text{Sol}$, то функция приспособленности принимает меньшее значение, чем в любом допустимом решении, что соответствует штрафу за нарушение ограничений задачи. Кроме того, будем предполагать, что в ГА $\Phi(\xi) \geq 0$ для любого $\xi \in \mathcal{B}$.

Построение новой особи (потомка) начинается с выбора пары родительских особей из X^t при помощи вероятностного оператора селекции $\text{Sel} : \mathcal{B}^N \rightarrow \{1, \dots, N\}$, в котором номера родительских особей выбираются с учетом их приспособленности. К генотипам выбранных особей применяется оператор кроссинговера $\text{Cross} : \mathcal{B} \times \mathcal{B} \rightarrow \mathcal{B} \times \mathcal{B}$ (в некоторых генетических алгоритмах $\text{Cross} : \mathcal{B} \times \mathcal{B} \rightarrow \mathcal{B}$), заменяющий часть генов одного генотипа генами другого. Полученный генотип подвергается действию оператора мутации $\text{Mut} : \mathcal{B} \rightarrow \mathcal{B}$, при этом часть его генов изменяется случайным образом. Здесь и далее под вероятностным оператором понимаем рандомизированную процедуру, такую что при заданном входном аргументе распределение вероятностей на ее выходе не зависит от предшествующей работы алгоритма.

В качестве критерия остановки ГА может использоваться, например, огра-

ничение на число итераций или условие достижения оптимума, когда это возможно установить. Результатом ГА является лучшее построенное решение.

В настоящей работе, наряду с пропорциональной селекцией, исследуется другой широко известный оператор *турнирной селекции*. При действии данного оператора из популяции равновероятно (с возвращением) выбираются s особей и в качестве родителя используется лучшая по приспособленности среди выбранных. Настраиваемый параметр s называется *размером турнира*. Параметры N и s , вообще говоря, зависят от индивидуальной задачи и их выбор может существенно влиять на время первого достижения решения требуемой точности.

Операторы Cross и Mut с некоторым упрощением моделируют процессы кроссинговера и мутации в живой природе, где они сводятся к случайным изменениям в молекуле ДНК. Степень воздействия кроссинговера и мутации в КГА регулируется вероятностью кроссинговера P_c и вероятностью мутации P_m . Известно множество вариантов генетического алгоритма, имеющих разные операторы селекции, мутации, кроссинговера и несколько различающихся в своих схемах.

В КГА на каждой итерации происходит полная замена популяции. Во многих других ГА используется *стационарная стратегия* управления популяцией, где на каждой итерации вычисляются только две или одна новая особь, заменяющие в популяции особи с наименьшей приспособленностью.

Далее приводится описание операторов ЭА и его общая схема. Определения фенотипа, функции представления решений и популяции, введенные выше для ГА, сохраняются без изменений. Функция приспособленности в случае ЭА может иметь более общий вид, чем в случае ГА. Известны эволюционные алгоритмы как для задач с одним критерием оптимизации, так и для многокритериальных задач, поэтому допускается и многокритериальная вектор-функция приспособленности. В качестве примеров известных ЭА в главе 1 описываются эволюционные стратегии и многокритериальные ЭА.

На практике общие схемы эволюционных алгоритмов зачастую используются в комбинации с эвристическими процедурами, учитывающими специфику решаемой задачи. Наиболее часто такими процедурами являются методы локальной оптимизации, жадные алгоритмы корректировки решений и декодирующие эвристики. Специфика задачи в ЭА может быть использована при решении вспомогательных задач математического программирования в

операторах кроссинговера и при построении начальной популяции.

В **главе 2** исследуется точность и трудоемкость эволюционных методов и проводится их сопоставление с методами локального поиска.

В разделе 2.1 найдены достаточные условия, при которых генетический алгоритм с полной заменой популяции при турнирной селекции достигает локального оптимума в среднем за полиномиально ограниченное время, и доказано выполнение этих условий на классе задач комбинаторной оптимизации с гарантированными локальными оптимумами (GLO). Предполагается, что двоичное представление решений совпадает с кодировкой решений задачи комбинаторной оптимизации, а при выполнении условия остановки происходит новая инициализация счетчика итераций и популяции X^0 , и вычислительный процесс продолжается. Данный вариант ГА обозначается далее через GA.

Пусть в результате кроссинговера с вероятностью не менее некоторой константы ε , $0 < \varepsilon \leq 1$, образуется пара генотипов $(\mathbf{x}', \mathbf{y}') = \text{Cross}(\mathbf{x}, \mathbf{y})$, хотя бы один из которых не уступает по приспособленности родительским особям $\mathbf{x}, \mathbf{y} \in B$ при любых $\mathbf{x}, \mathbf{y} \in B$, причем константа ε не зависит от I . Для одноточечного кроссинговера последнее неравенство выполняется с $\varepsilon = 1 - P_c$, если $P_c < 1$ – константа, не зависящая от I . Указанное условие также выполняется с $\varepsilon = 1$, если один из двух потомков – решение задачи оптимальной рекомбинации, рассматриваемой далее в главе 3.

Пусть имеется задача $\Pi = (\text{Inst}, \text{Sol}, f_I)$ и указана некоторая система окрестностей $\{\mathcal{N}_I(\mathbf{y}) : \mathbf{y} \in \text{Sol}(I)\}$. Обозначим через $h(I)$ число различных неоптимальных значений целевой функции f_I , т. е.

$$h(I) = |\{\phi \in \mathbb{R} : \phi = f_I(\mathbf{x}), \mathbf{x} \in \text{Sol}(I)\}| - 1.$$

Начиная с любого решения, локальный поиск достигает локального оптимума не более чем за $h(I)$ итераций, улучшающих значение целевой функции. Пусть $L(I)$ обозначает минимальную вероятность достижения любого заданного решения в пределах окрестности:

$$L(I) = \min_{\mathbf{x} \in \text{Sol}(I), \mathbf{x}' \in \mathcal{N}_I(\mathbf{x})} \mathbf{P}\{\text{Mut}(\mathbf{x}) = \mathbf{x}'\}.$$

Чем выше величина $L(I)$, тем больше согласованность оператора мутации Mut с системой окрестностей $\mathcal{N}_I$. Показано, что если начальная популяция X^0

содержит допустимое решение, $s \geq rN$, $r > 0$, $h(I) > 1$, $L(I) > 0$ и

$$N \geq \frac{2(1 + \ln h(I))}{\varepsilon L(I)(1 - e^{-2r})},$$

то ГА с условием остановки $t = h + 1$ имеет свойства: 1) локальный оптимум посещается к итерации $h(I)$ с вероятностью не менее $1/e$; 2) локальный оптимум достигается не более, чем за $eh(I)$ итераций в среднем.

