Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Сибирский государственный университет науки и технологий
имени академика М. Ф. Решетнева»

На правах рукописи

Рожнов Иван Павлович

АЛГОРИТМЫ ПОИСКА С ЧЕРЕДУЮЩИМИСЯ РАНДОМИЗИРОВАННЫМИ ОКРЕСТНОСТЯМИ ДЛЯ ЗАДАЧ АВТОМАТИЧЕСКОЙ ГРУППИРОВКИ ОБЪЕКТОВ

05.13.01 – Системный анализ, управление и обработка информации
(космические и информационные технологии)

ДИССЕРТАЦИЯ
на соискание ученой степени кандидата технических наук

Научный руководитель
доктор технических наук, доцент
Казаковцев Л.А.

Красноярск 2019

Оглавление

ВВЕДЕНИЕ

Актуальность. В связи с ускоренным ростом объемов данных растет и потребность в современных средствах и системах сбора, хранения и обработки массивов данных, вследствие чего увеличивается их многообразие. Всё возрастающее использование массивов данных большой размерности стимулирует повышенный интерес к разработке и применению методов и средств обработки и анализа этих массивов. Одним из перспективных направлений является кластерный анализ, который позволяет упорядочивать (группировать) объекты в однородные группы (кластеры), а решение задачи автоматической группировки (кластеризации) сводится к разработке алгоритма, способного обнаружить эти группы без использования предварительно маркированных данных.

Существуют производственные задачи автоматической группировки объектов, которые должны быть решены сравнительно быстро, при этом результат должен быть таким, чтобы известными методами его трудно было бы улучшить без значительного увеличения временных затрат.

Анализ существующих проблем применения методов автоматической группировки объектов, к которым предъявляются высокие требования по точности и стабильности результата, показывает дефицит алгоритмов, способных выдавать за фиксированное время результаты, которые было бы трудно улучшить известными методами, и которые бы обеспечивали стабильность получаемых результатов при многократных запусках алгоритма. При этом известные алгоритмы (например, метода жадных эвристик) требуют значительных вычислительных затрат. Отмечая некоторый дефицит компромиссных по качеству результата и времени счета методов автоматической группировки (под качеством будем понимать точность - близость значения целевой функции к глобальному оптимуму) в настоящем исследовании ставится задача разработать усовершенствованные алгоритмы для задач автоматической группировки, в

которых предъявляются высокие требования к точности и стабильности результата.

Степень разработанности темы. В настоящее время в любой дисциплине, предполагающей многомерный анализ данных, существуют задачи автоматической группировки (кластеризации). Существует множество различных методов и алгоритмов автоматической группировки. Наиболее известной моделью кластерного анализа является модель k-средних, которая была предложена Штейнгаузом (1957 г.). В то же время Ллойдом был разработан и составлен и сам алгоритм (хотя работа опубликована только в 1982). С тех пор алгоритм k-средних, его улучшение, модификация и сочетание с другими алгоритмами, становились темой работ многих исследователей.

В первую очередь среди ученых, в чьих трудах получила развитие автоматическая группировка объектов, необходимо выделить Дюрана Б., Оделла П., Манделя И., Маккуина Дж. Модели автоматической группировки часто имеют сходство с моделями теории размещения объектов, а иногда даже идентичны им, поэтому нередко рассматривались исследователями совместно. Существенный вклад в эти исследования внесли Дрезнер Ц., Хамахер Х., Бримберг Д., Младенович Н. (задачи размещения), Весоловский В. (широкий круг задач), Хакими С. (задачи на сети), Лав Р. (непрерывные задачи с различными метриками). В СССР Хачатуров В.Р. и Черенин В.П. занимались исследованием вопроса размещения предприятий. В Институте математики им. Соболева С.Л. СО РАН работы Гимади Э.Х., Береснева В.Л., Колоколова А.А., а позже Кочетова Ю.А., Еремеева А.В., Забудского Г.Г., Левановой Т.В. и др. при разработке моделей стандартизации и унификации заложили основу для разработки программно-математического аппарата решения задач автоматической группировки и теории размещения объектов.

Метод поиска с чередующимися окрестностями, разработанный Младеновичем Н. и Хансеном П. стал популярным методом решения задач дискретной оптимизации (что отражено в работах Кочетова Ю.А., Лопеса Ф.Г., Бримберна Дж., Левановой Т.В., Алексеевой Е.В. и др.), позволяющим находить

хорошие субоптимальные решения достаточно больших задач автоматической группировки.

Казаковцевым Л.А. и Антамошкиным А.Н. предложено применение алгоритмов с жадной эвристической процедурой и показано их преимущество над считающимися классическими алгоритмами автоматической группировки (k-средних, PAM, j-means и др.) для многомерных данных (2014 г.). Сам метод является расширенным подходом для построения процедур псевдобулевой оптимизации и кластеризации. Метод жадных эвристик использует эволюционные алгоритмы как один из возможных способов организации глобального поиска, в том числе подходы и Красноярской школы эволюционных алгоритмов.

Диссертация посвящена решению задачи разработки и исследования алгоритмов автоматической группировки с повышенными требованиями к точности и стабильности результата с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных эвристических алгоритмов автоматической группировки, в том числе для массивно-параллельных систем.

Основной **идеей** настоящей диссертации является комбинированное применение методов поиска с чередующимися окрестностями и жадных эвристик для задач автоматической группировки, в том числе разработка новых алгоритмов метода жадных эвристик с применением алгоритмов поиска с чередующимися рандомизированными окрестностями.

Объектом диссертационного исследования являются задачи автоматической группировки многомерных данных, **предметом исследования** - алгоритмы для решения данных задач.

Целью исследования является повышение эффективности систем автоматической группировки объектов, к которым предъявляются высокие требования по точности и стабильности результата (улучшение достигаемого значения целевой функции за заданное время).

Задачи, решаемые в процессе достижения поставленной цели:

1. Анализ существующих проблем при применении методов автоматической группировки объектов, к которым предъявляются высокие требования по точности и стабильности результата.

2. Разработка новых алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных эвристических процедур для задачи k-средних.

3. Разработка новых алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных эвристических процедур для задачи k-медоид.

4. Разработка комбинированного алгоритма на основе классификационного EM-алгоритма (CEM – Classification Expectation Maximization) с применением поиска с чередующимися рандомизированными окрестностями и жадных эвристических процедур.

5. Реализация алгоритмов метода жадных эвристик для задач автоматической группировки для массивно-параллельных систем.

6. Разработка процедуры составления ансамблей алгоритмов автоматической группировки, позволяющей повысить точность разделения (то есть уменьшить число ошибок) сборной партии промышленной продукции на однородные партии промышленной продукции с использованием данных неразрушающих тестовых испытаний.

Методы исследования. Для решения поставленных задач использовались методы системного анализа, исследования операций, теории оптимизации, параллельных вычислений.

Новые научные результаты и положения, выносимые на защиту:

1. Предложен новый подход к разработке алгоритмов автоматической группировки, основанных на параметрических оптимизационных моделях, с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных агломеративных эвристических процедур. Показано, что применение данного подхода позволяет создавать эффективные алгоритмы автоматической группировки (по достигаемому

значению целевой функции за фиксированное время), основанные на различных оптимизационных моделях.

2. С использованием нового подхода разработаны новые алгоритмы поиска с чередующимися рандомизированными окрестностями для задач k-средних, k-медоид, задачи четкой кластеризации на основе разделения смеси вероятностных распределений (с применением классификационного EM-алгоритма). Продемонстрировано, что новые алгоритмы позволяют получать более точный и стабильный результат (по достигаемому значению целевой функции) в сравнении с известными алгоритмами автоматической группировки за фиксированное время, позволяющее использовать алгоритмы в интерактивном режиме принятия решений для практических задач.

3. Предложены параллельные модификации алгоритмов с жадной агломеративной эвристической процедурой для больших задач автоматической группировки, адаптированные к архитектуре CUDA. Было выявлено, что параллельная реализация алгоритма локального поиска, а также отдельных шагов жадной агломеративной эвристической процедуры позволяет построить алгоритм автоматической группировки с высоким коэффициентом ускорения, сокращающим время расчетов в десятки раз без ухудшения достигаемого значения целевой функции.

4. Предложена процедура составления оптимальных ансамблей алгоритмов автоматической группировки с комбинированным применением генетического алгоритма метода жадных эвристик и согласованной матрицы бинарных разбиений для практических задач. Было выявлено, что точность разделения сборной партии промышленной продукции с особыми требованиями качества на однородные партии, выполненного с применением получаемых ансамблей, выше усредненной точности разделения с применением отдельных алгоритмов, отобранных для составления ансамбля.

Значение для теории. Теоретическая значимость результатов диссертационной работы состоит в разработке нового подхода к созданию алгоритмов автоматической группировки, основанных на параметрических

оптимизационных моделях, с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных агломеративных эвристических процедур, развивающем метод жадных эвристик, а также процедуры составления оптимальных ансамблей алгоритмов кластеризации.

Практическая ценность нового подхода решения задач автоматической группировки с повышенными требованиями к точности и стабильности результата обусловлена широким диапазоном сфер их применения в задачах кластерного анализа, в том числе непосредственно в практических задачах на производстве, где требуется обеспечение высокой точности разделения производственных партий промышленной продукции на однородные партии по результатам тестовых испытаний.

Практическая реализация результатов. Программная реализация новых алгоритмов и процедуры составления оптимальных ансамблей алгоритмов автоматической группировки была встроена в производственный процесс проведения испытаний электронной компонентной базы космических аппаратов в АО «ИТЦ - НПО ПМ» (г. Железногорск) и в состав «Автоматизированной системы управления технологическим процессом производства анодов», используемой на АО «РУСАЛ Саяногорск», что позволило обеспечить высокую точность разделения на однородные партии промышленной продукции, сократить время расчетов и требования к вычислительным ресурсам, а также (в АО «ИТЦ - НПО ПМ») обеспечить возможность принятия решений об отборе экземпляров продукции из каждой однородной партии для разрушающего анализа в интерактивном режиме.

Основная часть настоящего диссертационного исследования была проведена в рамках государственного задания Министерства образования и науки Российской Федерации № 2.5527.2017/БЧ «Методы комбинаторной оптимизации в системах автоматической группировки и классификации».

Апробация работы. Основные положения и результаты диссертационной работы докладывались и обсуждались на международных конференциях и

семинарах: «Решетневские чтения» (в 2017 и 2018 годах, г.Железногорск), «Optimization Problems and Their Applications OPTA-2018» (2018 г., г.Омск), «Актуальные проблемы электронного приборостроения АПЭП-2018» (2018 г., г.Новосибирск), «Передовые технологии в аэрокосмической отрасли, машиностроении и автоматизации MIST: Aerospace-2018» (2018 г., г.Красноярск), «Advanced Technologies in Material Science, Mechanical and Automation Engineering» (2019 г., г.Красноярск), «Science and education: experience, problems, development prospects» (2019 г., г.Красноярск) и во всероссийских «Лесной и химический комплексы - проблемы и решения» (2017 г., г.Красноярск), «Системы связи и радионавигации» (2018 г., г.Красноярск). Работа в целом обсуждалась на международном семинаре «Advanced Technologies in Material Science, Mechanical and Automation Engineering» (2019 г., г.Красноярск).

Публикации. Основные теоретические и практические результаты диссертации содержатся в 18 публикациях, среди которых 7 работ в ведущих рецензируемых журналах, рекомендуемых в действующем перечне ВАК, 5 - в международных изданиях, индексируемых в системах цитирования Web of Science и Scopus. Имеется свидетельство о государственной регистрации программы для ЭВМ.

Структура и объем диссертации. Диссертация состоит из введения, 4 глав, заключения и приложений. Она изложена на 176 листах машинописного текста, содержит список литературы из 255 наименований.

ГЛАВА 1. СОВРЕМЕННЫЕ МЕТОДЫ АВТОМАТИЧЕСКОЙ ГРУППИРОВКИ ОБЪЕКТОВ С ПОВЫШЕННЫМИ ТРЕБОВАНИЯМИ К ТОЧНОСТИ И СТАБИЛЬНОСТИ РЕЗУЛЬТАТА

Глава посвящена анализу текущего состояния и развития методов и задач автоматической группировки во взаимосвязи с задачами теории размещения. Обозначены проблемы, возникающие при решении задач автоматической группировки объектов с повышенными требованиями к точности и стабильности результата.

1.1 Общая постановка задач автоматической группировки объектов и области их применения

Современное бурное развитие технологий автоматического сбора данных, передачи и хранения информации, интеллектуального анализа данных (Data Mining), а также технологический рост во многих отраслях промышленности и экономики привели к появлению гигантских массивов многомерных данных.

Большая часть массивов данных или уже хранится в цифровой форме, или интенсивно оцифровывается. Одновременно увеличивается объем и качество современных средств и решений, в том числе систем сбора, хранения и обработки данных, и, как следствие, потребность в их качественном анализе и достоверных выводах для принятия эффективных управленческих решений. Все это требует новых достижений в способах восприятия, автоматической обработки и обобщения информации [1, 2].

В настоящее время для анализа данных существует множество статистических методов: факторный анализ, многомерное шкалирование, кластерный анализ, регрессия регрессионный анализ, дисперсионный анализ, дискриминантный анализ, анализ корреляции [2-4].

Как правило [5], исследователи выделяют два больших класса задач при анализе данных:

1. Классификация (обучение с учителем) – имеется обучающая выборка, в которой данные должны быть отнесены к тому или иному заранее определенному классу.

2. Кластеризация (без учителя) – принадлежность к той или иной группе (кластеру) данных заранее не известна, очень часто не известно и количество групп.

В обоих случаях происходит разбиение объектов на однородные группы, только разделение на кластеры намного сложнее. Как правило, под автоматической группировкой понимают именно кластеризацию.

Целью автоматической группировки данных является выделение таких однородных подмножеств (естественных группировок объектов) в исходных многомерных данных, чтобы объекты внутри групп были «похожи» друг на друга, а объекты из разных групп были не схожи по своим параметрам или характеристикам [6].

В аналитике интенсивных данных кластерный анализ (англ. Cluster Analysis) является одним из самых перспективных направлений [7]. Впервые термин «кластерный анализ» (англ. cluster – гроздь, пучок), как считает большинство исследователей, был предложен математиком Р. Трионом [8]. Сфера применения кластерного анализа очень широка – его используют во многих дисциплинах: археологии, медицине, психологии, химии, биологии, государственном управлении, филологии, антропологии, геологии и других.

Задача автоматической группировки формулируется следующим образом: имеется N объектов, нужно найти в них k групп (т.е. выполнить разбиение N объектов на k непересекающихся подмножеств) таким образом, чтобы, основываясь на некой мере подобия, объекты, принадлежащие одной и той же группе, были подобны (обладали схожими значениями параметров), а объекты, принадлежащие различным группам, различались бы значениями параметров. Это задача четкой кластеризации.

Задача нечеткой кластеризации отличается тем, что разбиение N объектов к каждой из k групп будет выполнено с некоторой условной вероятностью.

Данная формулировка задачи автоматической группировки в таком виде оставляет не решенными как минимум два вопроса: как определить общее количество групп объектов k, и какую меру подобия/различия объектов применить.

При использовании метрического определения подобия (расстояния в некотором пространстве числовых признаков между объектами) группы могут отличаться с точки зрения своего размера, формы и плотности. Присутствие шума в данных значительно усложняет задачу нахождения групп.

Решение задачи автоматической группировки неоднозначно по следующим причинам [9]:

– число групп, в основном, заранее неизвестно;

– наилучшего критерия качества автоматической группировки (без использования предварительно маркированных данных) не существует, в связи с чем разбиение может различаться от случая к случаю;

– если постановка задач включает меру различия, то есть расстояния между объектами, то результат зависит от выбранной метрики или меры расстояния.

Допустим, что объекты или элементы данных представлены точками в некотором пространстве характеристик. Тогда идеальная группа может быть определена как ряд точек, который изолирован и компактен. В действительности группа – сущность, восприятие которой зачастую субъективно и определение ее может потребовать еще и знаний в соответствующей области. В практических задачах число измерений данных может быть очень велико, например, в прикладной задаче группировки электрорадиоизделий, которая приведена в качестве примера в этой главе, размерность данных может варьироваться от нескольких десятков до тысяч измерений [10, 11].

Практически в любой дисциплине, предполагающей многомерный анализ данных, существуют задачи кластеризации. Трудно даже перечислить многочисленные научные области, в которых использованы методы автоматической группировки, а также множество существующих различных методов и алгоритмов.

Примерами задач автоматической группировки являются: категоризация документов для обеспечения быстрого доступа и поиска [12-15], сегментация изображения (в компьютерном зрении) [16-19], задачи распознавания рукописного [20] и печатного [21] текста, группировка потенциальных клиентов сервисных точек по географической/геометрической близости для эффективной организации обслуживания [22], биологические задачи [23, 24], работа с группами клиентов (потребителей) в CRM-системах [25, 26].

Решение задачи автоматической группировки объектов в наиболее распространенных постановках, как уже упоминалось, предполагает наличие некой меры сходства (подобия), либо наоборот – меры различия, которая, по сути, является расстоянием между объектами в некотором дискретном либо непрерывном пространстве характеристик. Мерой сходства может быть, например, обратная ей величина.

1.2 Теория размещения и задачи автоматической группировки объектов

Задача автоматической группировки в наиболее часто используемых ее постановках, как правило, оперирует положениями некоторых точек (объектов) в пространстве и расстояниями между ними, прослеживается ее связь с задачами теории размещения, которые чаще всего исследователями определяются как задачи, основными параметрами которых являются положения каких-либо объектов в пространстве и расстояния между ними [27-30]. Взаимосвязь теории размещения с задачами кластеризации сложилась достаточно давно [31-34], зародившись еще в рамках экономической теории [35].

Задачи размещения широко применяются как непосредственно (в архитектуре, при застройке городов, транспортировке и т.д.) [27, 36-39] так и в опосредованном применении (например, в стандартизации) [40-42]. Исторически в СССР (например, в Институте математики им. Соболева С.Л.) начиная с 1960-х годов, задачи размещения формулировались для определения оптимального состава технических систем, оптимального ассортимента изделий и формально

непосредственно с размещением в геометрическом понимании связаны не были [43-45]. Впоследствии, когда была выявлена связь этих двух направлений исследования, фокус внимания исследователей постепенно смещался именно в сторону задач размещения.

Задачи размещения можно классифицировать по зависимости целевой функции от расстояний между новыми и существующими объектами: непрерывная (если объект можно разместить в любой точке пространства), дискретная (если размещение возможно лишь в определенных точках), также выделяются задачи на сети [46].

Модели кластеризации часто имеют сходство с моделями теории размещения объектов, а иногда даже идентичны им, поэтому нередко рассматривались исследователями совместно. Параллельное развитие теории размещения и кластерного анализа дало одинаковые либо очень схожие методы. Например, один из наиболее распространенных алгоритмов в теории размещения – ALA-процедура (Alternating Location-Allocation – чередующееся распределение-размещение) для решения p-медианной задачи [47] и процедура k-средних [48] (довольно распространенный алгоритм в кластерном анализе) построены по одному и тому же шаблону (схеме).

Задача Ферма, вероятно, является простейшей из задач теории размещения: для данных трех точек найти новую точку, от которой до известных точек сумма расстояний будет минимальна [49, 50, 51]. Также этой проблемой занимались Т. Хайнен, Ф. Симпсон и др.

Позднее задача Ферма была развита А. Вебером. В новой задаче требовалось найти точку минимума суммы расстояний уже для произвольного числа известных точек. В задачу были добавлены весовые коэффициенты точек. Исследование было посвященное влиянию основных факторов производства на размещение предприятий с целью минимизации издержек [52]. Эта задача, называемая задачей Вебера (Ферма-Вебера), 1-медианной задачей [53] или задачей Штайнера, послужила исходной точкой развития теории размещения.

Заметим, что процедура решения задачи Вебера входит во многие методы решения задач кластеризации (является их составной частью), в частности – методов на основе жадных агломеративных эвристических алгоритмов, комбинации с которыми используются для исследований в настоящей работе. В своей работе А.Вайсфельд [54] доказал теорему, сформулированную Штурмом [55] и определил последовательность, которая сходилась бы к оптимальному решению задачи Вебера, что по сути являлось вариантом алгоритма градиентного спуска [56]. Данный алгоритм в его более совершенных вариация [57] и сегодня широко применяется при решении задач размещения. Хакими С.Л. определил возможность дискретизации непрерывной задачи Вебера [58, 59]. Отметим, что в случае использования квадрата евклидова расстояния в качестве меры расстояния задача Вебера решается элементарно [28]: решением является точка, координаты которой являются усредненными значениями координат известных точек. Этим обстоятельством объясняется популярность алгоритма k-средних, использующего именно квадрат евклидова расстояния.

Задачи кластеризации и размещения традиционно формулируются российскими (ранее советскими) исследователями в непрерывном пространстве и на сетях задачами линейного и целочисленного линейного программирования [60]. Доказана NP-трудность большинства таких задач, но, несмотря на это, для них разработан большой арсенал эффективных методов решения, большинство из которых можно отнести к точным методам [40, 41, 44, 61, 62-65].

В СССР Хачатуров В.Р. и Черенин В.П. занимались исследованием вопроса размещения предприятий [66-69]. В Институте математики им. Соболева С.Л. СО РАН работы Гимади Э.Х., Береснева В.Л., Колоколова А.А., а позже Кочетова Ю.А., Еремеева А.В., Забудского Г.Г., Левановой Т.В. [30, 40-43, 61, 70-72] и др. при разработке моделей стандартизации и унификации явились теоретическим базисом для разработки алгоритмического и математического аппарата решения задач автоматической группировки и теории размещения объектов.

В алгоритмах автоматической группировки локальный поиск реализуется путем последовательного улучшения заранее известного промежуточного

результата и поэтому наблюдается зависимость результата алгоритмов от выбранного начального решения. Так как поиск следующего решения осуществляется не обязательно в окрестности предыдущего, такие алгоритмы в строгом смысле нельзя отнести к оптимизационным методам локального поиска. Возможна организация работы в режиме множественного старта (мультистарта) данных методов с рандомизированными процедурами выбора начальных решений [73, 74] или более сложные подходы [4]. Также существуют подходы, основанные на идеях из живой природы: генетические и другие эволюционные алгоритмы [75, 76], нейронные сети [77], а также методы имитации отжига [78] и т.д.

Многие разновидности генетического алгоритма характеризуются тем, что нередко получают решение в виде глобального оптимума (хотя задача проверки найденного оптимума на глобальность в свою очередь также является трудной задачей), в то время как классические методы локального поиска легко находят собственно локальные оптимумы задачи [43].

Алп О., Эркут Е. и Дрезнер Ц. предложили генетический алгоритм, в котором используется специальная процедура рекомбинации (кроссинговера) – жадная (агломеративная) эвристическая процедура [79]. Под эвристическим алгоритмом или процедурой, называемыми в литературе «эвристикой», будем понимать алгоритм, не имеющий строгого обоснования, но дающий приемлемое решение задачи для большинства практических случаев. В [79] алгоритм предложен для решения p-медианной задачи на сети. Вместо перестановки последовательностей, которыми представлены родительские «особи», данная эвристика производит объединение родительских множеств индексов узлов сети [80, 81], выбранных в качестве центров групп – дочернее решение содержит больше центров, чем требуется по условиям задачи. Далее происходит последовательное удаление лишних центров (того элемента решения, удаление которого дает наименьший прирост целевой функции) до достижения допустимого решения.

В эволюционных алгоритмах (включая генетические) применяются принципы и терминология, связанная с природной эволюцией. Преимущество

данных алгоритмов перед классическими было исследовано экспериментально [82]. Но эволюционные и генетические алгоритмы лишь определяют общую схему организации поиска.

Реализация схемы поиска в генетическом алгоритме зависит от выбора процедур селекции, мутации и особенно кроссинговера, которая изначально была задумана как простая рандомизированная процедура [83]. Позже процедура кроссинговера (рекомбинации, скрещивания) была реализована в виде иногда очень сложных по своей структуре алгоритмов для конкретных задач оптимизации [84]. В ходе решения многих NP-трудных задач [85, 86] доказана эффективность подходов, в которых на смену рандомизированным процедурам рекомбинации (скрещивания) приходят способы рекомбинации, оптимизированные для конкретных задач. Таким образом, выбор процедуры скрещивания очень важен, хотя выбор оптимальной процедуры может быть задачей не менее трудной, чем собственно нахождение лучшего решения задачи оптимизации (лучшего значения целевой функции за ограниченное время).

1.3 Обзор основных методов кластерного анализа

Современные методы кластерного анализа предлагают широкий выбор средств выявления разнородных по совокупности параметров групп. Наиболее распространенным из подобных методов является метод k-средних (k-means) [2, 4, 6]. Собственно алгоритм k-средних является алгоритмом локальной оптимизации, и его результат зависит от выбора начальных значений (усредненных параметров центров или центроидов групп – кластеров). В то же время, метод выявления различных по параметрам групп объектов должен давать воспроизводимые результаты.

Задача k-средних, наряду с очень похожей p-медианной задачей, является одной из классических задач теории размещения [28]. Задача k-средних состоит в нахождении таких k центров кластеров $X_1 \ldots X_k$ в d-мерном пространстве, чтобы

сумма квадратов расстояний от них до заданных точек A_i ($A_1,\ldots,A_N$) была минимальна:

$$\arg\min F(X_1,\ldots,X_k) = \sum_{i=1}^{N} \min_{j\in\{1,k\}} \left\| X_j - A_j \right\|^2, \qquad (1.1)$$

Наиболее распространенным методом решения задачи k-средних является одноименный алгоритм k-средних, называемый также ALA-процедурой (Alternating Location-Allocation - чередующееся размещение-распределение). Это простой и быстрый алгоритм, применимый ко многим задачам автоматической группировки и размещения. Алгоритм включает всего два шага, чередующихся в ходе работы: разбиение на группы или кластеры (объект относится к той группе, центр которой является к нему ближайшим) и перерасчет центров групп. Алгоритм последовательно улучшает известное решение, позволяя найти локальный минимум (1.1).

Алгоритм имеет ограничения - в начале решения необходимо задать число групп k, на которые разбиваются объекты, а результат сильно зависит от начального решения, как правило, выбираемого случайно.

Алгоритм 1.1 Алгоритм k-средних

Дано: векторы данных $A_1 \ldots A_N$, k начальных центров кластеров $X_1 \ldots X_k$

выполнять

1: Составить кластер C_i векторов данных для каждого центра X_i так, чтобы для каждого вектора данных его центр был ближайшим.

2: Рассчитать новое значение центра X_i для каждого кластера.

пока шаги 1-2 приводят к каким-либо изменениям.

Идея алгоритма k-средних была предложена в 1956 году Штейнгаузом [87], а сам алгоритм был разработан Ллойдом [48, 88] С тех пор алгоритм k-средних (алгоритм Ллойда), его улучшение, модификация и сочетание с другими алгоритмами становились темой работ многих исследователей. В случае классической задачи k-средних центры кластеров обычно называют центроидами.

Альсабти и др. [89] предложили эффективный метод кластеризации с помощью паттерна в k-мерном дереве. Нигам и др. [90] представили алгоритм, использующий помеченные и непомеченные документы, основанный на классификаторе. Канунго и др. [91] описали применение k-средних как алгоритма фильтрации. Чунг [92] представил обобщенный алгоритм k-средних, дающий корректные результаты кластеризации без заранее известного количества кластеров. Сяоли и др. [93] предложили алгоритм, основанный на k-средних и работающий не со всем пространством данных, а только с репрезентативными точками, выбранными с помощью сэмплинга. Сюн и др. [94] исследовали влияние распределения данных на алгоритм k-средних. Они провели исследование мер оценки k-means и кластеризации с точки зрения распределения данных. Фактически, их внимание было сосредоточено на характеристике отношений между распределением данных и кластеризацией k-means в дополнение к мере энтропии и мере точности.

Чжан и др. [95] предложили простую и эффективную методологию, классификации защитников НБА (Национальной баскетбольной ассоциации), основанную на алгоритме k-means и евклидовых расстояниях как мере различия. Ванг и др. [96] представили улучшенный алгоритм k-means, фильтрующий шумы при кластеризации и преодолевающий недостатки оригинального метода. В оригинальный алгоритм встроены шаги анализа и обработки зашумленных данных, основанные на определении плотности. В работе Сингха и др. [97] описан модифицированный алгоритм k-средних, основанный на чувствительности начальных центров кластеров. В этом алгоритме пространство разделяется на сегменты, и вычисляется частота точек в каждом сегменте. Сегмент с максимальной частотой точек с максимальной вероятностью содержит центр кластера. Статья Ши На и др. [98] описывает алгоритм k-средних, улучшенный за счет шагов, сохраняющих информацию о расстояниях между объектами, полученную на предыдущих итерациях и экономящую время расчета. Подобный же подход применен Рани [99].

Важной частью алгоритма k-средних является выбор начальных центров для работы алгоритма, что зачастую является темой отдельных исследований. Бусаре и Бансод в работе [100] описывают алгоритм k-средних в сочетании с улучшенным пиллар-алгоритмом. Пиллар-алгоритм эффективен для выбора начальных центров, но имеет проблемы с выбросами, ведущие к снижению производительности. Улучшение алгоритма позволило решить эту проблему. Проблемой выбора начальных центров занимались также Каур и др. в работе [101]. Шунье Ванг и др. в своей работе [102] использовали для выбора начальных центров матрицу различий, строящуюся с помощью дерева Хаффмана. Махмуд и др. [103] в случае взвешенных многомерных данных использовали для выбора начальных центров эвристический метод, включающий в себя вычисление среднего показателя и сортировку слиянием. Абдул Назир и Себастиан в своей работе [104] описывают улучшенный алгоритм k-средних, включающий в себя специальные методы определения начальных центров и привязки точек к кластерам.

Говоря о задаче k-средних и ее решении, необходимо упомянуть также и алгоритм J-means, разработанный Хансеном и Младеновичем [105], и считающийся одним из наиболее эффективных и точных алгоритмов для данной задачи, а также для p-медианной задачи. Алгоритм заменяет центры на один (лучший с точки зрения целевой функции) из векторов данных и далее продолжает поиск с помощью стандартного k-means.

Алгоритм для задачи k-медоид – Partitioning Around Medoids (PAM) – был предложен Кауфманом (Leonard Kaufman) и Руссивом (Peter J. Rousseeuw) [106]. Он похож на алгоритм k-средних: работа обоих основана на попытках минимизировать ошибку, но PAM работает с медоидами – объектами, являющимися частью исходного множества и представляющими группу, в которую они включены, а k-means работает с центроидами – искусственно созданными объектами, представляющими кластер. Алгоритм PAM разделяет множество из N объектов на k кластеров (N и k являются входными данными алгоритма). Алгоритм работает с матрицей расстояний, его цель –

минимизировать расстояние между представителями каждого кластера и его членами.

PAM-процедура состоит из двух фаз: BUILD и SWAP:

1. BUILD – выполняется первичная группировка, в ходе которой последовательно выбираются *k* объектов в качестве медоидов.

2. SWAP – итерационный процесс, в ходе которого производятся попытки улучшить множество медоидов. Алгоритм выполняет поиск пары объектов (медоид, не-медоид), минимизирующих целевую функцию при замене, после чего обновляет множество медоидов.

На каждой итерации алгоритма выбирается пара (медоид, не-медоид) такая, что замена медоида на не-медоид дает лучшую кластеризацию из возможных. Оценка кластеризации выполняется с помощью целевой функции, вычисляемой как сумма расстояний от каждого объекта до ближайшего медоида. Пока есть возможность улучшения значения целевой функции, процедура изменения множества медоидов повторяется.

Алгоритм 1.2 PAM-процедура

Фаза Build:

1. Выбрать *k* объектов в качестве медоид.

2. Построить матрицу расстояний, если она не была задана.

3. Отнести каждый объект к ближайшей медоиде.

Фаза Swap:

4. Для каждого кластера найти объекты, снижающие суммарное расстояние, и если такие объекты есть, выбрать те, которые снижают его сильней всего, в качестве медоид.

5. Если хотя бы одна медоида поменялась, вернуться к шагу 3, иначе завершить алгоритм.

В число популярных методов автоматической группировки входит и алгоритм Expectation Maximization (EM-алгоритм – максимизация математического ожидания) [107]. Основная идея алгоритма состоит в

искусственном введении вспомогательного вектора скрытых переменных, что сводит сложную оптимизационную задачу к двум шагам:

1. Е-шаг - последовательность итераций по пересчету скрытых переменных по текущему приближению вектора параметров;

2. М-шаг - максимизации правдоподобия (для нахождения следующего приближения вектора).

Задача кластеризации при ее решении ЕМ-алгоритмом сводится к задаче разделения смеси вероятностных распределений. Общее описание ЕМ-алгоритма (для разделения смеси распределений) [107-109]:

Алгоритм 1.3 ЕМ -алгоритм

Дано: Выборка (массив) из N векторов d-мерных данных $X_i = \left(x_{i,1}, \dots, x_{i,d}\right)^T, i = \overline{1,N}$, предполагаемое число распределений в смеси k.

Шаг 1 (инициализация). Выбрать некоторые начальные значения параметров распределений. Как правило, в качестве векторов математических ожиданий μ для задачи разделения смеси гауссовых распределений выбираются значения случайно выбранных векторов данных, а значения дисперсий (или ковариационных матриц) устанавливаются одинаковыми для всех распределений и вычисляются для всей выборки, либо в качестве ковариационных матриц берутся единичные матрицы (аналогично, для экспоненциальных распределений или распределений Лапласа параметр α рассчитывается по всей выборке $X_1,\dots,X_N$).

Установить значения априорных вероятностей каждого из распределений равными для всех распределений $w_j = 1/k, j = \overline{1,k}$.

Шаг 2 (Е-шаг – классификация / разбиение на кластеры).

При нечеткой кластеризации для каждого распределения j и для каждого вектора данных i рассчитывается апостериорная вероятность того, что i-й вектор данных относится к j-му распределению: $g_{i,j} = \frac{f(x_i|j)w_j}{\sum_{l=1}^{k}(f(x_i|l)w_l)} \forall i = \overline{1,N}, j = \overline{1,k}$.

Здесь $f(x_i|j)$ – плотность j-го распределения в точке x_i.

Шаг 3 (М-шаг – модификация параметров распределений).

3.1. Пересчитать значения априорных вероятностей:

$$w_j = \frac{\sum_{i=1}^{N} g_{i,j}}{N} \forall j = \overline{1,k}.$$

3.2. Пересчитать оценки параметров каждого из распределений с учетом апостериорной вероятности того, что конкретный i-й вектор данных входит в j-й кластер с вероятностью $g_{i,j}$. Например, вектор средних значений $\mu_j = (\mu_{j,1}, \dots, \mu_{j,d})$ для каждого кластера рассчитывается по формуле:

$$\mu_{j,l} = \frac{1}{\sum_{q=1}^{N} g_{q,j}} \sum_{i=1}^{N} x_{i,l} g_{i,j} = \frac{1}{N w_j} \sum_{i=1}^{N} x_{i,l} g_{i,j} \quad \forall j = \overline{1,d}, l = \overline{1,k}.$$

Аналогично, оценки среднеквадратичных отклонений рассчитываются так:

$$\sigma_{j,l}{}^2 = \frac{1}{\sum_{q=1}^{N} g_{q,j}} \sum_{i=1}^{N} (x_{i,l} - \mu_{j,l})^2 g_{i,j} = \frac{1}{N w_j} \sum_{i=1}^{N} (x_{i,l} - \mu_{j,l})^2 g_{i,j} \quad \forall j = \overline{1,d}, l = \overline{1,k}.$$

Здесь $\sigma_{j,l}$ – среднеквадратичное отклонение по l- му измерению в j-м распределении (кластере).

При использовании многомерного гауссова распределения с полной ковариационной матрицей

$$\Sigma(j) = \begin{pmatrix} \sigma(j)_1^2 = \sigma_{1,1} & \sigma(j)_{1,2} & \dots & \sigma(j)_{1,d} \\ \sigma(j)_{2,1} & \sigma(j)_2^2 = \sigma(j)_{2,2} & \dots & \sigma(j)_{2,d} \\ \vdots & \vdots & \ddots & 0 \\ \sigma(j)_{d,1} & \sigma(j)_{d,2} & \dots & \sigma(j)_d^2 = \sigma(j)_{d,d} \end{pmatrix}$$

Ее элементы также рассчитываются с учетом апостериорных вероятностей:

$$\sigma(j)_{p,q} = \sigma(j)_{q,p} = \frac{1}{N w_j} \sum_{i=1}^{N} (x_{i,p} - \mu_{j,p})(x_{i,q} - \mu_{j,q}) g_{i,j}.$$

4. Вычислить значение целевой функции – логарифмической функции правдоподобия:

$$Q(w_1, \dots, w_1, \textit{параметры всех распределений}) = \sum_{i=1}^{N} ln \left(\sum_{j=1}^{k} w_j f(x_i | j) \right)$$

5. Проверить условия останова, перейти к Шагу 2.

В качестве условий останова используются следующие условия:

А) достижение предельного числа итераций *ITER*;

Б) достижение предельного времени работы алгоритма t_{max};

В) Отсутствие изменений в значении целевой логарифмической функции правдоподобия.

Отметим, что результатом EM-алгоритма является матрица вероятностей $g_{i,j}$, каждый элемент которой означает вероятность того, что i-й объект относится к j-му кластеру (т.е. порожден j-м распределением).

Классификационный EM-алгоритм (Classification Expectation Maximization - CEM) [108, 110] – модификация EM-алгоритма работает по принципу четкой классификатора данных выборки. В этом случае каждый объект относится к единственному кластеру. CEM-алгоритм почти совпадает с другой модификацией – SEM (Stochastic EM) [111, 112, 113], только у первого на каждом шаге вводится детерминированное правило, что данные относятся лишь к одному кластеру, для которого вычислили максимальную апостериорную вероятность. Таким образом, CEM-алгоритм, в отличие от EM, решает задачу четкой кластеризации.

Как уже отмечалось в предыдущем параграфе, связь задач автоматической группировки с задачами теории размещения очевидна. Кроме того, наиболее популярные алгоритмы решения таких задач как k-средних, k-медоид и EM-алгоритм, схожи по своей структуре. Каждый из них содержит два чередующихся шага. Во всех этих задачах целевые функции обладают свойством многоэкстремальности. Общность свойств подобных моделей и соответствующих алгоритмов дает обоснованную надежду на то, что схожие способы повышения точности и стабильности получаемых результатов решения задач окажутся эффективными.

Для дальнейшего изложения в настоящей диссертации каждую из процедур: k-средних, k-медоид и CEM обозначим как двухшаговый алгоритм локального поиска. Тем более что сами процедуры k-средних и k-медоид являются, по сути, алгоритмами поиска с чередующимися окрестностями. Далее при решении задач k-средних в качестве двухшагового алгоритма локального поиска будет

реализован Алгоритмом 1.1, соответственно для k-медоид - Алгоритмом 1.2 и максимизации функции правдоподобия – CEM-алгоритмом.

В считающихся классическими алгоритмах автоматической группировки, (например, k-средних, k-медоид) выполняется направленный поиск возможных решений в относительно небольшом подмножестве пространства, который не гарантирует нахождение строго оптимального решения при использовании различных ограничений (числа, формы получаемых групп и т.п.). Несмотря на то, что существует множество методов решения задач автоматической группировки на основе классических моделей [114], претендующих на нахождение глобально оптимального решения (практически неприменимых к очень большим задачам и не гарантирующих точное решение), все-таки основное направление современных исследований состоит в развитии эвристических методов и алгоритмов, находящих субоптимальные решения, но при этом близкие к истинному оптимуму задачи [115, 116]. Таким образом, при решении задач автоматической группировки с повышенными требованиями к точности и стабильности результата, известные алгоритмы показывают далеко не лучшие результаты, в особенности за ограниченное фиксированное время.

1.4 Пример актуальной задачи автоматической группировки с повышенными требованиями к точности и стабильности результата

Актуальность решения задач автоматической группировки с повышенными требованиями к точности и стабильности результата обусловлена диапазоном их применения, как в задачах кластерного анализа, так и непосредственно в практических задачах на производстве где требуется обеспечение высокой точности разделения на однородные партии промышленной продукции. Рассмотрим один из практических примеров.

В статье Орлова В.И. и Сергеевой Н.А. [117] сказано: «Современный космический аппарат – это сложная электронная система, которая, находясь в космосе в течение 10–15 лет, должна сама себя диагностировать, проверять,

принимать решение в рамках поставленных задач и выполнять различные возложенные на нее функции. Космос является агрессивной средой, которая обладает различными деструктивными характеристиками. В их числе глубокий вакуум, большой перепад температур, радиация, потоки заряженных частиц и т. д. Бортовая аппаратура в космическом пространстве не подлежит ремонту, именно поэтому она называется неремонтопригодной, и, соответственно, надежность такой аппаратуры должна быть максимальной. Требуемый уровень надежности обеспечивается за счет различных факторов, самым главным из которых является использование высоконадежных электронных компонентов. Космический аппарат (КА) содержит от 100 до 200 тыс. электронных компонентов (ЭКБ). К ним относятся микросхемы, транзисторы, диоды, конденсаторы, реле, кварцевые резонаторы, резисторы и т. д. С каждым годом габаритные размеры электронных компонентов становятся все меньше, а степень интеграции микросхем – все выше. Размеры безвыводных резисторов или конденсаторов достигают 1–2 мм, а вес – доли грамма. Интегральные микросхемы типа процессора обладают возможностями персонального компьютера, который упакован в корпус размерами 5×5 см. Комплектация бортовой аппаратуры КА высоконадежной ЭКБ является одной из основных задач современной космической отрасли. В первую очередь следует предотвратить попадание в аппаратуру низкосортной фальсифицированной продукции, которая не удовлетворяет требованиям, предъявляемым к надежности. В рамках решения этой проблемы необходимо обеспечить закупку ЭКБ у проверенных поставщиков, а также проведение входного контроля (ВК), дополнительных отбраковочных испытаний (ДОИ) и разрушающего физического анализа (РФА) ЭКБ. Особое значение приобретает индивидуальная отбраковка компонентов».

В отличие от США и западноевропейских государств, в нашей стране нет специализированных производств ЭКБ для космической отрасли, поэтому электрорадиоизделия (ЭРИ) должны подвергаться проверке для использования в аппаратуре космических аппаратов. С этой целью уже много лет в России существует принцип комплектования аппаратуры космических аппаратов через

специализированные испытательные технические центры [118, 119] с проведением различных испытаний ЭКБ (ВК, ДОИ, РФА и ДНК - диагностического неразрушающего контроля).

Такой подход сразу же дал ощутимые результаты. Так, если большинство из эксплуатируемых до 2000 года космических аппаратов АО «ИСС» (г. Железногорск) имели замечания по качеству функционирования, начиная с первых дней или месяцев эксплуатации, то на эксплуатируемом с апреля 2000 года космическом аппарате «Sesat» не выявлено существенных замечаний к ЭРИ в течение вот уже почти 20 лет эксплуатации. По мнению большинства экспертов, это произошло вследствие того, что впервые в практике все 100 процентов ЭКБ космического аппарата «Sesat» прошли ВК, ДОИ, ДНК и РФА [119-121].

В различных публикациях [122-128] изложены, например, задачи обеспечения радиационной стойкости ЭРИ, но они основываются на предположении, что радиационная стойкость любого электрорадиоизделия из производственной партии известна и, главное, одинакова. На практике характеристики ЭРИ (включая и радиационную стойкость) внутри производственной партии различны и зависят от различных причин [129].

Для того, чтобы распространить результаты испытаний на всю производственную партию изделий, необходимо быть уверенными в том, что мы имеем дело с партией электрорадиоизделий, изготовленной из единой (однородной) партии сырья. Поэтому выявление однородных производственных партий из сборных партий ЭРИ должно стать одним из важнейших этапов при проведении испытаний. На практике проводится ряд тестов, от десятков до нескольких тысяч для каждого изделия, результаты сводятся в таблицу и служат данными для анализа. Сама процедура разделения параметров должна быть регламентирована, повторный обсчет данных должен давать те же или очень близкие результаты. Из вышесказанного формируется задача: разделение на однородные производственные партии на основе данных тестовых испытаний (кластеризация). Используются все данные, собранные во время тестов на

отклонение от заданных параметров. Это означает, что кластеризация выполняется в пространстве от десятков до сотен тысяч векторов [130].

Современные методы кластерного анализа предлагают широкий выбор средств выявления разнородных по совокупности параметров групп. В то же время, метод выявления различных по параметрам групп (кластеров) электрорадиоизделий должен давать воспроизводимые результаты. Повысить точность методов автоматической группировки по выявлению различных по параметрам групп электрорадиоизделий позволяют алгоритмы, предложенные в главах 2 и 3.

1.5 Метод локального поиска с чередующимися окрестностями

Методы локального поиска получили дальнейшее развитие в метаэвристиках (методах оптимизации, многократно использующих простые правила или эвристики для достижения оптимального или субоптимального решения, характеризующихся большей устойчивостью) [131]. Рассмотрим одну из них, получившую название поиск с чередующимися окрестностями (VNS – Variable Neighborhoods Search) – популярном методе решения задач дискретной оптимизации Н. Младеновича и П. Хансена, который позволяет находить хорошие субоптимальные решения достаточно больших задач автоматической группировки [132, 133, 134]. Основная идея состоит в систематическом изменении вида окрестности в ходе локального поиска.

Существует много вариантов реализации метода поиска с чередующимися окрестностями для задач большой размерности. Гибкость и высокая эффективность объясняют ее конкурентоспособность при решении NP-трудных задач, что отражено в работах Кочетова Ю.А., Лопеса Ф.Г., Бримберна Дж., Левановой Т.В., Алексеевой Е.В. и др., в частности, для решения задач автоматической группировки и размещения [30, 61, 70, 71, 135, 136], множественной задачи Вебера [137], задачи о p-медиане [72, 138] и многих других.

Обозначим через N_k, k=1,..k_{max}, конечное множество видов окрестностей, предварительно выбранных для локального поиска. Предлагаемый метод с чередующимися окрестностями опирается на то, что локальный минимум в одной окрестности не обязательно является локальным минимумом в другой окрестности, при этом глобальный минимум является локальным в любой окрестности [139]. Кроме того, в среднем локальные минимумы ближе к глобальному, чем случайно выбранная точка, и они расположены близко друг к другу. Это позволяет сузить область поиска глобального оптимума, используя информацию об уже обнаруженных локальных оптимумах. Эта гипотеза лежит в основе различных операторов скрещивания (crossover) для генетических алгоритмов [140] и других подходов.

Реализация метода локального поиска с чередующимися окрестностями возможна одним из трех способов: детерминированным, вероятностным или смешанным, сочетающим в себе два предыдущих [139].

В детерминированном локальном спуске с чередующимися окрестностями (VND) предполагается фиксированный порядок смены окрестностей и поиск локального минимума относительно каждой из них. Вероятностный локальный спуск с чередующимися окрестностями (RVNS) отличается от предыдущего метода VND случайным выбором точек из окрестности $O_k(x)$. Этап поиска лучшей точки в окрестности опускается. Алгоритмы RVNS наиболее продуктивны при решении задач большой размерности, когда применение детерминированного варианта требует слишком много «машинного» времени для выполнения одной итерации.

Основная схема локального поиска с чередующимися окрестностями (VNS) является комбинацией двух предыдущих вариантов (VND и RVNS) [133].

Алгоритм 1.4 VNS **(**Variable Neighborhoods Search)

Шаг 1. Выбрать окрестности O_k, k=1,..k_{max}, и начальную точку x.

Шаг 2. Повторять, пока не выполнен критерий остановки.

2.1. $k \leftarrow 1$.

2.2. Повторять до тех пор, пока $k \leq k_{max}$:

2.2.1. случайно выбрать точку $x' \in O_k(x)$;

2.2.2. применить локальный спуск с начальной точки *x'*, не меняя координат, по которым *x* и *x'* совпадают. Полученный локальный оптимум обозначается *x''*;

2.2.3. если *F(x'')* < *F(x)*, то полагается *x*←*x''*, *k*←1, иначе *k*← *k* + 1.

Суть алгоритма поиска с чередующимися окрестностями [132] заключается в том, что для некоторого промежуточного решения определяется множество окрестностей этого решения. Из этого множества выбирается очередной вид окрестности, для поиска в которой применяется соответствующий алгоритм локального поиска. Если этим алгоритмом улучшенное решение найдено, промежуточное решение заменяется этим новым решением, и поиск в той же окрестности продолжается. Если очередной алгоритм локального поиска не смог улучшить решение, из множества окрестностей промежуточного решения выбирается новая окрестность поиска.

Критерием остановки может служить максимальное время счета или максимальное число итераций без смены лучшего найденного решения. При решении задач большой размерности сложность выполнения одной итерации становится весьма большой, и требуются новые подходы для разработки эффективных методов локального поиска.

1.6 Развитие метода жадных эвристик для задач автоматической группировки объектов

Основной отличительной особенностью жадных (англ. «greedy») агломеративных эвристических методов состоит в том, что они, являясь методами локального поиска (последовательно улучшающими известный результат) в некоторой окрестности известного решения, выбирают в качестве следующего решения тот вариант, который дает наибольшее уменьшение значения целевой функции (наибольший прирост значения – в случае максимизации).

Метод жадных эвристик был предложен Казаковцевым Л.А. и Антамошкиным А.Н. для задач автоматической группировки на моделях теории размещения [141, 142]. Алгоритмы данного метода получают для практических задач результаты, которые трудно существенно улучшить другими методами за сопоставимое время. При том, что алгоритмы метода жадных эвристик в основном рандомизированы, получаемые ими результаты достаточно стабильны, то есть дают очень близкие результаты при перезапусках.

В методе жадных эвристик используются эволюционные алгоритмы как один из возможных способов организации глобального поиска, в том числе подходы и Красноярской школы эволюционных алгоритмов (Семёнкин Е.С. и др.) [143-146].

Задачи *k*-средних и *p*-медианная, за исключением особых случаев, являются NP-трудными, требующими глобального поиска. С точки зрения обеспечения воспроизводимости результатов вычислений проблемой является то, что результат зависит от выбора начальных центров кластеров.

Возможно применение широкого круга стратегий глобального поиска.

Для задач автоматической группировки при разделении множества объектов на *k* групп (кластеров) метод жадных эвристик можно представить в виде трех вложенных циклов:

1) Цикл, осуществляющий итерации некоторой стратегии глобального поиска, Этот цикл генерирует промежуточные решения, представленные множествами, мощность которых выше *k*.

2) Выполнение жадной эвристической процедуры, в ходе которой могут запускаться алгоритмы локального поиска, приводящие промежуточные решения к допустимым и одновременно улучшающие промежуточные решения.

3) Цикл локального поиска, предусматривающий оценку последствий исключения элементов из промежуточного решения.

На Рисунке 1.1 в самом общем виде представлена блок-схема алгоритма метода жадных эвристик [142].

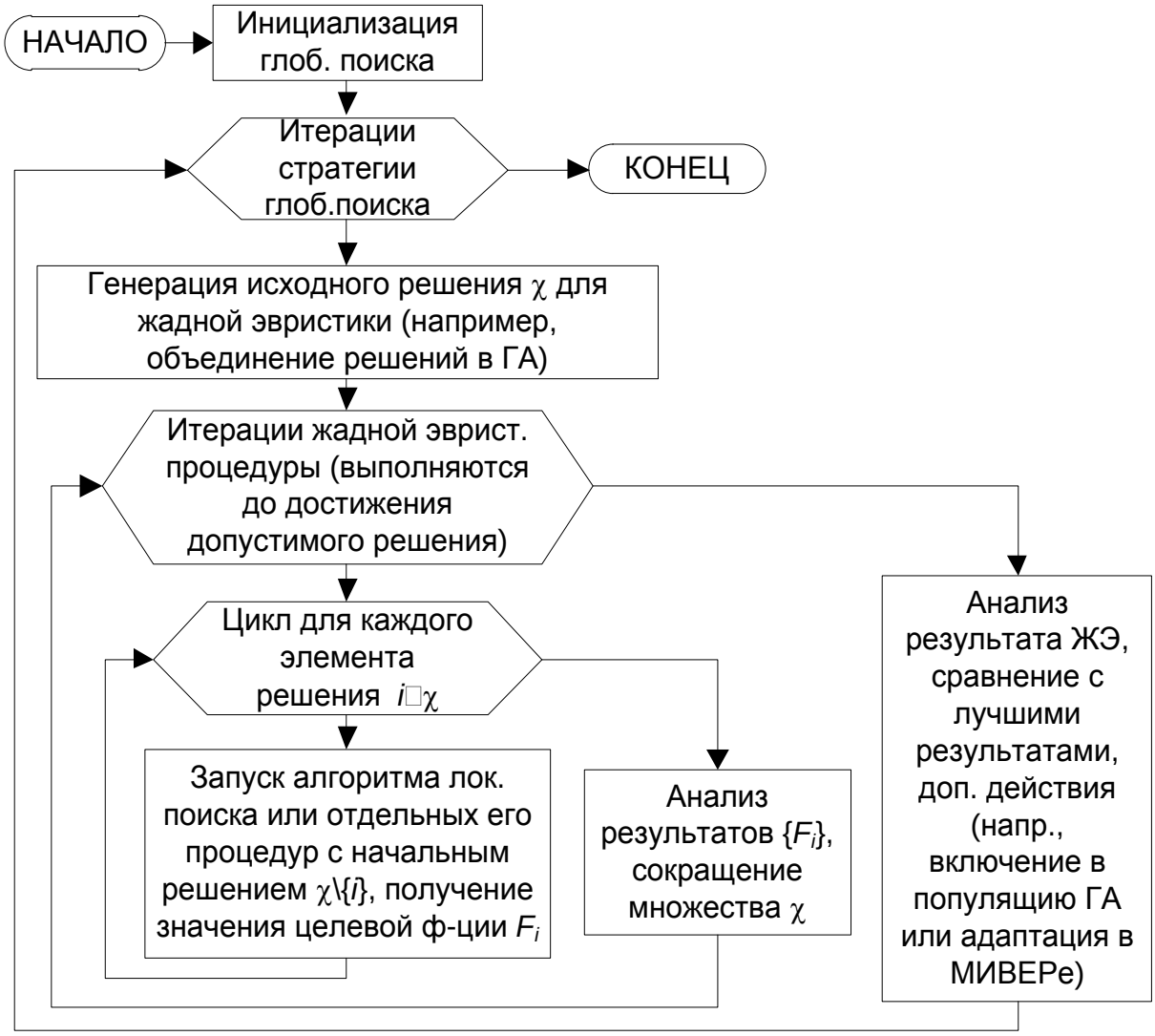
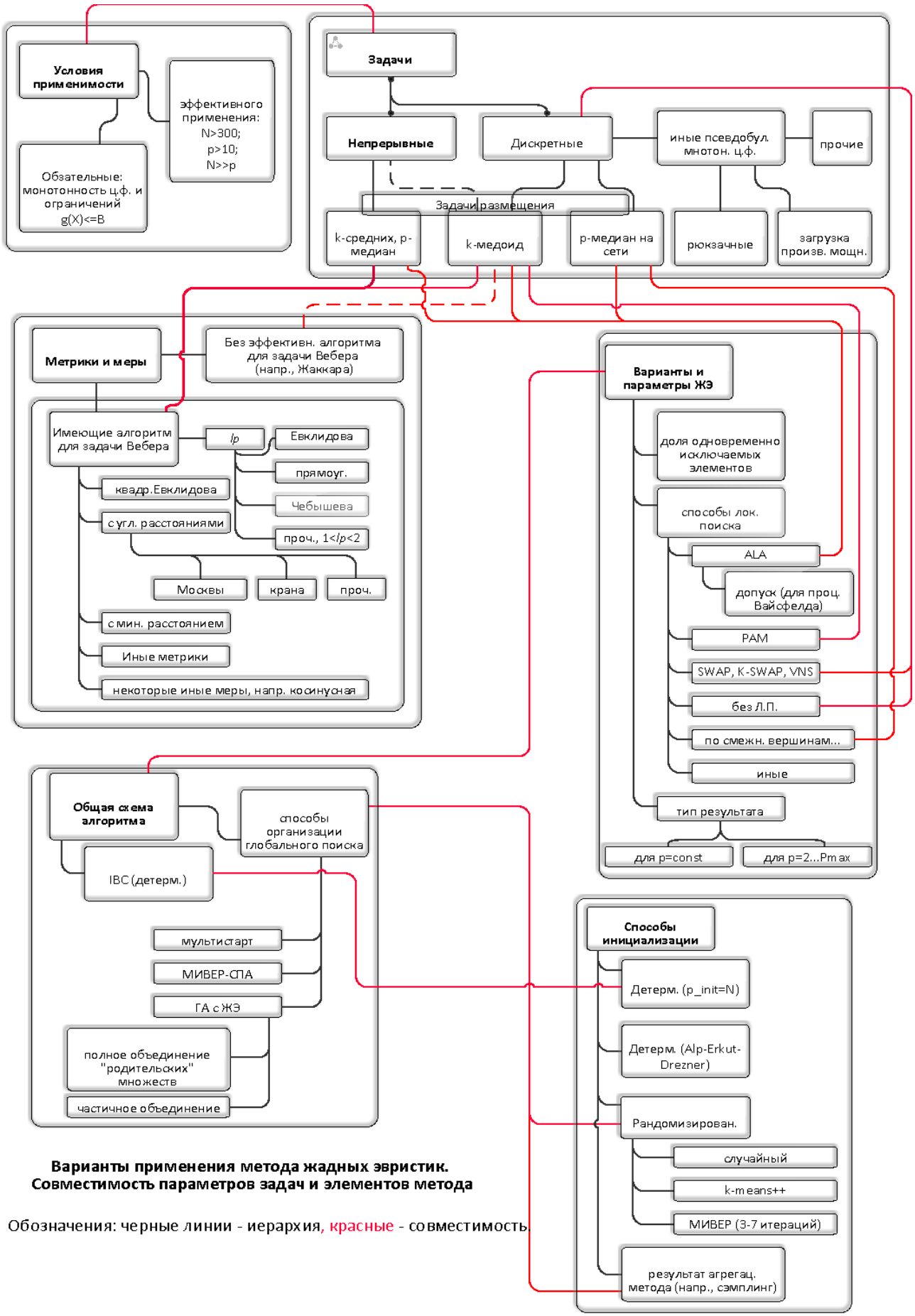


Рисунок 1.1 - Общая схема алгоритма метода жадных эвристик [142]

Схема на Рисунке 1.2 отражает взаимную совместимость различных постановок задач (дискретная/непрерывная задача размещения или группировки, задача псевдобулевой монотонной оптимизации), различных стратегий глобального поиска (МИВЕР / ГА / мультистарт / детерминированные методы), различных вспомогательных методов локального поиска, эффективных при решении той или иной конкретной задачи, а также используемых мер расстояния при их применении в составе метода жадных эвристик [142].



Варианты применения метода жадных эвристик. Совместимость параметров задач и элементов метода

Обозначения: черные линии - иерархия, красные - совместимость

Рисунок 1.2 - Схема совместимости компонентов метода жадных эвристик [142]

Алгоритм работы базовой жадной агломеративной эвристической процедуры представлен ниже:

Алгоритм 1.5 Базовая жадная агломеративная эвристическая процедура (Greedy)

Дано: начальное число кластеров K, требуемое число кластеров $k<K$, начальное решение $S=\{X_1,...,X_k\}$.

1: Передать решение S в двухшаговый алгоритм локального поиска, получить новое, улучшенное решение S.

пока $K \neq k$

для каждого $i' \in \{\overline{1,K}\}$

2.1: $S' = S \setminus \{X_{i'}\}$.

2.2: Передать решение S' в двухшаговый алгоритм локального поиска, выполнить от 1 до 3 итераций алгоритма, полученное значение (значение целевой функции) сохранить в $F'_{i'}$.

конец цикла

3: $i'' = \arg\max_{i'=1,k} F_{i'}$.

4: Получить решение $S'' = S \setminus \{X_{i''}\}$, улучшить его с помощью двухшагового алгоритма локального поиска.

конец цикла

Жадная агломеративная эвристическая процедура для задачи k-средних и аналогичных задач состоит из двух шагов. Пусть имеются два известных (родительских) решения задачи (первое из которых, например, является лучшим из известных), представленных множествами центров кластеров S.

Вначале множества родительских решений объединяются. Получаем промежуточное недопустимое (с избыточным числом кластеров) решение (Рисунок 1.3).

Затем производится последовательное уменьшение числа центров. Каждый раз отсекается тот центр, удаление которого дает наименее существенное ухудшение значения целевой функции.

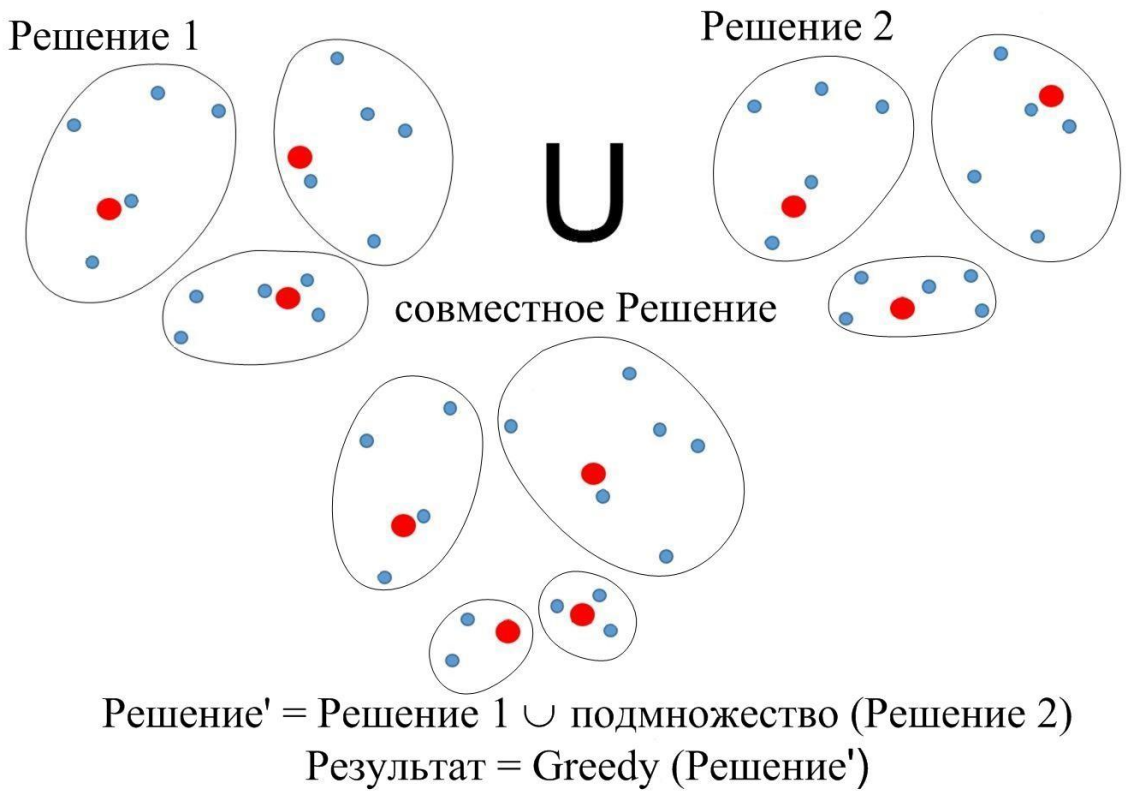


Рисунок 1.3 – Принцип объединения множеств родительских решений

Для повышения точности методов автоматической группировки с применением метода жадных эвристик для задач автоматической группировки с повышенными требованиями к точности и стабильности результата предлагается новый подход к разработке алгоритмов автоматической группировки, основанный на параметрических оптимизационных моделях, с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных агломеративных эвристических процедур, изложенный в Главах 2 и 3.

Выводы к Главе 1

Анализ литературных источников показал, что методы автоматической группировки объектов, несмотря на сходство с методами решения задач теории размещения, развивались во многом раздельно. Популярные алгоритмы автоматической группировки объектов основаны в основном на эвристических методах локального поиска и стратегиях глобального поиска.

Выбор в качестве средства локального поиска жадных агломеративных эвристических алгоритмов обусловлен тем, что эти методы позволяют получить результаты высокой точности, хотя и требуют при этом значительных вычислительных затрат. Сами по себе эти алгоритмы являются детерминированными процедурами, что дает надежду на получение более стабильных результатов при применении таких эвристических алгоритмов в составе различных стратегий глобального поиска, в том числе рандомизированных.

Отмечая некоторый дефицит компромиссных по качеству результата и времени счета методов автоматической группировки (под качеством будем понимать точность - близость значения целевой функции к глобальному оптимуму и стабильность - близость получаемых значений друг к другу при множественных запусках алгоритма) в настоящем исследовании ставится задача разработать усовершенствованные алгоритмы метода жадных эвристик для задач автоматической группировки с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями, к которым предъявляются высокие требования по точности и стабильности результата.

Приведенный обзор опубликованных работ в этой области дает обоснованную надежду, что предложенная задача разработки новых алгоритмов найдет экспериментальное подтверждение, а потенциал, заложенный в методе жадных эвристик, будет реализован в более полной мере, чему и посвящены главы 2 и 3.

ГЛАВА 2. АЛГОРИТМЫ МЕТОДА ЖАДНЫХ ЭВРИСТИК С ЧЕРЕДУЮЩИМИСЯ ОКРЕСТНОСТЯМИ ДЛЯ ЗАДАЧИ K-СРЕДНИХ

Данная глава посвящена разработке комбинированных алгоритмов метода жадных эвристик для задач автоматической группировки с повышенными требованиями к точности и стабильности результата, с совместным применением алгоритмов поиска с чередующимися рандомизированными окрестностями, а также параллельных жадных эвристических алгоритмов автоматической группировки для массивно-параллельных систем применительно к задаче k-средних.

2.1 Жадные агломеративные эвристические процедуры

Алгоритмы метода жадных эвристик, включая модификации жадной агломеративной эвристической процедуры в составе различных схем глобального поиска, в сочетании с условиями их применимости к тем или иным задачам, являются эффективным методом решения оптимизационных задач автоматической группировки, размещения, а также задач псевдобулевой оптимизации с большим объемом входных данных в зависимости от условий и параметров решаемых задач [142], а сам метод жадных эвристик состоит в составлении эффективной комбинации компонентов при построении автоматизированной системы.

На Рисунке 2.1 показана структурная схема компонентов метода жадных эвристик и варианты применения данного метода [142]. Порядок расположения компонентов по вертикали отражает вложенность алгоритмов. Схема также отображает применимость тактик локального и глобального поисков к различным классам задач.

Метод жадных эвристик для задач автоматической группировки объектов был рассмотрен в Главе 1. Для расчетов при количестве кластеров более 50 в настоящем исследовании предложен измененный Алгоритм 1.5 базовой жадной

агломеративной эвристической процедуры. Кроме запуска двухшагового алгоритма локального поиска, наиболее вычислительно требовательной частью жадной агломеративной эвристики, при большом числе кластеров является шаг 3, на котором Алгоритм 1.5 вычисляет суммарное расстояние после удаления одного кластера: $F'_{i'} = F(S')$, где $S' = S \setminus \{X_{i'}\}$.

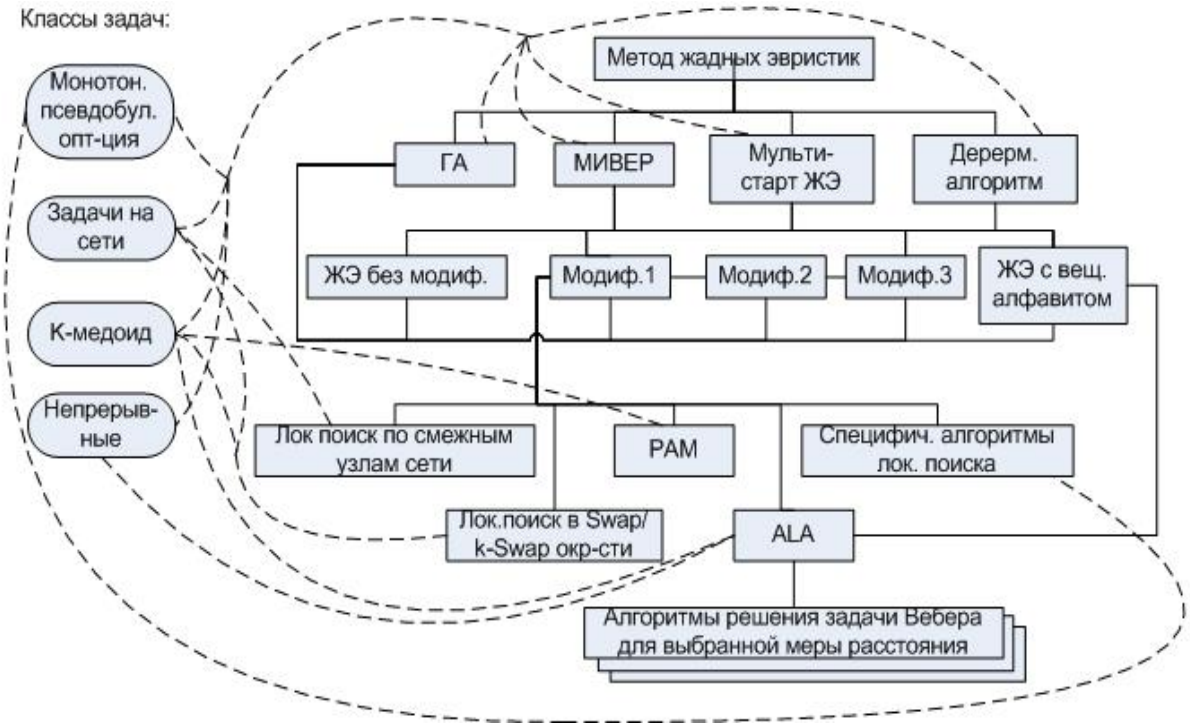


Рисунок 2.1 - Компоненты метода жадных эвристик, их взаимная совместимость (сплошные линии) и применимость к классам задач (курсивные линии) [142]

Производительность Алгоритма 1.5 при больших объемах данных во время выполнения расчетов становится проблемой, особенно когда найти правильный параметр *k* (число кластеров) в практических задачах можно только путем выполнения нескольких запусков с разным количеством кластеров. При увеличении количества кластеров Алгоритм 1.5 на шаге 2 начинает работать всё медленнее (алгоритм требует все большего количества итераций, и каждая итерация требует возрастающих вычислительных ресурсов), поэтому мы внесли изменения и реализовали удаление кластеров не по одному, а по несколько за одну итерацию.

Алгоритм 2.1 Базовая жадная агломеративная эвристическая процедура для задач с большим числом кластеров

Дано: начальное число кластеров K, необходимое количество кластеров $k<K$, $k>50$, первоначальное решение S, $|S|=K$.

1: Улучшить решение S двухшаговым алгоритмом локального поиска (если это возможно).

пока $K \neq k$

для каждого $i' \in \{\overline{1, K}\}$

2: $S' = S \setminus \{X_{i'}\}$. Вычислить $F'_{i'} = F(S')$, где $F(.)$ значение целевой функции (например, (1.1) для задачи k-средних).

конец цикла

3. Установить S_{elim} из n_{elim} центроидов, $S_{elim} \subset S$, $|S_{elim}| = n_{elim}$, с минимальными значениями $F'_{i'}$. Здесь, $n_{elim} = \max\{1, 0.2 \cdot (|S| - k)\}$.

4: Получить новое решение $S = S \setminus S_{elim}$, $K = K - 1$, и улучшить его с помощью двухшагового алгоритма локального поиска.

конец цикла

Алгоритм базовой жадной агломеративной эвристической процедуры лег в основу трех алгоритмов. Схожие жадные агломеративные эвристические процедуры использовались в качестве операторов кроссинговера в эволюционных (генетических) алгоритмах метода жадных эвристик [142, 147-149]. В настоящей работе данными процедурами формируются окрестности, используемые для модификации известного решения при поиске в алгоритме поиска с чередующимися окрестностями. Предложенные новые эвристические процедуры модифицируют известное решение с использованием второго известного решения (Алгоритмы 1.5 и 2.1).

Алгоритм 2.2 Жадная процедура 1

Дано: множества центров кластеров $S' = \{X'_1, \ldots, X'_k\}$ и $S'' = \{X''_1, \ldots, X''_k\}$

для каждого $i' \in \{\overline{1,K}\}$

1: Объединить *S'* с элементом множества *S''*: $S = S' \cup \{X''_{i'}\}$

2: Запустить базовую жадную агломеративную эвристическую процедуру (Алгоритм 1.5 или 2.1) с *S* в качестве начального решения. Полученный результат (полученное множество, а также значение целевой функции) сохранить.

3: Возвратить в качестве результата лучшее (по значению целевой функции) из решений, полученных на шаге 2.

конец цикла

Возможен вариант, в котором множества объединяются частично, при этом первое множество берется полностью, а из второго множества выбирается случайным образом случайное число элементов [147, 150].

Алгоритм 2.3 Жадная процедура 2

Дано: см. Алгоритм 2.2.

1: Выбрать случайное $r' \in [0;1)$. Присвоить $r=[(k/2-2)\, r'^2]+2$.

Здесь [.] – целая часть числа.

2: **для** *i* **от** 1 **до** *k-r*

2.1: Сформировать случайно выбранное подмножество *S'''* элементов множества *S''* мощности *r*. Объединить множества $S = S' \cup S'''$.

2.2: Запустить базовую жадную агломеративную эвристическую процедуру (Алгоритм 1.5 или 2.1) с этими объединенными множествами в качестве начального решения.

конец цикла

3: Возвратить в качестве результата лучшее (по значению целевой функции) из решений, полученных на шаге 2.2.

Более простой вариант Алгоритма 2.2 представлен ниже, уже с полным объединением множеств.

Алгоритм 2.4 Жадная процедура 3

Дано см. Алгоритм 2.2.

1: Объединить множества $S = S' \cup S''$.

2: Запустить базовую жадную агломеративную эвристическую процедуру (Алгоритм 1.5 или 2.1) с S в качестве начального решения.

Данные алгоритмы могут использоваться в составе различных стратегий глобального поиска, а в качестве окрестностей, в которых производится поиск решения, используются множества решений, производные («дочерние») по отношению к решению S', образованные комбинированием его элементов с элементами некоторого решения S'' и применением базовой жадной агломеративной эвристической процедуры (Алгоритм 1.5 или 2.1).

В следующем параграфе рассмотрим применение подходов метода жадных эвристик для задач автоматической группировки, с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями.

2.2 Принцип работы комбинированных алгоритмов поиска с чередующимися рандомизированными окрестностями для задачи k-средних

В качестве одного из вариантов локального поиска для метода жадных эвристик [142] возможно применение алгоритмов поиска с чередующимися окрестностями (VNS-алгоритмов) [134, 151, 152]. В работах Кочетова Ю.А., Младеновича Н. и Хансена П. [43, 132, 134, 139] приведен обзор методов локального поиска, основанных на идее чередующихся окрестностей, в том числе способы комбинирования этих методов с другими метаэвристиками.

На Рисунке 2.2 приведена основная схема локального поиска с чередующимися окрестностями (Алгоритм 1.4).

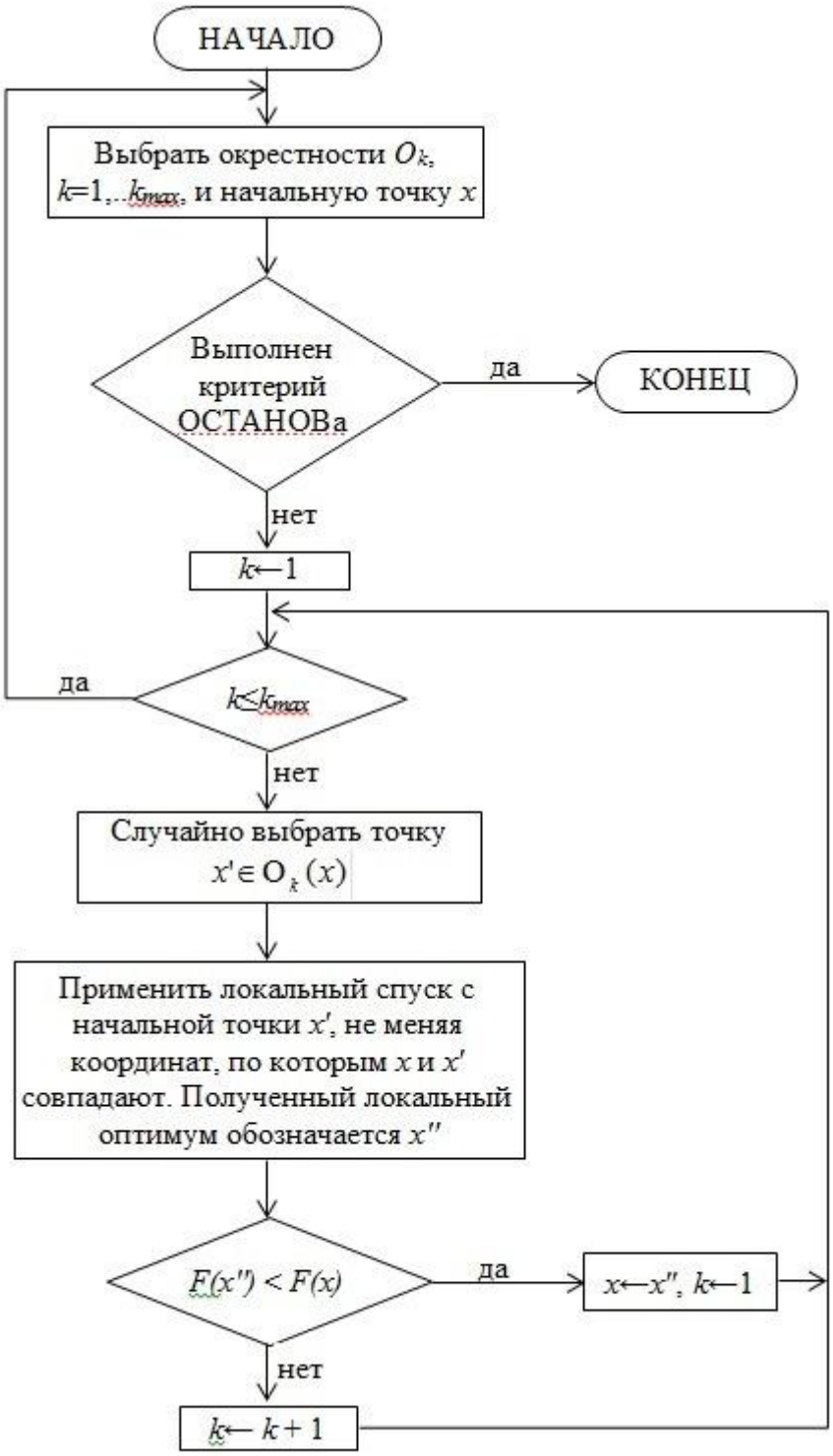


Рисунок 2.2 - Блок–схема VNS-алгоритма

Алгоритмы 2.2-2.4 могут осуществлять поиск в окрестностях известного промежуточное решение *S'*, где решения, принадлежащие окрестности, образуются добавлением элементов другого решения *S''* с последующим удалением «лишних» центров кластеров жадной агломеративной эвристической

процедурой. Таким образом, второе решение *S''* является параметром окрестности, выбираемым случайным образом (рандомизированным) [153].

Алгоритм автоматической группировки для задачи k-средних с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных агломеративных эвристических процедур можно описать следующим образом:

Алгоритм 2.5 k-GH-VNS (Greedy Heuristic in the Variable Neighborhood Search) для задачи k-средних

1: Получить решение *S*, запустив Алгоритм 1.1 из случайным образом сгенерированного начального решения.

2: $O=O_{start}$ номер окрестности поиска.

3: $i=0$, $j=0$.

пока $j < j_{\max}$

пока $i < i_{\max}$

4: **если** не выполняются условия ОСТАНОВа, **то** получить решение *S'*, запустив Алгоритм 1.1 из случайного начального решения.

повторять

5: В зависимости от значения *S* (возможны значения 1, 2 или 3), запустить Алгоритм Жадная процедура 1, 2 или 3 соответственно с начальными решениями *S* и *S'*. Так, окрестность определяется способом включения центров кластеров из второго известного решения и параметром окрестности – вторым известным решением.

если новое решение лучше, чем *S*, **то**

записать новый результат в *S*, $i=0$, $j=0$.

иначе выйти из цикла.

конец цикла

6: $i=i+1$.

конец цикла

7: i=0, j=j+1, O=O+1, **если** O>3, **то** O=1.

конец цикла

В данном алгоритме i_{max} - число безрезультатных поисков в окрестности, а j_{max} - число безрезультатных переключений окрестностей. Значения этих двух параметров важны при расчетах. Мы использовали значения: i_{max}=2k, j_{max}=2.

Как показали вычислительные эксперименты, параметр O_{start}, задающий номер окрестности, с которой начинается поиск, особенно важен. Были проведены вычислительные эксперименты со всеми его значениями. В зависимости от значения параметра O_{start}, версии алгоритмов обозначены k-GH-VNS1, k-GH-VNS2, k-GH-VNS3. Отметим, что работа алгоритма поиска может начинаться с разных окрестностей.

Важным является способ получения второго решения S’ на Шаге 4. По умолчанию второе решение содержит число центров, равное числу центров в решении S. Мы также использовали модификации Алгоритма 2.5, в которых число центров в решении S’ выбирается случайным образом из множества $\{\overline{2,|S|}\}$, где $|S|$ – число центров в решении S. В этом случае алгоритмы названы k-GH-VNS1-RND, k-GH-VNS2-RND, k-GH-VNS3-RND.

Отметим, что для задач k-средних весьма эффективной является процедура j-means [154], область применения которой ограничена довольно небольшими задачами (числом векторов данных обычно меньше 1000) [142]. Соседство текущего решения определяется всеми возможными заменами центроида на объект с последующими соответствующими изменениями. Движение осуществляется в таких окрестностях, пока не будет достигнут локальный оптимум. Данная процедура сводится к замене центров на один (лучший с точки зрения целевой функции) из векторов данных с последующим продолжением поиска стандартным алгоритмом k-средних.

Алгоритм 2.6 j-means

Дано: начальное решение *S*, представленное как множество центров.

цикл

1: Передать решение *S* в Алгоритм 1.1как начальное решение.

2: **если** на Шаге 1 значение целевой функции не улучшилось, **то** ОСТАНОВ и возврат решения *S*.

3: Сформировать массив $I = \overline{\{1, p\}}$, расположить элементы массива в случайном порядке. Это ускоряет процедуру.

для всех $i' \in I$

4.1: $i = I_i'$, $f' = +\infty$.

4.2: $j^{'} = \arg\min_{j \in \overline{\{1,N\}}} F((S \setminus \{S_i\} \cup \{A_j\}))$, где S_i – *i*-й центроид в решении, A_j-*j*-й вектор данных.

4.3: **если** $F((S \setminus \{S_i\} \cup \{A_j\})) < F(S)$, **то** $S = S \setminus \{S_i\} \cup \{A_j\}$ **и** завершить цикл.

конец цикла

конец цикла

Для вычислительных экспериментов мы использовали комбинированные версии алгоритмов k-GH-VNS и j-means, обозначив их: j-means-GH-VNS (Алгоритм 2.7). В нем уже не три значения номера окрестности *O* (как в Алгоритме 2.5), а четыре. Если значение равно 4, то запуск Алгоритма 2.6 (j-means) иначе Алгоритмов 2.2-2.4.

Алгоритм 2.7 j-means-GH-VNS (комбинированный алгоритм k-GH-VNS с j-means)

1: Получить решение *S*, запустив Алгоритм 1.1 из случайным образом сгенерированного начального решения.

2: $O=O_{start}$ (по номеру окрестности поиска)

3: $i=0$, $j=0$.

пока $j < j_{\max}$

пока $i < i_{\max}$

4: **если** не выполняются условия ОСТАНОВа, **то** получить решение *S'*, запустив Алгоритм 1.1 из случайного начального решения

повторять

5: В зависимости от значения *S* (возможны значения 1, 2, 3 и 4), запустить Алгоритм 2.2, 2.3, 2.4 или 2.6 соответственно с начальными решениями *S* и *S'*. Так, окрестность определяется способом включения центров кластеров из второго известного решения и параметром окрестности – вторым известным решением.