С использованием свойства 2) найдены полиномиальные верхние оценки на среднее время получения оптимума в семействе задач с целевой функцией из класса ONEMAX**, введенного S.Droste, T.Jansen и I.Wegener, и семействе задач вершинного покрытия графа $G(k)$, предложенном Л.А.Заозерской. Кроме того, из свойства 2) вытекает

Теорема 2.1 *Если задача $\Pi = (\text{Inst}, \text{Sol}, f_I) \in \text{NPO}$ полиномиально ограничена, функция $1/L(I)$ полиномиально ограничена, а популяция X^0 содержит допустимое решение, то при соответствующих параметрах ГА локальный оптимум впервые достигается в среднем за полиномиально ограниченное время.*

Согласно G.Ausiello и M.Protasi, задача Π из NPO принадлежит классу задач с гарантированными локальными оптимумами (GLO), если выполняются следующие два условия:

- 1) по крайней мере одно допустимое решение может быть вычислено за полиномиально ограниченное время для любого входа $I \in \text{Inst}$;
- 2) существует натуральное число k такое, что все локальные оптимумы задачи Π относительно некоторой k -ограниченной системы окрестностей имеют константную оценку точности.

Примерами задач из класса GLO являются задача максимальной выполнимости логической формулы, задачи о наибольшем независимом множестве, о наименьшем доминирующем множестве вершин и о наименьшем вершинном покрытии в графах со степенью вершин, ограниченной константой, а также задача о максимальном разрезе графа.

Теорема 2.1 применима для оценки возможностей ГА отыскивать приближенные решения с априорной оценкой точности на классе GLO.

Следствие 2.1 *Пусть $\Pi \in \text{GLO}$, $n > k$ и популяция X^0 содержит допустимое решение. Тогда при операторе мутации Mut^* и соответствующем*

выборе параметров ГА решение с константной оценкой точности впервые достигается в среднем за полиномиально ограниченное время.

В разделе 2.2 исследуется динамика численности особей высокой приспособленности в популяции генетического алгоритма с турнирной селекцией при отсутствии оператора кроссинговера. Для анализа ГА предлагается математическая модель эволюционного процесса, возникающего при его работе. С использованием данной модели получены оценки числа особей заданной приспособленности на итерации t и исследована зависимость числа таких особей от размера турнира.

Трудности применения известных моделей ГА (G.Rudolph, M.Vose) к анализу задач комбинаторной оптимизации связаны с большой размерностью матрицы перехода цепи Маркова. Предложенная в данном разделе модель включает в себя меньшее число состояний и позволяет найти нижние и верхние оценки числа особей с приспособленностью не ниже заданного уровня.

Пусть $\Phi_0 = \min_{\xi \in \mathcal{B}} \Phi(\xi)$ и заданы линии уровня функции приспособленности $\Phi_1, \dots, \Phi_d$, такие что $\Phi_0 < \Phi_1 < \Phi_2 \dots < \Phi_d$. Число линий уровня, а также их значения могут зависеть от индивидуальной задачи I . С выбранными линиями уровня сопоставляются лебеговы множества $H_i = \{\xi : \Phi(\xi) \geq \Phi_i\}$, $i = 0, \dots, d$. Кроме того, полагаем $H_{d+1} = \emptyset$.

Пусть для всех $i = 0, \dots, d$ и $j = 1, \dots, d$ априори известны нижние α_{ij} и верхние β_{ij} оценки вероятности перехода из $H_i \setminus H_{i+1}$ в H_j при действии оператора мутации, а именно для любого $\xi \in H_i \setminus H_{i+1}$

$$\alpha_{ij} \leq \mathbf{P}\{\text{Mut}(\xi) \in H_j\} \leq \beta_{ij}.$$

Обозначим матрицу с элементами α_{ij} , где $i = 0, \dots, d$ и $j = 1, \dots, d$ через $\mathbf{A}$; аналогичную матрицу верхних оценок обозначим через $\mathbf{B}$. Представим популяцию на шаге t при помощи вектора популяции $\mathbf{z}^{(t)} = (z_1^{(t)}, z_2^{(t)}, \dots, z_d^{(t)})$, где $z_i^{(t)} \in \mathbb{R}$ – доля генотипов из H_i в популяции X^t .

Будем называть $(d \times d)$ -матрицу с элементами γ_{ij} *монотонной*, если $\gamma_{i-1,j} \leq \gamma_{i,j}$ для всех i, j от 1 до d . Пусть $\mathbf{E}$ – символ математического ожидания; $\mathbf{W}$ – $(d \times d)$ -матрица с элементами $w_{ij} = \alpha_{ij} - \alpha_{i-1,j}$; $\mathbf{I}$ – единичная матрица; $\mathbf{a} = (\alpha_{01}, \dots, \alpha_{0d})$. Если матрица $\mathbf{A}$ монотонна и $\|\mathbf{W}\| < 1$ в некоторой матричной норме, то для любого натурального t имеет место нижняя оценка числа приближенных решений достаточно высокого качества

$$\mathbf{E}[\mathbf{z}^{(t)}] \geq \mathbf{E}[\mathbf{z}^{(0)}]\mathbf{W}^t + \mathbf{a}(\mathbf{I} - \mathbf{W})^{-1}(\mathbf{I} - \mathbf{W}^t).$$

При условии монотонности матрицы $\mathbf{B}$ имеет место оценка сверху

$$\mathbf{E}[z_j^{(t+1)}] \leq \beta_{dj} - \sum_{i=1}^d (\beta_{ij} - \beta_{i-1,j})(1 - \mathbf{E}[z_i^{(t)}])^s, \quad j = 1, \dots, d.$$

Итеративным применением последнего неравенства каждая компонента вектора $\mathbf{E}[\mathbf{z}^{(t)}]$ может быть оценена сверху при любом t .

Если при всех $i = 0, \dots, d, j = 1, \dots, d$, вероятность $\mathbf{P}\{\text{Mut}(\xi) \in H_j\}$ не зависит от выбора $\xi \in H_i \setminus H_{i+1}$, то существуют матрицы оценок $\mathbf{A}$ и $\mathbf{B}$, $\mathbf{A} = \mathbf{B}$. В таком случае оператор мутации будем называть *ступенчатым* относительно набора линий уровня $\Phi_1, \dots, \Phi_d$ и обозначать матрицу $\mathbf{A}$ через $\mathbf{M}$.

Ступенчатый оператор мутации будем называть *монотонным*, если матрица $\mathbf{M}$ монотонна. Для таких операторов мутации проведено исследование свойств вектора популяции, связанных с изменением размера турнира при $N \rightarrow \infty$. С использованием полученных асимптотических свойств доказана

Теорема 2.2 Пусть $\mathbf{z}^{(t)}$ и $\hat{\mathbf{z}}^{(t)}$ – векторы популяции ГА с турнирами величины s и $\hat{s}$ соответственно, причем $s \leq \hat{s}$. Тогда, если ГА имеет монотонный оператор мутации и особи начальных популяций одинаково распределены, то для любых $t = 0, 1, \dots$ и $i = 1, \dots, d$ при достаточно большом размере популяции имеет место неравенство $\mathbf{E}[z_i^{(t)}] \leq \mathbf{E}[\hat{z}_i^{(t)}]$.

Выделенный здесь случай монотонного оператора мутации характеризует ситуацию, в которой при $N = 1$ становятся точными нижние оценки, а при $N \rightarrow \infty$ – верхние оценки на математическое ожидание вектора популяции. Установлено, что для оператора мутации классического ГА условие монотонности во многих случаях эквивалентно условию принадлежности функции Φ к классу строго монотонных псевдобулевых функций.