если новое решение лучше, чем *S*, **то**

записать новый результат в *S*, *i*=0, *j*=0.

иначе выйти из цикла

конец цикла

6: Инкрементировать *i*.

конец цикла

7: *i*=0, *j*=*j*+1, *O*=*O*+1, **если** *O*>4, **то** *O*=1

конец цикла

Алгоритм j-means довольно «тяжелый» в плане вычислений и может очень долго искать решение при большом объеме данных и большом числе кластеров. На небольших наборах данных это свойство не так существенно, а при больших объемах данных требуется много времени и ресурсов. Особенно больших вычислительных ресурсов требует комбинация j-means-GH-VNS3, где используется Жадная процедура 3 (Алгоритм 2.4) с полным объединением множеств, требующая наибольших вычислительных ресурсов среди новых процедур, используемых для формирования окрестностей поиска. Поэтому данная комбинация, как показали вычислительные эксперименты, не дает существенных

преимуществ в сравнении с алгоритмом k-GH-VNS3, и ее результаты не приводятся.

2.3 Результаты вычислительных экспериментов с новыми алгоритмами для задачи k-средних

Для тестирования нашего нового алгоритма (k-GH-VNS) в трех различных его модификациях были использованы классические наборы данных из репозиториев UCI (Machine Learning Repository) [155] и Clustering basic benchmark [156]. Для корректности исследований наборы данных были подобраны из различных сфер жизни и различных объемов данных:

- Ionosphere (351 векторов данных, каждый размерностью 35) - классификация радиолокационных возвратов из ионосферы (прогнозирование высокоэнергетических структур в атмосфере по данным антенны);
- Mopsi-Joensuu (6014 векторов данных, каждый размерностью 2) - местоположение пользователей до 2012 года (Йоенсуу - город на востоке Финляндии);
- Chess (3196 векторов данных, каждый размерностью 36) - шахматные задачи (Король + Ладья против Короля + Пешка);
- Europe (169309 векторов данных, каждый размерностью 2) - классификация европейских навыков, компетенций, квалификаций и профессий;
- BIRCH3 (100001 векторов данных, каждый размерностью 2) - березовые наборы (кластеры случайного размера в случайных местах);
- KDDCUP04BioNormed (65536 векторов данных, каждый размерностью 74) - биологический набор данных [157].

Репозиторий UCI (г.Ирвин, Калифорния, США) является крупнейшим репозиторием модельных и реальных задач машинного обучения, основанных на реальных данных по прикладным задачам в области физики, техники, биологии, медицины, социологии и др. Наборами данных (data set) именно этого репозитория чаще всего пользуются исследователи для эмпирического анализа

алгоритмов машинного обучения [158]. Репозиторий Clustering basic benchmark – Школа машинного обучения Университета Восточной Финляндии (г.Йоэнсуу).

Для экспериментов использовалась вычислительная система Depo X8Sti (6-ядерное ЦПУ Xeon X5650 2.67 ГГц,12 Гб ОЗУ), технология hyperthreading отключена. Также эксперименты проводились на системе малой мощности с 2-ядерным ЦПУ Atom N270 1.6 ГГц, 1 Гб ОЗУ (время выполнения увеличивается в 16-25 раз, то есть, чтобы достичь тех же результатов, нужно в 16-25 раз большее фиксированное время).

Для всех наборов данных было выполнено по 30 попыток запуска каждого из алгоритмов. Фиксировались только лучшие результаты, достигнутые в каждой попытке, затем из этих результатов по каждому алгоритму были рассчитаны значения целевой функции: минимальное и максимальное значения (Min, Max), среднее значение (Среднее) и среднеквадратичное отклонение (СКО). Алгоритмы j-means и k-средних были запущены в режиме мультистарта.

Кроме этого на некоторых наборах данных проведены расчеты с различными вариантами предполагаемого числа кластеров, а также при варьирующемся лимите времени, отведенным на одну попытку запуска алгоритма (Таблицы 2.1, 2.3, 2.6).

Результаты наших вычислительных экспериментов представлены в Таблицах 2.1-2.6. Лучшие значения целевой функции (минимальное значение, среднее значение и среднеквадратичное отклонение) выделены полужирным курсивом.

Таблица 2.1 - Результаты вычислительных экспериментов по набору данных ionosphere (30 секунд, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Среднеквадратичное отклонение
10 кластеров				
j-means	1 590,34	1 598,83	1 594,77	2,41
k-средних	1 590,93	1 598,61	1 595,28	2,47
k-GH-VNS1	***1 586,38***	1 586,65	***1 586,52***	***0,12***
k-GH-VNS2	***1 586,38***	1 591,99	1 588,00	2,36
k-GH-VNS3	***1 586,38***	1 591,99	1 587,24	1,57
j-means-GH-VNS1	***1 586,38***	1 586,63	***1 586,44***	***0,10***
j-means-GH-VNS2	***1 586,39***	1 586,63	***1 586,48***	***0,12***
20 кластеров				
j-means	1 282,18	1 299,13	1 291,92	4,83
k-средних	1 286,30	1 310,54	1 301,98	5,81
k-GH-VNS1	1 239,16	1 259,56	***1 246,39***	5,18
k-GH-VNS2	1 243,94	1 263,11	1 252,26	4,96
k-GH-VNS3	***1 238,53***	1 265,28	1 252,53	6,47
k-GH-VNS1-RND	***1 237,58***	1 252,42	***1 245,53***	4,31
k-GH-VNS2-RND	1 245,93	1 264,07	1 254,72	5,15
k-GH-VNS3-RND	1 243,73	1 259,78	1 251,01	***3,96***
j-means-GH-VNS1	***1 236,21***	1 257,85	***1 245,97***	6,48
j-means-GH-VNS2	1 245,71	1 256,18	1 249,95	***3,09***

Таблица 2.2 - Результаты вычислительных экспериментов по набору данных mopsi-Joensuu (20 кластеров, 40 минут, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Среднеквадратичное отклонение
j-means	36,565	37,520	36,730	0,253
k-средних	47,891	52,759	50,387	1,359
k-GH-VNS1	***36,565***	36,565	***36,565***	***0,000***
k-GH-VNS2	***36,565***	36,565	***36,565***	***0,000***
k-GH-VNS3	***36,565***	36,565	***36,565***	***0,000***
k-GH-VNS1-RND	***36,565***	36,565	***36,565***	***0,000***
k-GH-VNS2-RND	***36,565***	36,565	***36,565***	***0,000***
k-GH-VNS3-RND	***36,565***	36,565	***36,565***	***0,000***
j-means-GH-VNS1	***36,565***	36,565	***36,565***	***0,000***
j-means-GH-VNS2	***36,565***	36,565	***36,565***	***0,000***

Таблица 2.3 - Результаты вычислительных экспериментов по набору данных chess (30 кластеров, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Среднеквадратичное отклонение
5 минут				
j-means	8 021,22	8 102,13	8 060,24	21,05
k-средних	7 989,20	8 038,81	8 019,24	13,68
k-GH-VNS1	***7 960,82***	7 978,85	7 967,27	4,99
k-GH-VNS2	***7 959,92***	7 989,01	7 974,27	8,61
k-GH-VNS3	7 998,48	8 007,96	8 003,84	***3,51***
k-GH-VNS1-RND	7 960,66	7 978,62	7 968,86	5,94
k-GH-VNS2-RND	7 961,89	7 996,10	7 972,31	8,76
k-GH-VNS3-RND	7 987,20	8 001,26	7 991,42	***3,55***
j-means-GH-VNS1	***7 958,25***	7 967,75	***7 961,82***	4,65
j-means-GH-VNS2	***7 959,03***	7 970,65	***7 963,63***	4,13
2 часа				
j-means	7 997,43	8 031,05	8 014,72	10,71
k-средних	7 970,88	8 005,28	7 990,12	9,31
k-GH-VNS1	***7 958,26***	7 969,10	7 962,73	3,89
k-GH-VNS2	***7 958,25***	7 961,61	***7 959,34***	1,21
k-GH-VNS3	***7 958,26***	7 963,07	7 960,22	2,03
k-GH-VNS1-RND	***7 958,24***	7 965,03	7 960,91	1,59
k-GH-VNS2-RND	***7 958,24***	7 963,09	***7 959,57***	1,56
k-GH-VNS3-RND	***7 958,24***	7 968,36	***7 959,49***	2,64
j-means-GH-VNS1	***7 958,25***	7 958,28	***7 958,26***	***0,02***
j-means-GH-VNS2	***7 958,25***	7 960,39	***7 958,68***	***0,85***

Таблица 2.4 - Результаты вычислительных экспериментов по набору данных europe (30 кластеров, 4 часа, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Средне-квадратичное отклонение
j-means	7,51477E+12	7,60536E+12	7,56092E+12	29,764E+9
k-средних	7,54811E+12	7,57894E+12	7,56331E+12	13,560E+9
k-GH-VNS1	***7,4918E+12***	7,49201E+12	***7,49185E+12***	***0,073E+9***
k-GH-VNS2	7,49488E+12	7,52282E+12	7,50082E+12	9,989E+9
k-GH-VNS3	***7,4918E+12***	7,51326E+12	7,49976E+12	9,459E+9
k-GH-VNS1-RND	***7,49181E+12***	7,49358E+12	7,49224E+12	0,688E+9
k-GH-VNS2-RND	***7,4918E+12***	7,51914E+12	7,49719E+12	9,776E+9
k-GH-VNS3-RND	7,49182E+12	7,51505E+12	7,4971E+12	8,159E+9
j-means-GH-VNS1	***7,4918E+12***	7,49211E+12	***7,49185E+12***	***0,112E+9***
j-means-GH-VNS2	7,49187E+12	7,51455E+12	7,4962E+12	8,213E+9

Таблица 2.5 - Результаты вычислительных экспериментов по набору данных birch3 (100 кластеров, 6 часов, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Средне-квадратичное отклонение
j-means	3,76222E+13	3,7965E+13	3,77715E+13	0,116211E+12
k-средних	7,92474E+13	8,87404E+13	8,31599E+13	3,088140E+12
k-GH-VNS1	***3,72537E+13***	3,77474E+13	3,74703E+13	0,171124E+12
k-GH-VNS2	4,21378E+13	5,01871E+13	4,52349E+13	4,333462E+12
k-GH-VNS3	***3,72525E+13***	3,74572E+13	***3,73745E+13***	0,074315E+12
k-GH-VNS1-RND	***3,72541E+13***	3,77687E+13	3,74943E+13	0,185483E+12
k-GH-VNS2-RND	3,83257E+13	4,61847E+13	4,0815E+13	2,543163E+12
k-GH-VNS3-RND	3,73131E+13	3,75242E+13	3,74164E+13	***0,061831E+12***
j-means-GH-VNS1	***3,71595E+13***	3,71807E+13	***3,71735E+13***	***0,012162E+12***
j-means-GH-VNS2	***3,72422E+13***	3,7456E+13	***3,7347E+13***	0,106977E+12

Таблица 2.6 - Результаты вычислительных экспериментов по набору данных KDDCUP04BioNormed (30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Среднеквадра-тичное отклонение
30 кластеров, 500 минут				
j-means	6 280 406	6 288 774	6 283 271	4 767,4
k-средних	6 310 843	6 429 357	6 370 635	63 853,5
k-GH-VNS1	6 385 012	6 385 150	6 385 047	***51,3***
k-GH-VNS2	6 385 196	6 430 515	6 418 326	17 204,4
k-GH-VNS3	***6 267 808***	6 286 808	6 283 641	7 756,6
k-GH-VNS1-RND	6 385 016	6 385 033	6 385 023	***6,2***
k-GH-VNS2-RND	6 385 149	6 429 426	6 410 598	16 538,2
k-GH-VNS3-RND	6 386 703	6 386 808	6 386 753	***50,4***
j-means-GH-VNS1	***6 267 205***	6 267 395	***6 267 300***	134,3
j-means-GH-VNS2	***6 267 217***	6 267 421	***6 267 319***	144,2
200 кластеров, 24 часа				
j-means	5 330 344	5 382 908	5 355 903	26 785,8
k-средних	5 336 446	5 381 386	5 366 144	25 722,4
k-GH-VNS1	***5 294 620***	5 307 828	***5 301 224***	***9 339,5***
k-GH-VNS2	5 440 814	5 476 140	5 458 477	24 979,5
k-GH-VNS3	нет резуль	тата		
k-GH-VNS1-RND	***5 310 067***	5 340 849	5 325 458	21 765,7
k-GH-VNS2-RND	5 368 527	5 399 695	5 384 111	22 039,6
k-GH-VNS3-RND	нет резуль	тата		
j-means-GH-VNS1	5 430 120	5 446 222	5 438 171	11 385,8
j-means-GH-VNS2	5 500 410	5 508 985	5 504 697	***6 063,4***

Для графического сравнения новых и известных алгоритмов по каждому набору данных на Рисунках 2.3-2.9 приведены графики сходимости алгоритмов, построенные по среднему значению целевой функции (Таблицы 2.1-2.6). По оси абсцисс – время, по оси ординат – усредненное по 30 запускам значение целевой функции.

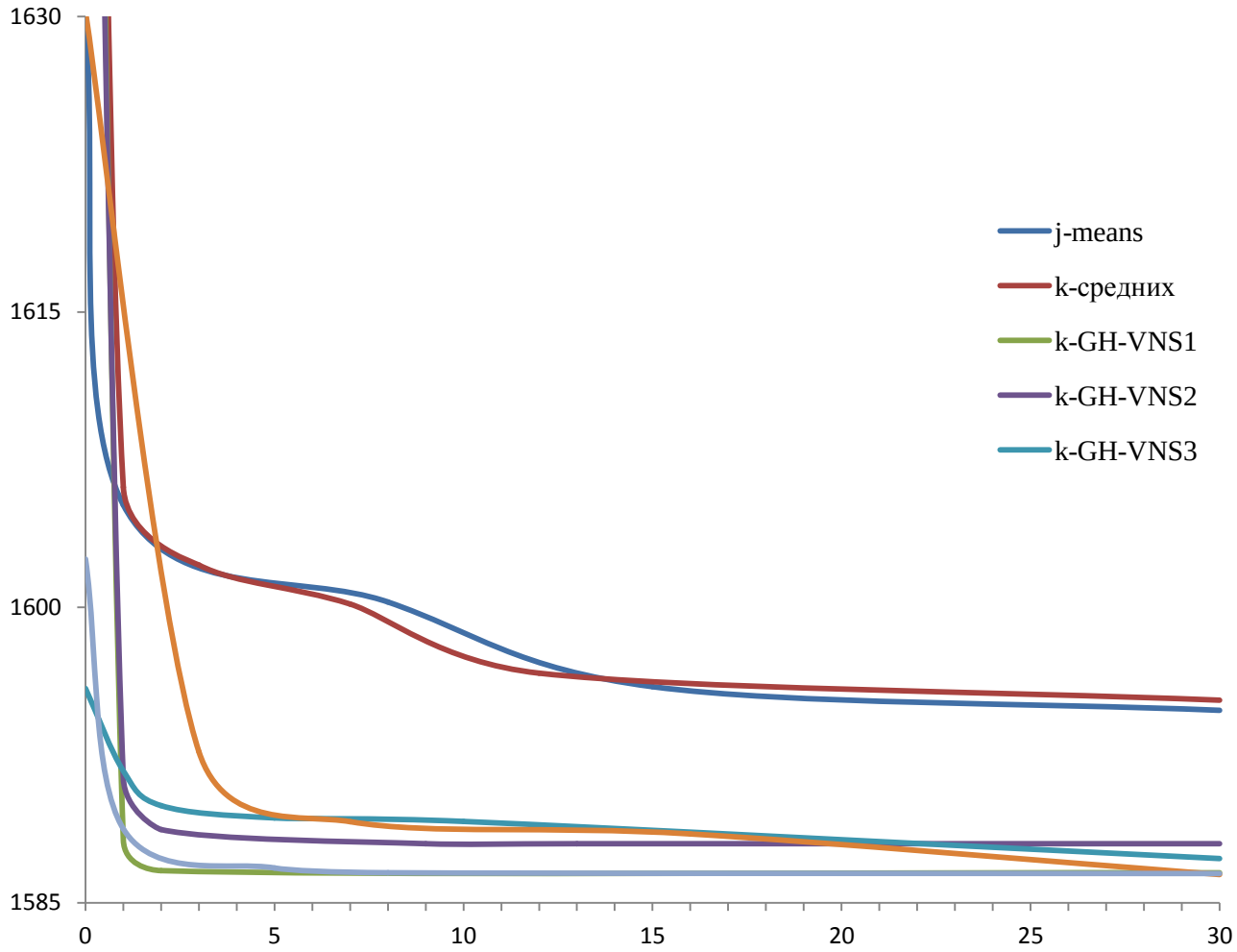


Рисунок 2.3 - Сравнение новых и известных алгоритмов для набора данных ionosphere (10 кластеров, 30 секунд)
по оси абсцисс – время в секундах, по оси ординат – достигнутое среднее значение целевой функции

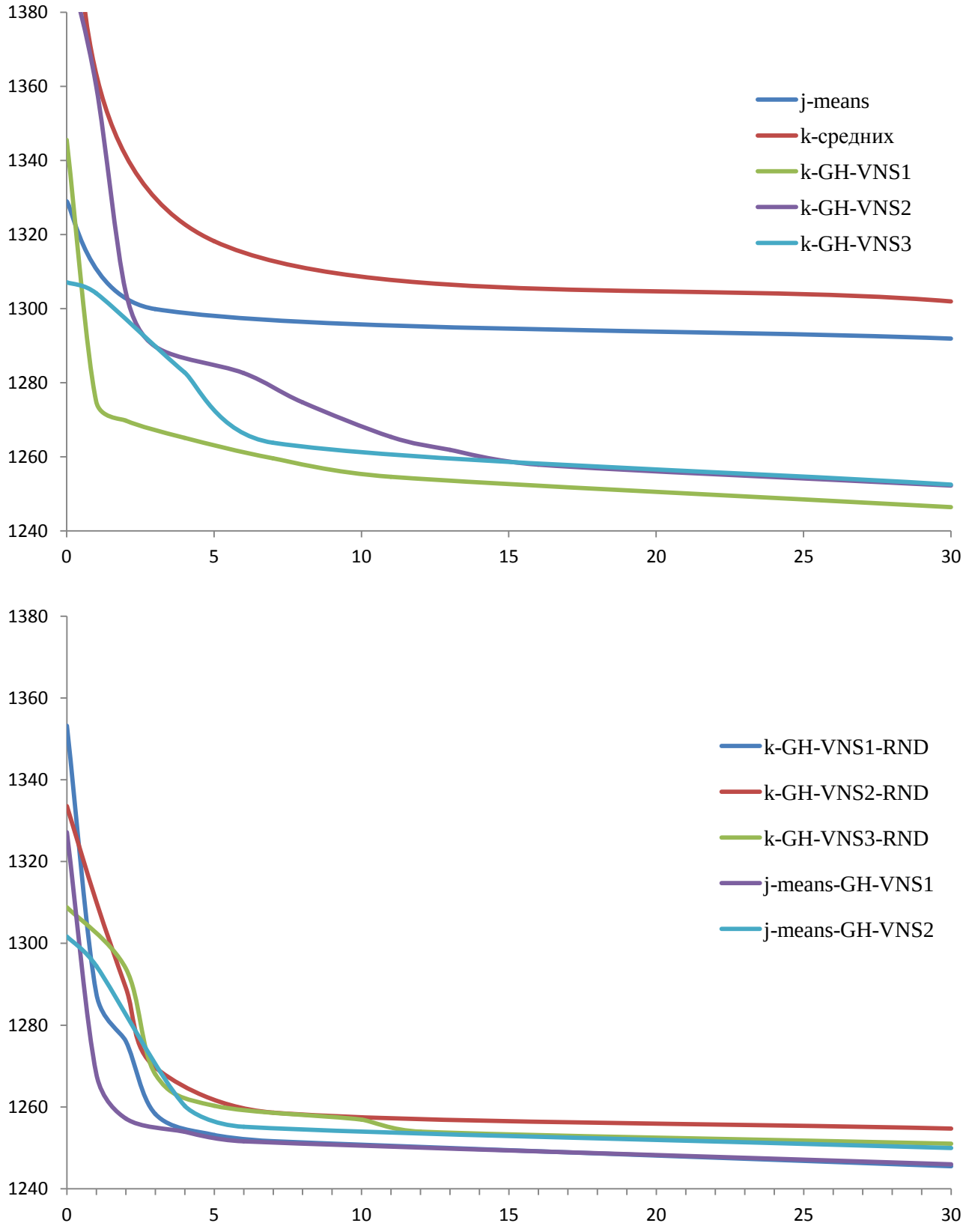
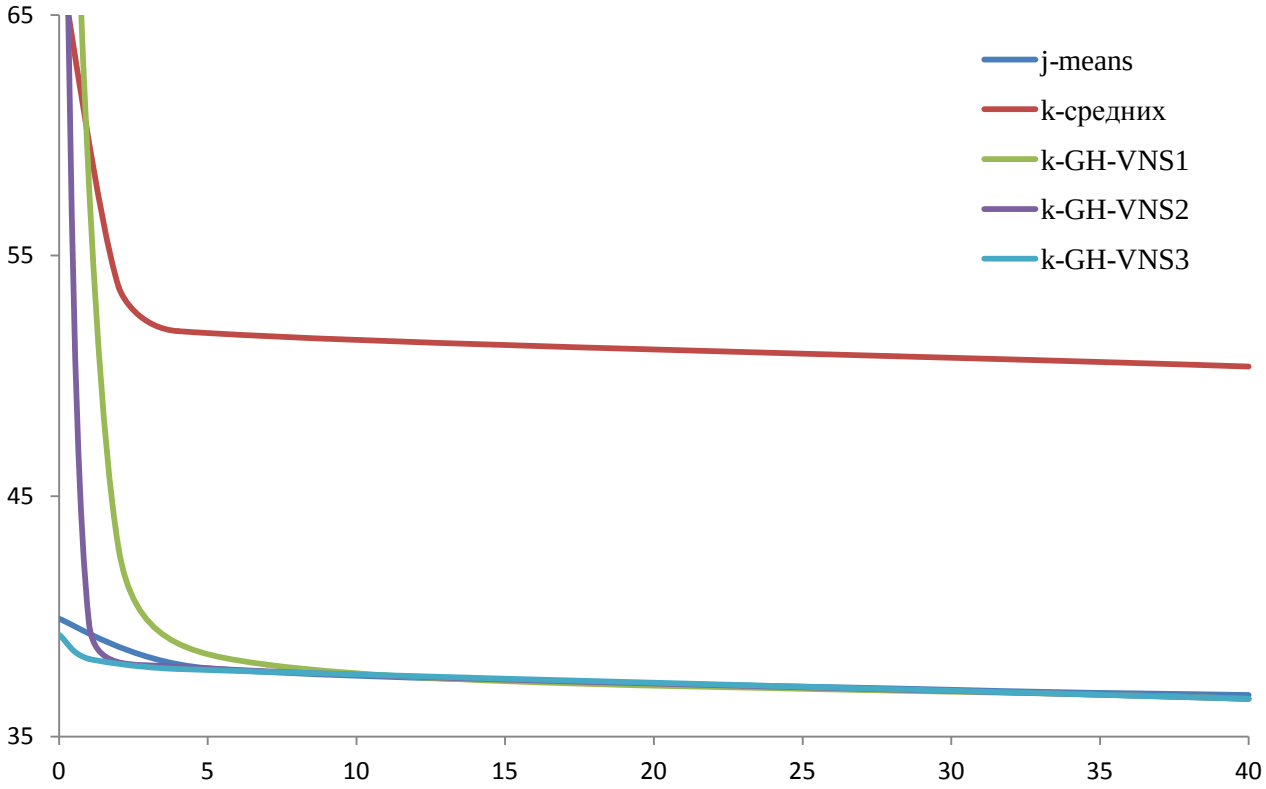


Рисунок 2.4 - Сравнение новых и известных алгоритмов для набора данных ionosphere (20 кластеров, 30 секунд)
по оси абсцисс – время в секундах, по оси ординат – достигнутое среднее значение целевой функции



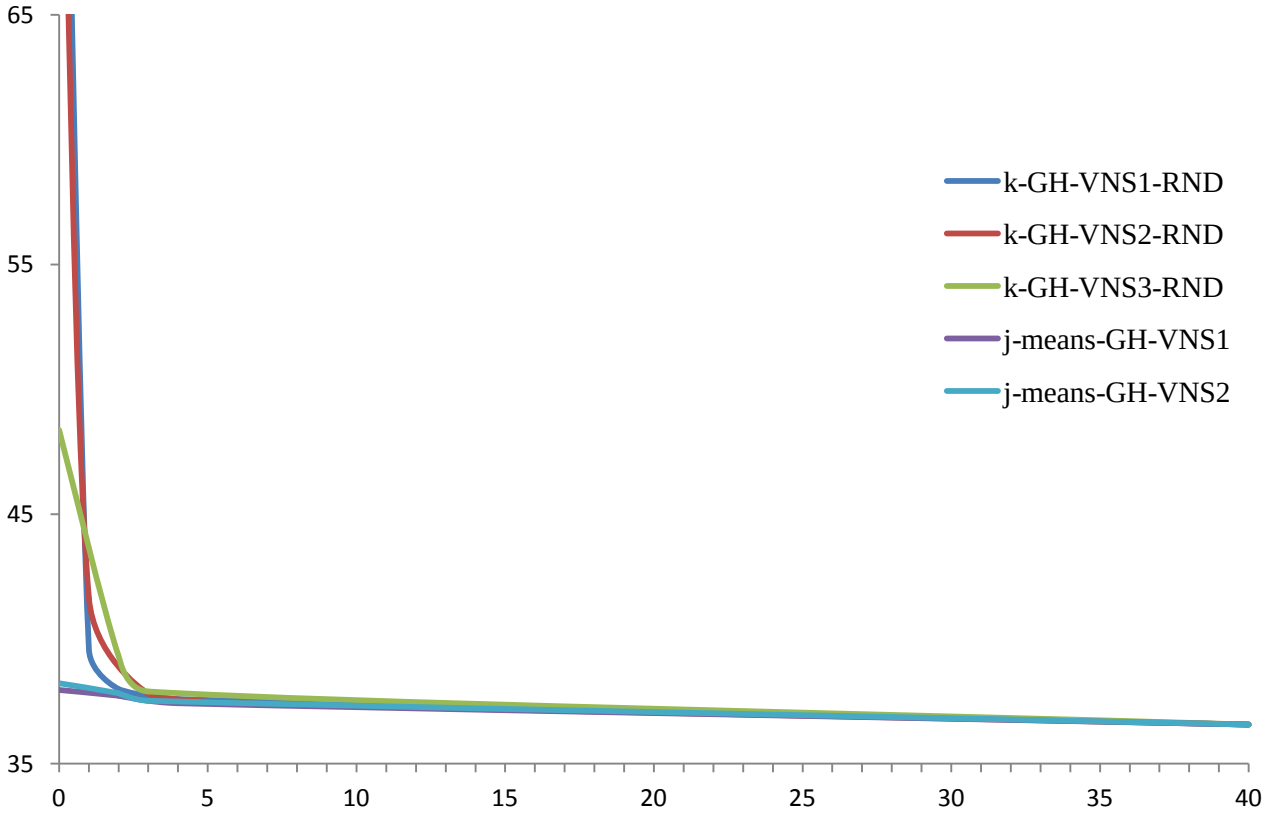


Рисунок 2.5 - Сравнение новых и известных алгоритмов для набора данных mopsi-Joensuu (20 кластеров, 40 минут)
по оси абсцисс – время в минутах, по оси ординат – достигнутое среднее значение целевой функции

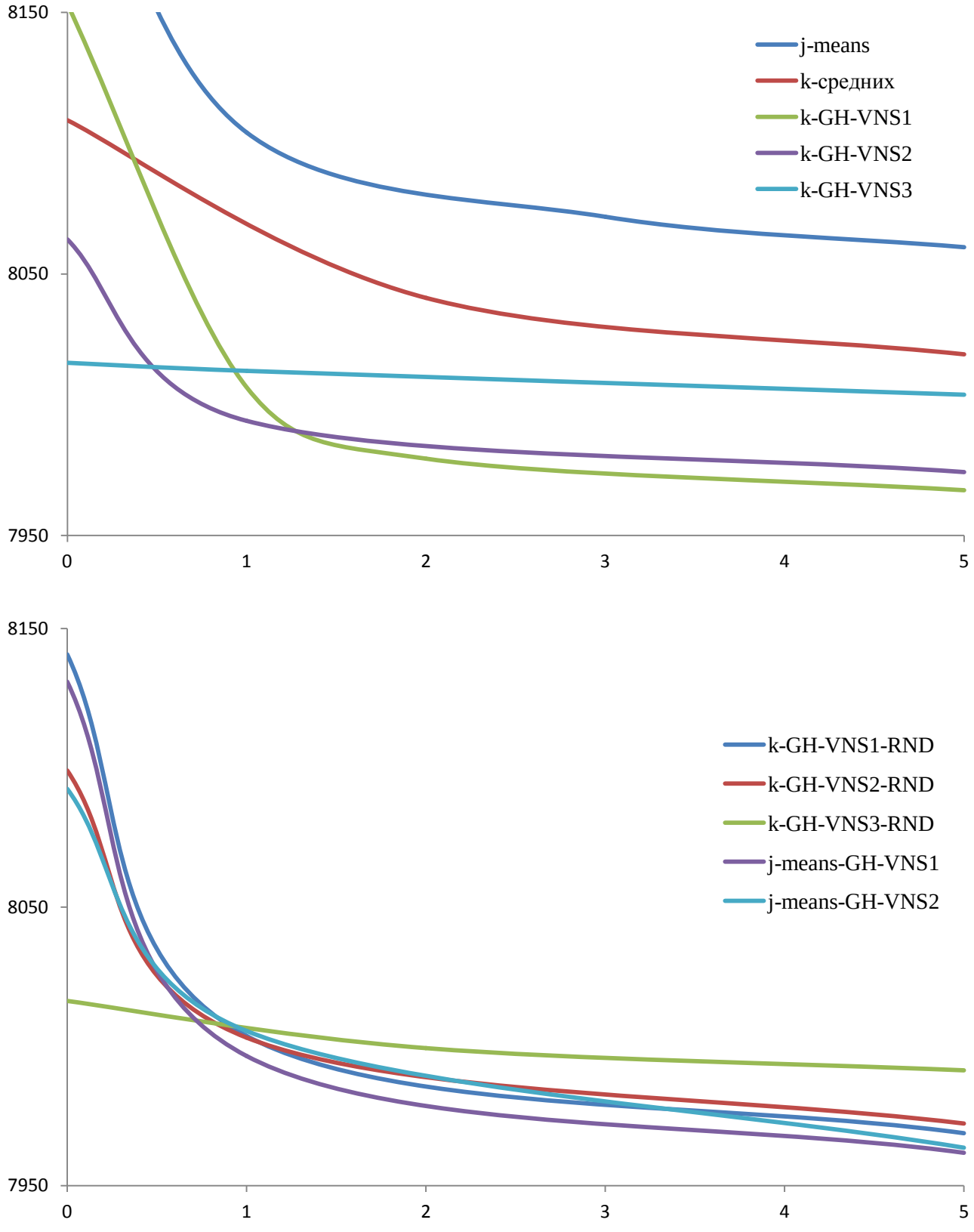


Рисунок 2.6 - Сравнение новых и известных алгоритмов для набора данных chess (30 кластеров, 5 минут)
по оси абсцисс – время в минутах, по оси ординат – достигнутое среднее значение целевой функции

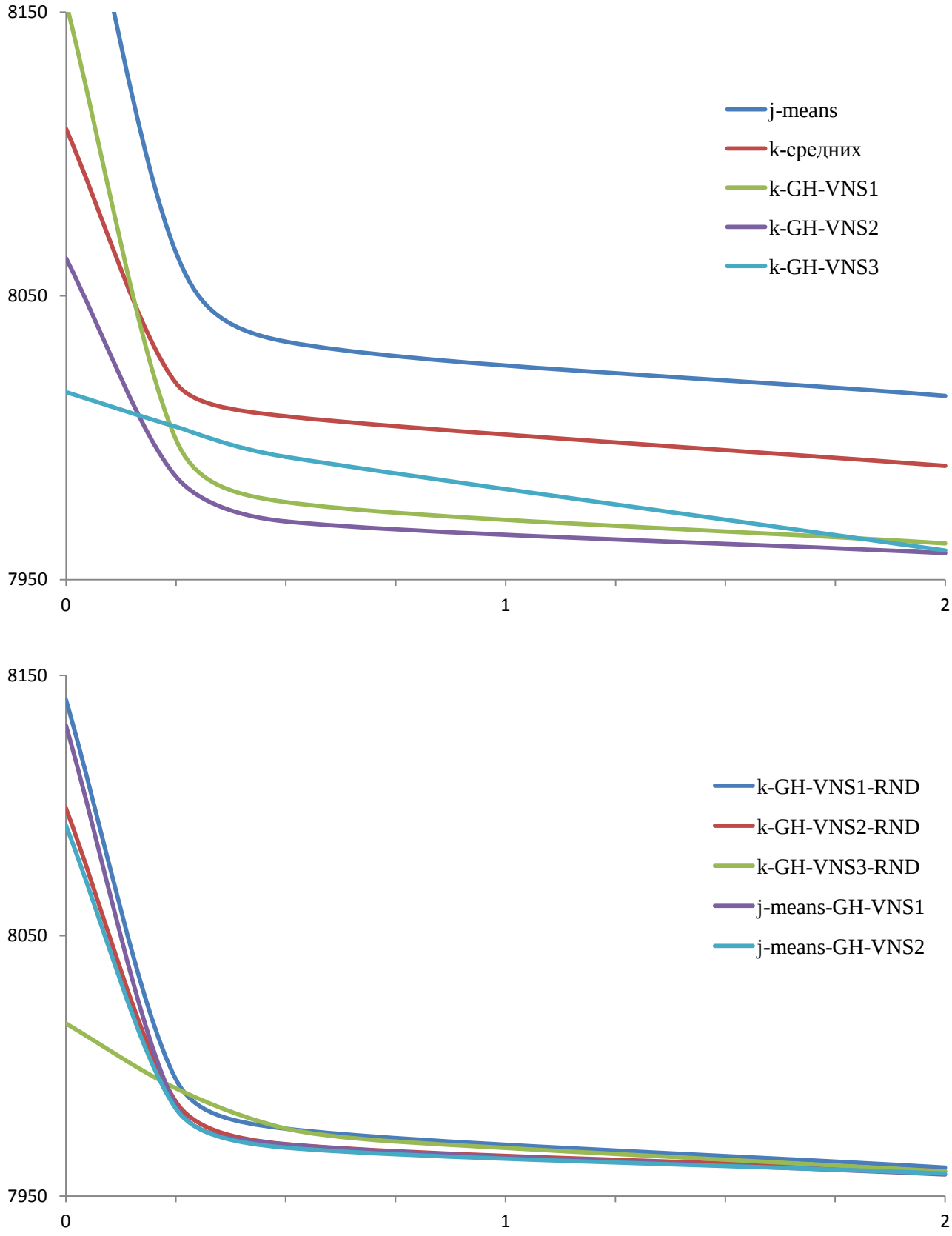


Рисунок 2.7 - Сравнение новых и известных алгоритмов для набора данных chess (30 кластеров, 2 часа)
по оси абсцисс – время в часах, по оси ординат – достигнутое среднее значение целевой функции

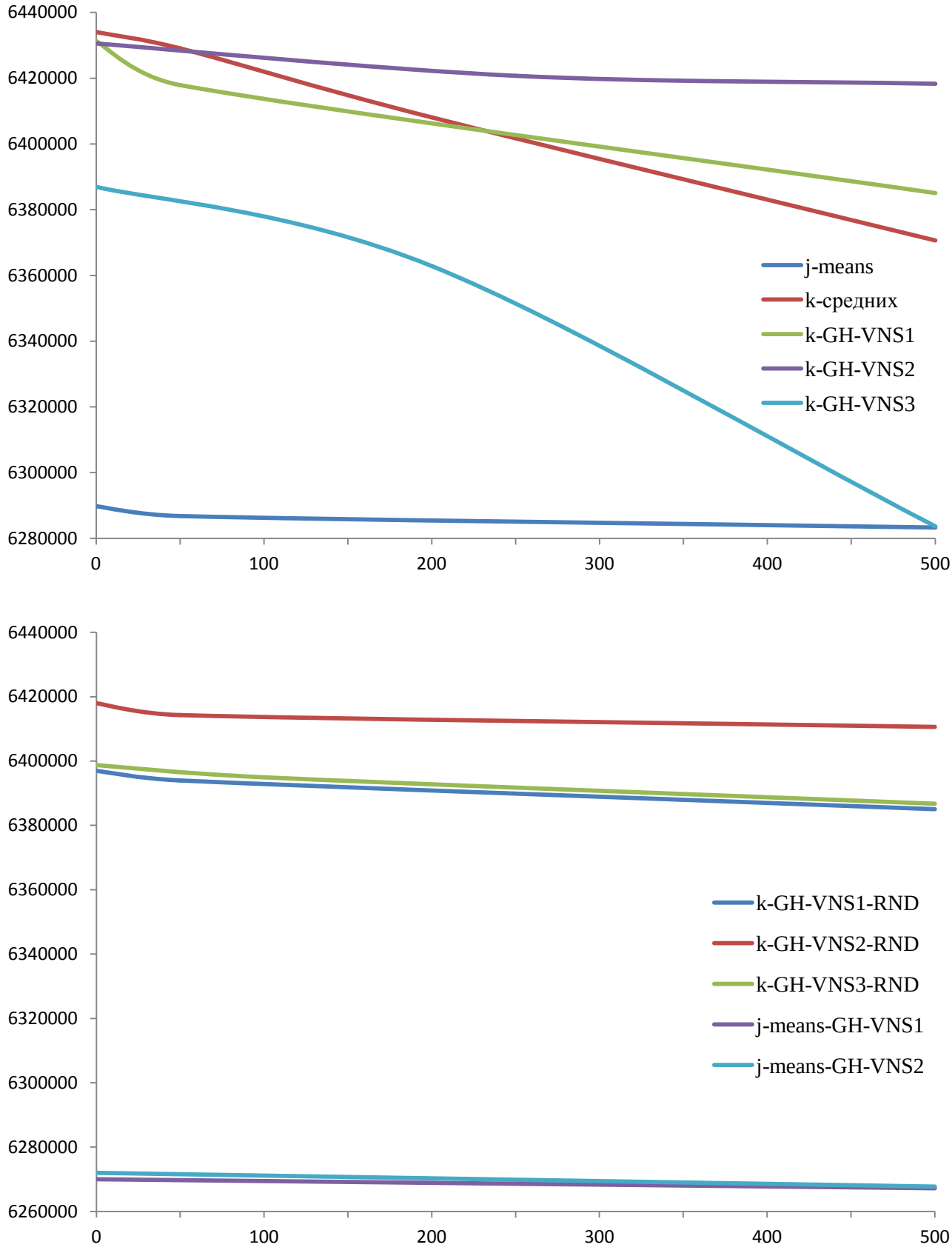


Рисунок 2.8 - Сравнение новых и известных алгоритмов для набора данных KDDCUP04BioNormed (30 кластеров, 500 минут)
по оси абсцисс – время в минутах, по оси ординат – достигнутое среднее значение целевой функции

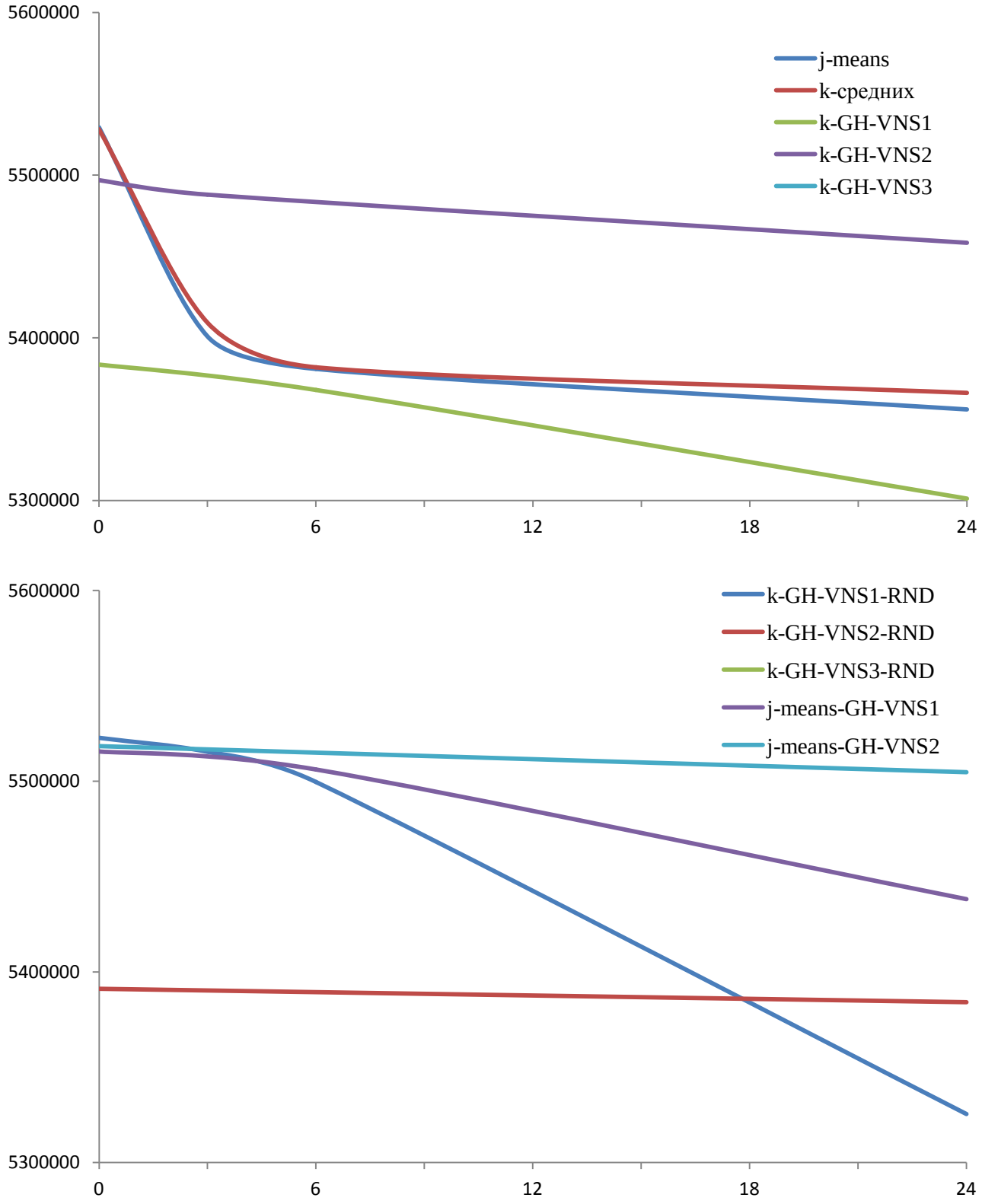


Рисунок 2.9 - Сравнение новых и известных алгоритмов для набора данных KDDCUP04BioNormed (200 кластеров, 24 часа)
по оси абсцисс – время в часах, по оси ординат – достигнутое среднее значение целевой функции

Результаты вычислительных экспериментов (Таблицы 2.1-2.6) показали, что новые алгоритмы поиска с чередующимися рандомизированными окрестностями (k-GH-VNS) имеют более стабильные результаты (дают меньшее минимальное значение и/или среднеквадратичное отклонение целевой функции, меньший разброс достигнутых значений) и, следовательно, лучшие показатели в сравнении с классическими алгоритмами j-means и k-средних (Рисунки 2.3-2.9) [149, 159]. При этом сложно отдать однозначное предпочтение какой-либо одной из версий алгоритма k-GH-VNS или его комбинированной версии j-means-GH-VNS. Однако отметим, что комбинация k-GH-VNS2 (в которой первой используется Жадная процедура 2 - Алгоритм 2.3) в большинстве случаев показывает худшие результаты в сравнении с остальными новыми алгоритмами, что ставит под сомнение целесообразность ее использования.

Результаты вычислительных экспериментов, приведенные в Таблицах 2.1-2.6, графически обобщены на рисунке 2.10.

На Рисунке 2.10 показаны количества наилучших достигнутых значений целевой функции среди всех решенных задач (наборов данных из репозиториев UCI и Clustering basic benchmark) по всем алгоритмам кластеризации (Таблицы 2.1-2.6). Количество лучших значений целевой функции среди алгоритмов кластеризации подсчитано по всем вычислительным экспериментам: отдельно по минимальному значению (Min) и среднеквадратичному отклонению (СКО). Так же, чтобы показать, какие из алгоритмов кластеризации дают лучшие из полученных значений целевой функции одновременно как по лучшему (рекордному) значению целевой функции, так и по среднеквадратичному отклонению достигнутого значения целевой функции, на рисунке есть столбец (Min + СКО).

В качестве тестовых наборов данных также были использованы результаты неразрушающих тестовых испытаний сборных производственных партий электрорадиоизделий (ЭРИ), проведенных в специализированном тестовом центре АО «ИТЦ - НПО ПМ» (г. Железногорск) для комплектации бортовой аппаратуры космических аппаратов, состав которых заранее известен:

- 3OT122A - 2 производственные партии (767 векторов данных, каждый размерностью 13);

- 5514BC1T2-9A5 - 2 партии (91 векторов данных, каждый размерностью 173);

- 1526TL1 - 3 партии (1234 векторов данных, каждый размерностью 157).

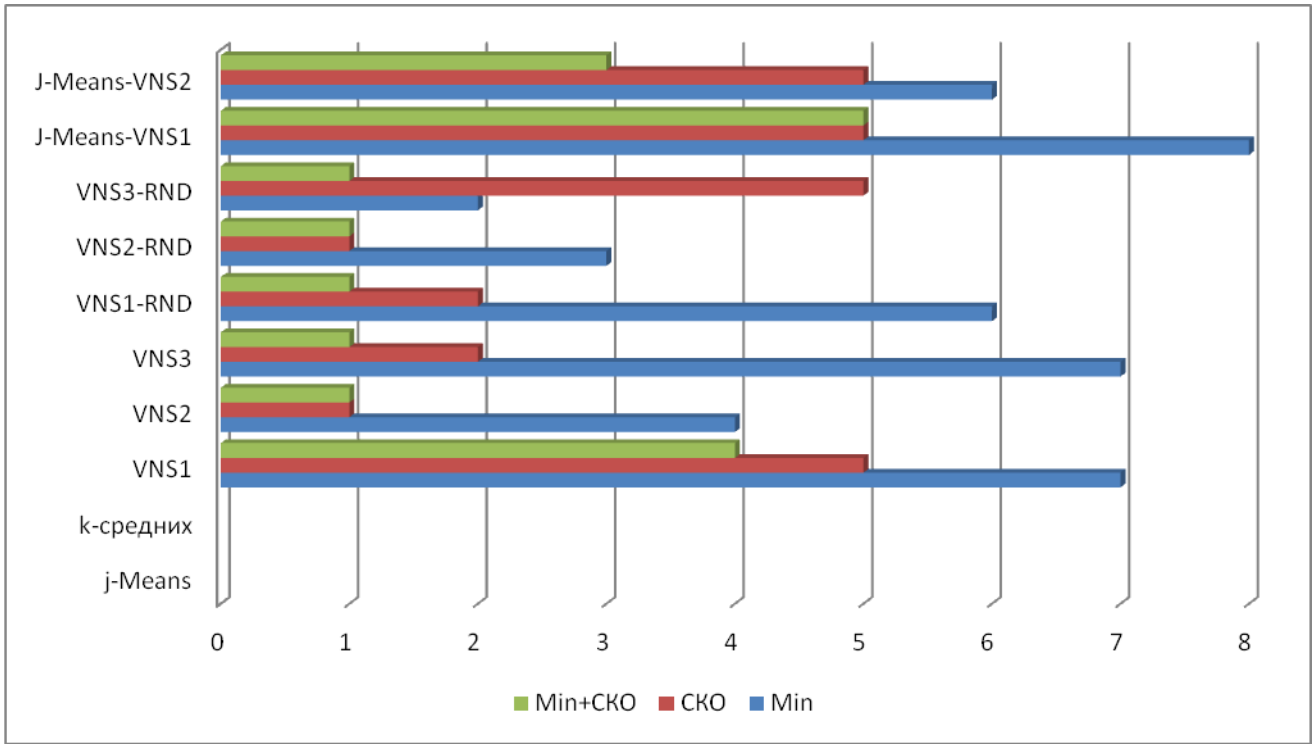


Рисунок 2.10 - Количества достигнутых лучших рекордных и лучших усредненных значений целевой функции каждым из алгоритмов, подсчитанные по всем вычислительным экспериментам со всеми наборами данных из репозиториев UCI и Clustering basic benchmark, а также количества одновременно достигнутых рекордов, как по значению целевой функции, так и по СКО

В качестве задачи ставилось разделение составленной сборной партии на кластеры, соответствующие однородным партиям, с последующим анализом качества этого разделения. Лучшие значения целевой функции (минимальное значение, среднее значение и среднеквадратичное отклонение) выделены полужирным курсивом (Таблицы 2.7-2.9). Графическая реализация сходимости алгоритмов построенных по среднему значению целевой функции представлена на Рисунках 2.11-2.13.

Таблица 2.7 - Результаты вычислительных экспериментов над производственными партиями электрорадиоизделий 3ОТ122А (2 минуты, 30 попыток)

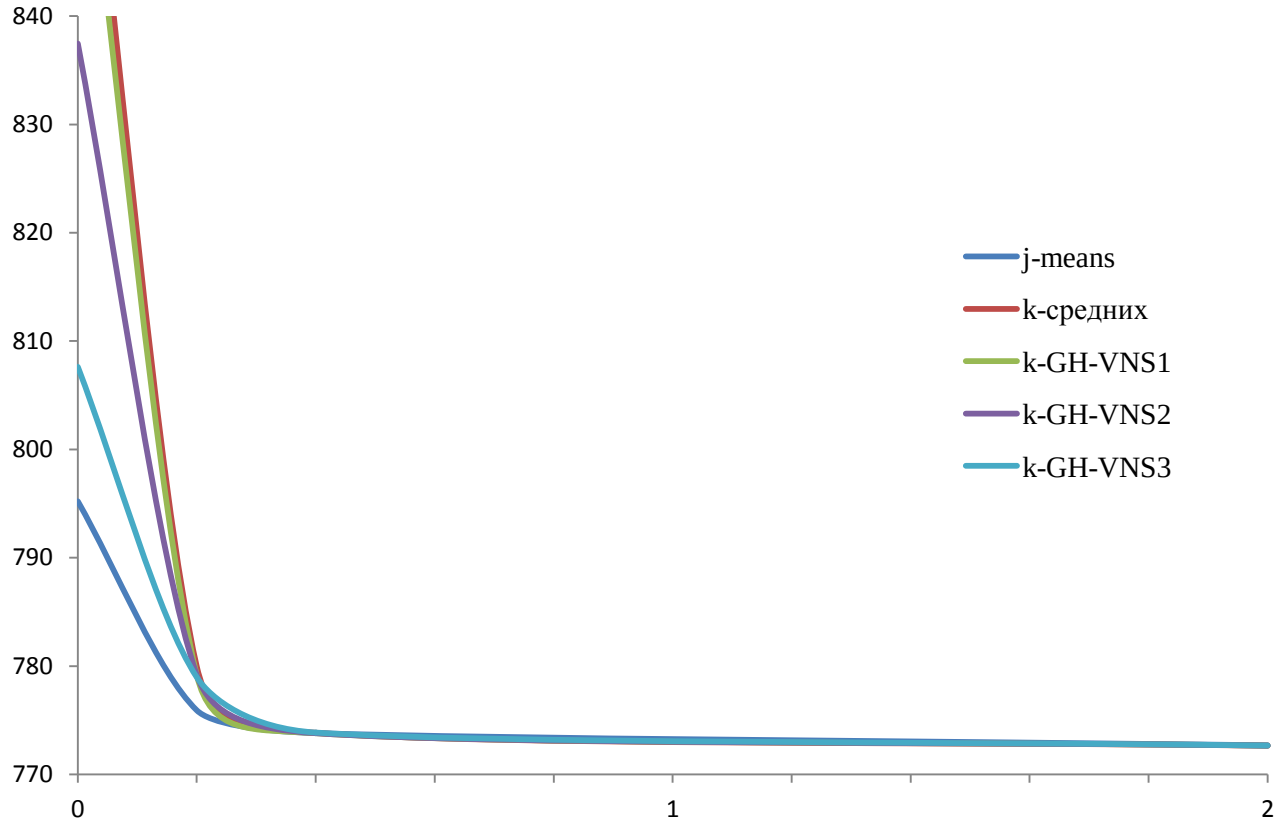
Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Средне-квадратичное отклонение
2 кластера				
j-means	***3 978,31***	3 978,31	***3 978,31***	***0,0000***
k-средних	***3 978,31***	3 978,31	***3 978,31***	***0,0000***
k-GH-VNS1	***3 978,31***	3 978,31	***3 978,31***	***0,0000***
k-GH-VNS2	***3 978,31***	3 978,31	***3 978,31***	***0,0000***
k-GH-VNS3	***3 978,31***	3 978,31	***3 978,31***	***0,0000***
k-GH-VNS1-RND	***3 978,31***	3 978,31	***3 978,31***	***0,0000***
k-GH-VNS2-RND	***3 978,31***	3 978,31	***3 978,31***	***0,0000***
k-GH-VNS3-RND	***3 978,31***	3 978,31	***3 978,31***	***0,0000***
j-means-GH-VNS1	***3 978,31***	3 978,31	***3 978,31***	***0,0000***
j-means-GH-VNS2	***3 978,31***	3 978,31	***3 978,31***	***0,0000***
5 кластеров				
j-means	***1 420,48***	1 420,49	***1 420,48***	0,0041
k-средних	***1 420,48***	1 420,48	***1 420,48***	***0,0000***
k-GH-VNS1	***1 420,49***	1 420,50	***1 420,49***	0,0061
k-GH-VNS2	***1 420,49***	1 420,50	***1 420,49***	0,0061
k-GH-VNS3	***1 420,49***	1 420,50	***1 420,49***	0,0057
k-GH-VNS1-RND	***1 420,49***	1 420,49	***1 420,49***	***0,0000***
k-GH-VNS2-RND	***1 420,49***	1 420,50	***1 420,49***	0,0057
k-GH-VNS3-RND	***1 420,49***	1 420,50	***1 420,49***	0,0050
j-means-GH-VNS1	***1 420,49***	1 420,50	***1 420,49***	0,0061
j-means-GH-VNS2	***1 420,49***	1 420,50	***1 420,49***	0,0057
10 кластеров				
j-means	***772,66***	772,70	772,68	0,0191
k-средних	***772,66***	772,66	***772,66***	***0,0000***
k-GH-VNS1	***772,66***	772,66	***772,66***	***0,0000***
k-GH-VNS2	***772,66***	772,66	***772,66***	***0,0000***
k-GH-VNS3	***772,66***	772,66	***772,66***	***0,0000***
k-GH-VNS1-RND	***772,66***	772,66	***772,66***	***0,0000***
k-GH-VNS2-RND	***772,66***	772,66	***772,66***	***0,0000***
k-GH-VNS3-RND	***772,66***	772,66	***772,66***	***0,0000***
j-means-GH-VNS1	***772,66***	772,66	***772,66***	***0,0000***
j-means-GH-VNS2	***772,66***	772,66	***772,66***	***0,0000***

Таблица 2.8 - Результаты вычислительных экспериментов над производственными партиями электрорадиоизделий 5514ВС1Т2-9А5 (2 минуты, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Средне-квадратичное отклонение
2 кластера				
j-means	***10 516,87***	10 516,87	***10 516,87***	***0,0000***
k-средних	***10 516,87***	10 516,87	***10 516,87***	***0,0000***
k-GH-VNS1	***10 516,87***	10 516,87	***10 516,87***	***0,0000***
k-GH-VNS2	***10 516,87***	10 516,87	***10 516,87***	***0,0000***
k-GH-VNS3	***10 516,87***	10 516,87	***10 516,87***	***0,0000***
k-GH-VNS1-RND	***10 516,87***	10 516,87	***10 516,87***	***0,0000***
k-GH-VNS2-RND	***10 516,87***	10 516,87	***10 516,87***	***0,0000***
k-GH-VNS3-RND	***10 516,87***	10 516,87	***10 516,87***	***0,0000***
j-means-GH-VNS1	***10 516,87***	10 516,87	***10 516,87***	***0,0000***
j-means-GH-VNS2	***10 516,87***	10 516,87	***10 516,87***	***0,0000***
5 кластеров				
j-means	***8 287,83***	8 287,83	***8 287,83***	***0,0000***
k-средних	***8 287,83***	8 287,83	***8 287,83***	***0,0000***
k-GH-VNS1	***8 287,83***	8 287,83	***8 287,83***	***0,0000***
k-GH-VNS2	***8 287,83***	8 287,83	***8 287,83***	***0,0000***
k-GH-VNS3	***8 287,83***	8 287,83	***8 287,83***	***0,0000***
k-GH-VNS1-RND	***8 287,83***	8 287,83	***8 287,83***	***0,0000***
k-GH-VNS2-RND	***8 287,83***	8 287,83	***8 287,83***	***0,0000***
k-GH-VNS3-RND	***8 287,83***	8 287,83	***8 287,83***	***0,0000***
j-means-GH-VNS1	***8 287,83***	8 287,83	***8 287,83***	***0,0000***
j-means-GH-VNS2	***8 287,83***	8 287,83	***8 287,83***	***0,0000***
10 кластеров				
j-means	7 060,45	7 085,67	7 073,55	8,5951
k-средних	7 046,33	7 070,83	7 060,11	8,8727
k-GH-VNS1	***7 001,12***	7 009,53	7 004,48	4,3453
k-GH-VNS2	***7 001,12***	7 010,59	***7 002,26***	***2,9880***
k-GH-VNS3	***7 001,12***	7 009,53	7 003,01	***3,1694***
j-means-GH-VNS1	***7 001,12***	7 001,12	***7 001,12***	***0,0000***
j-means-GH-VNS2	***7 001,12***	7 011,94	7 003,88	4,4990

Таблица 2.9 - Результаты вычислительных экспериментов над производственными партиями электрорадиоизделий 1526TL1 (2 минуты, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Средне-квадратичное отклонение
3 кластера				
j-means	***86 599,77***	86 599,77	***86 599,77***	***0,0000***
k-средних	***86 599,77***	86 599,77	***86 599,77***	***0,0000***
k-GH-VNS1	***86 599,77***	86 599,77	***86 599,77***	***0,0000***
k-GH-VNS2	***86 599,77***	86 599,77	***86 599,77***	***0,0000***
k-GH-VNS3	***86 599,77***	86 599,77	***86 599,77***	***0,0000***
k-GH-VNS1-RND	***86 599,77***	86 599,77	***86 599,77***	***0,0000***
k-GH-VNS2-RND	***86 599,77***	86 599,77	***86 599,77***	***0,0000***
k-GH-VNS3-RND	***86 599,77***	86 599,77	***86 599,77***	***0,0000***
j-means-GH-VNS1	***86 599,77***	86 599,77	***86 599,77***	***0,0000***
j-means-GH-VNS2	***86 599,77***	86 599,77	***86 599,77***	***0,0000***
5 кластеров				
j-means	***63 337,29***	63 337,56	***63 337,46***	0,1211
k-средних	***63 337,29***	63 337,29	***63 337,29***	***0,0000***
k-GH-VNS1	***63 337,47***	63 337,56	***63 337,55***	0,0280
k-GH-VNS2	***63 337,56***	63 337,56	***63 337,56***	***0,0000***
k-GH-VNS3	***63 337,56***	63 337,56	***63 337,56***	***0,0000***
k-GH-VNS1-RND	***63 337,56***	63 337,56	***63 337,56***	***0,0000***
k-GH-VNS2-RND	***63 337,56***	63 337,56	***63 337,56***	***0,0000***
k-GH-VNS3-RND	***63 337,56***	63 337,56	***63 337,56***	***0,0000***
j-means-GH-VNS1	***63 337,56***	63 337,56	***63 337,56***	***0,0000***
j-means-GH-VNS2	***63 337,56***	63 337,56	***63 337,56***	***0,0000***
10 кластеров				
j-means	***43 841,97***	43 843,51	***43 842,59***	0,4487
k-средних	***43 842,10***	43 844,66	43 843,38	0,8346
k-GH-VNS1	***43 841,97***	43 844,18	***43 842,34***	0,9000
k-GH-VNS2	***43 841,97***	43 844,18	43 843,46	1,0817
k-GH-VNS3	***43 841,97***	43 842,10	***43 841,99***	***0,0424***
j-means-GH-VNS1	***43 841,97***	43 841,97	***43 841,97***	***0,0000***
j-means-GH-VNS2	***43 841,97***	43 844,18	***43 842,19***	***0,6971***



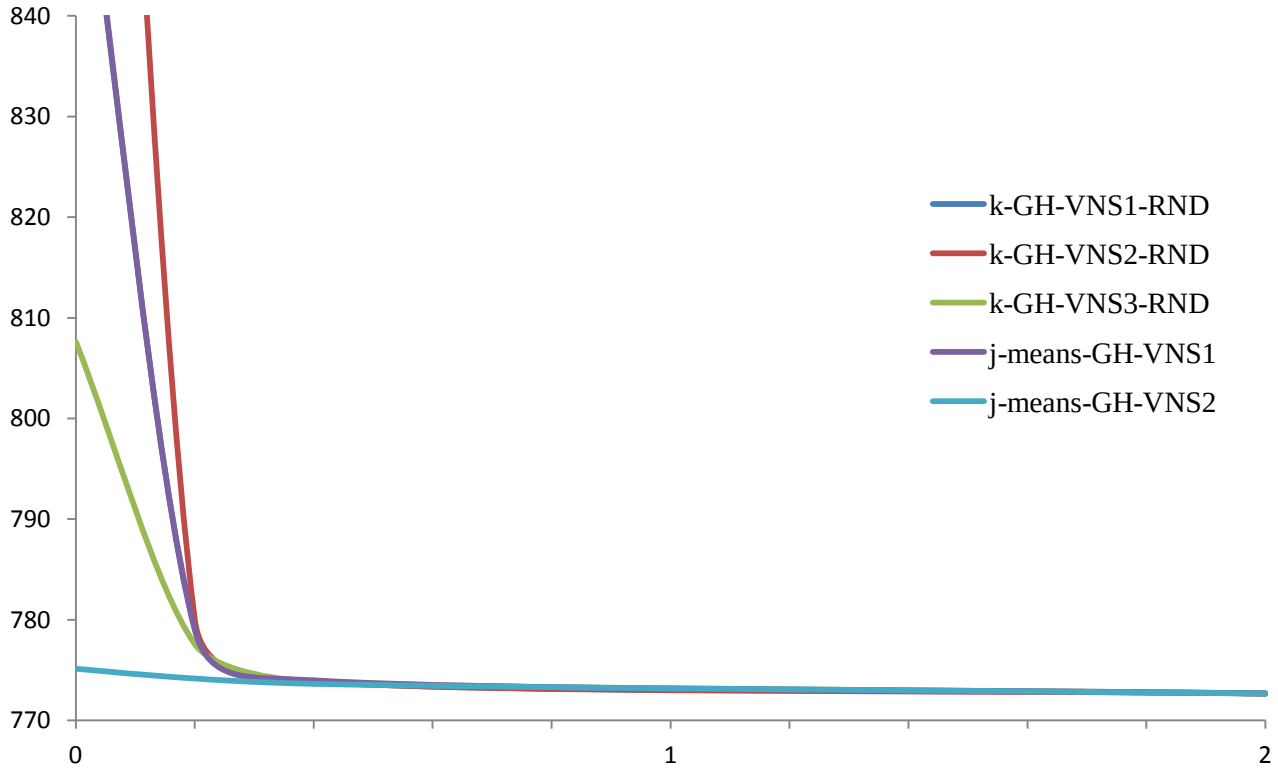


Рисунок 2.11 - Сравнение новых и известных алгоритмов для набора данных 3OT122A (10 кластеров, 2 минуты)
по оси абсцисс – время в минутах, по оси ординат – достигнутое среднее значение целевой функции

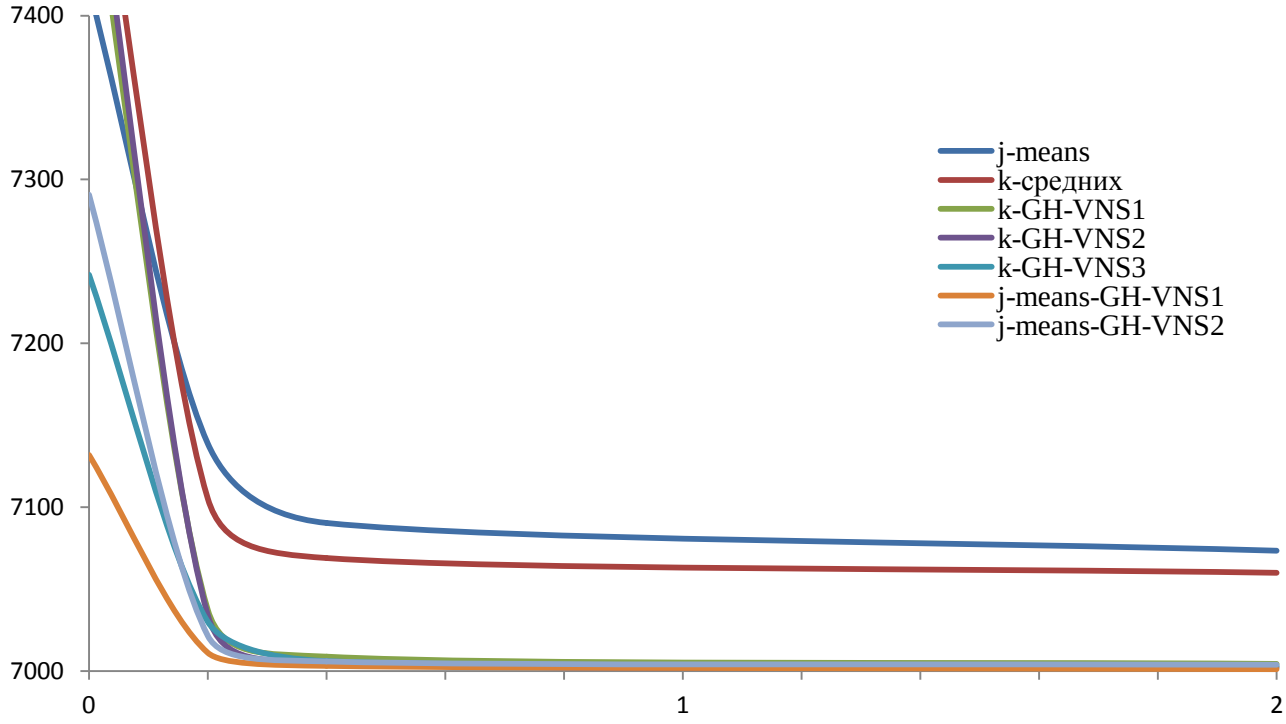


Рисунок 2.12 - Сравнение новых и известных алгоритмов для набора данных 5514BC1T2-9A5 (10 кластеров, 2 минуты)
по оси абсцисс – время в минутах, по оси ординат – достигнутое среднее значение целевой функции

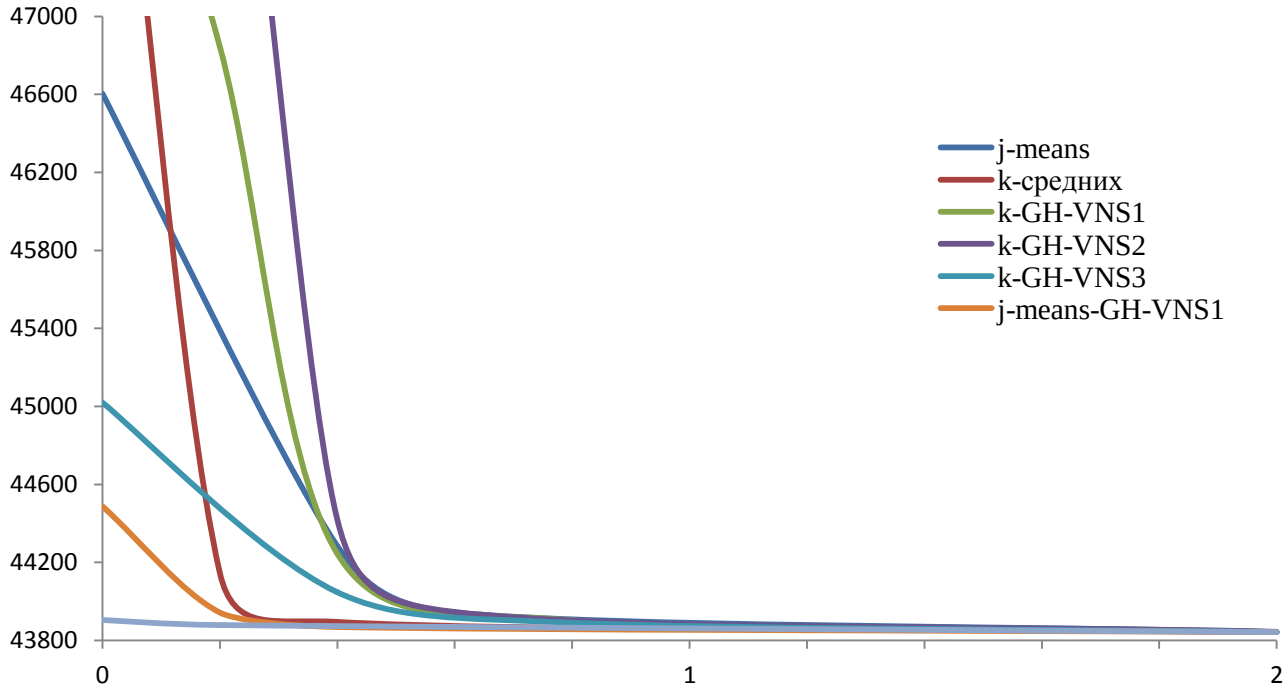


Рисунок 2.13 - Сравнение новых и известных алгоритмов для набора данных 1526TL1 (10 кластеров, 2 минуты)
по оси абсцисс – время в минутах, по оси ординат – достигнутое среднее значение целевой функции

Как видно по результатам вычислительных экспериментов по наборам данных электрорадиоизделий (Таблицы 2.7-2.9), новые комбинированные алгоритмы поиска с чередующимися рандомизированными окрестностями снова дали более стабильные результаты (Рисунок 2.14). Стоит отметить, что модификации Алгоритма 2.5 (k-GH-VNS-RND), в которых число центров в решении *S'* выбирается случайным образом из множества $\{\overline{2,|S|}\}$, показывают себя не самым лучшим образом, а нередко и вообще не дают результата. Поэтому эти алгоритмы не показаны в некоторых таблицах, в которых они не имели решения.

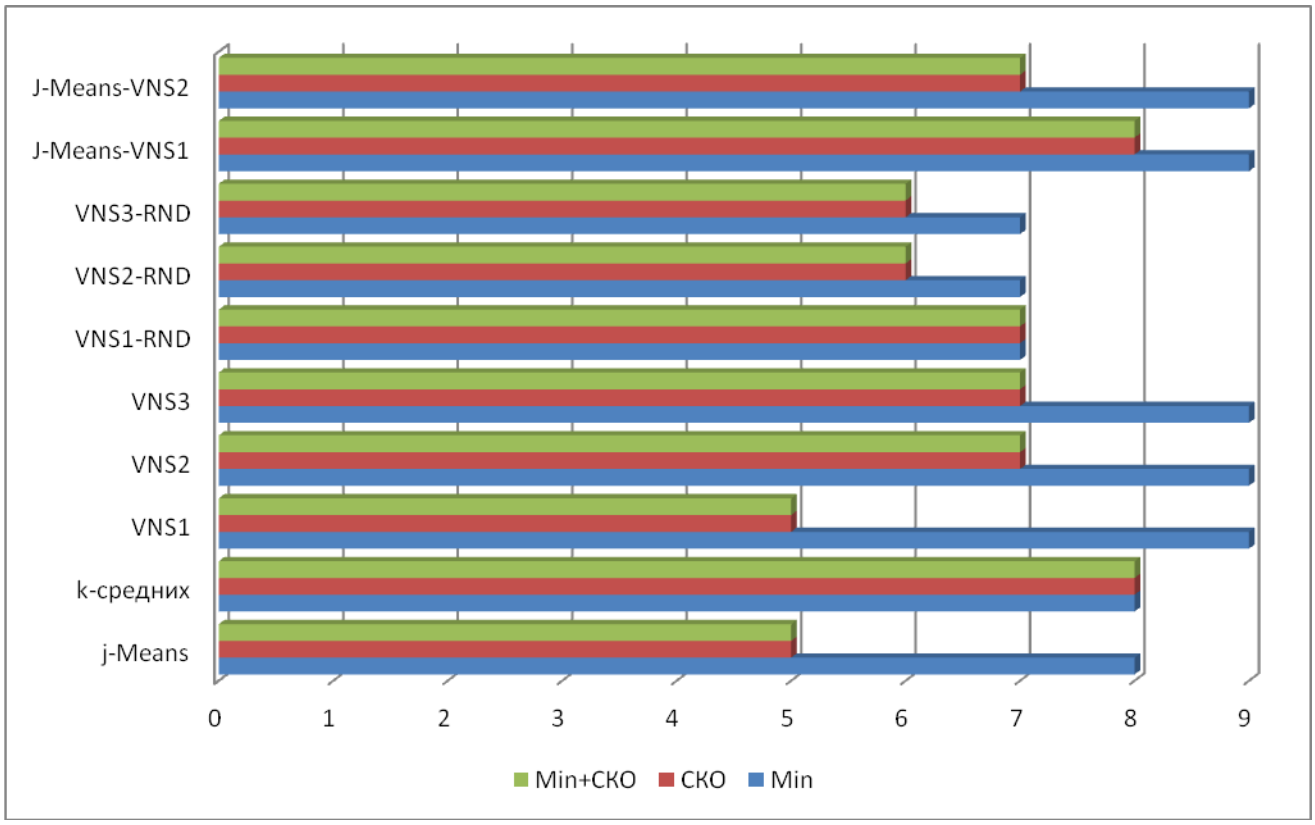


Рисунок 2.14 - Количества достигнутых лучших рекордных и лучших усредненных значений целевой функции каждым из алгоритмов, подсчитанные по всем вычислительным экспериментам со всеми наборами данных результатов испытаний электрорадиоизделий, а также количества одновременно достигнутых рекордов, как по значению целевой функции, так и по СКО

Для более полного сравнения полученных результатов вычислительных экспериментов новых алгоритмов с известными алгоритмами были использованы

результаты проведенных ранее вычислительных экспериментов над наборами данных электрорадиоизделий различными модификациями генетического алгоритма [142]. Сравнительные результаты приведены в Приложении А. Для расчетов были использованы наборы данных, представляющие собой результаты тестовых испытаний сборных партий электрорадиоизделий:

- 1526TL1 - 3 партии (1234 векторов данных, каждый размерностью 157);
- 2Д522Б - 5 партий (3711 векторов данных, каждый размерностью 10);
- Н5503ХМ1 - 5 партий (3711 векторов данных, каждый размерностью 229).

Как видно из Приложения А, значения целевой функции, полученные новыми алгоритмами в ряде случаев оказываются существенно лучше (по достигнутому значению целевой функции) и стабильнее результатов генетического алгоритма. При этом в некоторых задачах новые алгоритмы уступают генетическим алгоритмам, но не существенно. Тем не менее, можно говорить о конкурентоспособности новых алгоритмов как в сравнении с классическими алгоритмами k-средних и j-means, так и с генетическими алгоритмами, включая алгоритмы метода жадных эвристик, а также с детерминированными алгоритмами.

В следующем параграфе представлена реализация алгоритмов метода жадных эвристик с применением архитектуры CUDA и исследование их свойств при решении задач большой размерности.

2.4 Реализация жадных эвристических алгоритмов автоматической группировки для массивно-параллельных систем

CUDA (от английского Compute Unified Device Architecture) - платформа параллельных вычислений и модель программирования, специально разработанная фирмой NVIDIA для общих вычислений на графических процессорах (GPU), которая позволяет существенно увеличить вычислительную производительность [160]. Все больше новых решений и инициатив в различных областях исследований реализуются с CUDA, включая обработку видео и

изображений, вычислительную биологию и химию, моделирование динамики жидкостей, сейсмический анализ, восстановление изображений (полученных путем компьютерной томографии), трассировку лучей и многое другое.

Алгоритмы, использующие параллельную обработку данных (когда одна и та же последовательность математических операций применяется к большому объему данных), при расчетах на GPU показывают замечательные результаты, особенно если алгоритм в принципе хорошо распараллеливается и достаточно велико отношение числа арифметических инструкций к числу обращений к памяти. Кроме этого, большой объем данных и высокая плотность математических операций отменяют необходимость в больших кэшах, как на центральном процессоре (CPU).

С появлением единой модели шейдеров (программ для GPU) разделение процессоров вершинных и фрагментных шейдеров в аппаратном обеспечении исчезло. Теперь можно настроить шейдерные процессоры для выполнения обеих задач в зависимости от требований приложения. Кроме того, был представлен особый тип шейдера, геометрический шейдер, который позволяет генерировать геометрические элементы в аппаратных средствах на компьютере [161]. Начиная с семейства графических процессоров G80, NVIDIA поддерживает эту новую модель шейдеров, что приводит к отходу от предыдущих разработок графических процессоров. Графический процессор теперь состоит из так называемых многопроцессорных систем, в которых размещается ряд потоковых процессоров, которые идеально подходят для массовых параллельных вычислений.

Графический процессор рассматривается как набор мультипроцессоров, выполняющих параллельные потоки параллельно (Рисунок 2.15). Потоки сгруппированы в блоки данных и выполняют те же инструкции по разным данным параллельно. Один или несколько блоков напрямую связаны с аппаратным мультипроцессором, где распределение времени определяет порядок выполнения. Внутри одного блока потоки могут быть синхронизированы в любой точке выполнения. Определенное исполнение заказа на блокировку не гарантируется. Блоки далее группируются в сети, связь и синхронизация по

блокам невозможны, исполнение по заказу блоков с сетью не определено. Потоки и блоки могут быть организованы в трех и двух измерениях соответственно. Потоку присваивается идентификатор в зависимости от его положения в блоке, блоку также присваивается идентификатор в зависимости от его положения в сетке. Поток и идентификатор блока потока доступны во время выполнения, что позволяет задавать конкретные шаблоны доступа к памяти на основе выбранных макетов. Каждый поток в GPU выполняет ту же процедуру, известную как ядро [160, 162].

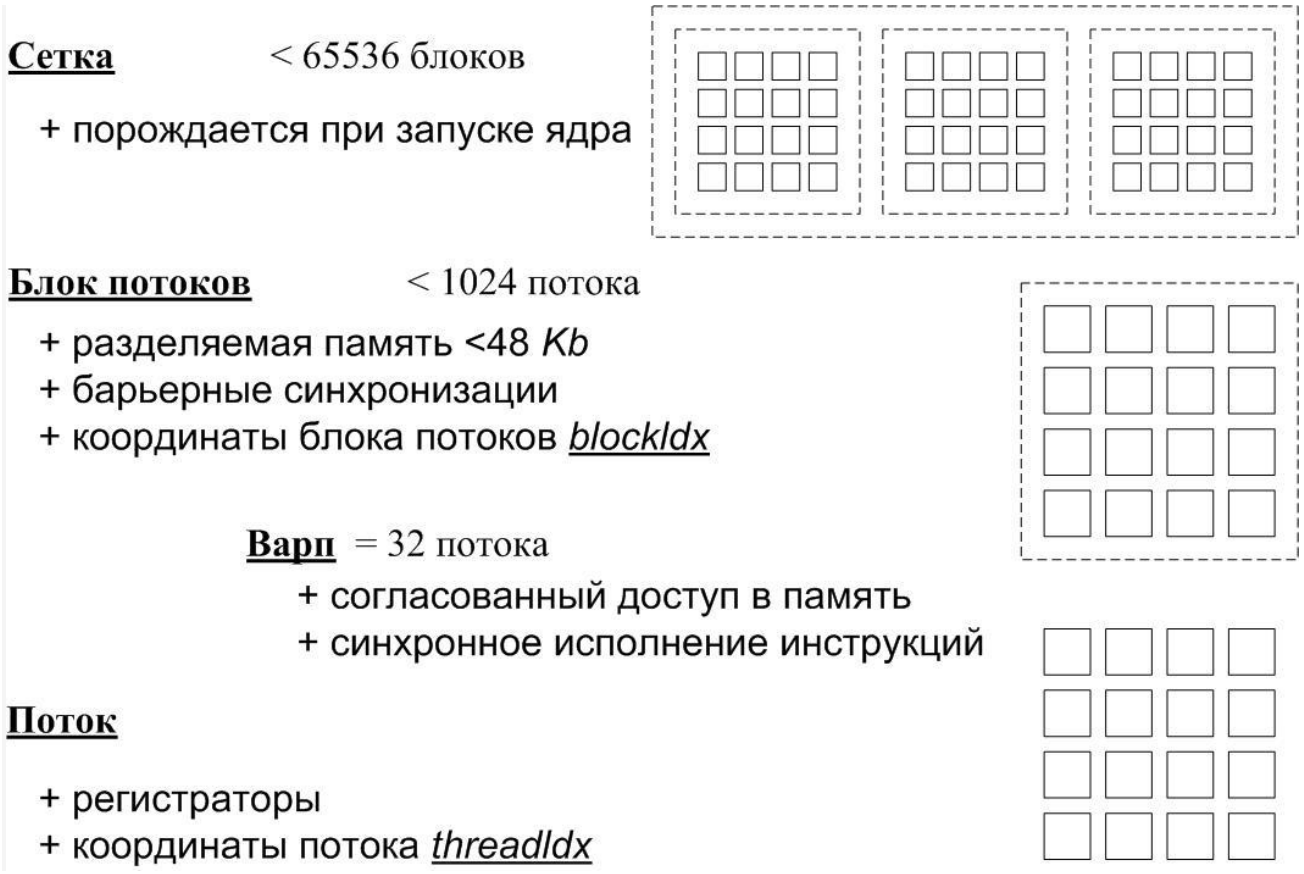


Рисунок 2.15 - Модель CUDA

Потоки имеют доступ к различным видам памяти. Каждая нить (тред) очень быстро записывается в локальный регистр, и ему назначается локальная память. Внутри одного блока все потоки имеют доступ к блоку локальной разделяемой памяти, доступ к которому возможен так же быстро, как и к регистрам, в зависимости от шаблонов доступа. Регистры, локальная память и общая память ограничены ресурсами. Части памяти устройства могут использоваться в качестве

текстуры или постоянной памяти, которые выигрывают от кэширования на кристалле. Постоянная память оптимизирована для операций только для чтения, текстурная память – для конкретных шаблонов доступа. Потоки также имеют доступ к некэшированной памяти устройства общего назначения или глобальной памяти [160].

Различные ошибки могут привести к ухудшению производительности графического процессора. Во-первых, совместное использование памяти несколькими параллельными потоками может привести к так называемым банковским конфликтам, сериализующим выполнение этих потоков и, следовательно, уменьшающим параллелизм. Во-вторых, при обращении к глобальным адресам памяти должно быть кратно 4, 8 или 16, в противном случае доступ может быть скомпилирован для нескольких инструкций и, следовательно, обращений. Кроме того, адреса, к которым одновременно обращается несколько потоков в глобальной памяти, должны быть расположены так, чтобы доступ к памяти можно было объединить в единый непрерывный выравниваемый доступ к памяти. Это часто называют объединением памяти. Другим фактором является так называемая заполняемость. Заполняемость определяет, сколько блоков и, следовательно, потоков фактически работают параллельно. Поскольку разделяемая память и регистры являются ограниченными ресурсами, графический процессор может выполнять только определенное количество блоков параллельно. Поэтому обязательно оптимизировать использование общей памяти и регистрировать как можно больше параллельных блоков и потоков [160, 162].