В разделе 2.3 исследуются условия, при которых эволюционная стратегия (1+1)-ES (рандомизированный алгоритм локального поиска) оказывается «наилучшим» методом в классе эволюционных алгоритмов.

Будем называть *эволюционным алгоритмом с неограниченной памятью* следующий рандомизированный алгоритм: начальная популяция $X^0 \in \mathcal{B}^N$ строится некоторым случайным или детерминированным способом. Последовательность генотипов, построенных в ЭА с неограниченной памятью к началу итерации t , обозначим через σ^{t-1} , а множество всех входящих в σ^{t-1} элементов – через $A^{(t-1)}$. На каждой итерации t , $t = 1, 2, \dots$, вычисляется

промежуточная популяция X^t из N'' генотипов потомков с помощью оператора воспроизведения $\text{Rep}(\eta^1, \dots, \eta^{N'})$, где $\eta^1, \dots, \eta^{N'}$ – некоторые из ранее построенных особей. Родительские особи $\eta^1, \dots, \eta^{N'}$ на итерации t выбираются с помощью рандомизированного оператора *селекции с неограниченной памятью* Sel_∞ , так что $\text{Sel}_\infty(\sigma^{t-1}) \subseteq A^{(t-1)}$. Работа ЭА с неограниченной памятью представляет собой итерации следующей композиции случайных отображений: $X^t = \text{Rep}(\text{Sel}_\infty(X^0, \dots, X^{t-1}))$, $t = 1, 2, \dots$. Алгоритм указанного вида обозначим через ЕА.

Эволюционная стратегия (1+1)-ES является рандомизированным алгоритмом локального поиска и начинает работу с начального генотипа, построенного случайно или детерминированно с помощью некоторой процедуры инициализации. На каждой итерации t , $t = 0, 1, \dots$, по текущему генотипу ζ^t строится новый генотип ζ' посредством оператора мутации Mut . В случае, если $\Phi(\zeta') \geq \Phi(\zeta^t)$, полагаем $\zeta^{t+1} := \zeta'$. В противном случае $\zeta^{t+1} := \zeta^t$.

Обозначим максимум функции приспособленности на последовательности генотипов $\sigma = (\xi^1, \dots, \xi^k)$ через $\check{\Phi}(\sigma)$, т.е. $\check{\Phi}(\sigma) = \max_{i=1, \dots, k} \Phi(\xi^i)$. Будем говорить, что оператор воспроизведения Rep *доминируется* оператором мутации Mut , если для любой N' -элементной последовательности генотипов X' и любого $\eta \in \mathcal{B}$, таких что $\Phi(\eta) \geq \check{\Phi}(X')$, при всех $\phi > \Phi(\eta)$:

$$\mathbf{P} \{ \Phi(\text{Mut}(\eta)) \geq \phi \} \geq \mathbf{P} \{ \check{\Phi}(\text{Rep}(X')) \geq \phi \}.$$

Основным результатом данного раздела является следующая теорема о сравнении эволюционных алгоритмов с эволюционной стратегией (1+1)-ES. Здесь, как и ранее, предполагается, что в эволюционном алгоритме ЕА используется оператор воспроизведения Rep , а в алгоритме (1+1)-ES – оператор мутации Mut .

Теорема 2.3 *Для любого эволюционного алгоритма ЕА с неограниченной памятью при $\Phi(\zeta^0) \geq \check{\Phi}(X^0)$ выполняются неравенства*

$$\mathbf{P}\{\Phi(\zeta^t) \geq \phi\} \geq \mathbf{P}\{\check{\Phi}(\sigma^t) \geq \phi\}, \quad t \in \mathbb{Z}_+, \quad \phi \in \mathbb{R},$$

в том и только в том случае, когда оператор Mut доминирует оператор Rep .

Показано, что в условиях применимости данной теоремы (1+1)-ES не уступает никакому другому ЭА в терминах математического ожидания приспособленности лучшего полученного генотипа на любой заданной итерации, а также по среднему времени поиска оптимального генотипа.

В разделе 2.4 предлагаются и обосновываются статистические методы построения доверительных интервалов для числа локальных оптимумов ν на основе результатов многократного выполнения локального поиска.

Во многих ЭА перед добавлением в популяцию каждый новый генотип проходит процедуру локальной оптимизации (см., например, [15, 17]). При использовании данного подхода возникает вопрос о мощности множества всех генотипов, получаемых в результате локальной оптимизации. Кроме того, существует большое число эвристических методов решения задач комбинаторной оптимизации, в которых локальный поиск используется многократно с различными начальными точками. Сложность решения задачи алгоритмами такого типа характеризуется целым рядом параметров, одним из которых является общее число локальных оптимумов.

В данном разделе построены нижние границы параметра ν , обеспечивающие заданную доверительную вероятность. Кроме того, в предположении равновероятного попадания локального поиска во все локальные оптимумы построены консервативные двусторонние доверительные интервалы для ν . Предлагаемые методы предназначены для исследования индивидуальных задач комбинаторной оптимизации, в которых большая размерность пространства решений не позволяет осуществить полный перебор всех его элементов за приемлемое время. Вместе с тем S.Sahoo и A.Albrecht показана перспективность использования этих методов в биоинформатике при анализе стабильных состояний молекул РНК.

Результаты второй главы опубликованы в [1, 5, 12, 20, 25, 26, 32, 34, 38].

В **главе 3** исследуются сложность и методы решения задачи оптимальной рекомбинации, состоящей в отыскании наилучшего возможного результата кроссинговера при заданных двух родительских генотипах, представляющих допустимые решения задачи комбинаторной оптимизации. Предполагается, что для рассматриваемых индивидуальных задач множество допустимых решений не пусто.

Определение 3.1 Задачей оптимальной рекомбинации для задачи комбинаторной оптимизации $\Pi = (\text{Inst}, \text{Sol}, f)$ является задача комбинаторной оптимизации $\bar{\Pi} = (\bar{\text{Inst}}, \bar{\text{Sol}}, \bar{f})$, такая что всякая индивидуальная задача $\bar{I} \in \bar{\text{Inst}}$ имеет вид $\bar{I} = (I, \mathbf{p}^1, \mathbf{p}^2)$, где $I \in \text{Inst}$, $\mathbf{p}^1 = (p_1^1, \dots, p_{n(I)}^1) \in \text{Sol}(I)$, $\mathbf{p}^2 = (p_1^2, \dots, p_{n(I)}^2) \in \text{Sol}(I)$, и

$$\bar{\text{Sol}}(\bar{I}) = \{\mathbf{x} \in \text{Sol}(I) \mid x_j = p_j^1 \text{ или } x_j = p_j^2, j = 1, \dots, n(I)\}. \quad (1)$$

Критерий оптимизации в $\bar{I}$ тот же, что и в I , т. е. $\bar{f}_{\bar{I}} \equiv f_I$.