В настоящее время уже немало параллельных алгоритмов, адаптированных к архитектуре CUDA, реализовано на графических процессорах [162-165].

Как правило, каждый поток (тред) при использовании архитектуры CUDA занят очень простыми операциями по обработке информации, связанной с простейшим объектом. Поскольку архитектура изначально разрабатывалась для обработки изображений, таким объектом является пиксель. В различных частях алгоритмов автоматической группировки таким объектом являются либо объекты данных, либо кластеры, представленные, например, координатами центроидов.

Таким образом, каждый поток вычислений является интеллектуальным агентом, отвечающим за обработку данных, связанных со своим элементарным объектом – объектом данных или центром кластера (для модели k-средних). Шаги алгоритма k-средних, отвечающие за разбиение множества объектов данных на кластеры, представленные координатами их центроидов, чередуются с шагами, связанными с перерасчетом центроидов кластеров. В первом случае при реализации для архитектуры CUDA могут запускаться потоки для каждого из объектов данных («агент», связанный с объектом данных, должен определить, к какому центроиду он ближе всего и «приписать» себя к соответствующему кластеру). Во втором случае агенты-центроиды, используя данные «приписанных» к ним объектов, перерасчитывают свои координаты – поток запускается для каждого центроида.

Рассмотрим параллельную реализацию алгоритма k-средних (Алгоритма 1.1) для архитектуры CUDA на GPU [162, 165, 166].

Мы использовали следующий вариант реализации Шага 1 Алгоритма 1.1.

Для первой части параллельного алгоритма, которая реализует 1-й шаг Алгоритма 1.1, мы использовали один поток вычислений (фактически без распараллеливания).

Алгоритм 2.8 CUDA-реализация шага 1 Алгоритма 1.1, часть 1

X'_j=0 для всех $j \in \{\overline{1,k}\}$. // Здесь, X'_j векторы, используемые для расчета новых кластерных центров/центроидов.

$counter_j$=0 для всех $j \in \{\overline{1,k}\}$. // счетчики объектов для каждого кластера

Для второй части алгоритма, которая реализует 1-й шаг Алгоритма 1.1, мы использовали $N_{trreads}$ = 512 потоков для каждого блока CUDA. Количество блоков рассчитывается как

$$N_{blocks}=(N+N_{threads}-1)/N_{threads}. \quad (2.1)$$

Таким образом, каждый поток обрабатывает только один объект (вектор) данных.

Алгоритм 2.9 CUDA-реализация шага 1 Алгоритма 1.1, часть 2

$i = blockIdx.x \times blockDim.x + threadIdx.x$.

Если $i>N$ тогда возврат.

j'=arg $\min_j \left\|A_j - X_i\right\|^2$. // номер кластера

$X'_{j'}=X'_{j'}+A_i$.

$C_i=j'$. // приписать A_i для кластера j'.

$counter_{j'}=counter_{j'}+1$.

Синхронизировать потоки.

Для части алгоритма, которая реализует 2-й шаг Алгоритма 1.1, мы использовали $N_{trreads}$ = 512 потоков для каждого блока CUDA. Количество блоков рассчитывается как $N_{blocks2} = (k + N_{threads}-1) / N_{threads}$.

Алгоритм 2.10 CUDA-реализация шага 2 Алгоритма 1.1

$j = blockIdx.x * blockDim.x + threadIdx.x$.

Если $j>k$ тогда возврат.

$X_{j'}= X'_i/counter_j$.

Синхронизировать потоки.

Производительность алгоритма k-средних при больших объемах данных становится проблемой, особенно когда найти правильный параметр k можно только путем выполнения нескольких запусков с разным количеством кластеров. Кроме того, Алгоритм 2.1 предполагает многократный запуск алгоритма k-средних (или другого метода локального поиска), и число этих запусков растет с ростом числа кластеров (квадратичная зависимость). Мы предлагаем использовать оптимизированную для GPU стратегию для k-средних, а также адаптированную к архитектуре CUDA процедуру исключения кластеров из решения, которая является обязательным и наиболее вычислительно затратным шагом в жадной агломеративной эвристической процедуре [166, 167]. Для этого мы реализовали Шаг 2 Алгоритма 2.1 на графическом процессоре (GPU). На этом этапе Алгоритм 2.1 вычисляет общее расстояние после удаления одного кластера:

$F'_{i'} = F(S')$, где $S' = S \setminus \{X_{i'}\}$. Вычислив $F(S)$, мы можем рассчитать $F'_{i'} = F(S') = F(S) + \sum_{l=1}^{N} \Delta D_l \cdot$, где

$$\Delta D_l \cdot = \begin{cases} 0, & C_{i'} \neq l, \\ \left(\min_{j \in \{\overline{1,k}\},\, j \neq i'} \left\| A_j - X_j \right\|^2\right) - \left\| A_j - X_{C_{i'}} \right\|^2, & C_{i'} = l. \end{cases} \quad (2.2)$$

где l – номер кластера. Здесь мы использовали 512 потоков (число подобрано экспериментальным путем) для каждого блока CUDA, количество блоков рассчитывается в соответствии с (2.1). Сначала переменная *sumD* инициализируется со значением 0. Затем для каждого вектора данных запускается следующий алгоритм и вычисляется ΔD_l (Рисунок 2.16).

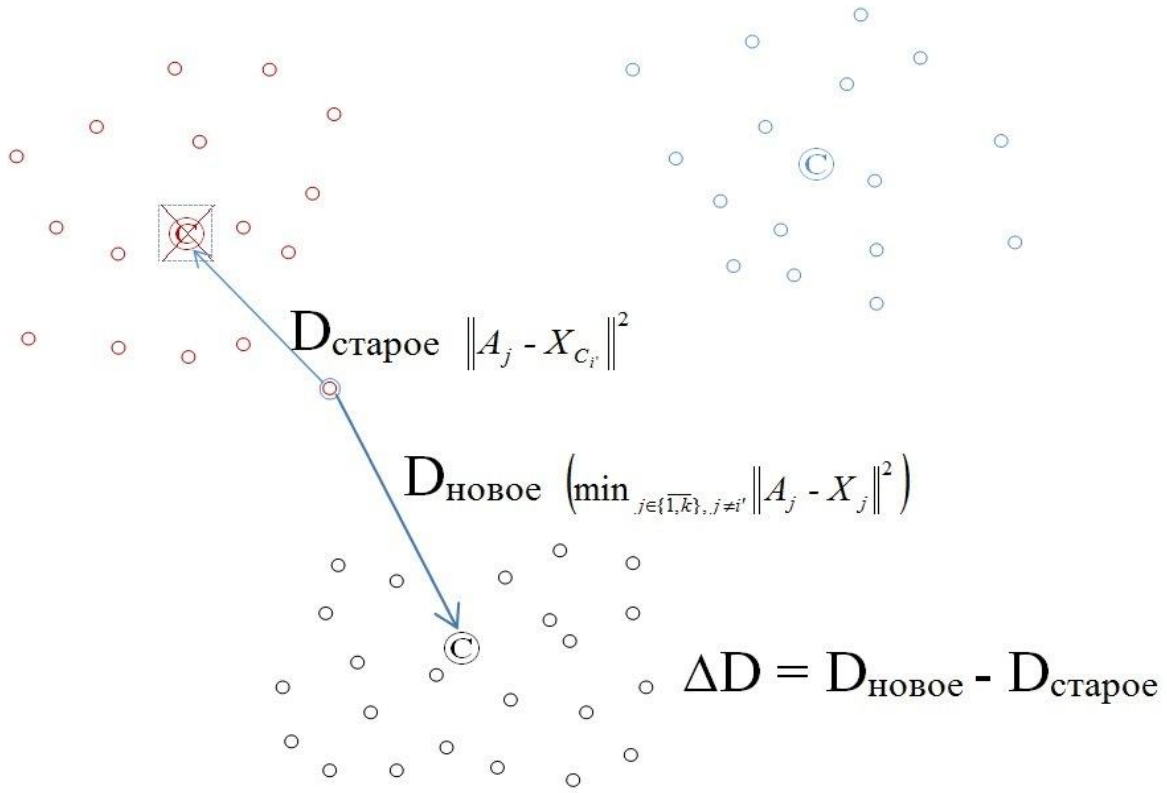


Рисунок 2.16 - Расчет приращения расстояния ΔD при удалении центроида

Алгоритм 2.11 CUDA-реализация шага 2 Алгоритма 2.1

---

l = blockIdx.x × *blockDim.x* + *threadIdx.x* .

Если *l*>*k* тогда возврат.

Рассчитать ΔD_l в соответствии с (2.2).

Если ΔD_l >0 то atomicAdd(*sumD*, ΔD_l).

Синхронизировать потоки.

Таким образом, каждый поток алгоритма 2.11 выполняет функцию интеллектуального агента, определяющего вклад каждого вектора данных ΔD_l в приращение целевой функции после удаления l-го кластера.

Все остальные алгоритмы работают на центральном процессоре (CPU).

Таким образом, были предложены параллельные алгоритмы с жадной агломеративной эвристической процедурой для решения задач автоматической группировки большого объема данных, адаптированные к архитектуре CUDA, в которых каждый поток выполняет роль интеллектуального агента, связанного с определенным вектором данных и реагирующего на такие события, как изменение координат или удаление центроидов.

2.5 Анализ результатов вычислительных экспериментов для массивно-параллельных систем

Для нашего исследования, как и раньше, были использованы классические наборы данных из репозиториев UCI и Clustering basic benchmark [155, 156].

Тестовая система состояла из Intel Core 2 Duo E8400CPU, 4GBRAM. Графический процессор NVIDIA GeForce 9600 GT, с 2048 МБ ОЗУ.

Необходимо отметить, что все алгоритмы, используемые для данных вычислительных расчетов, были реализованы для параллельных вычислений на графическом процессоре (CUDA-реализация). Поэтому во избежание путаницы с алгоритмами, указанными ранее в настоящей главе, к их названием был добавлен индекс «^{G}» (k-средних^G, k-GH-VNS1^G, k-GH-VNS2^G, k-GH-VNS3^G, k-GH-VNS1-RND^G, k-GH-VNS2-RND^G, k-GH-VNS3-RND^G, j-means^G, j-means-GH-VNS1^G, j-means-GH-VNS2^G, GA-FULL^G, GA-MIX^G, GA-ONE^G). Для всех наборов данных было выполнено по 30 попыток запуска каждого из 13 алгоритмов.

Фиксировались только лучшие результаты, достигнутые в каждой попытке, затем из этих результатов по каждому алгоритму были рассчитаны: минимальное

и максимальное значения (Min, Max), среднее значение (Среднее) и среднеквадратичное отклонение (СКО). Алгоритмы j-means и k-means были запущены в режиме мультистарта.

Лучшие значения целевой функции (минимальное значение, среднее значение и среднеквадратичное отклонение) выделены полужирным курсивом (Таблицы 2.10-2.13).

Таблица 2.10 - Результаты вычислительных экспериментов по набору данных Mopsi-Joensuu (180 секунд, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Среднеквадратичное отклонение
100 кластеров				
k-средних^G	20,2234	25,1256	22,6732	1,9230
k-GH-VNS1^G	1,8518	2,0704	1,9320	0,0996
k-GH-VNS2^G	***1,6519***	1,7969	1,7335	0,0504
k-GH-VNS3^G	1,6745	1,7950	1,7301	***0,0444***
k-GH-VNS1-RND^G	1,9142	2,9365	2,2084	0,3680
k-GH-VNS2-RND^G	1,7589	2,0456	1,8427	0,1026
k-GH-VNS3-RND^G	1,6558	1,8107	1,7204	0,0646
j-means^G	1,8600	10,2344	4,0787	3,4959
j-means-GH-VNS1^G	1,7801	2,1694	1,9197	0,1543
j-means-GH-VNS2^G	1,7337	2,0676	1,9031	0,1471
GA-FULL^G	***1,6544***	1,7569	***1,6760***	***0,0398***
GA-MIX^G	1,6600	17,7807	5,4884	6,5581
GA-ONE^G	19,0837	33,0772	26,8381	4,5549
300 кластеров				
k-средних^G	5,6141	8,9812	7,7135	1,1162
k-GH-VNS1^G	2,0335	3,4027	2,6656	0,4973
k-GH-VNS2^G	5,1070	11,1468	8,9344	2,2980
k-GH-VNS3^G	***0,1432***	0,2974	***0,1836***	***0,0582***
k-GH-VNS1-RND^G	2,2020	4,3911	2,7338	0,8446
k-GH-VNS2-RND^G	6,7474	14,6131	10,9959	2,6691
k-GH-VNS3-RND^G	***0,1533***	14,4612	9,1619	5,6364
j-means^G	2,3443	7,1081	4,1037	1,7900
j-means-GH-VNS1^G	2,4097	12,8224	9,9201	3,9420
j-means-GH-VNS2^G	3,7229	6,9412	5,4652	1,3822
GA-FULL^G	0,2073	3,6894	1,2855	1,5409
GA-MIX^G	0,7039	2,5733	1,4348	0,6968
GA-ONE^G	8,0874	15,9837	11,8232	3,1623

Таблица 2.11 - Результаты вычислительных экспериментов по набору данных BIRCH3 (100 кластеров, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Средне-квадратичное отклонение
60 секунд				
k-средних^{G}	8,18676E+13	9,96542E+13	8,98255E+13	8,37212E+12
k-GH-VNS1^{G}	3,71973E+13	3,76732E+13	3,73639E+13	0,18509E+12
k-GH-VNS2^{G}	3,73240E+13	4,06161E+13	3,91485E+13	1,14305E+12
k-GH-VNS3^{G}	3,72082E+13	3,72550E+13	***3,72422E+13***	***0,01998E+12***
k-GH-VNS1-RND^{G}	3,71993E+13	3,76607E+13	3,73757E+13	0,18322E+12
k-GH-VNS2-RND^{G}	3,98574E+13	5,17877E+13	4,47900E+13	4,74952E+12
k-GH-VNS3-RND^{G}	***3,71558E+13***	3,73328E+13	***3,72362E+13***	0,06507E+12
j-means^{G}	5,30805E+13	13,2286E+13	7,91183E+13	28,2000E+12
j-means-GH-VNS1^{G}	нет результа	та		
j-means-GH-VNS2^{G}	нет результа	та		
GA-FULL^{G}	3,74076E+13	3,84774E+13	3,75950E+13	0,34167E+12
GA-MIX^{G}	3,76402E+13	4,13519E+13	3,84577E+13	1,44968E+12
GA-ONE^{G}	6,36816E+13	9,10870E+13	7,47659E+13	11,6766E+12
600 секунд				
k-средних^{G}	7,98405E+13	9,96542E+13	8,93187E+13	9,04845E+12
k-GH-VNS1^{G}	***3,71474E+13***	3,71933E+13	***3,71778E+13***	***0,02348E+12***
k-GH-VNS2^{G}	***3,71474E+13***	3,72261E+13	***3,71834E+13***	0,02595E+12
k-GH-VNS3^{G}	***3,71473E+13***	3,72453E+13	***3,71817E+13***	0,03723E+12
k-GH-VNS1-RND^{G}	***3,71474E+13***	3,71932E+13	***3,71775E+13***	***0,02326E+12***
k-GH-VNS2-RND^{G}	***3,71474E+13***	3,72275E+13	***3,71853E+13***	0,03177E+12
k-GH-VNS3-RND^{G}	***3,71474E+13***	3,72275E+13	***3,71857E+13***	0,03163E+12
j-means^{G}	4,03266E+13	4,5392E+13	4,23065E+13	1,77787E+12
j-means-GH-VNS1^{G}	нет результа	та		
j-means-GH-VNS2^{G}	нет результа	та		
GA-FULL^{G}	3,72332E+13	3,74141E+13	3,72741E+13	0,06510E+12
GA-MIX^{G}	3,71525E+13	3,72071E+13	3,71949E+13	***0,02097E+12***
GA-ONE^{G}	3,71495E+13	3,7233E+13	3,71906E+13	0,04180E+12

Таблица 2.12 - Результаты вычислительных экспериментов по набору данных chess (50 кластеров, 120 секунд, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Среднеквадратичное отклонение
k-средних^{G}	6 926,22	6 958,36	6 941,13	11,2781
k-GH-VNS1^{G}	***6 851,11***	6 855,66	***6 853,08***	***1,5482***
k-GH-VNS2^{G}	***6 851,07***	6 857,08	***6 853,96***	2,4912
k-GH-VNS3^{G}	***6 851,15***	6 859,06	6 854,82	3,5286
k-GH-VNS1-RND^{G}	***6 851,29***	6 859,57	***6 853,93***	2,9739
k-GH-VNS2-RND^{G}	***6 851,25***	6 861,01	6 857,15	3,6077
k-GH-VNS3-RND^{G}	***6 851,30***	6 855,86	***6 853,93***	1,7071
j-means^{G}	6 938,97	6 987,53	6 962,71	18,6573
j-means-GH-VNS1^{G}	6 931,44	6 994,70	6 963,11	23,9752
j-means-GH-VNS2^{G}	6 962,87	6 994,55	6 980,95	10,77
GA-FULL^{G}	6 864,33	6 867,14	6 865,68	***1,2282***
GA-MIX^{G}	***6 851,41***	6 858,32	6 854,64	2,9540
GA-ONE^{G}	***6 851,44***	6 860,75	6 856,01	4,0019

Таблица 2.13 - Результаты вычислительных экспериментов по набору данных KDDCUP04BioNormed (2000 кластеров, 14 часов, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Среднеквадратичное отклонение
k-средних^{G}	4 424 475	4 426 251	4 425 137	***786,5***
k-GH-VNS1^{G}	4 358 583	4 386 584	4 367 311	12 966,3
k-GH-VNS2^{G}	4 338 584	4 419 181	4 378 916	42 724,9
k-GH-VNS3^{G}	***4 311 992***	4 318 547	***4 315 658***	2 721,5
k-GH-VNS1-RND^{G}	нет ре	зультата		
k-GH-VNS2-RND^{G}	нет ре	зультата		
k-GH-VNS3-RND^{G}	нет ре	зультата		
j-means^{G}	4 390 323	4 404 301	4 396 180	5 912,2
j-means-GH-VNS1^{G}	нет ре	зультата		
j-means-GH-VNS2^{G}	нет ре	зультата		
GA-FULL^{G}	***4 314 647***	4 319 851	***4 316 581***	2 847,4
GA-MIX^{G}	4 332 422	4 354 462	4 342 210	11 224,5
GA-ONE^{G}	4 426 306	4 431 211	4 428 233	2 615,5

По результатам вычислительных экспериментов (Таблицы 2.10-2.13) видно, что новый алгоритм k-GH-VNS3^{G} (использующий жадную процедуру с полным объединением множеств) стабильно показывает лучшие результаты на всех

наборах данных при параллельной реализации на графическом процессоре и при достаточном времени работы, так как данная эвристика, как правило, при первых же итерациях попадает в область достаточно «хороших» значений целевой функции. В то же время, данная эвристика часто «застревает» в этой области, и улучшает известное решение скачкообразно, в то время как другие варианты жадной эвристической процедуры способны шаг за шагом улучшать существующее решение. Поэтому очень важно, с какой из трех эвристик начинается поиск (параметр O_{start}, задающий номер окрестности в Алгоритме GH-VNS^{G}). При дефиците времени другие варианты алгоритма GH-VNS^{G} имеют преимущество.

Таблица 2.14 - Сравнение результатов работы алгоритмов на CPU и GPU по набору данных birch3 (100 кластеров, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Среднеквадратичное отклонение
GPU 1 минута				
k-средних^{G}	8,18676E+13	9,96542E+13	8,98255E+13	8,37212E+12
j-means^{G}	5,30805E+13	13,2286E+13	7,91183E+13	28,2000E+12
k-GH-VNS1^{G}	***3,71973E+13***	3,76732E+13	3,73639E+13	0,18509E+12
k-GH-VNS2^{G}	3,73240E+13	4,06161E+13	3,91485E+13	1,14305E+12
k-GH-VNS3^{G}	3,72082E+13	3,72550E+13	***3,72422E+13***	***0,01998E+12***
GPU 10 минут				
k-средних^{G}	7,98405E+13	9,96542E+13	8,93187E+13	9,04845E+12
j-means^{G}	4,03266E+13	4,5392E+13	4,23065E+13	1,77787E+12
k-GH-VNS1^{G}	***3,71474E+13***	3,71933E+13	***3,71778E+13***	***0,02348E+12***
k-GH-VNS2^{G}	***3,71474E+13***	3,72261E+13	3,71834E+13	0,02595E+12
k-GH-VNS3^{G}	***3,71473E+13***	3,72453E+13	3,71817E+13	0,03723E+12
CPU 6 часов				
k-средних	7,92474E+13	8,87404E+13	8,31599E+13	3,088140E+12
j-means	3,76222E+13	3,7965E+13	3,77715E+13	0,116211E+12
k-GH-VNS1	3,72537E+13	3,77474E+13	3,74703E+13	0,171124E+12
k-GH-VNS2	4,21378E+13	5,01871E+13	4,52349E+13	4,333462E+12
k-GH-VNS3	***3,72525E+13***	3,74572E+13	***3,73745E+13***	***0,074315E+12***

В исследованиях по набору данных BIRCH3 без использования технологии CUDA (Таблица 2.5) было получено лучшее минимальное значение целевой функции 3.72525E+13 при условии 6 часов на каждую попытку. При расчетах с использованием графического процессора (Таблица 2.11) получено минимальное значение целевой функции 3.71473E+13 тем же алгоритмом, но за 10 минут и 3.71973E+13 за 1 минуту. Как видим (Рисунок 2.17) результат при использовании графического ускорителя получился точнее (Таблица 2.14), а время затрачено на несколько порядков меньше (в данном случае в 360 раз).

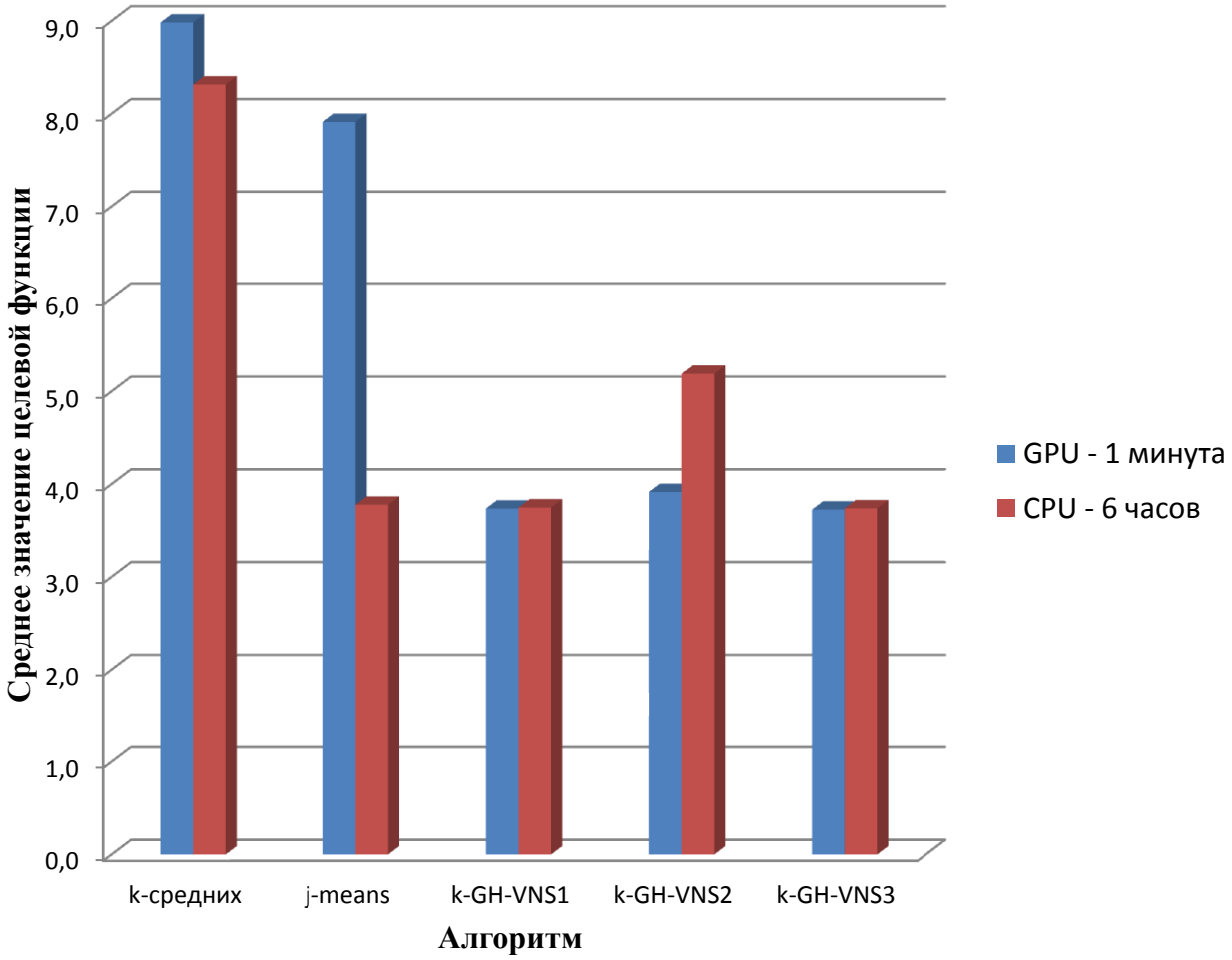


Рисунок 2.17 - Сравнение результатов работы алгоритмов на CPU и GPU по набору данных birch3

Также в качестве тестовых наборов данных также были использованы результаты неразрушающих тестовых испытаний сборных производственных

партий электрорадиоизделий, проведенных в специализированном тестовом центре для комплектации бортовой аппаратуры космических аппаратов, состав которых заранее известен (Таблица 2.15). Лучшие значения целевой функции (минимальное значение, среднее значение и среднеквадратичное отклонение) выделены полужирным курсивом.

Таблица 2.15 - Результаты вычислительных экспериментов по набору данных 1526IE10 (10 кластеров, 20 секунд, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Среднеквадратичное отклонение
k-средних^{G}	3 925,08	3 925,09	3 925,09	0,0050
k-GH-VNS1^{G}	***3 925,04***	3 925,04	***3 925,04***	***0,0000***
k-GH-VNS2^{G}	3 925,08	3 925,74	3 925,25	0,2756
k-GH-VNS3^{G}	3 925,08	3 926,32	3 925,29	0,5060
k-GH-VNS1-RND^{G}	***3 925,04***	3 925,04	***3 925,04***	***0,0000***
k-GH-VNS2-RND^{G}	3 925,07	3 925,12	3 925,09	0,0148
k-GH-VNS3-RND^{G}	3 925,05	3 925,08	3 925,07	0,0128
j-means^{G}	3 926,38	4 100,77	3 981,19	69,7025
j-means-GH-VNS1^{G}	нет ре	зультата		
j-means-GH-VNS2^{G}	3 926,54	4 118,59	4 054,57	110,87
GA-FULL^{G}	3 925,07	3 925,10	3 925,08	0,0089
GA-MIX^{G}	***3 925,04***	3 925,08	3 925,05	0,0164
GA-ONE^{G}	***3 925,04***	3 925,07	3 925,05	0,0151

Алгоритмы кластеризации, которые показывают лучшие результаты целевой функции с небольшим числом кластеров, не всегда являются лучшими с увеличением числа кластеров. Однако преимущество семейства жадных эвристических алгоритмов над алгоритмом k-средних, а так же j-means (считающимся одним из лучших) остается после перехода к архитектуре CUDA. Использование графического процессора показывает преимущество в достижении скорости по сравнению с вычислениями на процессоре, и преимущество увеличивается для больших наборов данных и большого количества кластеров в десятки и сотни раз [166, 167].

Результаты Главы 2

Метод жадных эвристик [142] может быть успешно применен в составлении эффективной комбинации алгоритмов для решения задачи k-средних.

Новые комбинированные алгоритмы поиска с чередующимися рандомизированными окрестностями (k-GH-VNS) имеют более стабильные результаты (дают меньшее минимальное значение и/или среднеквадратичное отклонение целевой функции, меньший разброс достигнутых значений) и, следовательно, лучшие показатели в сравнении с известными (классическими) алгоритмами j-means и k-средних. При параллельной реализации новых алгоритмов на графическом процессоре для больших задач автоматической группировки, адаптированных к архитектуре CUDA они также стабильно показывают хорошие результаты на всех наборах данных, конкурируя с генетическими алгоритмами метода жадных эвристик. Параллельные алгоритмы сохраняют важное свойство алгоритмов метода жадных эвристик: высокая точность получаемых результатов.

Как видно из результатов вычислительных экспериментов, значения целевой функции новых алгоритмов в ряде случаев оказываются существенно лучше и стабильнее результатов генетического алгоритма. При этом в некоторых задачах новые алгоритмы уступают генетическим алгоритмам, но не существенно. Следует отметить, что для задач среднего размера (примерно до 10000 векторов данных, до 100 кластеров) новые алгоритмы утрачивают преимущество в сравнении с генетическими алгоритмами при значительном увеличении времени счета. Для больших задач выполнение лишь нескольких итераций генетического алгоритма за приемлемое время (например, сутки) не всегда возможно даже на современной вычислительной технике, например, массивно-параллельных системах. При этом алгоритмы k-GH-VNS демонстрируют хорошие результаты.

Таким образом, в данной главе были решены задачи разработки новых алгоритмов поиска с чередующимися рандомизированными окрестностями для

задачи k-средних и реализации жадных эвристических алгоритмов автоматической группировки для массивно-параллельных систем. Показано, что параллельная реализация алгоритма локального поиска, а также отдельных шагов жадной агломеративной эвристической процедуры позволяет построить алгоритм автоматической группировки с высоким коэффициентом ускорения, сокращающим время расчетов в десятки раз без ухудшения достигаемого значения целевой функции.

ГЛАВА 3. АЛГОРИТМЫ МЕТОДА ЖАДНЫХ ЭВРИСТИК С ЧЕРЕДУЮЩИМИСЯ ОКРЕСТНОСТЯМИ ДЛЯ ЗАДАЧ K-МЕДОИД И МАКСИМИЗАЦИИ ФУНКЦИИ ПРАВДОПОДОБИЯ

Глава посвящена разработке комбинированных алгоритмов метода жадных эвристик для задач автоматической группировки с повышенными требованиями к точности и стабильности результата с применением алгоритмов поиска с чередующимися рандомизированными окрестностями применительно к более широкому кругу задач: задаче k-медоид и максимизации математического ожидания.

3.1 Комбинированные алгоритмы поиска с чередующимися рандомизированными окрестностями для задачи k-медоид

Одной из классических моделей теории размещения является p-медианная задача. Целью непрерывной задачи размещения [27] является нахождение местоположения одной или нескольких точек (в зависимости от конкретной постановки задачи - центров, центроидов, медоидов и т.д.) в непрерывном пространстве (рассматривается бесконечное множество возможных местоположений искомых точек). Также существует промежуточный класс задач, дискретных фактически (число возможных местоположений является конечным), но при этом оперирующих понятиями, которые характерны для непрерывных задач. К таким относится и задача p-медоид [168, 169] (в литературе также называемая задачей k-медоид или дискретной p-медианной задачей [170]). Основными параметрами задач размещения являются координаты объектов и расстояния между ними [28, 171, 172].

Целью непрерывной p-медианной задачи [171] является нахождение таких k точек (медоидов, центров, центроидов), чтобы сумма взятых с весовыми коэффициентами расстояний от N известных точек (называемых точками требования, векторами данных или потребителями в зависимости от постановки и

предметной области задачи) до ближайшего из *k* центров достигала минимума.

В настоящее время предложено множество алгоритмов решения задачи Вебера для непрерывных задач размещения с евклидовой, манхэттенской (прямоугольной), чебышевской метриками (все эти метрики являются частными случаями метрик, основанных на l_p-нормах Минковского [173]). В частности, для метрик, основанных на нормах Минковского, была обобщена известная процедура Вайсфелда [54].

В традиционном представлении, в случае евклидовой метрики $L(X_j,A_i)=\sqrt{\sum_{k=1}^{d}(x_{j,k}-a_{i,k})^2}$ мы имеем собственно p-медианную задачу. Здесь $X_j=(x_{j,1},...,x_{j,k}) \quad \forall j=\overline{1,p}$, $A_i=(a_{i,1},...,a_{i,k}) \quad \forall i=\overline{1,N}$. В случае квадратичной евклидовой метрики $L(X_j,A_i)=\sum_{k=1}^{d}(x_{j,k}-a_{i,k})^2$ при $w_i=1 \forall i=\overline{1,N}$ мы имеем задачу k-средних. Здесь $x_1,\ldots,x_N$ булевы переменные, L – функция расстояния, N векторов данных $A_1,\ldots,A_N$ в d-мерном пространстве, $A_i=(a_{i,1},\ldots,a_{i,d})$, $A_i \in \mathrm{R}^d$.

Задача (модель) k-медоид отличается тем, что центры кластеров, называемые медоидами, отыскиваются исключительно среди известных точек A_i, то есть она относится к задачам дискретной оптимизации.

Методы локального поиска при решении задач дискретной оптимизации являются наиболее естественными и наглядными [139]. Эти идеи использовались и для решения задач размещения, коммивояжера, построения сетей, расписаний и др. [174-177]. Простой локальный спуск не позволяет находить глобальный оптимум задачи, но такие методы, как правило, являются довольно быстрыми. С появлением новых бионических алгоритмических схем и новых теоретических результатов в области локального поиска, интерес к ним не пропал.

Классический алгоритм локального спуска начинает работу с начального решения x_0 (в нашем случае – с начального множества медоидов), которое выбирается случайно или с помощью какого-либо вспомогательного алгоритма. До тех пор пока не будет достигнут локальный оптимум на каждом шаге локального спуска происходит переход от текущего решения к соседнему решению с меньшим значением целевой функции.

Функция окрестности O на каждом шаге локального спуска задает множество возможных направлений движения локального поиска. Довольно часто это множество состоит из нескольких элементов и при выборе следующего решения имеется определенная свобода, но правило выбора может оказать существенное влияние на результат работы алгоритма. Чтобы сократить трудоемкость одного шага при выборе окрестности желательно иметь множество $O(X)$ как можно меньшей мощности. Хотя с другой стороны, более широкая окрестность может привести к лучшему локальному оптимуму. Одним из путей решения этого противоречия является разработка сложных окрестностей, размер которых можно варьировать в ходе локального поиска [43, 139].

В настоящей работе мы предлагаем комбинированное использование алгоритмов локального поиска, содержащих в своем составе жадные агломеративные эвристические процедуры, а также известного PAM-алгоритма (Алгоритм 1.2), с применением схемы поиска с чередующимися рандомизированными окрестностями [132, 178].

Алгоритм 3.1 PAM-GH-VNS

1: Получить решение S, запустив Алгоритм 1.2 из случайным образом сгенерированного начального решения.

2: $O=O_{start}$ **(**номер окрестности поиска).

3: $i=0$, $j=0$**.**

пока $j < j_{\max}$

пока $i < i_{\max}$

4: **если** не выполняются условия ОСТАНОВа, **то** получить решение S', запустив Алгоритм 1.2 из случайного начального решения.

повторять

5: В зависимости от значения S (возможны значения 1, 2 или 3), запустить Алгоритм Жадная процедура 1, 2 или 3 соответственно с начальными решениями S и S'. Так, окрестность определяется способом включения центров

кластеров из второго известного решения и параметром окрестности – вторым известным решением.

если новое решение лучше, чем S, **то**

записать новый результат в S, i=0, j=0.

иначе выйти из цикла.

конец цикла

6: $i=i+1$.

конец цикла

7: i=0, $j=j+1$, $O=O+1$, **если** $O>3$, **то** $O=1$.

конец цикла

Далее для сравнительных вычислительных экспериментов мы использовали новые генетические алгоритмы GA-FULL и GA-ONE. Описание алгоритмов дано в [109, 142].

3.2 Результаты вычислительных экспериментов с новыми алгоритмами для задачи k-медоид

В нижеследующих таблицах использовали следующие аббревиатуры и сокращения алгоритмов:

PAM - классический PAM-алгоритм;

PAM-GH-VNS1, PAM-GH-VNS2, PAM-GH-VNS3 - вариации алгоритма поиска с чередующимися рандомизированными окрестностями (Алгоритм 2.8);

PAM-GH-VNS1-RND, PAM-GH-VNS2-RND, PAM-GH-VNS3-RND - вариации алгоритма поиска с чередующимися рандомизированными окрестностями со случайным образом определенным начальным решением;

GA-FULL - генетический алгоритм с жадной эвристикой с вещественным алфавитом [109];

GA-ONE - генетический алгоритм, в котором Алгоритм 2.3 (Жадная процедура 2) используется в качестве процедуры кроссинговера.

В качестве тестовых наборов данных были проанализированы (Таблицы 3.1-3.3) результаты неразрушающих тестовых испытаний сборных производственных партий электрорадиоизделий, проведенных в специализированном тестовом центре для комплектации бортовой аппаратуры космических аппаратов [178].

Для экспериментов использовалась вычислительная система DEXP OEM (4-ядерное ЦПУ Intel® Core™ i5-7400 CPU 3.00 ГГц, 8 Гб ОЗУ).

Для всех наборов данных было выполнено по 30 попыток запуска каждого из 9 алгоритмов. Фиксировались только лучшие результаты, достигнутые в каждой попытке, затем из этих результатов по каждому алгоритму были рассчитаны значения целевой функции: минимальное значение (Min), среднее значение (Среднее) и среднеквадратичное отклонение.

Результаты наших вычислительных экспериментов представлены в Таблицах 3.1-3.3. Лучшие значения целевой функции (минимальное значение, среднее значение и среднеквадратичное отклонение) выделены полужирным курсивом.

Сравнивать алгоритмы можно по среднему значению целевой функции или по медиане. В данном исследовании будем по среднему значению целевой функции, потому что если какой-то алгоритм случайно выдаст один раз лучший результат целевой функции (минимальное значение, среднее значение или среднеквадратичное отклонение), а в остальных экспериментах хуже, то такой алгоритм, конечно же, не будет являться лучшим.

Графическое сравнение новых и известных алгоритмов по каждому набору данных (Таблицы 3.1-3.3) показано на графиках сходимости алгоритмов построенных по среднему значению целевой функции (Рисунки 3.1-3.3). По оси абсцисс - время, по оси ординат - достигнутое среднее значение целевой функции.

Таблица 3.1 - Результаты вычислительных экспериментов по набору данных 3OT122A (767 векторов данных, каждый размерностью 13) 10 кластеров, 60 секунд, 30 попыток, расстояние Manhattan

Алгоритм	Значение целевой функции		
	Min (рекорд)	Среднее	Среднеквадратичное отклонение
PAM	1 654,39	1 677,35	12,2445
PAM-GH-VNS1	***1 554,26***	***1 566,70***	7,4928
PAM-GH-VNS2	1 558,02	***1 566,06***	***5,0686***
PAM-GH-VNS3	***1 555,09***	***1 563,99***	***3,9161***
GA-FULL	1 599,23	1 637,58	25,5365
GA-ONE	1 589,87	1 614,78	13,5342

Таблица 3.2 - Результаты вычислительных экспериментов по набору данных 5514BC1T2-9A5 (91 векторов данных, каждый размерностью 173) 10 кластеров, 60 секунд, 30 попыток, расстояние Manhattan

Алгоритм	Значение целевой функции		
	Min (рекорд)	Среднее	Среднеквадратичное отклонение
PAM	7 623,81	7 629,74	8,4124
PAM-GH-VNS1	***7 604,49***	***7 604,49***	***0,0000***
PAM-GH-VNS2	***7 604,49***	***7 604,49***	***0,0000***
PAM-GH-VNS3	***7 604,49***	***7 604,49***	***0,0000***
GA-FULL	***7 604,49***	***7 604,49***	***0,0000***
GA-ONE	***7 604,49***	7 606,43	6,1597

Таблица 3.3 - Результаты вычислительных экспериментов по набору данных 1526TL1 (1234 векторов данных, каждый размерностью 157) 10 кластеров, 60 секунд, 30 попыток, расстояние Manhattan

Алгоритм	Значение целевой функции		
	Min (рекорд)	Среднее	Среднеквадратичное отклонение
PAM	50 184,01	50 883,73	472,4409
PAM-GH-VNS1	***45 440,37***	***45 553,02***	***95,8004***
PAM-GH-VNS2	45 453,68	***45 657,68***	153,3286
PAM-GH-VNS3	***45 444,42***	***45 637,87***	177,5864
GA-FULL	46 660,86	48 391,17	845,0838
GA-ONE	47 081,34	48 125,99	766,5659

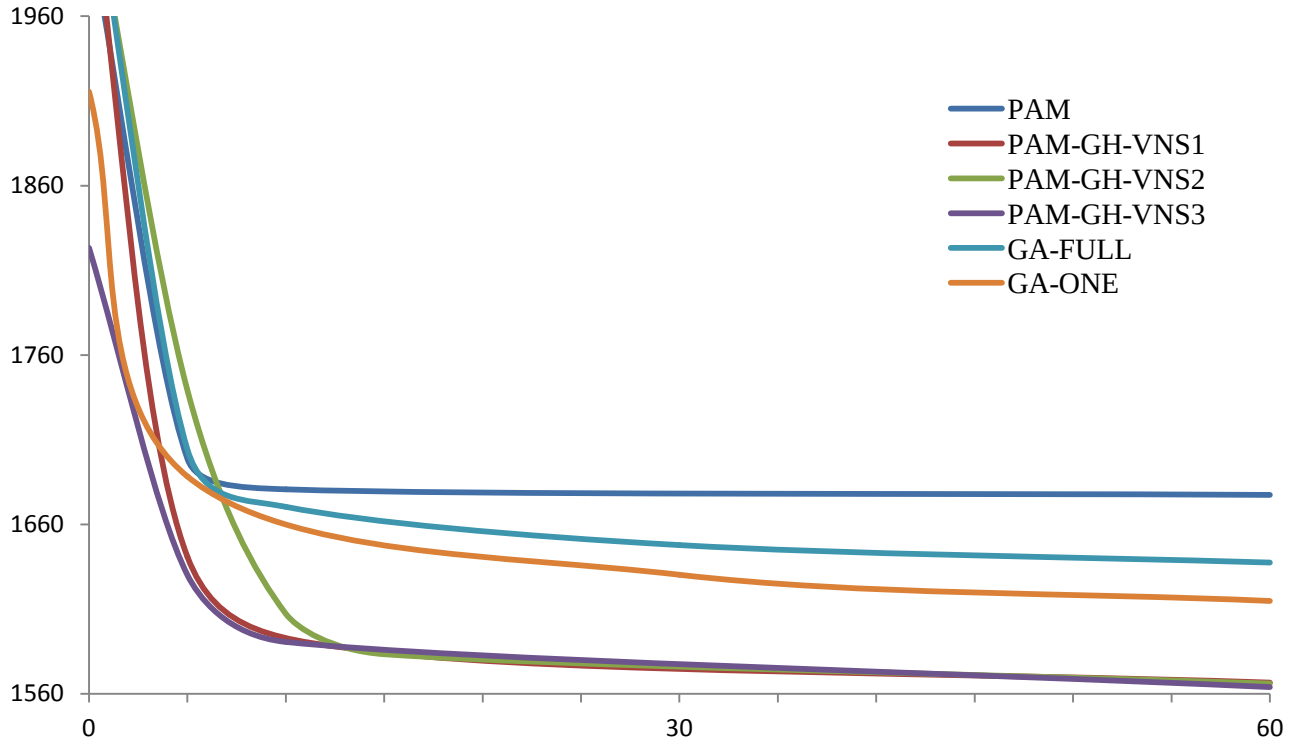


Рисунок 3.1 - Сравнение новых и известных алгоритмов для набора данных 3OT122A (10 кластеров, 60 секунд)
по оси абсцисс – время в секундах, по оси ординат – достигнутое среднее значение целевой функции

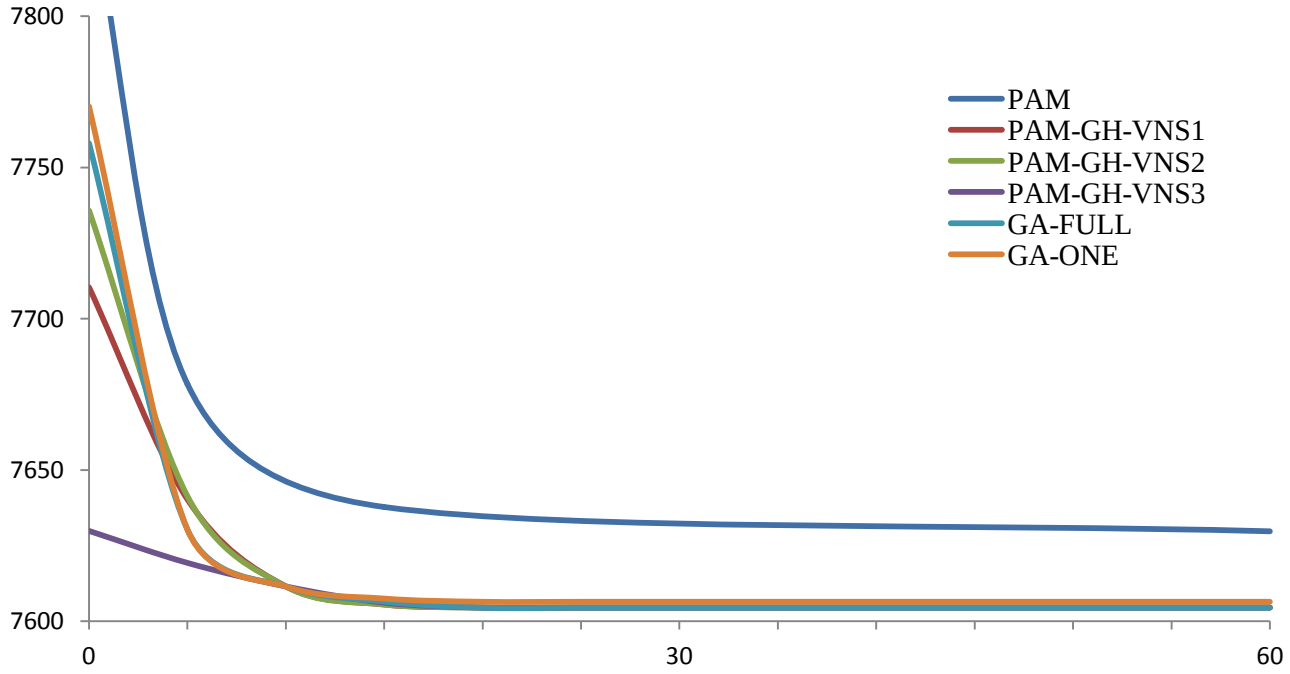


Рисунок 3.2 - Сравнение новых и известных алгоритмов для набора данных 5514BC1T2-9A5 (10 кластеров, 60 секунд)
по оси абсцисс – время в секундах, по оси ординат – достигнутое среднее значение целевой функции

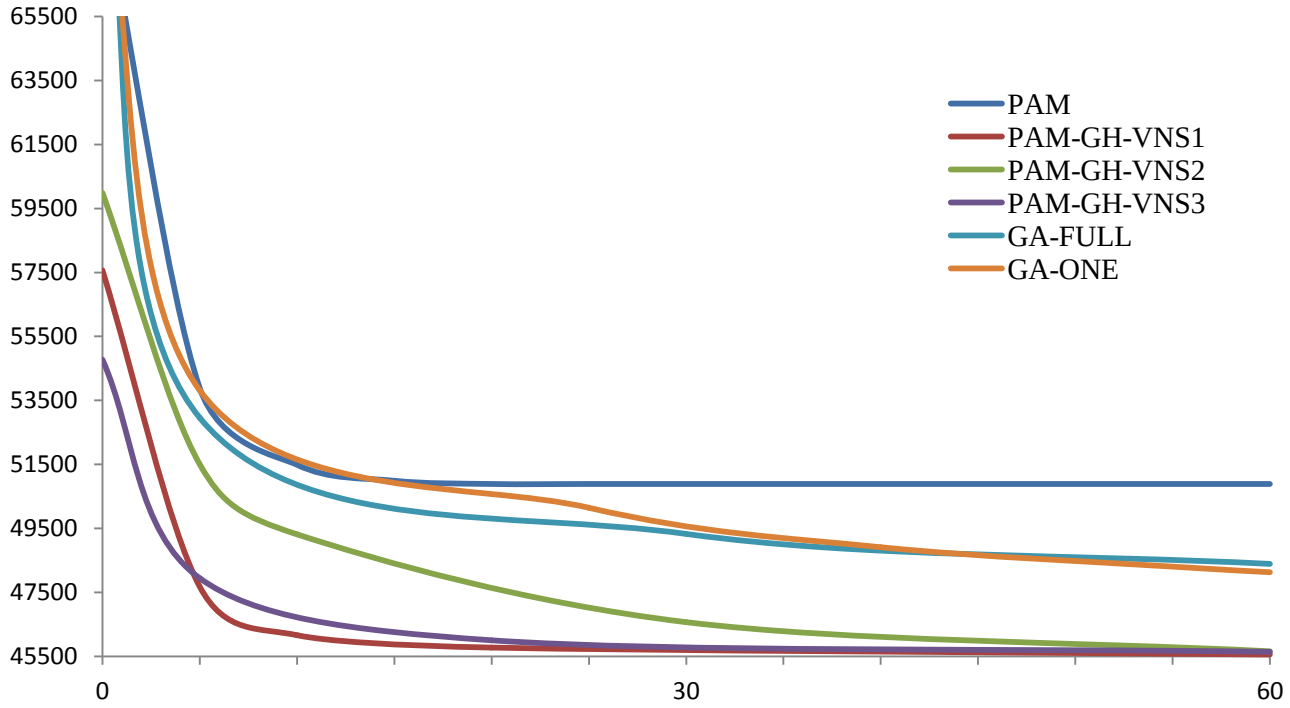


Рисунок 3.3 - Сравнение новых и известных алгоритмов для набора данных 1526TL1 (10 кластеров, 60 секунд)
по оси абсцисс – время в секундах, по оси ординат – достигнутое среднее значение целевой функции

Для более полного сравнения полученных результатов вычислительных экспериментов использованы результаты проведенных ранее вычислительных экспериментов над наборами данных электрорадиоизделий различными модификациями генетического алгоритма [142]. Сравнительные результаты приведены в Таблице 3.4. Для расчетов были использованы сборные партии электрорадиоизделий 1526TL1.

В Таблице 3.4 использованы следующие аббревиатуры и сокращения [142]: ГА - генетический алгоритм, ЖЭ - жадная эвристика, ГАЖЭ – генетический с жадной эвристикой с вещественным алфавитом, ЛП - локальный поиск, ГА ФП – генетический алгоритм с рекомбинацией подножеств фиксированной длины [37], IBC – Information Bottleneck Clustering, ЖЛ – мультистарт жадной эвристики с включенным локальным поиском, k-средн. мультистарт – мультистарт ALA-процедуры.

Таблица 3.4 - Результаты вычислительных экспериментов по набору данных 1526TL1 (1234 векторов данных, каждый размерностью 157) 10 кластеров, 60 секунд, 30 попыток, расстояние squared Euclidean

Алгоритм	Значение целевой функции		
	Min (рекорд)	Среднее	Среднеквадратичное отклонение
PAM	64 232,02	66 520,18	991,9938
PAM-GH-VNS1	55 373,00	55 906,02	416,4050
PAM-GH-VNS2	***55 361,75***	55 858,35	359,4161
PAM-GH-VNS3	55 383,81	55 755,00	353,9469
GA-FULL	58 789,34	60 629,52	1 187,0953
GA-ONE	58 300,15	60 165,43	1 388,6209
ГАЖЭ+ЛП	***55 361,75***	55 364,10	6,2204
ГАЖЭ вещ., σe=0.25	***55 361,75***	***55 361,75***	***7,86E-12***
ГАЖЭ вещ.частичн., σe=0.25	***55 361,75***	***55 361,75***	***7,86E-12***
ГА ФП	***55 361,75***	55 452,68	240,5632
ГА классич.	***55 361,75***	55 364,10	6,2204
IBC, σe=0.25	нет результата		
Детерм. ЖЭ, σe=0.25	57 131,00	57 131,00	***0,0000***
Детерм. ЖЭ, σe=0.001	55 998,22	55 998,22	***0,0000***
IBC, σe=0.001	нет результата		
ЖЭ адапт. σe=0.25	***55 361,75***	***55 361,75***	***7,86E-12***
ЖЭ адапт. σe=0.001	***55 361,75***	55 381,31	33,3953
ЖЭ, σe=0.25, β=0.5	***55 361,75***	***55 361,75***	***7,86E-12***
ЖЭ, σe=0.25, β=1	***55 361,75***	***55 361,75***	***7,86E-12***
ЖЭ, σe=0.25, β=3	***55 361,75***	55 371,53	25,8679
ЖЭ, σe=0.001, β=0.5	***55 361,75***	55 366,45	8,0305
ЖЭ, σe=0.001, β=1	***55 361,75***	***55 361,75***	***7,86E-12***
ЖЭ, σe=0.001, β=3	***55 361,75***	55 371,53	25,8679
ЖЛ, σe=0.25, β=0.5	***55 361,75***	55 604,47	294,1579
ЖЛ, σe=0.25, β=1	***55 361,75***	55 455,03	239,6050
ЖЛ, σe=0.25, β=3	***55 361,75***	55 907,30	240,5632
ЖЛ, σe=0.001, β=0.5	***55 361,75***	55 548,22	241,5122
ЖЛ, σe=0.001, β=1	***55 361,75***	55 634,52	340,2077
ЖЛ, σe=0.001, β=3	***55 361,75***	55 907,30	240,5632
k-средн. мультистарт	***55 361,75***	55 364,10	6,220381

Для окончательной проверки и выводов по полученным результатам вычислительных экспериментов с производственными партиями электрорадиоизделий для космических аппаратов и для возможности применения наших новых алгоритмов в дальнейшем использовали общедоступные и

известные наборы данных из репозиториев UCI [155] и Clustering basic benchmark [156].

Кроме этого, произвели расчеты с разным количеством кластеров и разными расстояниями (Таблицы 3.5-3.7). Графическая реализация сходимости алгоритмов построенных по среднему значению целевой функции представлена на Рисунках 3.4-3.6.

Таблица 3.5 - Результаты вычислительных экспериментов по набору данных ionosphere (351 векторов данных, каждый размерностью 35) 10 кластеров, 60 секунд, 30 попыток, расстояние Manhattan

Алгоритм	Значение целевой функции		
	Min (рекорд)	Среднее	Среднеквадратичное отклонение
PAM	2 688,57	2 704,17	12,3308
PAM-GH-VNS1	***2 607,21***	***2 607,25***	***0,1497***
PAM-GH-VNS2	***2 607,21***	***2 607,43***	***0,4303***
PAM-GH-VNS3	***2 607,21***	***2 607,34***	***0,4159***
GA-FULL	***2 608,22***	2 624,97	9,5896
GA-ONE	***2 608,69***	2 625,18	10,7757

Таблица 3.6 - Результаты вычислительных экспериментов по набору данных Mopsi-Joensuu (6015 векторов данных, каждый размерностью 2) 20 кластеров, 60 секунд, 30 попыток, расстояние Euclidean

Алгоритм	Значение целевой функции		
	Min (рекорд)	Среднее	Среднеквадратичное отклонение
PAM	319,84	343,44	***15,3004***
PAM-GH-VNS1	278,63	390,43	82,6086
PAM-GH-VNS2	333,26	471,15	100,2594
PAM-GH-VNS3	***273,91***	***334,98***	51,2440
PAM-GH-VNS1-RND	301,91	428,14	129,2156
PAM-GH-VNS2-RND	384,62	475,92	53,0972
PAM-GH-VNS3-RND	***265,96***	***325,49***	42,1440
GA-FULL	315,57	383,41	60,1489
GA-ONE	343,21	433,01	66,0360

Таблица 3.7 - Результаты вычислительных экспериментов по набору данных Chess (3196 векторов данных, каждый размерностью 37) 50 кластеров, 60 секунд, 30 попыток, расстояние squared Euclidean

Алгоритм	Значение целевой функции		
	Min (рекорд)	Среднее	Среднеквадратичное отклонение
PAM	10 763,0	10 822,4	***47,1268***
PAM-GH-VNS1	10 357,0	***10 530,9***	122,9620
PAM-GH-VNS2	10 803,0	11 107,1	174,1184
PAM-GH-VNS3	10 429,0	10 594,6	114,7192
PAM-GH-VNS1-RND	10 400,0	10 659,0	161,2982
PAM-GH-VNS2-RND	10 891,0	11 097,0	187,9113
PAM-GH-VNS3-RND	***10 310,0***	10 623,3	214,1288
GA-FULL	***10 252,0***	***10 381,3***	***72,9110***
GA-ONE	10 944,0	11 098,0	112,0813

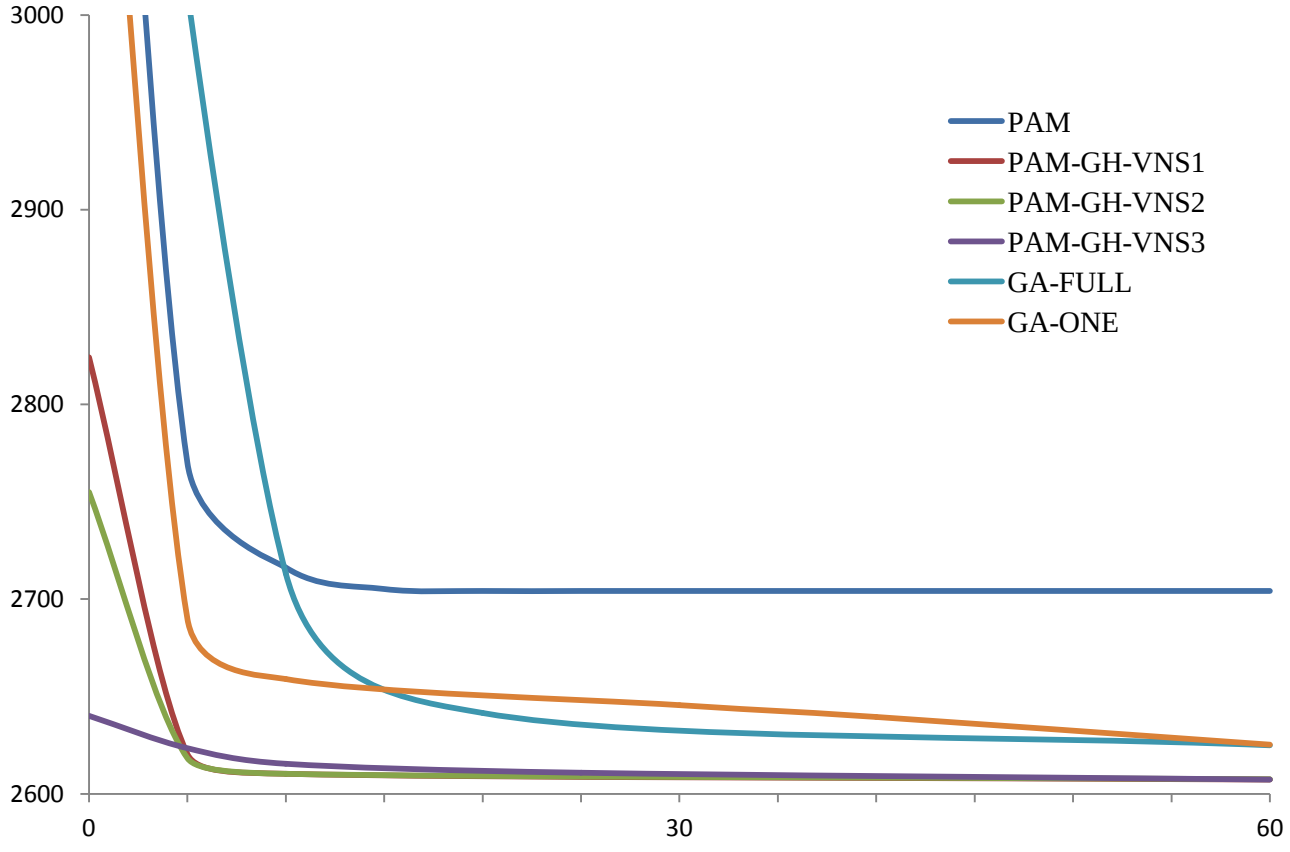


Рисунок 3.4 - Сравнение новых и известных алгоритмов для набора данных ionosphere (10 кластеров, 60 секунд, расстояние Manhattan)
по оси абсцисс – время в секундах, по оси ординат – достигнутое среднее значение целевой функции

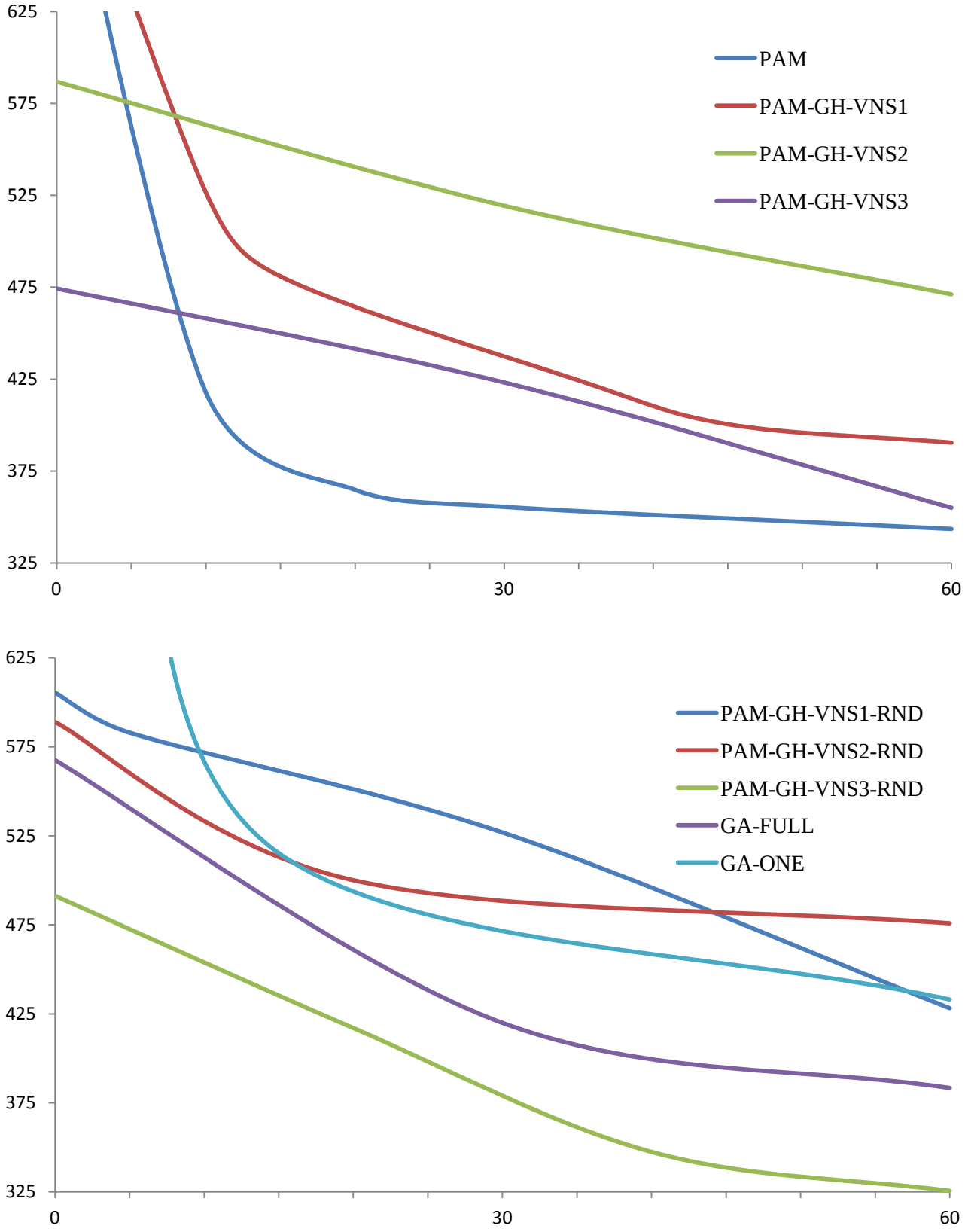


Рисунок 3.5 - Сравнение новых и известных алгоритмов для набора данных Mopsi-Joensuu (20 кластеров, 60 секунд, расстояние Euclidean)
по оси абсцисс – время в секундах, по оси ординат – достигнутое среднее значение целевой функции

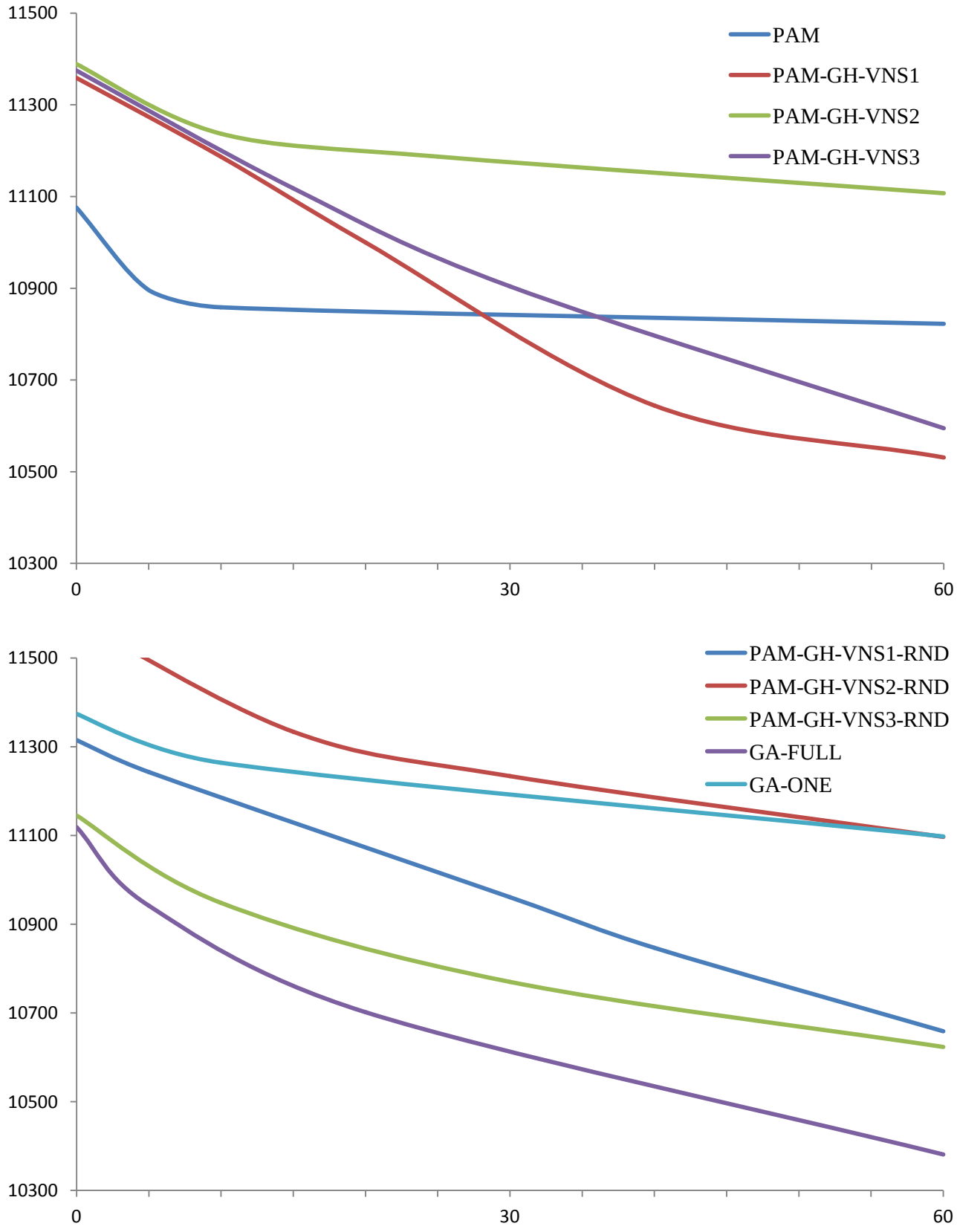


Рисунок 3.6 - Сравнение новых и известных алгоритмов для набора данных Chess (50 кластеров, 60 секунд, расстояние squared Euclidean)
по оси абсцисс – время в секундах, по оси ординат – достигнутое среднее значение целевой функции

Результаты вычислительных экспериментов показали, что новые комбинированные алгоритмы поиска с чередующимися рандомизированными окрестностями (PAM-GH-VNS) при небольшом числе кластеров имеют более стабильные результаты (дают меньшее среднее значение и/или среднеквадратичное отклонение достигаемого значения целевой функции, меньший разброс достигнутых значений) и, следовательно, лучшие показатели в сравнении с известными алгоритмами (на наборах данных ionosphere и электрорадиоизделий). При числе кластеров более 20 на булевых данных как новые, так и генетические алгоритмы метода жадных эвристик, показывая преимущество в достижении лучших средних достигаемых значений целевой функции при многократных запусках, не демонстрируют при этом преимущества по стабильности получаемого значения целевой функции (PAM-алгоритм демонстрирует более стабильный, но при этом стабильно плохой результат в сравнении с некоторыми из новых алгоритмов).

3.3 Комбинированный классификационный EM-алгоритм

В настоящее время имеется большое количество методов кластеризации данных [180]. EM-алгоритм (Expectation Maximization - максимизация математического ожидания) входит в число популярных. Он используется в случае анализа неполных данных[107, 110, 181]:

- в силу каких-либо причин отсутствуют некоторые статистические данные;
- функция правдоподобия имеет вид, допускающий серьезные упрощения при введении дополнительных «скрытых» величин, но не допускающий «удобных» методов исследования.

При решении EM-алгоритмом задачи кластеризации она сводится к задаче разделения смеси вероятностных распределений. Общая постановка задачи разделения смеси распределений состоит в следующем.

Пусть плотность распределения на множестве X имеет вид смеси k распределений (предполагаем, что распределения гауссовы):

$$\rho(x)=\sum_{j=1}^{k}\alpha_j\rho_j(x),\ \sum_{j=1}^{k}\alpha_j=1,\ \alpha_j\geq 0,$$

где $\rho_j(x)$ - функция правдоподобия *j*-ой компоненты смеси, α_j - ее априорная вероятность («вес» в составе смеси).

Главным достоинством EM-алгоритма является простота исполнения. Вдобавок ко всему, он может оптимизировать не только параметры модели, но и делать предположения относительно значений отсутствующих данных.

Это делает EM отличным методом для кластеризации и создания моделей с параметрами. Зная кластеры и параметры модели можно предполагать, что содержит кластер и куда стоит отнести новые данные.

Хотя у EM-алгоритма есть свои недостатки:

1. С ростом количества итераций падает производительность алгоритма.
2. EM не всегда находит оптимальные параметры и может застрять в локальном оптимуме, так и не найдя глобальный.

EM-алгоритм является так называемым «жадным» алгоритмом, смысл которого в принятии локально оптимальных решений на каждом этапе. А локальный максимум может сильно отличаться от глобального. Для этого в CEM-алгоритме (Classification EM) реализовано рандомизированное, но целенаправленное «встряхивание» выборки на каждой итерации. Это позволяет «выбить» оптимизационный процесс из локальных максимумов.

В CEM-алгоритме работает детерминированное правило, по которому объект приписывается одному кластеру, номер которого совпадает с номером наибольшего из чисел. В целом CEM работает относительно быстро, в сравнении с EM и, как правило, CEM находит экстремум, близкий к глобальному.

С целью сравнения результатов EM и CEM алгоритмов проведено исследование проверки значимости информационных признаков, предположительно имеющих экспоненциальное распределение для задачи выделения однородных партий электрорадиоизделий (подробно в Главе 4).

Исходными данными для анализа при решении задачи являются результаты тестовых воздействий на электрорадиоизделия по контролю вольт-амперных характеристик входных и выходных цепей микросхем.

Количество ошибок (ошибка - это неверно определенная партия) при запуске EM и CEM алгоритмов показано в Таблице 3.8. В знаменателе дроби - объем сборной партии.

Таблица 3.8 - Сравнение результатов EM и CEM алгоритмов (ошибок в среднем за 10 запусков)

Алгоритм	Микросхемы 1526ТЛ1 (4 признака, 1 – экспоненциальный) 3 партии	Микросхемы 1526ИЕ10 (6 признаков, 2 – экспоненциальные) 5 партий	Диоды 3ОТ122А (4 признака, 1 – экспоненциальный) 3 партии
EM	152/626	217/870	68/279
CEM	96/626	130/870	40/279

Как можно заметить CEM в среднем работает лучше EM на 10%. Экспериментально установлено, что основной EM-алгоритм сильнее неустойчив по начальным данным [107, 109, 182].

Для окончательного понятия различий между EM и CEM алгоритмами приведем описание EM-алгоритма (Алгоритм 1.3) с конкретным указанием изменений в нем при CEM-алгоритме (выделено курсивом).

Алгоритм 3.2 CEM-алгоритм (классификационный EM-алгоритм)

Дано: Выборка (массив) из N векторов d-мерных данных $X_i = \left(x_{i,1}, \dots, x_{i,d}\right)^T, i = \overline{1,N}$, предполагаемое число распределений в смеси k.

Шаг 1 (инициализация). Выбрать некоторые начальные значения параметров распределений. Как правило, в качестве векторов математических ожиданий μ выбираются значения случайно выбранных векторов данных, а значения дисперсий (или ковариационных матриц) устанавливаются одинаковыми для всех распределений и вычисляются для всей выборки, либо в качестве ковариационных матриц берутся единичные матрицы (аналогично, для

экспоненциальных распределений или распределений Лапласа параметр α рассчитывается по всей выборке $X_1,\ldots,X_N$).

Установить значения априорных вероятностей каждого из распределений равными для всех распределений $w_j = 1/k, j = \overline{1,k}$.

Шаг 2 (Е-шаг – классификация / разбиение на кластеры).

При нечеткой кластеризации для каждого распределения j и для каждого вектора данных i рассчитывается апостериорная вероятность того, что i-й вектор данных относится к j-му распределению: $g_{i,j} = \frac{f(x_i|j)w_j}{\sum_{l=1}^{k}(f(x_i|l)w_l)} \forall i = \overline{1,N}, j = \overline{1,k}$.

Здесь $f(x_i|j)$ – плотность j-го распределения в точке x_i.

При выполнении алгоритма CEM для каждого вектора данных все значения $g_{i,j}$ *для всех распределений устанавливаются равными 0, кроме одного распределения j', для которого* $\frac{f(x_i|j')w_{j'}}{\sum_{l=1}^{k}(f(x_i|l)w_l)}$ *имеет максимальное значение. Для этого распределения устанавливается значение* $g_{i,j'} = 1$. *Считаем, что векторы данных, для которых* $g_{i,j'} = 1$, *образуют j'-й кластер.*

Шаг 3 (М-шаг – модификация параметров распределений).

3.1. Пересчитать значения априорных вероятностей:

$$w_j = \frac{\sum_{i=1}^{N} g_{i,j}}{N} \forall j = \overline{1,k}.$$

3.2. Пересчитать оценки параметров каждого из распределений с учетом апостериорной вероятности того, что конкретный i-й вектор данных входит в j-й кластер с вероятностью $g_{i,j}$. Например, вектор средних значений $\mu_j = (\mu_{j,1}, \ldots, \mu_{j,d})$ для каждого кластера рассчитывается по формуле:

$$\mu_{j,l} = \frac{1}{\sum_{q=1}^{N} g_{q,j}} \sum_{i=1}^{N} x_{i,l} g_{i,j} = \frac{1}{N w_j} \sum_{i=1}^{N} x_{i,l} g_{i,j} \quad \forall j = \overline{1,d}, l = \overline{1,k}.$$

Аналогично, оценки среднеквадратичных отклонений рассчитываются так:

$$\sigma_{j,l}{}^2 = \frac{1}{\sum_{q=1}^{N} g_{q,j}} \sum_{i=1}^{N} (x_{i,l} - \mu_{j,l})^2 g_{i,j} = \frac{1}{N w_j} \sum_{i=1}^{N} (x_{i,l} - \mu_{j,l})^2 g_{i,j} \quad \forall j = \overline{1,d}, l = \overline{1,k}.$$

Здесь $\sigma_{j,l}$ – среднеквадратичное отклонение по l- му измерению в j-м распределении (кластере).

При выполнении алгоритма CEM вектор средних значений и среднеквадратичные отклонения рассчитываются для каждого кластера по отдельности (несложно убедиться, что две приведенные выше формулы подходят и для случая CEM, но расчет по каждому кластеру по отдельности – на практике быстрее и точнее).

При использовании стандартного многомерного распределения с полной ковариационной матрицей

$$\Sigma(j) = \begin{pmatrix} \sigma(j)_1^2 = \sigma_{1,1} & \sigma(j)_{1,2} & \dots & \sigma(j)_{1,d} \\ \sigma(j)_{2,1} & \sigma(j)_2^2 = \sigma(j)_{2,2} & \dots & \sigma(j)_{2,d} \\ \vdots & \vdots & \ddots & 0 \\ \sigma(j)_{d,1} & \sigma(j)_{d,2} & \dots & \sigma(j)_d^2 = \sigma(j)_{d,d} \end{pmatrix}$$

Ее элементы также рассчитываются с учетом апостериорных вероятностей:

$$\sigma(j)_{p,q} = \sigma(j)_{q,p} = \frac{1}{Nw_j}\sum_{i=1}^{N}\left(x_{i,p} - \mu_{j,p}\right)\left(x_{i,q} - \mu_{j,q}\right)g_{i,j}.$$

4. Вычислить значение целевой функции – логарифмической функции правдоподобия:

$$Q(w_1, \dots, w_1, \textit{параметры всех распределений}) = \sum_{i=1}^{N} ln\,(\sum_{j=1}^{k} w_j f(x_i|j))$$

5. Проверить условия останова, перейти к Шагу 2.

Применение различных вероятностных моделей в ЕМ-алгоритме для задачи разделения партий промышленной продукции на однородные партии исследовалось в [109]. Показано, что в случае многомерных данных модель с многомерными некоррелируемыми гауссовыми измерениями наиболее адекватна (дает наименьшее число ошибок при проверке на наборах данных с заранее маркированными данными) в сравнении с многомерными гауссовыми распределениями с полной ковариационной матрицей и в сравнении со сферическими гауссовыми распределениями.

Многомерное гауссово распределение с независимыми (некоррелированными) признаками (измерениями) отличается от многомерного гауссово распределения только тем, что не требует работы с матрицами.

Дано: *N* векторов *d*-мерных данных $X_i = \left(x_{i,1}, \ldots, x_{i,d}\right)^T \quad , i = \overline{1,N}$.

Существует вектор $\mu \in R^d$ и неотрицательно определенная симметричная ковариационная матрица:

$$\Sigma = \begin{pmatrix} \sigma_1^2 = \sigma_{1,1} & \sigma_{1,2} & \ldots & \sigma_{1,d} \\ \sigma_{2,1} & \sigma_2^2 = \sigma_{2,2} & \ldots & \sigma_{2,d} \\ \vdots & \vdots & \ddots & 0 \\ \sigma_{d,1} & \sigma_{d,2} & \ldots & \sigma_d^2 = \sigma_{d,d} \end{pmatrix} \text{p}$$

размерности $d \times d$, такие что плотность вероятности вектора *X* имеет вид:

$$f(X) = \frac{\alpha}{\sqrt{(2\pi)^d |\Sigma|}} exp\ (-\frac{1}{2}(X-\mu)^T \Sigma^{-1}(X-\mu)),$$

где | Σ |— определитель матрицы Σ, а Σ^{-1}— матрица обратная к Σ.Компоненты вектора μ вычисляются по отдельности для каждого измерения*:* $\mu_j = \frac{1}{N}\sum_{i=1}^{N} x_{i,j}$

Здесь $x_{i,j}$ – й компонент (*j*-е измерение) *i*-го вектора данных.

Ковариационная матрица диагональная (поэтому может быть заменена вектором), состоит из дисперсий по каждому измерению:

$$\Sigma = \begin{pmatrix} \sigma_1^2 & 0 & \ldots & 0 \\ 0 & \sigma_2^2 & \ldots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \ldots & \sigma_d^2 \end{pmatrix}.$$

Дисперсии рассчитываются независимо*:* $\sigma_j^2 = \frac{1}{N}\sum_{i=1}^{N} \quad \left(x_{i,j} - \mu_j\right)^2$.

Плотность распределения вычисляется как произведение плотностей по каждому измерению:

$$f(X) = \prod_{j=1}^{d} f_j(x_j) = \prod_{j=1}^{d} \frac{1}{\sigma_j \sqrt{(2\pi)}} exp\ (-\frac{1}{2}\left(x_j - \mu_j\right)^2 / \sigma_j^2),$$

где – *j*-й компонент вектора *X*.

CEM-алгоритм, как модификация EM-алгоритма вполне успешно может использоваться в качестве метода локального поиска [107, 109, 112, 183, 184]. В

сравнении со случайно выбранными решениями, решения образованные из элементов различных решений, являющихся локальными оптимумами, с большей вероятностью окажутся ближе к глобальному оптимуму [110]. Поэтому предложено в данном случае также использовать VNS-алгоритм в качестве расширенного локального поиска [150, 183, 184].

Таким образом, усовершенствованный алгоритм на основе классификационного EM-алгоритма (Алгоритм 2.9) с применением поиска с чередующимися рандомизированными окрестностями будет выглядеть следующим образом [183, 184]:

Алгоритм 3.3 CEM-GH-VNS

1: Получить решение *S*, запустив CEM-алгоритм из случайным образом сгенерированного начального решения.

2: $O=O_{start}$ **(**номер окрестности поиска).

3: $i=0$, $j=0$**.**

пока $j < j_{\max}$

пока $i < i_{\max}$

4: **если** не выполняются условия ОСТАНОВа, **то** получить решение *S'*, запустив CEM-алгоритм из случайного начального решения.

повторять

5: В соответствии со значенем переменной *S* (возможны значения 1, 2 или 3), запустить Алгоритм Жадная процедура 1, 2 или 3 соответственно с начальными решениями *S* и *S'*. Так, окрестность определяется способом включения центров кластеров из второго известного решения и параметром окрестности – вторым известным решением.

если новое решение лучше, чем *S*, **то**

записать новый результат в *S*, $i=0$, $j=0$.

иначе выйти из цикла.

конец цикла

6: $i=i+1$.

конец цикла

7: $i=0$, $j=j+1$, $O=O+1$, **если** $O>3$, **то** $O=1$.

конец цикла

В качестве тестовых наборов данных были использованы (Таблица 3.9) результаты неразрушающих тестовых испытаний сборных производственных партий электрорадиоизделий, проведенных в специализированном тестовом центре для комплектации бортовой аппаратуры космических аппаратов.

Для экспериментов использовалась вычислительная система DEXP OEM (4-ядерное ЦПУ Intel® Core™ i5-7400 CPU 3.00 ГГц, 8 Гб ОЗУ).

Таблица 3.9 - Результаты вычислительных экспериментов по наборам данных тестовых испытаний партии изделий промышленной продукции (10 кластеров, 2 минуты, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Среднеквадратичное отклонение
3ОТ122А (767 векторов данных, каждый размерностью 13)				
CEM	120 947,6	146 428,5	135 777,6	***7 985,6992***
CEM-GH-VNS1	121 256,5	152 729,1	***143 956,0***	8 708,6293
CEM-GH-VNS2	123 664,4	158 759,2	***143 028,5***	10 294,3992
CEM-GH-VNS3	***128 282,2***	155 761,9	***143 506,9***	10 058,8266
1526TL1 (1234 векторов данных, каждый размерностью 157)				
CEM	354 007,3	416 538,4	384 883,4	***20 792,8068***
CEM-GH-VNS1	376 137,1	477 124,5	438 109,4	29 964,0641
CEM-GH-VNS2	345 072,6	487 498,3	444 378,1	43 575,3282
CEM-GH-VNS3	***379 352,3***	516 777,8	***456 271,4***	38 323,0246
5514ВС1Т2-9А5 (91 векторов данных, каждый размерностью 173)				
CEM	4 504,1	7 284,2	5 776,4	987,9598
CEM-GH-VNS1	3 977,6	9 620,5	***6 981,3***	1 990,3690
CEM-GH-VNS2	***4 528,9***	13 545,5	6 342,4	2 632,7929
CEM-GH-VNS3	4 415,6	7 112,9	5 966,3	***904,9495***

Для всех наборов данных было выполнено по 30 попыток запуска каждого из алгоритмов. Фиксировались только лучшие результаты, достигнутые в каждой

попытке, затем из этих результатов по каждому алгоритму были рассчитаны значения целевой функции: минимальное значение (Min), среднее значение (Среднее) и среднеквадратичное отклонение. Лучшие значения целевой функции (минимальное значение, среднее значение и среднеквадратичное отклонение) выделены полужирным курсивом. Графическая реализация сходимости алгоритмов построенных по среднему значению целевой функции представлена на Рисунках 3.7-3.9.