Допустимые решения $\mathbf{p}^1, \mathbf{p}^2$ задачи I называются *родительскими* решениями для задачи $\bar{I} = (I, \mathbf{p}^1, \mathbf{p}^2)$. Обозначим $D(\mathbf{p}^1, \mathbf{p}^2) = \{j \mid p_j^1 \neq p_j^2\}$.

Определение 3.1 может интерпретироваться как «фиксация» в ЗОР тех значений генов, в которых оба родительских генотипа совпадают. В литературе, однако, известны и другие постановки задач рекомбинации. Например, в работах W.Cook, C.Cotta, P.Seymour, J.Troya обнадеживающие экспериментальные результаты показали ГА, в которых решается задача рекомбинации с «фиксацией» только тех значений генов, в которых оба родительских генотипа имеют значение, равное 0. Такая постановка задачи далее будет называться *ослабленной задачей оптимальной рекомбинации*, и для ее формулировки достаточно заменить условие (1) в определении 3.1 на

$$\overline{\text{Sol}}(\bar{I}) = \{\mathbf{x} \in \text{Sol}(I) \mid x_j \leq p_j^1 \text{ или } x_j \leq p_j^2, j = 1, \dots, n(I)\}. \quad (2)$$

В главе 3 ряд задач комбинаторной оптимизации рассматриваются как задачи булевого линейного программирования:

$$\max f(\mathbf{x}) = \sum_{j=1}^n c_j x_j, \quad (3)$$

$$\sum_{j=1}^n q_{ij} x_j \leq b_i, \quad i = 1, \dots, m, \quad (4)$$

$$x_j \in \{0, 1\}, \quad j = 1, \dots, n. \quad (5)$$

Здесь $\mathbf{x} \in \{0, 1\}^n$ – вектор булевых переменных, а исходные данные c_j, b_i, q_{ij} , $i = 1, \dots, m$, $j = 1, \dots, n$ – рациональные числа.

В разделе 3.2 с использованием полиномиальных сводимостей между задачами оптимальной рекомбинации доказывается, что ЗОР полиномиально разрешима для задач упаковки множества наибольшего веса, разбиения множества минимального веса и простейшей задачи размещения предприятий в постановке булевого линейного программирования. Для задачи булевого линейного программирования (3)-(5) введем обозначение $U_i = \{j \mid q_{ij} \neq 0, j = 1, \dots, n\}$, $i = 1, \dots, m$.

Теорема 3.2 *ЗОР для задачи булевого линейного программирования (3)-(5) сводится к задаче о независимом множестве в 2-раскрашиваемом*

гиперграфе, где 2-раскраска является частью входных данных. Каждое ребро в получаемом гиперграфе содержит не более чем $U_{\max}$ вершин, где $U_{\max} = \max_{i=1, \dots, m} |U_i|$, и трудоемкость данной сводимости есть $O(m(2^{U_{\max}} + n))$.

Из теоремы 3.2 следует существование полиномиально вычислимого оператора оптимальной рекомбинации для задач булевого линейного программирования, где в каждое ограничение входят не более двух переменных. Таким образом, получено обобщение результата Е. Balas и W. Niehaus об эффективной разрешимости ЗОР для задачи о клике наибольшего веса.

В разделе 3.3 показана сложность ряда задач оптимальной рекомбинации. Доказана NP-трудность оптимальной рекомбинации для задач булевого линейного программирования с тремя переменными в каждом ограничении. Далее рассмотрена ЗОР для задач об одномерном булевом рюкзаке и одномерной упаковке в контейнеры в постановке булевого линейного программирования. Задача об одномерном булевом рюкзаке имеет формулировку

$$\max \left\{ \sum_{j=1}^n c_j x_j \mid \sum_{j=1}^n a_j x_j \leq A, x_j \in \{0, 1\}, j = 1, \dots, n \right\},$$

где $a_j \geq 0, c_j \geq 0, j = 1, \dots, n$, и $A \geq 0$ целочислены.

В задаче об одномерной упаковке в контейнеры указан размер контейнера A , число предметов k и целые числа $a_1, \dots, a_k$, представляющие собой размеры предметов. Требуется разместить предметы в минимальное число контейнеров так, чтобы сумма размеров предметов в каждом контейнере не превышала A .

Пусть булева переменная y_j является индикатором использования контейнера с номером j , $j = 1, \dots, k$, а булева переменная x_{ij} — индикатором упаковки предмета i в контейнер j при $i, j = 1, \dots, k$. Требуется найти

$$\begin{aligned} \min \sum_{j=1}^k y_j, \\ \sum_{j=1}^k x_{ij} = 1, \quad i = 1, \dots, k, \\ \sum_{i=1}^k a_i x_{ij} \leq A, \quad j = 1, 2, \dots, k, \end{aligned}$$

$$y_j \geq x_{ij}, \quad i = 1, \dots, k, \quad j = 1, \dots, k,$$

$$x_{ij}, y_j \in \{0, 1\}, \quad i = 1, \dots, k, \quad j = 1, 2, \dots, k.$$

Пусть решения задачи об одномерной упаковке в контейнеры кодируются посредством $(k \times k)$ -матрицы с компонентами x_{ij} (соответствующий вектор $(y_1, \dots, y_k)$ вычисляется однозначно). Тогда справедлива

Теорема 3.3 *ЗОР и ослабленная ЗОР для одномерных задач о булевом рюкзаке и об упаковке в контейнеры являются NP-трудными.*

Рассмотрим задачу коммивояжера (ЗК): дан ориентированный граф G без петель и кратных дуг с множеством вершин V и множеством дуг $\mathcal{A}$, где длина дуги $(i, j) \in \mathcal{A}$ равна $c_{ij} \geq 0$. Найти гамильтонов контур минимальной длины.

Если для любой дуги $(i, j) \in \mathcal{A}$ существует обратная, и $c_{ij} = c_{ji}$, то ЗК называется *симметрической* и граф G можно считать неориентированным. Если же такое свойство не предполагается, то имеет место *общий случай* ЗК.

Допустимое решение задачи коммивояжера в ЭА может кодироваться как строка, в которой последовательно записаны все компоненты матрицы перестановки, отвечающей маршруту коммивояжера. В таком случае ЗОР состоит в поиске кратчайшего маршрута коммивояжера, совпадающего с двумя родительскими допустимыми решениями по тем дугам (или ребрам), по которым проходят оба контура родительских решений, и не проходящего по дугам (или ребрам), отсутствующим в обоих из них. NP-трудность ЗОР в симметрическом случае ЗК доказана с использованием результата A.Itai, C.Papadimitriou и J.Szwarcfiter об NP-полноте задачи проверки свойства гамильтоновости решеточных графов.

Теорема 3.4 *ЗОР для задачи коммивояжера в симметрическом случае NP-трудна в сильном смысле.*

Аналогично доказывается теорема 3.5 об NP-трудности ЗОР для задачи отыскания кратчайшей гамильтоновой цепи в графе.

В общем случае ЗК задача оптимальной рекомбинации не является обобщением рассмотренной в теореме 3.4. Даже при симметрической матрице расстояний (c_{ij}) пара родительских «маршрутов», понимаемых как контуры, обуславливает иное множество допустимых решений ЗОР, чем та же пара

«маршрутов», понимаемых как циклы. Таким образом, общий случай требует специального анализа сложности.