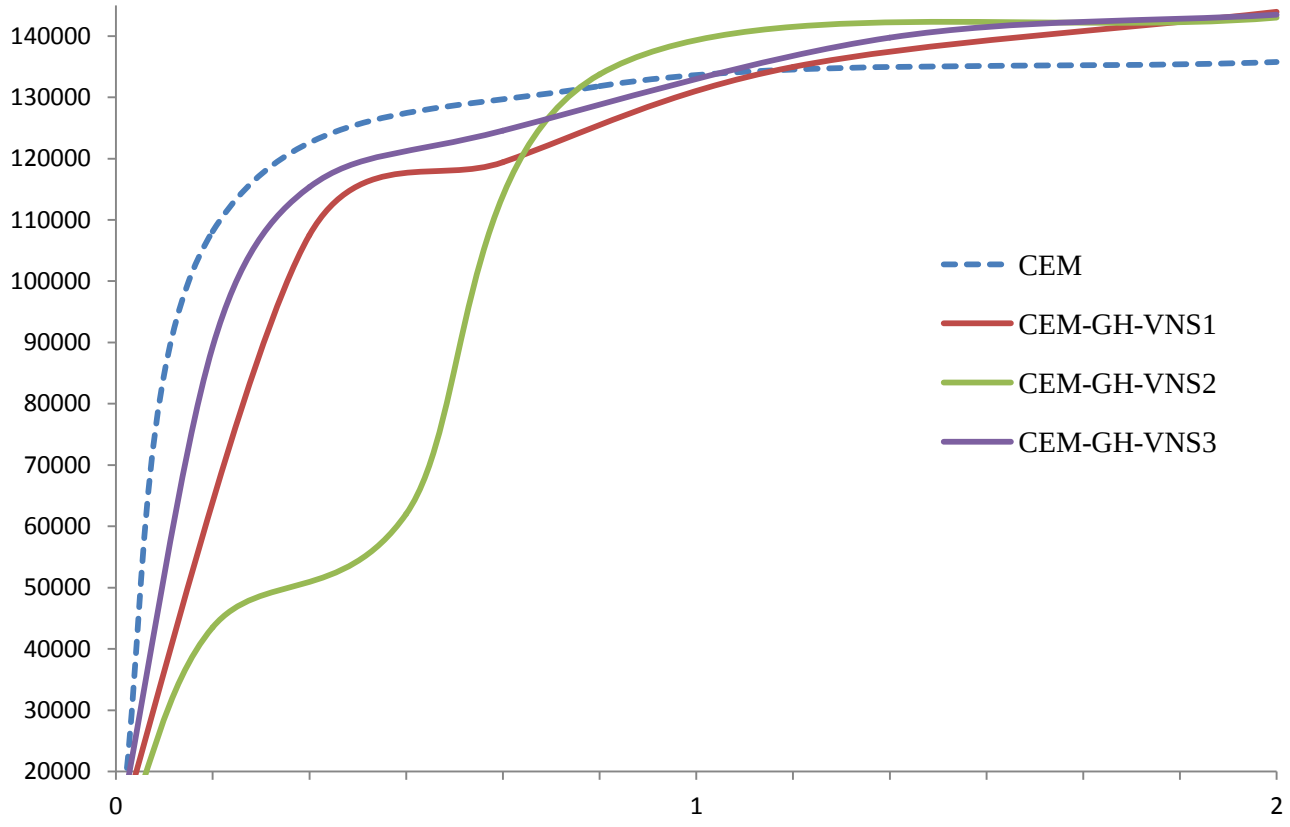


Рисунок 3.7 - Сравнение новых и известных алгоритмов для набора данных 3OT122A (10 кластеров, 2 минуты)
по оси абсцисс – время в минутах, по оси ординат – достигнутое среднее значение целевой функции

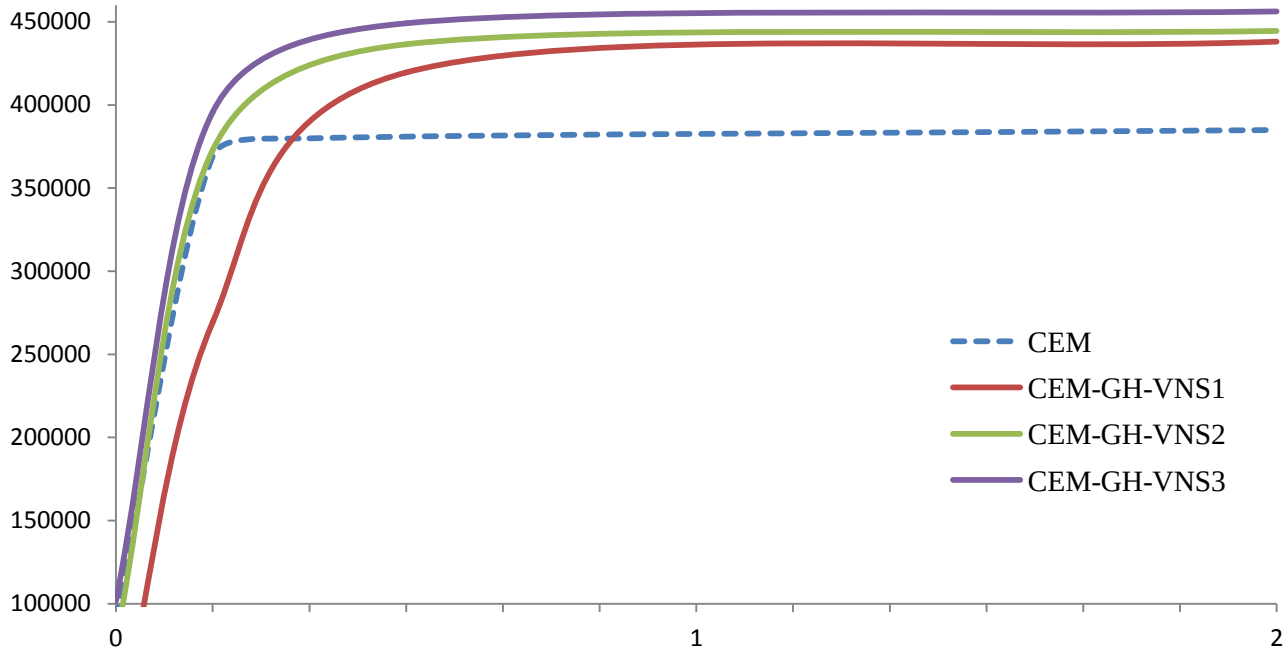


Рисунок 3.8 - Сравнение новых и известных алгоритмов для набора данных 1526TL1 (10 кластеров, 2 минуты)
по оси абсцисс – время в минутах, по оси ординат – достигнутое среднее значение целевой функции

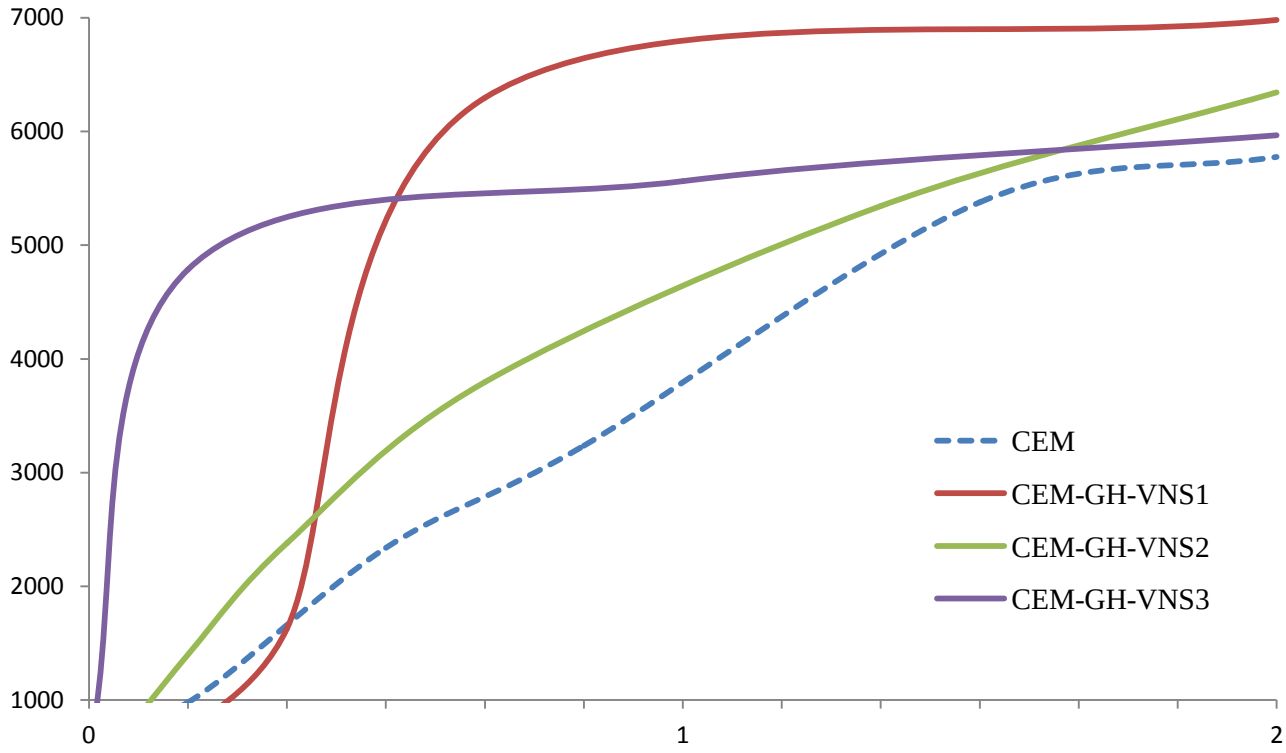


Рисунок 3.9 - Сравнение новых и известных алгоритмов для набора данных 5514BC1T2-9A5 (10 кластеров, 2 минуты)
по оси абсцисс – время в минутах, по оси ординат – достигнутое среднее значение целевой функции

Как показывают вычислительные эксперименты [148, 182-185], стабильность результатов при многократных запусках CEM-алгоритма выше (среднеквадратичное отклонение целевой функции меньше), чем у новых алгоритмов, в то же время результат во многих случаях далек от истинного оптимума функции правдоподобия. Об имеющихся возможностях улучшения результатов говорит то, что при многократном проведении вычислительных экспериментов результаты лучших попыток запуска CEM-алгоритма иногда отличаются на десятки процентов по значению целевой функции правдоподобия от усредненных значений всего множества попыток. Поэтому новые алгоритмы поиска с чередующимися рандомизированными окрестностями (CEM-GH-VNS) имеют в сравнении с классическим CEM-алгоритмом преимущество по среднему достигаемому значению целевой функции при многократных запусках.

3.4 Подход к разработке алгоритмов автоматической группировки, основанных на параметрических оптимизационных моделях

Таким образом, были рассмотрены комбинации алгоритмов метода жадных эвристик с чередующимися окрестностями для задач k-средних, k-медоид, и известных алгоритмов j-means и CEM.

Блок-схема нового подхода к разработке алгоритмов автоматической группировки, основанных на параметрических оптимизационных моделях, с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных агломеративных эвристических процедур представлена на Рисунке 3.10.

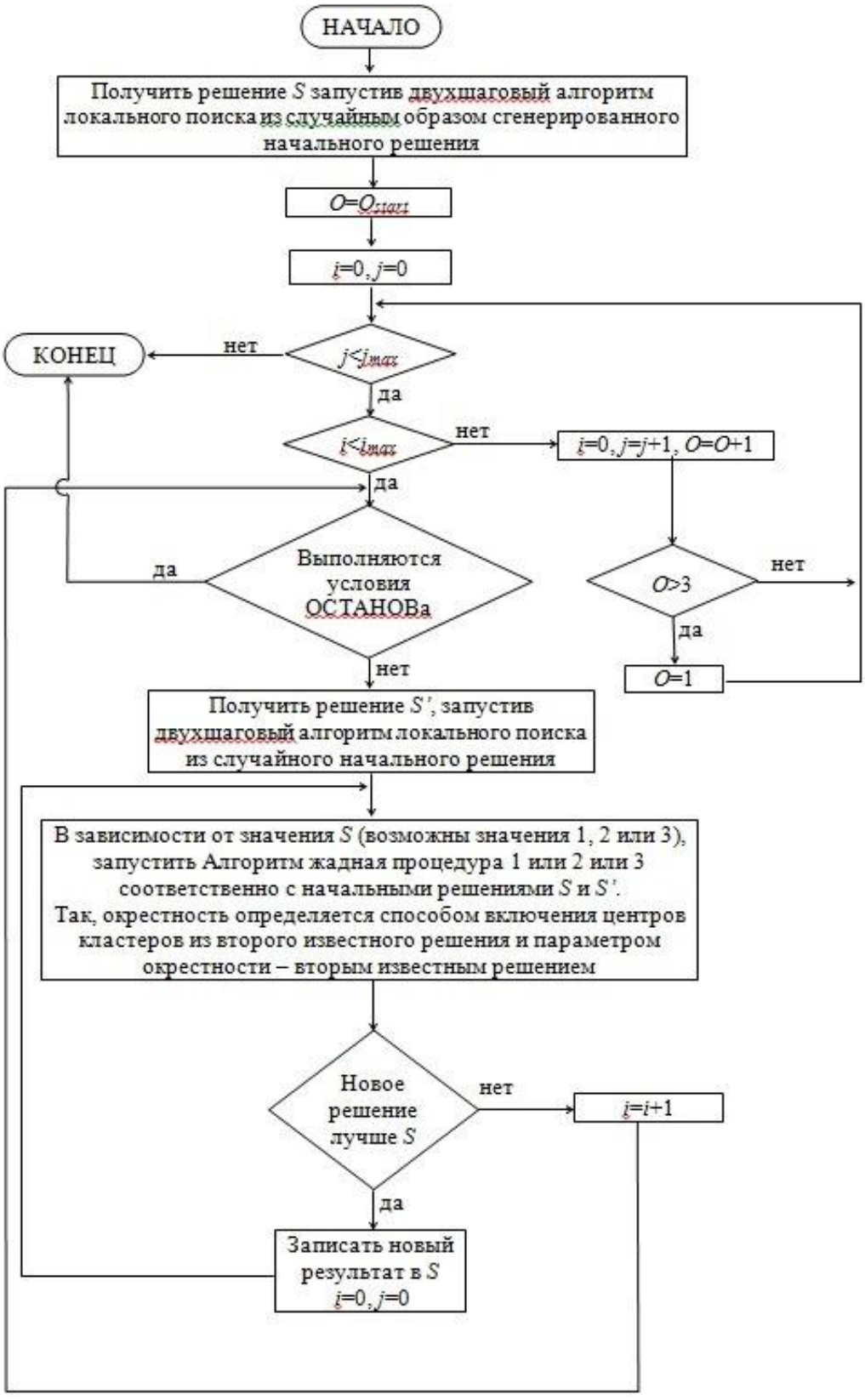


Рисунок 3.10 - Общая схема подхода к разработке алгоритмов автоматической группировки, основанных на параметрических оптимизационных моделях, с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных агломеративных эвристических процедур

Общую схему предлагаемого нового подхода к разработке алгоритмов автоматической группировки, основанных на параметрических оптимизационных моделях, с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных агломеративных эвристических процедур можно описать следующим образом:

Алгоритм 3.4 GH-VNS (Greedy Heuristic in the Variable Neighborhood Search)

1: Получить решение S, запустив двухшаговый алгоритм локального поиска из случайным образом сгенерированного начального решения.

2: $O=O_{start}$ **(**номер окрестности поиска).

3: $i=0$, $j=0$ (количество безрезультатных итераций в конкретной окрестности и в целом по алгоритму).

пока $j < j_{\max}$

пока $i < i_{\max}$

4: **если** не выполняются условия ОСТАНОВа (превышение лимита времени), **то** получить решение S', запустив двухшаговый алгоритм локального поиска из случайного начального решения.

повторять

5: В зависимости от значения S (возможны значения 1, 2 или 3), запустить Алгоритм Жадная процедура 1 или 2 или 3 соответственно с начальными решениями S и S'. Так, окрестность определяется способом включения центров кластеров из второго известного решения и параметром окрестности – вторым известным решением.

если новое решение лучше, чем S, **то**

записать новый результат в S, $i=0$, $j=0$.

иначе выйти из цикла.

конец цикла

6: $i=i+1$.

конец цикла

7: i=0, j=j+1, O=O+1, **если** O>3, **то** O=1.

конец цикла

В зависимости от значения O_{start} алгоритмы в настоящей диссертации обозначены GH-VNS1, GH-VNS2, GH-VNS3 (для задачи k-средних соответственно - k-GH-VNS1, k-GH-VNS2, k-GH-VNS3; для решения задачи p-медоид: PAM-GH-VNS1, PAM-GH-VNS2, PAM-GH-VNS3; для решения задач с применением CEM-алгоритма: CEM-GH-VNS1, CEM-GH-VNS2, CEM-GH-VNS3).

Результаты Главы 3

Результаты вычислительных экспериментов показали, что новые алгоритмы метода жадных эвристик для задач автоматической группировки с повышенными требованиями к точности результата (по значению целевой функции), с применением алгоритмов поиска с чередующимися рандомизированными окрестностями (GH-VNS) имеют более стабильные (меньшее среднеквадратичное отклонение целевой функции) и более точные (меньшее среднее значение целевой функции) результаты, и следовательно, лучшие показатели в сравнении с классическими алгоритмами (k-средних, j-means, PAM и CEM).

В то же время, с ростом числа кластеров и объема выборки сравнительная эффективность нового подхода, основанного на параметрических оптимизационных моделях, с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных агломеративных эвристических процедур, повышается, и для больших наборов данных новые алгоритмы имеют преимущество при фиксированном времени работы алгоритма.

Однако стоит отметить, что при значительном увеличении времени расчетов известные генетические алгоритмы метода жадных эвристик показывают хоть и незначительно, но лучшие результаты в сравнении с предложенными новыми алгоритмами. Тем не менее, можно говорить о

конкурентоспособности новых алгоритмов как в сравнении с классическими алгоритмами k-средних, PAM и j-means, так и с генетическими алгоритмами, включая алгоритмы метода жадных эвристик, а также с детерминированными алгоритмами.

На Рисунке 3.11 изображена структурная схема метода жадных эвристик с добавленными новыми компонентами, разработанными в рамках данного диссертационного исследования. Порядок расположения компонентов по вертикали отражает вложенность выполнения алгоритмов.

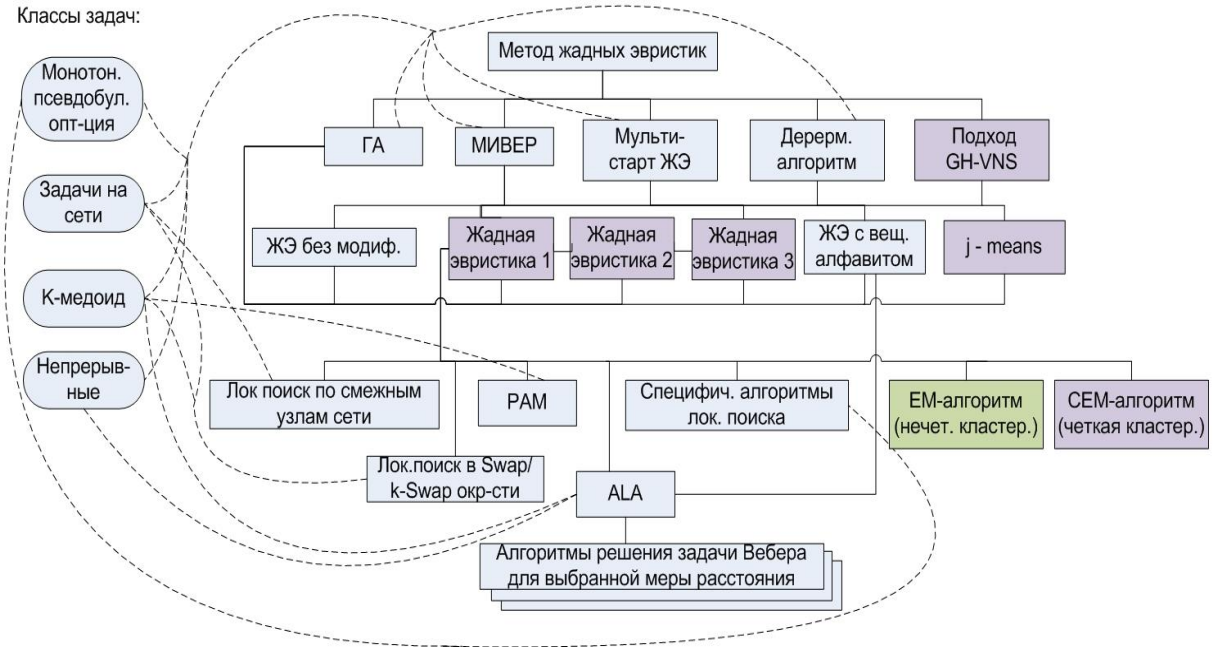


Рисунок 3.11 - Компоненты метода жадных эвристик, их взаимная совместимость (сплошные линии) и применимость к классам задач (курсивные линии). Сиреневым цветом выделены новые компоненты.

Новый подход к разработке алгоритмов автоматической группировки, основанных на параметрических оптимизационных моделях, с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных агломеративных эвристических процедур был использован в деятельности АО «РУСАЛ Саяногорск» и АО «Испытательный технический центр - НПО ПМ» (Приложение Б).

В настоящее время в кластерном анализе проявляется тенденция к применению коллективных методов [186]. Алгоритмы кластерного анализа не

являются универсальными: каждый алгоритм имеет свою особую область применения. В том случае, если рассматриваемая область содержит различные типы наборов данных, для выделения кластеров приходится применять не один определенный алгоритм, а набор различных алгоритмов. Ансамблевый (коллективный) подход позволяет снизить зависимость конечного решения от выбранных параметров исходных алгоритмов и получить более устойчивое решение [187-190]. В Главе 4 рассмотрим ансамбли алгоритмов автоматической группировки.

ГЛАВА 4. ПРИМЕНЕНИЕ МЕТОДА ЖАДНЫХ ЭВРИСТИК В ЗАДАЧАХ АВТОМАТИЧЕСКОЙ ГРУППИРОВКИ ПРОМЫШЛЕННОЙ ПРОДУКЦИИ

Глава посвящена описанию задачи выделения однородных партий для формирования электронной компонентной базы космического применения (как примеру актуальной задачи автоматической группировки с повышенными требованиями к точности и стабильности результата) и разработке процедуры составления оптимальных ансамблей алгоритмов автоматической группировки с комбинированным применением генетического алгоритма метода жадных эвристик и согласованной матрицы бинарных разбиений, позволяющей повысить точность разделения на однородные партии продукции для практических задач автоматической группировки промышленной продукции с применением изложенного в главах 2 и 3 подхода к разработке алгоритмов автоматической группировки.

4.1 Постановка задачи выделения однородных партий промышленной продукции

Одной из важнейших составляющих задачи повышения надежности системы в целом является комплектация критически важных узлов системы компонентной базой с повышенными требованиями к качеству (например, в случае электронной аппаратуры). Для обеспечения согласованной работы однотипных элементов системы важно, чтобы они имели очень близкие характеристики (были однородны). Однородность характеристик одинаковых элементов системы достигается в случае, если эти элементы были изготовлены из одной партии сырья в одной производственной партии. Поэтому при комплектации критически важных узлов системы с повышенными требованиями качества и надежности необходимо использовать и соответствующие компоненты, изготовленные отдельными «специальными» партиями, к которым предъявляются повышенные требования качества.

Решение задач автоматической группировки с повышенными требованиями к точности и стабильности результата являются актуальными, что обусловлено широким диапазоном их применения, как в задачах кластерного анализа, так и непосредственно в практических задачах на производстве где требуется обеспечение высокой точности воспроизводимости результата (например, для задачи разделения на однородные партии промышленной продукции с особыми требованиями качества).

В Главе 1 был рассмотрен пример актуальной задачи автоматической группировки с повышенными требованиями к точности и стабильности результата. Рассмотрим его более подробно.

Вопросы прогнозирования отказов, связанных с возникновением дефектов на этапе производства электрорадиоизделий (ЭРИ) подробно были рассмотрены и в российской и зарубежной литературе [191-199]. Все проводимые работы по предотвращению отказов в основном направлены на выявление и устранение дефектов непосредственно на этапе изготовления электрорадиоизделий [11, 200, 201]. В то же время как в примере с электрорадиоизделиями, так и в других областях важной задачей является разработка методов контроля качества уже выпущенных партий поступившей промышленной продукции (с повышенными требованиями качества) на предмет соответствия заявленных и фактических характеристик по результатам тестовых испытаний и прогнозирование отказоустойчивости, в том числе и с применением ретроспективного анализа о ранее выпущенных партиях. Особенно контроль необходим при использовании промышленной продукции зарубежного производства, когда непосредственно проконтролировать продукцию на этапе выпуска невозможно.

Отметим, что поставляемые промышленные партии ЭРИ [202] могут быть неоднородными (состоящими из нескольких производственных партий пластин), например, интегральные схемы одного наименования, но разной категории качества («ОС», «ВП», «V(S)», «Q(B)») [203, 204]. Поэтому для того чтобы распространить результаты испытаний на всю производственную партию изделий необходимо быть уверенными в том, что мы имеем дело с партией изделий,

изготовленной из единой (однородной) партии сырья или, что разброс параметров будет в пределах допустимой нормы. Поэтому выявление однородных производственных партий из сборных партий изделий является одним из важнейших мероприятий при проведении испытаний с целью недопущения ошибок в оценке качества, что напрямую влияет на срок функционирования бортовой аппаратуры космического аппарата.

Для того чтобы принять обоснованное решение о приемлемости качества изделий, необходимо провести предусмотренные разрушающие испытания для каждой производственной партии, состоящей из нескольких различных групп (партий). Для этого необходимо провести исследования по выявлению таких групп [117, 205, 206]. В случае с ЭРИ для оценки качества компонентной базы было предложено следующее формирование выборок:

1) электрорадиоизделия изготовлены из одной кристальной партии пластин – тогда одна выборка;

2) электрорадиоизделия изготовлены более чем из одной кристальной партии пластин - тогда необходимо определение количества однородных групп, которые и будут равняться количеству выборок.

Отметим, что выборки формируются после прохождения промышленной продукцией дополнительных отбраковочных испытаний и разрушающего физического анализа, в результате чего отсеиваются изделия с потенциальными дефектами.

Таким образом, автоматическая группировка производственных партий электрорадиоизделий важна для обеспечения надежности, а в особенности радиационной стойкости [202], что во многом определяет срок активного существования космических аппаратов [207, 208].

Поэтому взаимодействие с заводами-изготовителями по производству специальных партий электрорадиоизделий специально для космической отрасли («спецпартий») является актуальной задачей [189]. Характеристики входящих в спецпартию изделий должны быть лучше изделий обычной партии ЭКБ (даже категорий качества «ВП» или «ОС»), как и характеристики всей совокупности

входящих в спецпартию изделий должны отличаться в лучшую сторону. Таким образом, специальная партия является фактически прообразом компонентной базы космического уровня качества.

Как выяснилось [209], электрорадиоизделия зарубежного производства категорий качества «Space» и «Military» имеют два отличия: контроль наличия посторонних частиц в подкорпусном пространстве и оценку дрейфа параметров при электротермотренировке изделий.

При отсутствии производства спецпартий изделий с повышенными требованиями качества возможно единственным выходом представляется их формирование в специализированных тестовых центрах с применением методов кластерного анализа с повышенными требованиями к точности и стабильности результата (воспроизводимости результата разделения на однородные партии промышленной продукции) [210].

Ситуация с разделением на однородные партии промышленной продукции в системе управления технологическим процессом производства анодов в чем то аналогична описанному выше процессу разделения производственных партий электрорадиоизделий, но имеет свою специфику.

При производстве алюминия одним из ключевых материалов является обожженный анод [211, 212]. Минимальное изменение параметров анода влечет значительные колебания технико-экономических параметров процесса электролиза (доля обожженного анода в себестоимости при производстве алюминия составляет порядка 15 процентов) [213]. Сырье для производства анодов отличается самым большим разбросом параметров свойств, определяющих качество продукции. Низкокачественные аноды не только приводят к увеличению затрат на производство алюминия (доходящих до 170 долларов на тонну), но также к увеличению выбросов парниковых газов. Следовательно, совершенствование производственного процесса производства анодов дает большие экономические перспективы для предприятия [213, 214].

Применяемая на конкретном предприятии система контроля качества обожженных анодов будет считаться эффективной только в том случае, если она

достаточно полно будет моделировать работу анода в алюминиевом электролизере (достаточно чувствительна к изменению свойств анода). Чем полнее и качественнее выполнен анализ параметров сырья, тем более надежен получаемый результат.

В анодах, к примеру, могут быть дефекты структуры (например, трещины) образовавшиеся ещё на стадии формования «зеленых» блоков или при плохих условиях обжига. Зеленый анод (англ. Green Anode) - анод алюминиевой электролитической ванны, не подвергшийся обжигу. Поскольку аноды подвергаются сильной тепловой атаке в электролизерах, очень большое значение имеет их стойкость к образованию трещин. Отказ анода в электролизере из-за появления трещин приводит к серьезным нежелательным побочным эффектам, в результате которых возможны серьезные убытки. Поэтому задача разделения партий зеленых анодов до процесса обжига имеет одно из важнейших значений при производстве алюминия [211, 212, 214].

Повысить точность методов автоматической группировки с повышенными требованиями к точности и стабильности результата позволяют предложенные в главах 2 и 3 алгоритмы, которые могут стать основой автоматизированной системы по выявлению различных по параметрам групп любых промышленных изделий.

4.2 Применение алгоритмов поиска с чередующимися окрестностями для промышленной продукции с повышенными требованиями к качеству

Концептуальная схема системы разделения сборных партий промышленной продукции (на примере электрорадиоизделий космического применения) по результатам тестовых испытаний проводимых в АО «Испытательный технический центр - НПО ПМ» (АО «ИТЦ - НПО ПМ») приведена на Рисунке 4.1 [210].

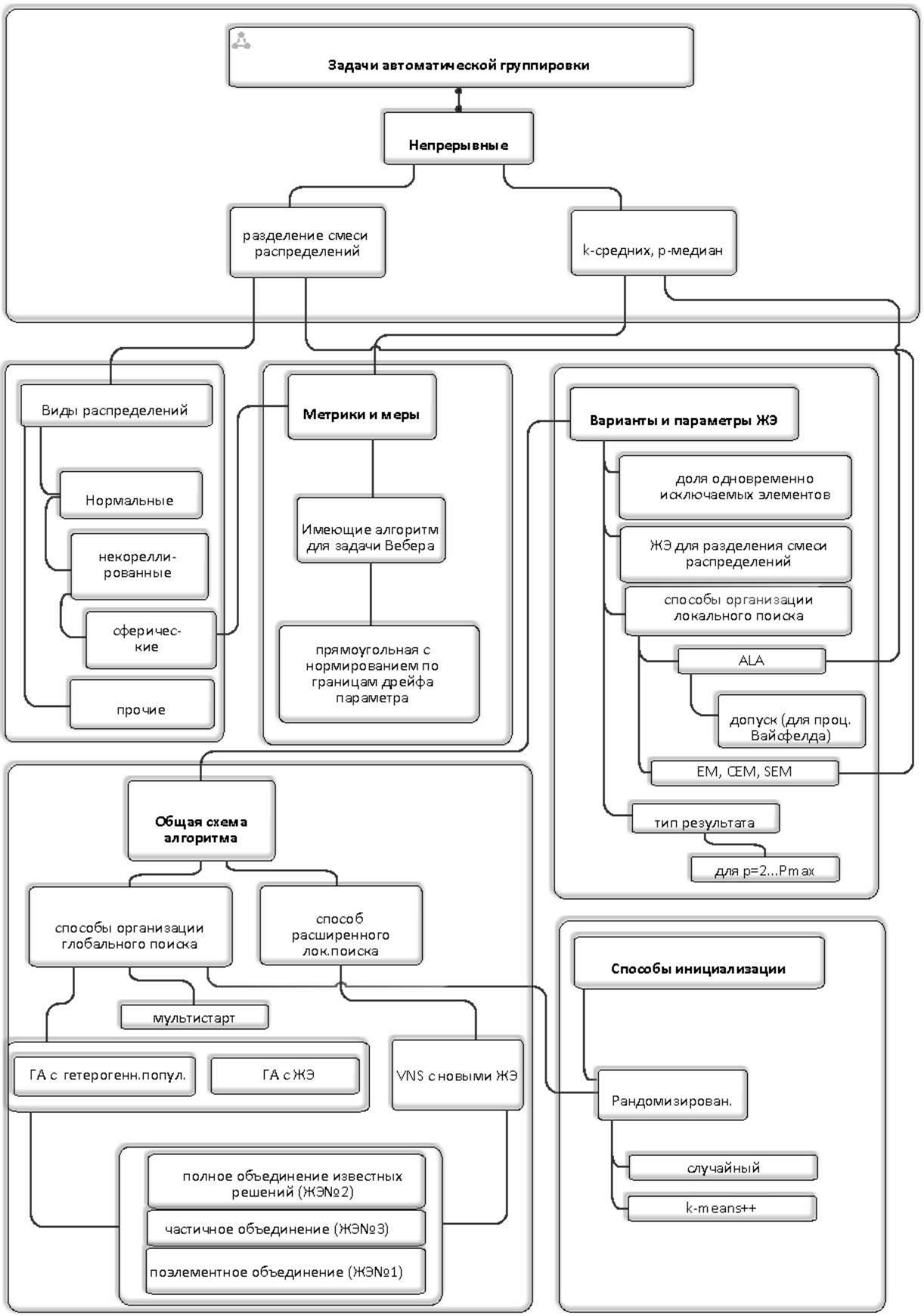


Рисунок 4.1 - Концептуальная схема системы разделения сборных партий промышленной продукции с повышенными требованиями к качеству по результатам тестовых испытаний [109]

На схеме показаны задачи, модели, алгоритмы и их возможные взаимосвязи, которые могут быть задействованы при построении эффективной системы автоматической группировки электрорадиоизделий по однородным производственным партиям [109, 142].

Ранее [215] было показано, что задача выделения однородных партий промышленной продукции может быть сведена к задаче кластерного анализа, где каждая группа (кластер) будет представлять однородную партию, изготовленную из одного вида сырья. Для решения задачи выявления однородных партий в работах [216-218] было предложено применение алгоритма кластеризации k-средних. В [219] рассмотрен метод нечеткой кластеризации на основе EM-алгоритма. Предложена модель разделения однородных производственных партий на основе смеси сферических или некоррелированных гауссовых распределений [220]. В [215] рассмотрено применение генетических алгоритмов с жадной эвристикой, а также модификаций EM-алгоритма для разделения однородных партий изделий.

Исходные данные о результатах испытаний промышленной продукции представляют собой многомерный набор параметров изделий, измеренных по результатам проведения нескольких сотен неразрушающих тестов [221]. В целях понижения размерности входных данных для кластеризации изделий по однородным партиям предпринимались попытки применения методов факторного анализа [222, 223, 224].

Было показано, что количество выделяемых факторов зависит от количества рассматриваемых изделий в выборке, а также от входных измеряемых параметров изделия в данной выборке [222, 223]. Однако, так и не удалось выделить оптимальный универсальный набор факторов для разделения сборной партии, состоящей из произвольного числа однородных партий. Таким образом, несмотря на то, что методы факторного анализа позволяют несколько сократить размерность данных, все же использование массива данных достаточно большой размерности является необходимым при применении методов кластерного анализа для разделения сборной партии (данные остаются многомерными).

Одной из проблем в кластеризации данных является автоматическое определение числа кластеров (групп). В большинстве случаев задача определения количества кластеров сводятся к проблеме выбора модели. Как правило, алгоритмы автоматической группировки запускаются в некотором допустимом пределе числа возможных групп, а наилучшее значение (число кластеров) выбирается на основании критерия компактности.

В кластерном анализе существуют следующие основные критерии определения числа кластеров: индекс Калински-Харабаша [225], индекс Дэвиса-Боулдина (DBI) [226], индекс Кржановски-Лая [227], критерий Хартигана [228], информационный критерий Байеса (BIC – Bayesian Information Criterion) [229], GAP-критерий [230], информационный критерий Акаике (AIC) [231], критерий силуэта [232].

Были проведены эксперименты по применению каждого из перечисленных критериев для промышленной продукции на примере результатов неразрушающих тестовых испытаний сборных производственных партий изделий [109, 142]. Наиболее информативным и при этом не требующим подстройки значений каких-либо параметров оказался критерий силуэта. Фактически в производстве специальных партий ЭРИ задействован исключительно критерий силуэта, хотя в программной реализации алгоритма автоматической группировки электрорадиоизделий по производственным партиям применена оценка результатов по критериям внутрикластерного расстояния, силуэта и Байесовского информационного критерия.

Применение критерия силуэта дало наименьшее число ошибок при определении числа производственных партий [142]. Критерий силуэта также служит для подтверждения результатов автоматической группировки, как по партиям, так и по каждому изделию.

Таким образом, применение критерия силуэта позволяет эффективно определять число производственных партий в сборной партии, что, в свою очередь, позволяет повысить эффективность алгоритма автоматической

группировки и более точно провести разделение на однородные партии электрорадиоизделий.

Модели автоматической группировки не являются универсальными: каждый алгоритм имеет свою область применения. В том случае, если рассматриваемая область содержит различные типы наборов данных, для выделения кластеров приходится применять не один определенный алгоритм, а набор различных алгоритмов, так как заранее неизвестно какой из алгоритмов покажет лучший результат на конкретном наборе данных (или партии промышленной продукции). Это видно и в данном исследовании, когда алгоритмы в рамках одного подхода демонстрировали различные результаты при различных задачах. Ансамблевый (коллективный) подход позволяет снизить зависимость конечного решения от выбранных параметров исходных алгоритмов и получить более устойчивое (по воспроизводимости результата) решение [187-190].

4.3 Ансамбли алгоритмов автоматической группировки

Для интеллектуального анализа данных, включая задачи автоматической группировки, предложено множество статистических и иных методов, но по-прежнему важной задачей остается разработка технологии (метода), подходящей для решения максимально широкого круга задач кластеризации [190, 233]. Например, после проведенных многократных исследований, применение ансамблей алгоритмов кластеризации позволяет сделать вывод об их сравнительной эффективности для решения широкого круга задач [190]. Но тогда возникает вопрос о методе формирования ансамбля. Как показывает практика, формирование эффективных ансамблей сопряжено с трудностями, так как выбор для формирования ансамбля алгоритмов, демонстрирующих лучшие результаты, не всегда приводит к формированию ансамбля дающего наилучшую точность [189, 234, 235].

Существуют две основные методики получения ансамбля алгоритмов [187, 236]:

1. Вычисление согласованной матрицы сходства/различий (co-occurrence matrix).

2. Нахождение консенсусного разбиения, то есть согласованного разбиения при имеющихся нескольких решениях, оптимального по некоторому критерию.

При формировании окончательного решения используются результаты, полученные различными алгоритмами автоматической группировки.

Рассмотрим пример ансамбля алгоритмов [237, 238]. Он представляет собой сочетание последовательных алгоритмов k-средних (каждый из которых предлагает свое разбиение) и иерархического агломеративного алгоритма, объединяющего полученные решения с помощью особого механизма.

На первом шаге каждый алгоритм, используя свою метрику расстояния, разбивает данные на кластеры. Далее, рассчитывается точность и вес мнения алгоритма в ансамбле по формуле:

$$W_i = \frac{Acc_i}{\sum_{i=1}^{L} Acc_i}, \qquad (3.1)$$

где Acc_i – точность алгоритма i, то есть отношение количества правильно кластеризованных объектов к объёму всей выборки, а L – количество алгоритмов в ансамбле.

Для каждого полученного разбиения составляется предварительная бинарная матрица различий размера n x n (где n - количество объектов) необходимая для определения занесения объектов разбиения в один класс. Далее рассчитывается согласованная матрица различий, каждый элемент которой представляет собой взвешенную сумму элементов предварительных матриц (с использованием веса из формулы 3.1). Полученная таким образом матрица используется в качестве входных данных для алгоритма иерархической агломеративной кластеризации. После этого с помощью обычных приемов (таких как определение скачка расстояния агломерации) можно выбрать наиболее подходящее кластерное решение [237, 238].

Как было сказано выше для получения наилучшего разбиения на кластеры необходимо составить бинарную матрицу сходства/различий на каждое L разбиение в ансамбле:

$$H_i=<h_i(i,j)>,$$

где $h_i(i,j)$ равен нулю, если элемент i и элемент j попали в один кластер, и 1, если нет.

Следующим шагом в составлении ансамбля алгоритмов автоматической группировки является составление согласованной матрицы бинарных разбиений:

$$H^* = \left\langle h^*(i,j) \right\rangle, \qquad h^*(i,j) = \sum w_i h_i(i,j),$$

где w_i – вес алгоритма. Мы принимаем вес, равный усредненной точности алгоритма, примененного на тестовых задачах.

Генетические алгоритмы показало высокую эффективность при построении ансамблей нейронных сетей [83, 239-243] применяемых, в том числе, для решения задач автоматической группировки. Мы применили генетический алгоритм метода жадных эвристик [150, 190] для формирования ансамбля произвольных алгоритмов. Выбор данного метода обусловлен тем, что алгоритмами данного метода для практических задач получаются результаты, которые трудно существенно улучшить другими методами за сопоставимое время. Кроме того вычислительные эксперименты показывают хорошие результаты (по значению целевой функции и стабильности этих значений) для задач автоматической группировки большого количества объектов (сотни тысяч) и векторами данных большой размерности.