Теорема 3.6 *ЗОР для задачи коммивояжера в общем случае NP-трудна в сильном смысле.*

Рассмотренные выше задачи оптимальной рекомбинации сводятся к ЗК с предписанными ребрами на графах, где степень вершин равна не более 4 в симметрическом и не более 3 – в общем случае. Решение таких задач может быть получено алгоритмами, предложенными D.Eppstein. В частности, трудоемкость решения ЗОР для общего случая ЗК составляет $O(|V| \cdot 1.42^{|V|})$.

В работе [8] рассмотрен иной подход к кодировке решений для задач на перестановках и соответствующая ему ЗОР. Как показано в [9], в случае задачи о кратчайшем гамильтоновом пути этот способ представления решений также приводит к NP-трудной ЗОР, однако для «почти всех» пар родительских решений ЗОР полиномиально разрешима.

Результаты третьей главы опубликованы в [4, 18, 21, 33].

В **главе 4** предложены эволюционные алгоритмы с возможностями методов динамического программирования (ДП).

В разделе 4.1 формулируется достаточно общая схема динамического программирования. С целью единообразного описания алгоритмов ДП здесь вводится дополнительное предположение о том, что решаемая задача комбинаторной оптимизации Π может быть преобразована во вспомогательную задачу многокритериальной оптимизации специального вида Π' .

Пусть имеется d' критериев оптимизации. Индивидуальная задача I' из многокритериальной задачи Π' определяется четверкой $(d', g, \mathcal{S}, \mathcal{D})$. Здесь $\mathcal{S}$ – пространство поиска, $g: \mathcal{S} \rightarrow (\mathbb{R} \cup \{\infty\})^{d'}$ – векторно-значная целевая функция, и $\mathcal{D} \subseteq \mathcal{S}$ – множество допустимых решений. Введем частичный порядок $\succeq$ для обозначения доминирования по Парето: $(y'_1, \dots, y'_{d'}) \succeq (y_1, \dots, y_{d'})$ тогда и только тогда, когда $y'_j \leq y_j$ для всех j , таких что g_j – критерий минимизации, и $y'_j \geq y_j$ для всех j , таких что g_j – критерий максимизации ($\mathbf{y}' \succ \mathbf{y}$, если $\mathbf{y}' \succeq \mathbf{y}$ и $\mathbf{y} \not\succeq \mathbf{y}'$). Решение задачи I' состоит в отыскании полного множества альтернатив (ПМА).

Далее предполагается, что ПМА задачи I' с помощью некоторой эффективной процедуры (например, метода обратного хода) может быть преобразовано в оптимальное решение $\mathbf{x}^* \in \text{Sol}(I)$ индивидуальной задачи I из Π .

Алгоритм ДП для задачи Π' выполняется за $\bar{n}$ итераций, называемых *стадиями*. На каждой стадии i вычисляется и записывается в память некоторое множество *состояний* $\mathcal{S}_i \subseteq \mathcal{S}$. Будем говорить, что алгоритм решает индивидуальную задачу I' , если множество состояний, вычисленных на последней стадии ДП, представляет собой ПМА для I' .

Пусть имеются $\bar{n}$ конечных множеств отображений $\mathcal{F}_i$, $i = 1, \dots, \bar{n}$, состоящих из переходных функций вида $F: \mathcal{S} \rightarrow \mathcal{S}'$. Здесь $\mathcal{S}'$ содержит пространство $\mathcal{S}$ или совпадает с ним. С целью исключения элементов, принадлежащих $\mathcal{S}' \setminus \mathcal{S}$, на каждой стадии i используется семейство функционалов H_F , $H_F: \mathcal{S}' \rightarrow \mathbb{R}$, таких что при любом $F \in \mathcal{F}_i$ выполнено $F(S) \in \mathcal{S}$ тогда и только тогда, когда $H_F(S) \leq 0$.

Сначала рассматривается *упрощенный* алгоритм ДП, в котором на этапе инициализации формируется множество $\mathcal{S}_0$, являющееся конечным подмножеством $\mathcal{S}$. На i -той стадии вычисляется множество состояний $\mathcal{S}_i$ по формуле

$$\mathcal{S}_i = \{F(S) \mid S \in \mathcal{S}_{i-1}, F \in \mathcal{F}_i, H_F(S) \leq 0\}.$$

Результатом работы упрощенного алгоритма ДП является минимальное по включению подмножество $\mathcal{S}'_{\bar{n}} \subseteq \mathcal{S}_{\bar{n}}$, такое что $g(\mathcal{S}'_{\bar{n}}) = g(\mathcal{S}_{\bar{n}})$.

Для снижения трудоемкости в приложениях ДП, как правило, используется принцип оптимальности Р.Беллмана или его обобщения, что позволяет удалить из рассмотрения неперспективные состояния, не нарушая оптимальности результата. Во многокритериальном случае классический принцип Р.Беллмана, как правило, неприменим, однако неперспективные состояния могут быть удалены с помощью соответствующего отношения доминирования на множестве состояний ДП (см., например, в алгоритмы, предложенные В.С.Михалевичем и Н.З.Шором, R.Klötzler, G.Woeginger и др.).

Рассмотрим предпорядок $\preceq_{\text{dom}}$ на $\mathcal{S}'$, такой что $S \preceq_{\text{dom}} S'$ тогда и только тогда, когда $g(S) \preceq g(S')$ или $S \notin \mathcal{S}$, $S' \in \mathcal{S}$. Установлено, что два следующих условия позволяют корректно использовать отношение $\preceq_{\text{dom}}$ для исключения неперспективных состояний в ДП.

С.1. При любых $S, S' \in \mathcal{S}$, $i = 1, \dots, \bar{n}$, если $S \preceq_{\text{dom}} S'$, то $F(S) \preceq_{\text{dom}} F(S')$ для всех $F \in \mathcal{F}_i$ таких, что $F(S') \in \mathcal{S}$.

С.2. Для любых $S, S' \in \mathcal{S}$, $i = 1, \dots, \bar{n}$ и $F \in \mathcal{F}_i$, если $S \preceq_{\text{dom}} S'$, то $H_F(S') \leq H_F(S)$.

При выполненных условиях С.1 и С.2 в алгоритме ДП достаточно хранить не все множество $\mathcal{S}_i$, а лишь минимальное по включению его подмножество,

доминирующее $\mathcal{S}_i$, и общая схема ДП представима в следующем виде.

Алгоритм 4.1 Динамическое программирование для задачи Π'

1. $\mathcal{T}_0 := \mathcal{S}_0$.
2. Цикл для всех $i = 1, \dots, \bar{n}$.
3. $\mathcal{T}_i := \emptyset$.
4. Цикл для всех $S \in \mathcal{T}_{i-1}$, $F \in \mathcal{F}_i$.
5. Если $H_F(S) \leq 0$ и $\nexists S' \in \mathcal{T}_i$, такой что $F(S) \preceq_{\text{dom}} S'$, то
6. $\mathcal{T}_i := (\mathcal{T}_i \setminus \{S'' \in \mathcal{T}_i \mid S'' \prec_{\text{dom}} F(S)\}) \cup \{F(S)\}$.
7. Конец цикла.
8. Конец цикла.
9. Результат: $\mathcal{T}_{\bar{n}}$.

Если упрощенный алгоритм ДП вычисляет множество состояний $\mathcal{S}'_{\bar{n}}$, представляющее собой ПМА для I' , и условия С.1 и С.2 выполнены, то алгоритм 4.1 также находит ПМА для I' .

В разделе 4.2 при условии применимости к задаче Π' алгоритма ДП предлагается многокритериальный ЭА с аналогичными свойствами. Для выявления эффектов, связанных со спецификой ЭА, необходимо ввести дополнительные ограничения на трудоемкость каждого из операторов многокритериального ЭА. Такой подход к сравнению ЭА и ДП реализован в следующей теореме 4.1. Обозначим через $T_{\text{ДР}}$ трудоемкость алгоритма ДП, а через W_{max} – наибольшую из мощностей множеств состояний, вычисленных за все время работы ДП. Под *обработкой состояния* в ДП будем понимать итерацию внутреннего цикла, т. е. строки 5 и 6 алгоритма 4.1.

Теорема 4.1 Пусть задача Π' разрешима алгоритмом ДП, условия С.1, С.2 выполнены, а обработка состояния в ДП имеет трудоемкость $\Theta(1)$. Тогда существует многокритериальный ЭА, где трудоемкость каждого оператора есть $O(1)$, и ПМА задачи Π' вычисляется в среднем за время $O(T_{\text{ДР}} \bar{n} \log W_{\text{max}})$.

Подобный результат имеет место и в более общем случае, когда время обработки состояния ДП и трудоемкости операторов ЭА полиномиально ограничены. В этом случае верхняя оценка средней трудоемкости многокритериального ЭА отличается от трудоемкости алгоритма динамического программирования на множитель, ограниченный полиномом от длины входных данных и числа состояний ДП.

В разделе 4.3 установлено (теорема 4.2), что для класса задач, удовлетворяющих условиям G.Woeginger существования вполне полиномиальной аппроксимационной схемы (FPTAS), на основе многокритериального ЭА может быть построено семейство алгоритмов, являющееся вполне полиномиальной рандомизированной аппроксимационной схемой (FPRAS).

Результаты четвертой главы опубликованы в [3, 14, 22, 23, 29].

Глава 5 посвящена применению оптимальной рекомбинации в ГА.

В разделе 5.1 разработаны и исследованы генетический алгоритм и точный гибридный алгоритм для решения задачи наименьшего покрытия множества (ЗНП). Задача имеет следующую постановку. Пусть даны множество $M = \{1, \dots, m\}$ и набор его подмножеств $M_j \subseteq M$, где $j \in U = \{1, \dots, n\}$. Подмножество $J \subseteq U$ называется *покрытием* M , если $\bigcup_{j \in J} M_j = M$. Каждому M_j приписан вес $c_j > 0$. Требуется найти покрытие минимального суммарного веса. ЗНП NP-трудна в сильном смысле. Кроме того, существование алгоритма полиномиальной трудоемкости, который гарантированно находит решение, не более чем в константу раз превышающее оптимум, влечет $P=NP$.

Для ЗНП предлагается новый вариант ГА (GANP) с недвоичным представлением решений и оператором ЛП-кроссинговера, при действии которого ослабленная ЗОР решается с использованием методов линейного программирования. Кроме того, в данном разделе описывается гибридный алгоритм решения ЗНП, который представляет собой комбинацию метода перебора L -классов с GANP и эвристикой лагранжевой релаксации. ГА используется для нахождения начального решения, а с помощью метода лагранжевой релаксации строятся нижние и верхние оценки целевой функции на начальном этапе и на подзадачах, возникающих в процессе перебора L -классов.

При решении задач библиотеки OR-Library предложенный ГА показал результаты, не уступающие результатам ГА с двоичным представлением J.Beasley и P.Chu. В частности, для всех 50 тестовых задач со случайно сгенерированными данными при размерности до 5000 переменных данным ГА найдены оптимальные решения. Кроме того, для двух индивидуальных задач, связанных с вопросом P.Erdős о раскраске гиперграфа, получены решения, улучшающие известные из литературы.

Эксперимент показал, что применение ЛП-кроссинговера в GANP целесообразно при решении задач, где разрыв двойственности не велик. В противном случае предпочтительнее использовать равномерный кроссинговер.

Гибридный алгоритм тестировался на задачах со случайно сгенерированными данными и показал существенное ускорение по сравнению с перебором L -классов.

В разделе 5.2 рассматривается *задача управления поставками продукции*, состоящая в минимизации стоимости доставки продукции от множества поставщиков до множества потребителей. Допустимый размер каждой открытой поставки ограничен снизу и сверху, размер потребления для каждого потребителя ограничен снизу, а функции стоимости поставки линейны при ненулевых объемах поставки. Задача имеет следующую постановку.

$$\min \sum_{i=1}^m \sum_{j=1}^n z_{ij} a_{ij} + c_{ij} x_{ij},$$

$$\sum_{i=1}^m x_{ij} \geq A_j, \quad j = 1, \dots, n,$$

$$\sum_{j=1}^n x_{ij} \leq M_i, \quad i = 1, \dots, m,$$

$$z_{ij} \in \{0, 1\}, \quad i = 1, \dots, m, \quad j = 1, \dots, n,$$

$$m_{ij} z_{ij} \leq x_{ij} \leq M_i z_{ij}, \quad i = 1, \dots, m, \quad j = 1, \dots, n,$$

где m – число поставщиков; n – число потребителей; переменные z_{ij} являются индикаторами наличия поставки от поставщика i потребителю j , а переменные x_{ij} обозначают размер соответствующей поставки. A_j – минимальное количество продукта, требуемое потребителю j ; m_{ij} – минимальное количество продукта, которое поставщик i готов доставить потребителю j ; M_i – максимальное общее количество продукта, которое поставщик i может доставить. В целевую функцию входят фиксированные доплаты a_{ij} , связанные с открытием поставки (i, j) и стоимости транспортировки единицы продукции c_{ij} . Параметры $a_{ij}, c_{ij}, m_{ij}, M_i, A_j$ целочисленны и неотрицательны при всех i, j .

Теорема 5.1 *Нахождение $p(m, n)$ -приближенного решения задачи управления поставками продукции при любом полиноме $p(m, n)$ представляет собой NP-трудную в сильном смысле задачу.*

Для случая $n = 1$ в данном разделе предложен жадный алгоритм трудоемкости $O(m^2)$ и доказано, что он является 2-приближенным алгоритмом,

если стоимость поставки от каждого поставщика i является вогнутой неубывающей функцией на отрезке $[0, M_i]$ (теорема 5.2). В [7] нами предложена FPTAS для случая $n = 1$ в более общей постановке, где предполагается, что объем поставки от каждого поставщика принадлежит объединению конечного числа отрезков, функции стоимости поставок являются вогнутыми и неубывающими в пределах каждого интервала. В [13] построена FPTAS для более широкого класса непрерывных целевых функций.

Для рассматриваемой задачи предлагаются два гибридных генетических алгоритма и проводится их исследование и сравнение. Первый из предложенных ГА, обозначаемый через GAgrd, основывается на декодировании генотипов с использованием жадного 2-приближенного алгоритма.

Во втором предложенном ГА, обозначаемом через GAmipr, генотип представляет собой $(m \times n)$ -матрицу $\mathbf{Z} = (z_{ij})$. Если при заданной $\mathbf{Z}$ допустимое решение существует, то объемы поставок x_{ij} вычисляются посредством решения транспортной задачи с нижними и верхними ограничениями на объем каждой перевозки. Если же такого решения не существует, то для оценки приспособленности используется штрафная функция.

В GAmipr разработан новый оператор ЦЛП-кроссинговера, в котором перед решением ЗОР в родительские генотипы вносятся случайные изменения. После этого решается (ослабленная) ЗОР для задачи управления поставками продукции в приведенной выше постановке частично целочисленного линейного программирования (ЧЦЛП). Из теоремы 3.3 вытекает, что ЗОР и ослабленная ЗОР для задачи управления поставками продукции являются NP-трудными даже в случае одного потребителя. Оператор ЦЛП-кроссинговера реализован с использованием метода ветвей и границ из пакета CPLEX.

Вычислительный эксперимент показал, что при заданном ограничении на время вычислений GAmipr имеет преимущество по качеству получаемых решений в сравнении с GAgrd и пакетом CPLEX.

В разделе 5.3 рассматривается NP-трудная задача балансировки автоматизированной производственной линии и предлагается ГА с оператором ЦЛП-кроссинговера для ее решения. Исследуемая задача имеет ряд отличий от известной задачи балансировки автоматической сборочной линии, таких как параметризованное время выполнения операции, нестрогие отношения предшествования и параллельное выполнение операций.

В предлагаемом ГА применяется стационарная стратегия управления по-

пуляцией и турнирная селекция. В качестве генотипа используется совокупность булевых переменных в терминах модели А.Б.Долгого и соавт., кодирующая назначение операций на блоки и станции. Генотипы начальной популяции формируются случайным образом посредством N -кратного применения эвристики GRASP, также реализованной с использованием модели ЧЦЛП.

Действие ЦЛП-кроссинговера состоит в решении ЗОР, получаемой фиксацией значений части булевых переменных в исходной задаче. Как следует из теоремы 3.3, данная ЗОР является NP-трудной задачей.

В вычислительном эксперименте с предложенным ГА сравнивались эвристические и точные алгоритмы, разработанные О.Н.Гущинской, Н.Н.Гущинским и А.Б.Долгим, а также эвристика GRASP и метод ветвей и границ с отсечениями из пакета CPLEX.

Эксперименты [19, 30] показали перспективность использования ЗОР в ГА для рассматриваемой задачи и позволили выявить классы задач, на которых ГА с ЦЛП-кроссинговером имеет преимущество перед другими известными алгоритмами. Важной отличительной особенностью данного подхода является возможность его применения к широкому классу задач за счет использования гибких средств построения моделей ЧЦЛП и универсальных пакетов программ для решения такого рода задач.

Результаты пятой главы дают экспериментальное подтверждение целесообразности решения ЗОР в операторах кроссинговера генетических алгоритмов даже в тех случаях, когда ЗОР является NP-трудной задачей. Результаты этой главы опубликованы в [2, 6, 10, 11, 24, 27, 28, 30, 31, 36].

В **заключении** приведены основные результаты диссертации.

Публикации автора по теме диссертации

Статьи в журналах из списка ВАК

- [1] Борисовский П. А., Еремеев А. В. О сравнении некоторых эволюционных алгоритмов // Автомат. и телемех. – 2004. – № 3. – С. 3–9.
- [2] Еремеев А. В. Генетический алгоритм для задачи о покрытии // Дискрет. анализ и исслед. операций. Сер. 2. – 2000. – Т. 7. – № 1. – С. 47–60.
- [3] Еремеев А. В. Вполне полиномиальная рандомизированная аппроксимационная схема на основе эволюционного алгоритма // Дискрет. анализ и исслед. операций. – 2010. – Т. 17. – № 4. – С. 3–17.
- [4] Еремеев А. В. О сложности оптимальной рекомбинации для задачи коммивояжера // Дискрет. анализ и исслед. операций. – 2011. – Т. 18. – № 1. – С. 27–40.

- [5] Еремеев А. В. Генетический алгоритм с турнирной селекцией как метод локального поиска // Дискрет. анализ и исслед. операций. – 2012. – Т. 19. – № 2. – С. 41–53.
- [6] Еремеев А. В., Заозерская Л. А., Колоколов А. А. Задача о покрытии множества: сложность, алгоритмы, экспериментальные исследования // Дискрет. анализ и исслед. операций. Сер. 2. – 2000. – Т. 7. – № 2. – С. 22–46.
- [7] Еремеев А. В., Ковалев М. Я., Кузнецов П. М. Приближенное решение задачи управления поставками со многими интервалами и вогнутыми функциями стоимости // Автомат. и телемех. – 2008. – № 7. – С. 90–97.
- [8] Еремеев А. В., Коваленко Ю. В. О задаче составления расписаний с группировкой машин по технологиям // Дискрет. анализ и исслед. операций. – 2011. – Т. 18. – № 5. – С. 54–79.
- [9] Еремеев А. В., Коваленко Ю. В. О сложности оптимальной рекомбинации для одной задачи составления расписаний с переналадками // Дискрет. анализ и исслед. операций. – 2012. – Т. 19. – № 3. – С. 13–26.
- [10] Еремеев А. В., Романова А. А., Сервах В. В., Чаухан С. С. Приближенное решение одной задачи управления поставками // Дискрет. анализ и исслед. операций. Сер. 2. – 2006. – Т. 13. – № 1. – С. 27–39.
- [11] Borisovsky P., Dolgui A., Eremeev A. Genetic algorithms for a supply management problem: MIP-recombination vs greedy decoder // Europ. Journ. of Operational Research – 2009. – Vol. 195. – N 3. – P. 770–779.
- [12] Borisovsky P. A., Eremeev A. V. Comparing evolutionary algorithms to the (1+1)-EA // Theoret. Computer Science – 2008. – Vol. 403. – N 1. – P. 33–41.
- [13] Chauhan S. S., Eremeev A. V., Romanova A. A., Servakh V. V., Woeginger G. J. Approximation of the supply scheduling problem // Operations Research Letters – 2005. – Vol. 33. – N 3. – P. 249–254.
- [14] Doerr B., Eremeev A., Neumann F., Theile M., Thyssen C. Evolutionary algorithms and dynamic programming // Theoret. Computer Science – 2011. – Vol. 412. – N 43. – P. 6020–6035.
- [15] Dolgui A., Eremeev A., Kolokolov A., Sigaev V. A Genetic Algorithm for the Allocation of Buffer Storage Capacities in a Production Line with Unreliable Machines // Journ. of Mathematical Modelling and Algorithms. – 2002. – Vol. 1. – N 2. – P. 89–104.
- [16] Dolgui A., Eremeev A. V., Kovalyov M. Y., Kuznetsov P. M. Multi-product lot sizing and scheduling on unrelated parallel machines // IIE Transactions. – 2010. – Vol. 42. – N 7. – P. 514–524.
- [17] Dolgui A., Eremeev A. V., Sigaev V. S. HBBA: hybrid algorithm for buffer allocation in tandem production lines // Journ. of Intelligent Manufacturing. – 2007. – Vol. 18. – N 3. – P. 411–420.
- [18] Eremeev A. V. On complexity of optimal recombination for binary representations of solutions // Evolutionary Computation. – 2008. – Vol. 16. – N 1. – P. 127–147.
- [19] Guschinskaya O., Gurevsky E., Dolgui A., Eremeev A. Metaheuristic approaches for the design of machining lines // The Intern. Journ. of Advanced Manufacturing Technology. – 2011. – Vol. 55. – N 1-4. – P. 11–22.

- [20] Reeves C. R., Eremeev A. V. Statistical analysis of local search landscapes // Journ. of the Operational Research Soc. – 2004. – Vol. 55. – N 7. – P. 687–693.

Главы монографий, статьи в трудах конференций и препринты

- [21] Долгий А., Еремеев А. В. О сложности оптимальной рекомбинации для задачи об одномерной упаковке в контейнеры // Материалы VIII международной научно-технической конференции «Динамика систем, механизмов и машин». – Омск: Изд-во ОмГТУ, 2012. – Кн. 3. – С. 25–27.
- [22] Еремеев А. В. О связи динамического программирования и многокритериальных эволюционных алгоритмов: препринт. – Омск: Изд-во ОмГУ, 2008. – 20 с.
- [23] Еремеев А. В. Эволюционные алгоритмы с возможностями динамического программирования // IV Всероссийская конференция «Проблемы оптимизации и экономические приложения»: материалы конференции. – Омск: Полигр. центр КАН, 2009. – С. 35–39.
- [24] Borisovsky P., Dolgui A., Eremeev A. Genetic algorithms for supply management problem with lower-bounded demands // Information Control Problems in Manufact.: Proc. of 12th IFAC Intern. Symp. / Ed. by Dolgui A. [et al.]. – St Etienne: Elsevier Science, 2006. – Vol. 3. – P. 521–526.
- [25] Borisovsky P. A., Eremeev A. V. On performance estimates for two evolutionary algorithms // Applications of Evolutionary Computing: Proc. of EvoWorkshops 2001 / Ed. by Boers E. J. W. [et al.]. – Berlin: Springer-Verlag, 2001. – P. 161–171. – (Lect. Notes Comput. Sci.; Vol. 2037).
- [26] Borisovsky P. A., Eremeev A. V. A study on performance of the (1+1)-evolutionary algorithm // Foundations of Genetic Algorithms 7 / Ed. by De Jong K., Poli R., Rowe J. – San Francisco: Morgan Kaufmann, 2003. – P. 271–287.
- [27] Chauhan S. S., Eremeev A. V., Kolokolov A. A., Servakh V. V. Concave cost supply management for single manufacturing unit // Supply chain optimisation. Product/process design, factory location and flow control. – N. Y.: Springer-Verlag, 2005. – P. 167–174. – (Applied Optim.; Vol. 94).
- [28] Chauhan S. S., Eremeev A. V., Romanova A. A., Servakh V. V. Approximation of linear cost supply management problem with lower-bounded demands // Proc. of Discrete Optimization Methods in Production and Logistics (DOM'2004). – Омск: Nasledie Dialog-Sibir, 2004. – P. 16–21.
- [29] Doerr B., Eremeev A., Horoba C., Neumann F., Theile M. Evolutionary algorithms and dynamic programming // Proc. of Genetic and Evolutionary Comput. Conf. (GECCO). – New York: ACM Press, 2009. – P. 771–777.
- [30] Dolgui A., Eremeev A., Guschinskaya O. MIP-based GRASP and genetic algorithm for balancing transfer lines // Matheuristics. Hybridizing Metaheuristics and Mathematical Programming / Ed. by Maniezzo V., Stutzle T., Voss S. – Berlin: Springer-Verlag, 2010. – P. 189–208.
- [31] Eremeev A. V. A Genetic algorithm with a non-binary representation for the set covering problem // Proc. of Operations Research (OR'98). – Berlin: Springer-Verlag, 1999. – P. 175–181.

- [32] Ereemeev A. V. Modeling and analysis of genetic algorithm with tournament selection // Proc. of Artificial Evolution Conference (AE'99) / Ed. by Fonlupt C. [et al.] – Berlin: Springer Verlag, 2000. – P. 84–95. – (Lect. Notes Comput. Sci.; Vol. 1829).
- [33] Ereemeev A. V. On complexity of optimal recombination for the travelling salesman problem // Proc. of Evolutionary Computation in Combinatorial Optimization (EvoCOP 2011) / Ed. by Merz P., Hao J.-K. – Berlin: Springer Verlag, 2011. – P. 215–225. – (Lect. Notes Comput. Sci.; Vol. 6622).
- [34] Ereemeev A. V. Non-elitist genetic algorithm as a local search method: Preprint (arXiv:1307.3463v2 [cs.NE]). – Cornell: Cornell University, 2013. – 9 p. URL: <http://arxiv.org/abs/1307.3463>.
- [35] Ereemeev A. V., Kolokolov A. A. On some genetic and L -class enumeration algorithms in integer programming // Proc. of the First International Conference on Evolutionary Computation and its Applications. – Moscow: Russian Academy of Sciences, 1996. – P. 297–303.
- [36] Ereemeev A. V., Kolokolov A. A., Zaozerskaya L. A. A hybrid algorithm for set covering problem // Proc. of International Workshop on Discrete Optimization Methods in Scheduling and Computer-Aided Design. – Minsk: National Academy of Sciences Belarus, 2000. – P. 123–129.
- [37] Ereemeev A. V., Reeves C. R. Evolutionary algorithms in discrete optimisation // Book of abstracts of Discrete Analysis and Oper. Res. Conf. – Novosibirsk: Institute of Mathematics, 2002. – P. 40–45.
- [38] Ereemeev A. V., Reeves C. R. On confidence intervals for the number of local optima // Applications of Evolutionary Computing: Proc. of EvoWorkshop 2003 / Ed. by Raidl G. R. [et al.] – Heidelberg: Springer-Verlag, 2003. – P. 224–235. – (Lect. Notes Comput. Sci.; Vol. 2611).

Подписано в печать . . . 2013. Формат 60×84/16. Бумага офсетная.

Оперативный способ печати.

Усл. печ. л. 2.0. Тираж 120 экз. Заказ №

Отпечатано в «Полиграфическом центре КАН»

644122, г.Омск, ул. Красный Путь, 30

тел. (3812) 24-70-79, 8-904-585-98-84

Лицензия ПЛД № 58-47 от 21.04.97