Точность отдельных алгоритмов кластеризации и их ансамблей можно оценить по имеющейся размеченной выборке - то есть требуется выборка, в которой принадлежность объектов к фактическим группам известна заранее.

Точность алгоритмов и их ансамблей будем оценивать следующим образом:

$$Fit^1 = A/N \to \max, \qquad (3.2)$$

где A – количество правильно кластеризованных объектов; N – общее количество объектов.

Общую схему предлагаемой процедуры составления оптимальных ансамблей алгоритмов автоматической группировки с комбинированным применением генетического алгоритма метода жадных эвристик и согласованной матрицы бинарных разбиений для практических задач можно описать следующим образом [189, 190]. Алгоритмы представлены результатами их работы на m тестовых задачах, то есть матрицами бинарных разбиений. Маркированные данные используются на этапе составления ансамбля алгоритмов автоматической группировки, потом расчеты идут непосредственно на сборной промышленной партии продукции (которую необходимо разделить на однородные партии), используя ансамбль отобранных в каждой модели лучших алгоритмов.

Алгоритм 4.1 Процедура составления оптимальных ансамблей алгоритмов автоматической группировки с комбинированным применением генетического алгоритма метода жадных эвристик и согласованной матрицы бинарных разбиений для практических задач

Дано: набор m тестовых задач с маркированными данными (фактическая разбивка данных на группы заранее известна), набор n алгоритмов кластеризации C_i, размер популяции q, количество алгоритмов в ансамбле p.

Решения («особи») в алгоритме - подмножества S выбранных для составления ансамбля алгоритмов кластеризации мощности p.

Шаг 1. Сгенерировать случайным образом q начальных решений – «особей» алгоритма.

Шаг 2. Для каждой особи оценить значение критерия (3.2), усреднённого по m задачам, применив к каждой задаче ансамбль, представленный «особью» - множеством алгоритмов. Сохранить значение усредненного критерия в переменной Fit_j, где j – номер «особи».

Шаг 3. Проверить условия останова (превышение лимита времени), ОСТАНОВ при достижении условий.

Шаг 4. Выбрать случайным образом с равной вероятностью два номера «особей» i, j. Составить ансамбль: $S = S_i \cup S_j$.

Шаг 5. Пока $|S| > p$ выполнять:

Шаг 5.1. Для каждого *i*: $C_i \in S$ выполнять:

Шаг 5.1.1. Исключить *i*-й алгоритм из ансамбля *S*: $S' = S \setminus C_i$.

Шаг 5.1.2. Для *S'* оценить значение критерия (3.2), усреднённого по *m* задачам, применив к каждой задаче ансамбль *S'*. Сохранить значение усреднённого критерия в переменной Fit'_i.

Шаг 5.1.3. Перейти к следующей итерации цикла 5.1.

Шаг 5.2. Удалить из *S* алгоритм C_i, которому соответствует наименьшее значение Fit'_i. $S' = S \setminus C_i$.

Шаг 5.3. Следующая итерация цикла 5.

Шаг 6. Для *S* оценить значение критерия (3.2), усреднённого по *m* задачам, применив к каждой задаче ансамбль *S*. Сохранить значение усреднённого критерия в переменной Fit_{new}.

Шаг 6. Выбрать номер «особи» *k* с наименьшим значением Fit_k. Если $Fit_{new} > Fit_k$, то заменить *k*-ю особь на *S*. $S_k = S$; $Fit_k = Fit_{new}$.

Перейти к Шагу 2.

Схема процедуры составления оптимальных ансамблей алгоритмов автоматической группировки представлена на Рисунке 4.2. Вначале производятся вычисления всеми алгоритмами по каждому набору данных, после чего отбирается один алгоритм каждой модели, показавший лучшие показатели целевой функции и из них уже составляется ансамбль алгоритмов автоматической группировки. Отметим, что с помощью генетического алгоритма фактически составляется ансамбль моделей, а в рамках каждой модели выбор происходит без участия генетического алгоритма.

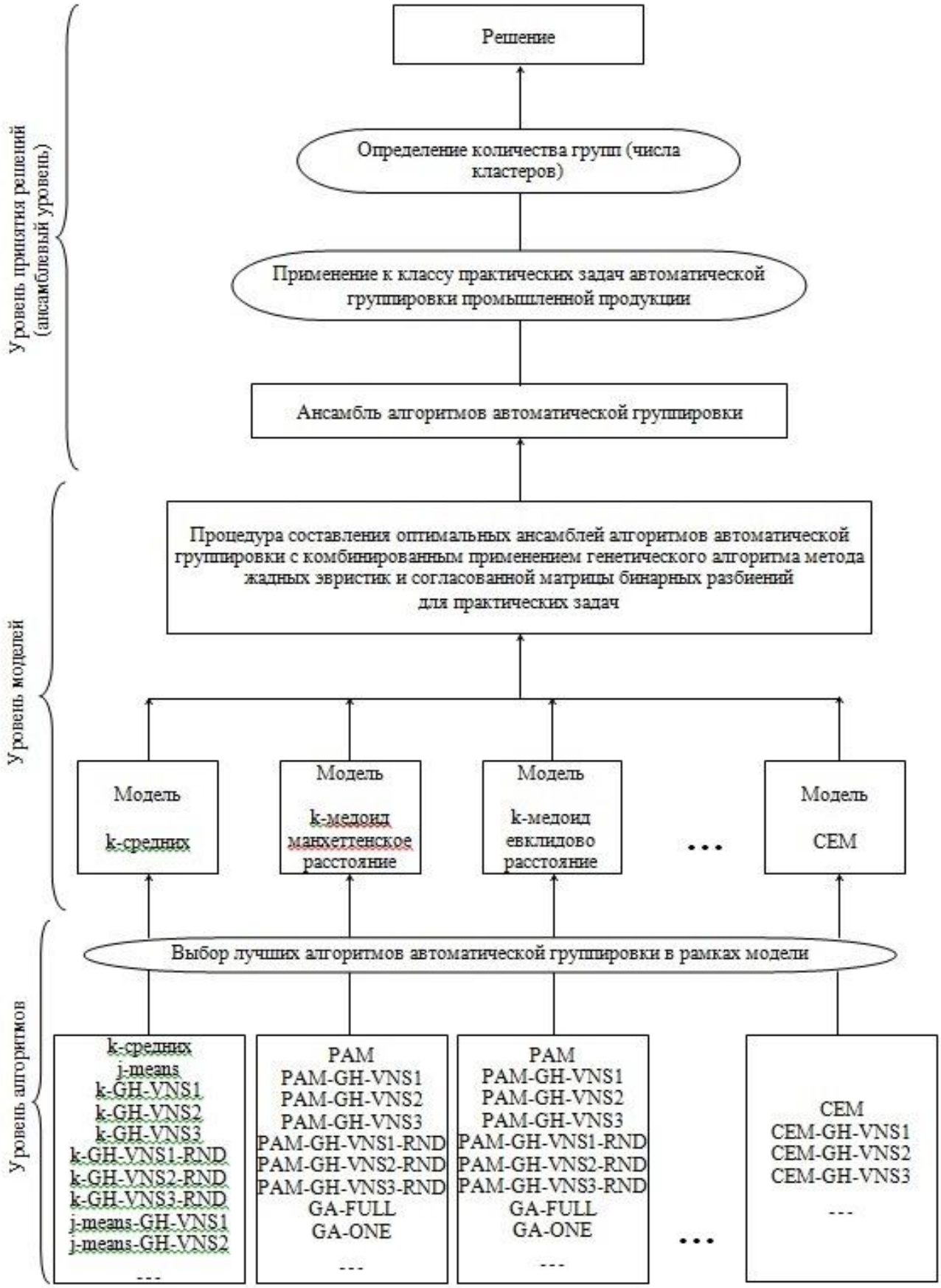


Рисунок 4.2 - Схема процедуры составления оптимальных ансамблей алгоритмов автоматической группировки

На схеме показано четыре модели алгоритмов автоматической группировки с использованием новых алгоритмов, описанных в главах 2 и 3. На самом деле моделей (как впрочем, и алгоритмов в каждой модели) может быть любое количество (на что указывают многоточия на схеме). Их количество зависит от конкретной решаемой задачи, вычислительных ресурсов и времени, которое имеется в распоряжении исследователя (или специалиста на конкретном предприятии).

Процедура составления оптимальных ансамблей алгоритмов автоматической группировки была использована при реализации программы для ЭВМ «Система составления оптимальных ансамблей алгоритмов кластеризации для задачи выделения производственных партий электрорадиоизделий» (Приложение В - Свидетельство о государственной регистрации программы для ЭВМ № 2019610095 от 09.01.2019).

Данную процедуру мы применяем к описанной выше задаче составления оптимальных ансамблей алгоритмов автоматической группировки для разделения электрорадиоизделий по производственным партиям. Генетические алгоритмы метода жадных эвристик не требуют большой популяции для своей работы. Мы использовали q=10 для составления ансамблей из 3 и 5 алгоритмов (p=3, p=5).

В качестве тестовых наборов данных были проанализированы результаты неразрушающих тестовых испытаний сборных производственных партий электрорадиоизделий, проведенных в специализированном тестовом центре АО «ИТЦ - НПО ПМ» (г. Железногорск), для комплектации бортовой аппаратуры космических аппаратов, состав которых заранее известен [188, 236, 238]. При этом сборные партии искусственно комплектовались из нескольких заведомо однородных партий электрорадиоизделий:

- 140УД25АВК - 2 производственные партии (кластера) и относительно небольшой объём данных (56 векторов каждый размерностью 18);

- 3ОТ122А - 2 партии (767 векторов каждый размерностью 10);

- 1526LE5 - 6 партий (963 векторов каждый размерностью 41).

В качестве задачи ставилось разделение составленной сборной партии на однородные компоненты с последующим анализом качества этого разделения.

Для исследований мы использовали основные классические алгоритмы автоматической группировки [244] для задач k-средних и k-медоид, а также EM-алгоритм: k-Means (метод k-средних) [208, 245-247], k-Means-fast (метод быстрых k-средних) [248], k-Means-kernel (метод ядра k-средних [249], k-Medoids (метод k-медоид) [106], EM (Expectation Maximization – максимизация математического ожидания) [250].

Кроме собственно вида алгоритма кластеризации на результат существенно влияют параметры алгоритмов, значения которых можно оптимизировать. Под оптимизацией мы понимаем подбор таких значений оптимизируемых параметров, при которых обеспечивается максимальная точность кластеризации, то есть наилучшее соответствие результата кластеризации истинному разбиению сборной партии на однородные партии электрорадиоизделий.

На выходе процесса оцениваем кластеризацию по параметру точности (Accuracy). Под точностью мы понимаем долю объектов данных, отнесенных к «правильному» кластеру. Эту «правильность» можно оценить, имея выборку размеченных данных, для которых заранее известно их отнесение к тому или иному кластеру. В данном случае наши выборки скомбинированы из данных отдельных однородных партий электрорадиоизделий. Результаты сведены в Таблицу 4.1.

Алгоритмы автоматической группировки были использованы в двух вариантах реализации: 1 – классическом и 2 – варьируемом. Во втором варианте мы пытаемся улучшить точность кластеризации путем изменения варьируемого параметра - в алгоритмах k-Means, k-Means(fast) и k-Medoids мы использовали тип меры расстояния. Для алгоритма k-Means(kernel) – тип ядра (точечное / радиальное ядро – dot/radial kernel).

Как видно из Таблицы 4.1 алгоритмы кластеризации при относительно небольших объёмах данных и количестве производственных партий (числе k)

показывают довольно высокую точность, а с увеличением объёмов данных и числа кластеров точность кластеризации уменьшается.

Таблица 4.1 - Результаты вычислительных экспериментов над производственными партиями электрорадиоизделий отдельными алгоритмами автоматической группировки

Алгоритм	Точность / значение оптимизируемого параметра			
	140УД2 5АВК 2 партии	3ОТ122А 2 партии	1526 LE5 6 партий	1526LE10 7 партий
k-Means-1	100,00 (Euclidean distance)	76,53 (Euclidean distance)	50,57 (Euclidean distance)	39,89 (Euclidean distance)
k-Means (fast)-1	100,00 (Euclidean distance)	67,67 (Euclidean distance)	50,57 (Euclidean distance)	39,89 (Euclidean distance)
k-Means (kernel)-1	100,00 (radial kernel)	59,19 (radial kernel)	47,14 (radial kernel)	46,83 (radial kernel)
k-Medoids-1	100,00 (Euclidean distance)	60,63 (Euclidean distance)	48,60 (Euclidean distance)	37,73 (Euclidean distance)
EM-1	96,43	90,09	нет результата	нет результата
k-Means-2	100,00 (Euclidean distance)	76,53 (Euclidean distance)	63,03 (Overlap Similarity)	52,83 (Overlap Similarity)
k-Means (fast)-2	100,00 (Euclidean distance)	76,53 (Euclidean distance)	50,99 (Kernel Euclidean distance)	46,84 (Correlation similarity)
k-Means (kernel)-2	53,57 (dot kernel)	67,67 (dot kernel)	30,22 (dot kernel)	46,83 (dot kernel)
k-Medoids-2	100,00 (Euclidean distance)	91,79 (Euclidean distance)	55,97 (Manhattan distance)	46,83 (Dice Similarity)
EM-2	96,43	95,44	нет результата	нет результата

При этом для моделей автоматической группировки важнейшим параметром, влияющим на результат, является используемая мера расстояния. Использование специальных мер иногда позволяет приспособить простые модели наподобие k-средних к довольно сложным задачам кластеризации. При этом достаточным условием применимости меры расстояния является наличие алгоритма решения соответствующей задачи Вебера – задачи отыскания центра

кластера [251, 252]. Проблема высокой вычислительной сложности некоторых из подобных алгоритмов при этом частично компенсируется распараллеливанием их выполнения, показанным в Главе 2.

Составим ансамбли из трёх и пяти соответственно лучших по точности алгоритмов кластеризации (Таблица 4.2) для каждого набора данных (Таблица 4.1).

Таблица 4.2 - Результаты вычислительных экспериментов с составленными ансамблями алгоритмов кластеризации

Производственная партия / ансамбль	140УД 25АВК 2 партии	3ОТ122А 2 партии	1526LE5 6 партий	1526LE10 7 партий
Ансамбль из трёх	100,00	95,04	57,01	49,09
Ансамбль из пяти	100,00	95,44	52,54	47,53

Фрагмент расчёта результата ансамбля из пяти алгоритмов кластеризации для набора данных 3ОТ122А приведён в Таблице 4.3.

Таблица 4.3 - Фрагмент результатов вычислительных экспериментов производственных партий электрорадиоизделий 3ОТ122А ансамблем из пяти алгоритмов кластеризации (указаны истинный и предполагаемые номера партий ЭРИ по результатам кластеризации)

Партия фактич.	EM-2	k-Medoids-2	EM-1	k-Means-1	k-Means (fast)-2	Ансамбль
1	1	2	1	1	1	1
1	1	2	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	2	1	2	2	2
1	1	2	1	1	1	1
1	1	1	1	1	1	1
…	…	…	…	…	…	…
2	2	2	2	1	1	2
2	2	2	2	2	2	2
2	2	2	2	2	2	2
2	2	2	2	1	1	2
…						

Для формулирования дальнейших выводов по полученным нами результатам вычислительных экспериментов с производственными партиями электрорадиоизделий для космических аппаратов и для исследования возможности использования ансамблей алгоритмов при дальнейшем применении мы взяли из репозиториев общедоступные и известные наборы данных:

- Cryotherapy [253, 254] - 2 кластера (90 векторов каждый размерностью 6);
- pima-indians-diabete - 2 кластера (768 векторов каждый размерностью 8);
- ionosphere - 2 кластера (351 векторов каждый размерностью 34);
- Iris - 3 кластера (150 векторов каждый размерностью 4);
- Zoo - 7 кластеров (101 векторов каждый размерностью 16).

Результаты полученных вычислений приведены в Таблице 4.4.

Возьмем соответственно так же по три и пять алгоритмов кластеризации, показавших лучшие результаты для каждого набора данных (Таблица 4.4), и составим из них ансамбли алгоритмов кластеризации (Таблица 4.5). Результаты ансамблей приведены в Таблице 4.6.

По результатам вычислительных экспериментов видно, что любые алгоритмы автоматической группировки для задачи разделения сборной партии электрорадиоизделий или набора данных из репозитория на две однородные партии показывают довольно высокую точность. При увеличении числа однородных производственных партий в сборной партии точность падает. При этом для разных наборов данных наилучшие результаты демонстрируются разными алгоритмами.

Использование ансамблевого подхода может быть более эффективно в сравнении с отдельными алгоритмами кластеризации. При этом отдельные алгоритмы способны показывать результаты, превосходящие по точности результаты ансамбля, но точность ансамбля все же выше, чем усреднённая точность отдельных алгоритмов [189, 190, 234].

Таблица 4.4 - Результаты вычислительных экспериментов над наборами данных отдельными алгоритмами кластеризации

Алгоритм	Точность / значение оптимизируемого параметра				
	Cryotherapy 2 кластера	pima-indians-diabetes 2 кластера	ionosphere 2 кластера	Iris 3 кластера	Zoo 7 кластеров
k-Means-1	56,67 (Euclidean Distance)	66,02 (Euclidean Distance)	71,23 (Euclidean Distance)	89,33 (Euclidean Distance)	75,25 (Euclidean Distance)
k-Means(fast)-1	56,67 (Euclidean Distance)	66,02 (Euclidean Distance)	71,23 (Euclidean Distance)	89,33 (Euclidean Distance)	75,25 (Euclidean Distance)
k-Means (kernel)-1	55,56 (radial kernel)	51,17 (radial kernel)	55,56 (radial kernel)	93,33 (radial kernel)	54,46 (radial kernel)
k-Medoids-1	57,78 (Euclidean Distance)	54,43 (Euclidean Distance)	68,09 (Euclidean Distance)	76,67 (Euclidean Distance)	79,21 (Euclidean Distance)
EM-1	56,67	65,62	нет результата	96,67	нет результата
k-Means-2	75,56 (CamberraDistance)	66,28 (Manhattan Distance)	нет результата	96,67 (CosineSimilarity)	83,17 (ManhattanDistance)
k-Means (fast)-2	75,56 (CamberraDistance)	66,28 (Manhattan Distance)	нет результата	96,67 (CosineSimilarity)	83,17 (ManhattanDistance)
k-Means (kernel)-2	53,33 (dot kernel)	65,10 (dot kernel)	64,10 (dot kernel)	33,33 (dot kernel)	40,59 (dot kernel)
k-Medoids-2	73,33 (CamberraDistance)	66,02 (DynamicTimeWarping Distance)	72,36 (JaccardSimilarity)	97,33 (CosineSimilarity)	80,20 (CosineSimilarity)
EM-2	56,67 (1-й шаг)	66,28 (1-й шаг)	нет результата	96,67 (100-й шаг)	нет результата

Так же необходимо для конкретной задачи учитывать количество алгоритмов применяемых в ансамбле, в связи с тем, что точность ансамбля алгоритмов автоматической группировки для разных наборов данных меняется при изменении числа алгоритмов в ансамбле. Поскольку на практике точность кластеризации определить невозможно вследствие отсутствия информации о

фактическом составе выборки, и невозможно априорно предсказать, какой из алгоритмов в конкретном случае покажет наиболее адекватные результаты, использование ансамблевого подхода к решению подобных задач является перспективным и актуальным. В частности, применение ансамблевого подхода в сочетании с новыми алгоритмами автоматической группировки GH-VNS (рассмотренными в главах 2 и 3), обеспечивающими наилучший результат в рамках заданной модели позволит получать результаты не только более адекватные, но и воспроизводимые при многократных запусках алгоритма.

Таблица 4.5 - Алгоритмы кластеризации, показавшие лучшие результаты для каждого набора данных

Набор данных	Cryotherapy 2 кластера	pima-indians-diabetes 2 кластера	ionosphere 2 кластера	Iris 3 кластера	Zoo 7 кластеров
1	k-Means-2	k-Means-2	k-Medoids-2	k-Medoids-2	k-Means-2
2	k-Means(fast)-2	k-Means(fast)-2	k-Means-1	EM-1	k-Means (fast)-2
3	k-Medoids-2	EM-2	k-Means (fast)-1	k-Means-2	k-Medoids-2
4	k-Medoids-1	k-Means-1	k-Medoids-1	k-Means (fast)-2	k-Medoids-1
5	EM-1	k-Means(fast)-1	k-Means (kernel)-2	EM-2	k-Means-1

Таблица 4.6 - Результаты вычислительных экспериментов над наборами данных ансамблями алгоритмов кластеризации

Набор данных	Cryotherapy 2 кластера	pima-indians-diabetes 2 кластера	ionosphere 2 кластера	Iris 3 кластера	Zoo 7 кластеров
Ансамбль из трёх	75,56	66,28	71,23	96,71	83,17
Ансамбль из пяти	75,56	65,89	68,66	96,67	81,15

4.4 Общая схема принятия решений по приемке партий промышленной продукции с повышенными требованиями к качеству

В рамках диссертационной работы была обновлена концептуальная схема системы разделения сборных партий промышленной продукции с повышенными требованиями к качеству по результатам тестовых испытаний [109, 142].

На дополненной концептуальной схеме (Рисунок 4.3) показаны задачи автоматической группировки, модели и алгоритмы с взаимосвязями задействованными при построении эффективной системы разделения сборных партий промышленной продукции с повышенными требованиями к качеству по однородным производственным партиям.

В действующую систему автоматической группировки разделения сборных партий промышленной продукции по однородным производственным партиям на основе модели k-средних добавлена программная подсистема на основе модели k-медоид с различными мерами расстояния, а также измененная жадная эвристика (с частичным объединением). Такой подход позволяет использовать дополнительные конкурентоспособные математические модели автоматической группировки для принятия решения о приемке партий и объединять их в ансамбли. Одна модель автоматической группировки позволяет верифицировать результаты другой модели, а при несовпадении результатов в условиях наивысших требований к точности и стабильности результата выделения однородных партий, предлагается отказаться от использования спорных экземпляров изделий при комплектовании бортовой аппаратуры космических аппаратов [109, 130].

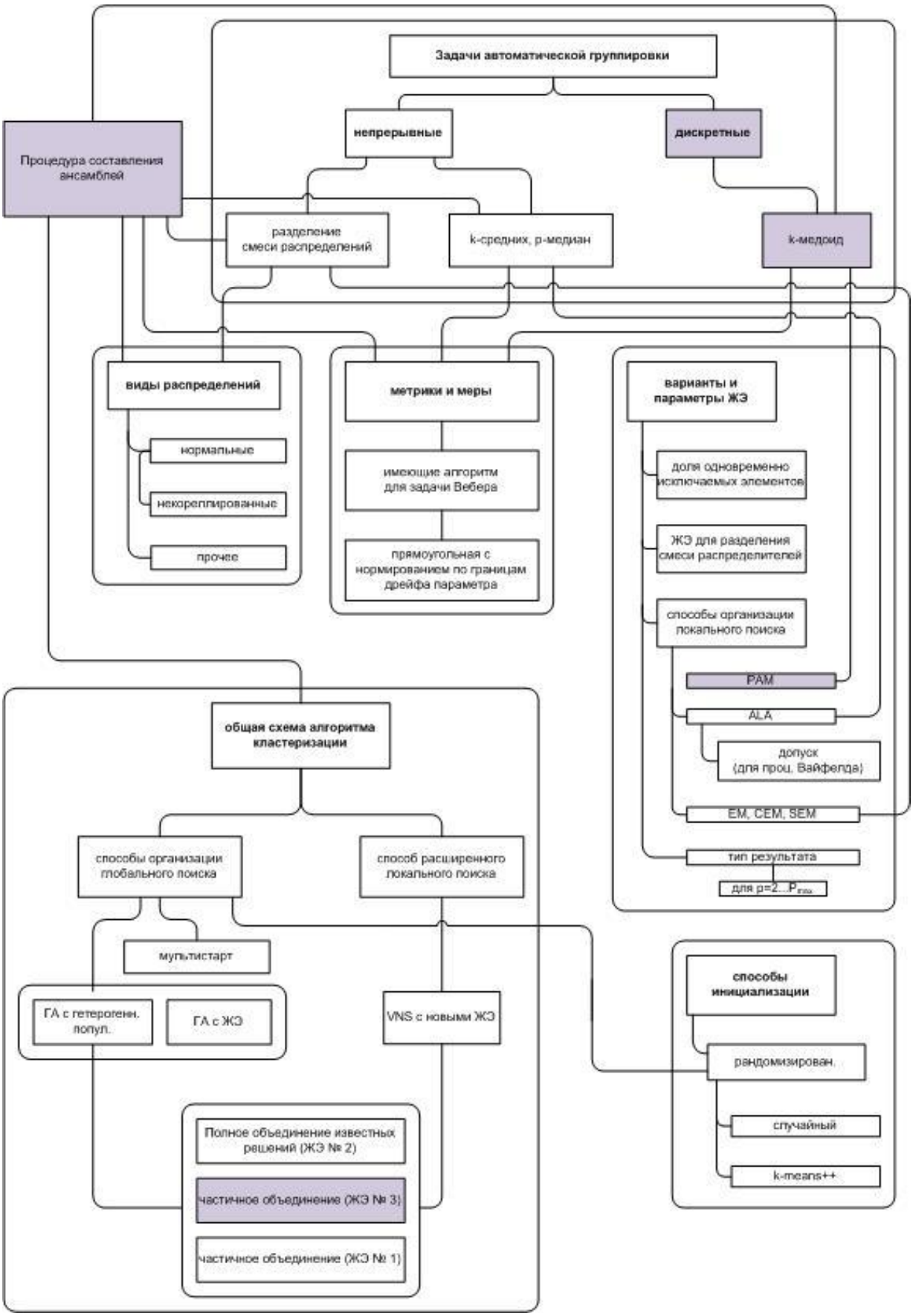


Рисунок 4.3 – Дополненная концептуальная схема системы разделения сборных партий промышленной продукции с повышенными требованиями к качеству по результатам тестовых испытаний (новые компоненты выделены цветом)

В базу данных заносятся данные проводимых тестовых испытаний, производители электрорадиоизделий, наименования изделий (номенклатура тестируемых изделий), состав тестов для каждого изделия с указанием диапазона допустимых значений каждого измерения и результаты испытаний каждого экземпляра в партии. Каждая партия того либо иного изделия в базе данных регистрируется с указанием возможного (предполагаемого изготовителем) количества производственных партий в сборной партии. Результаты проведенных тестов также фиксируются в базе данных, после чего автоматизированная система анализирует результаты и выдает их графическое представление.

Проверив данные, подготовленные автоматизированной системой и результаты визуализации, специалист принимает решение о приемке или же отклонении производственной партии продукции. Для сбора и анализа статистической информации по изготовителям и видам изделий данные о забракованных (с указанием причины) партиях также заносятся в базу данных.

Для оценки технологического процесса изготовления и для оценки технологических дефектов, которые обычно не выявляются на этапе отбраковочных испытаний, а проявляются со временем, от каждой партии электрорадиоизделий берутся образцы, на которых проводится разрушающий физический анализ. Основываясь на результаты проведенных тестовых испытаний, после организации всех необходимых и взаимосогласованных работ на заводе-изготовителе и в ОАО «ИТЦ-НПО ПМ» получается специальная партия промышленной продукции с повышенным требованием к качеству.

Результаты Главы 4

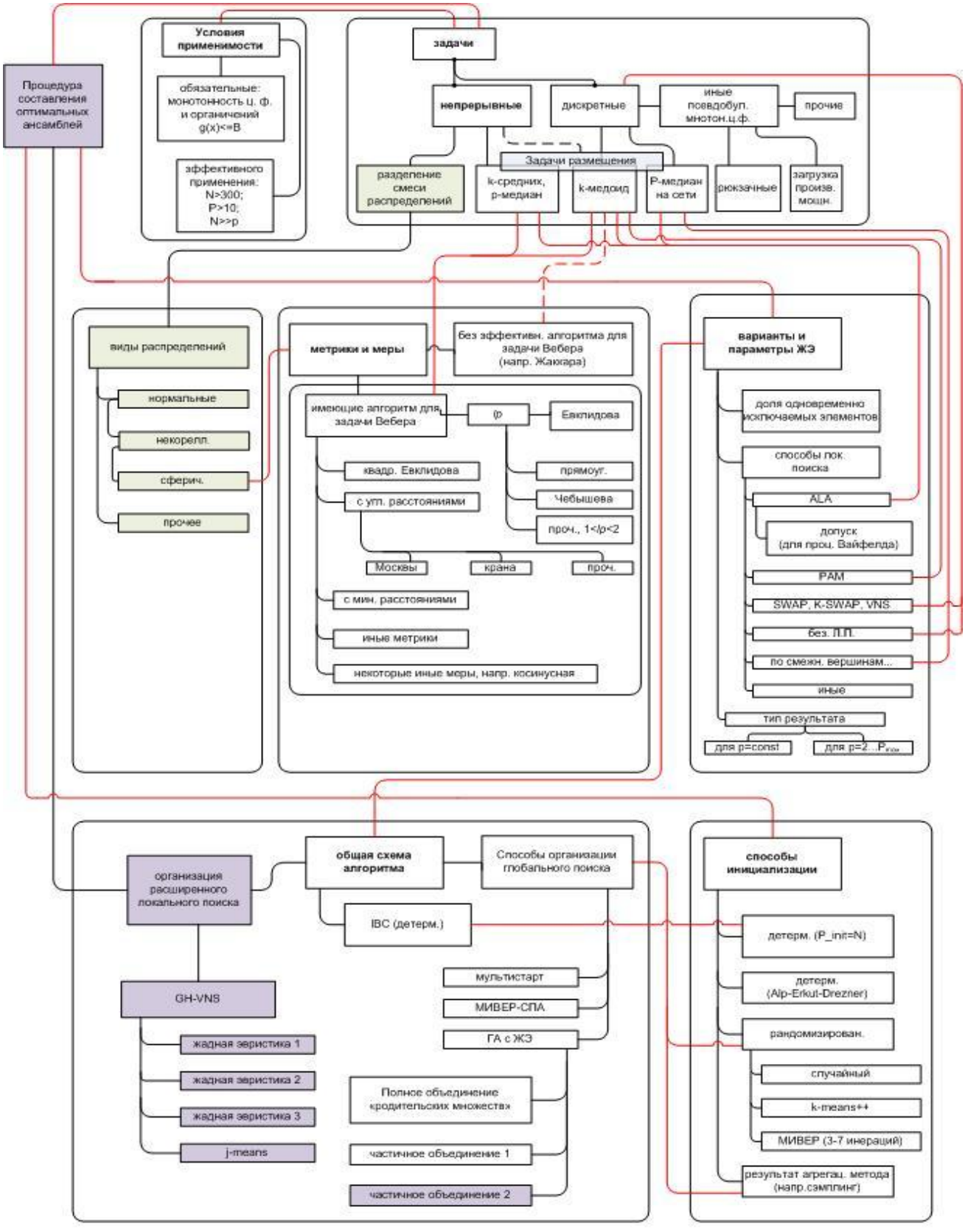
В Главе 4 рассмотрена задача выделения однородных партий для промышленной продукции с повышенными требованиями к качеству (в том числе для космического применения).

Процедура составления оптимальных ансамблей алгоритмов автоматической группировки с комбинированным применением генетического алгоритма метода жадных эвристик и согласованной матрицы бинарных разбиений (предложенная в данной главе), а также новый подход к разработке алгоритмов автоматической группировки, основанных на параметрических оптимизационных моделях, с комбинированным применением алгоритмов поиска с чередующимися рандомизированными окрестностями и жадных агломеративных эвристических процедур (изложенный в Главе 3) были использованы при разработке системы составления оптимальных ансамблей алгоритмов кластеризации для задачи выделения производственных партий [255] и успешно используются в деятельности АО «Испытательный технический центр - НПО ПМ» (г. Железногорск).

Применение новых алгоритмов поиска с чередующимися рандомизированными окрестностями (в том числе для массивно-параллельных систем) с использованием вышеуказанного подхода и внедрение системы составления оптимальных ансамблей алгоритмов кластеризации для задачи выделения производственных партий промышленной продукции, разработанных в рамках диссертационного исследования позволили повысить точность и стабильность результатов оценки точности разделения на однородные партии промышленной продукции одновременно снизив временные затраты (Приложение Б).

На Рисунке 4.4 представлена дополненная схема совместимости компонентов метода жадных эвристик [109] с новыми компонентами, которые расширяют возможности метода для решения задач автоматической группировки.

Ранее по результатам исследований Сташкова Д.В. [109] схема была дополнена еще одной непрерывной задачей – разделения смеси распределений с блоком видов распределений.



Варианты применения метода жадных эвристик.
Совместимость параметров задач и элементов метода

Обозначения: черные линии – иерархия, красные – совместимость.

Рисунок 4.4 – Дополненная схема совместимости компонентов метода жадных эвристик

В схеме совместимости компонентов метода жадных эвристик в результате настоящего исследования были добавлены (выделены сиреневым цветом) процедура составления оптимальных ансамблей и в общей схеме алгоритма подсистема организации расширенного локального поиска, а также измененная жадная эвристика (с частичным объединением 2). Это позволило расширить возможности метода жадных эвристик для задач автоматической группировки с повышенными требованиями к точности и стабильности результата.

ЗАКЛЮЧЕНИЕ

В диссертации предложены новые алгоритмы метода жадных эвристик (в том числе параллельные) для решения задач автоматической группировки (кластеризации) объектов, сочетающие применение жадных агломеративных эвристических процедур и расширенный локальный поиск с чередующимися рандомизированными окрестностями, позволяющие решать круг практических задач с повышенной точностью результата (по достигаемому значению целевой функции), а также процедура составления ансамблей алгоритмов автоматической группировки.

Цель диссертации достигается путем решения поставленных задач, а именно:

1. Анализ существующих проблем при применении методов автоматической группировки объектов, к которым предъявляются высокие требования по точности и стабильности результата, выявил дефицит алгоритмов, способных выдавать за фиксированное время результаты, которые было бы трудно улучшить известными методами, и которые бы обеспечивали стабильность получаемых результатов при многократных запусках алгоритма. При этом известные алгоритмы метода жадных эвристик требуют значительных вычислительных затрат.

2. Разработаны новые алгоритмы автоматической группировки объектов в соответствии с оптимизационной моделью k-средних, основанные на совместном применении алгоритма k-средних, жадных агломеративных эвристических процедур и расширенного локального поиска с чередующимися рандомизированными окрестностями. При этом вид окрестности поиска определяется видом применяемой жадной агломеративной эвристической процедуры, а случайным образом генерируемое известное решение является параметром данной окрестности. Показано, что новые алгоритмы позволяют получать более точный и стабильный результат (по достигаемому значению целевой функции) в сравнении с известными алгоритмами, являясь

конкурентоспособными в сравнении с известными алгоритмами метода жадных эвристик при фиксированном лимите времени работы алгоритма, позволяющем использовать алгоритмы в интерактивном режиме принятия решений.

3. Разработаны новые алгоритмы автоматической группировки объектов, основанной на модели k-медоид, также основанные на совместном применении жадных агломеративных эвристических процедур, расширенного локального поиска с чередующимися рандомизированными окрестностями и алгоритма Partition around Medoids. Показано, что новые алгоритмы также позволяют получать более точный и стабильный результат (по достигаемому значению целевой функции) в сравнении с известными алгоритмами.

4. Разработаны новые алгоритмы четкой кластеризации объектов, основанной на модели разделения смеси вероятностных распределений с применением жадных агломеративных эвристических процедур, расширенного локального поиска с чередующимися рандомизированными окрестностями и известного классификационного EM-алгоритма, также обладающие преимуществами по получаемому значению целевой функции за фиксированное время. Это позволяет говорить о новом подходе к разработке эффективных алгоритмов автоматической группировки, основанном на комбинированном применении известных для соответствующих задач алгоритмов локального поиска, жадных агломеративных эвристических процедур и алгоритмов поиска с чередующимися рандомизированными окрестностями, образуемыми применением одной из жадных агломеративных эвристических процедур к лучшему известному решению и второму решению, генерируемому случайным образом и являющемуся параметром окрестности.

5. Впервые предложены параллельные модификации алгоритмов метода жадных эвристик для архитектуры CUDA, позволяющие существенно расширить рамки применения метода жадных эвристик и охватить достаточно большие задачи – до сотен тысяч векторов многомерных данных.

6. Разработана процедура составления оптимальных ансамблей алгоритмов автоматической группировки с комбинированным применением генетического

алгоритма метода жадных эвристик и согласованной матрицы бинарных разбиений для практических задач, позволяющая уменьшить число ошибок при разделении сборной партии промышленной продукции на однородные партии с использованием данных неразрушающих тестовых испытаний.

СПИСОК ЛИТЕРАТУРЫ

1. Gantz, J.F. The diverse and exploding digital universe. IDC White Paper [Электронный ресурс]/ J.F. Gantz// Framingham: IDC. – 2008. Режим доступа: URL http://www.emc.com/collateral/analyst-reports/diverse-exploding-digitaluniverse.pdf (дата обращения: 01.12.2018).

2. Jain, A.K. Data clustering: 50 years beyond K-means/A.K. Jain// Pattern Recognition Letters.- 2010.- Vol. 31.- P. 651-666.

3. Большакова, Л.В. Современные математико-статистические методы обработки информации в научной и практической работе / Л.В.Большакова, Н.А.Яковлева // Проблемы современной науки и образования.- 2016.- № 7. С. 49-52.

4. Бериков, В.Б. Современные тенденции в кластерном анализе [Электронный ресурс]/ В.Б. Бериков, Г.С. Лбов// Всероссийский конкурсный отбор обзорно-аналитических статей по приоритетному направлению "Информационно-телекоммуникационные системы". Новосибирск: Институт математики им. С.Л. Соболева СО РАН.- 2008.- С. 26. Режим доступа:

URL http://www.ict.edu.ru/ft/005638/62315e1-st02.pdf (дата обращения 21.07.2018).

5. Duda, R. Pattern Classification, second ed./ R. Duda., P. Hart, D. Stork.// New York: John Wiley and Sons.- 2001.- P. 680.

6. Мандель, И.Д. Кластерный анализ/ И.Д. Мандель// М.: Финансы и статистика.– 1988.– С. 176.

7. Дюк, В.А. Применение технологий интеллектуального анализа данных в естественнонаучных, технических и гуманитарных областях/ В.А. Дюк, А.В. Флегонтов, И.К. Фомина// Известия РГПУ им. А.И. Герцена.– 2011.– № 138.– С. 77-84.

8. Tryon, R.C. Cluster analysis/ R.C. Tryon// London: Ann Arbor Edwards Bros. -1939. – P. 139.

9. Воронцов, К.В. Алгоритмы кластеризации и многомерного шкалирования [Электронный ресурс]/ К.В. Воронцов// Курс лекций.– МГУ.– 2007.– Режим доступа: http://www.ccas.ru/voron/ download/Clustering.pdf.

10. Лукьяненко, М.В. Надежность изделий электронной техники в аппаратуре космических аппаратов: учеб. пособие/ М.В. Лукьяненко, Н.П. Чурляева, В.В. Федосов// Сиб. гос. аэрокосмич. ун-т.– Красноярск.- 2016.- С. 188.

11. Федосов, В.В. Минимально необходимый объем испытаний изделий микроэлектроники на этапе входного контроля/ В.В. Федосов, В.И. Орлов// Известия высших учебных заведений. Приборостроение.– 2011.– Т.54. № 4.– С. 58-62.

12. Iwayama, M. Cluster-based text categorization: A comparison of category search strategies/ M. Iwayama, T. Tokunaga// Proc. 18th ACM Internat. Conf. on Research and Development in Information Retrieval.– 1995.– P. 273–281.

13. Барахнин, В.Б. Кластеризация текстовых документов на основе составных ключевых термов/ В.Б. Барахнин, Д.А. Ткачев// Вестник НГУ. Серия: Информационные технологии.– 2010.– Т. 8/ вып. 2.– С. 5-14.

14. Барахнин, В.Б. Проектирование информационной системы представления результатов комплексного анализа поэтических текстов / В.Б.Барахнин, О.Ю.Кожемякина, Ю.С.Борзилова // Вестник Новосибирского государственного университета. Серия: Информационные технологии.- 2019.- Т. 17. № 1. С. 5-17.

15. Bhatia, S. Conceptual clustering in information retrieval/ S. Bhatia, J. Deogun// IEEE Trans. Systems Man Cybernet.– 1998.– Vol. 28 (B).– P. 427–436.

16. Jain, A.K. Image segmentation using clustering/ A.K. Jain, P. Flynn// Advances in Image Understanding.- IEEE Computer Society Press.- 1996.- P. 65–83.

17. Арлазаров, В.В. Структурный анализ текстовых полей в системах потокового ввода оцифрованных документов/ В.В. Арлазаров, В.М. Кляцкин, О.А. Славин// Труды ИСА РАН.- 2015.- Т. 65.- вып. 1.- С. 75-81.

18. Shi, J. Normalized cuts and image segmentation/ J. Shi, J. Malik// IEEE Trans. Pattern Anal. Machine Intell.- 2000.- Vol. 22.- P. 888–905.

19. Борисенко, В.И. Сегментация изображения (состояние проблемы)/ В.И. Борисенко, А.А. Златопольский, И.Б. Мучник// Автомат. и телемех.- 1987.- вып. 7.- С. 3-56.

20. Connell, S.D. Writer adaptation for online handwriting recognition/ S.D. Connell, A.K. Jain// IEEE Trans. Pattern Anal. Machine Intell.- 2002.- Vol. 24.- Issue 3.- P. 329–346.

21. Андреева, Е.И. Сравнение оцифрованных страниц деловых документов на основе распознавания / Е.И.Андреева, Т.В.Манжиков, О.А.Славин // Сенсорные системы. 2018.- Т. 32. № 1. С. 35-41.

22. Hu, J. Statistical methods for automated generation of service engagement staffing plans/ J. Hu, B.K. Ray, M. Singh// IBM J. Res. Dev.– 2007.– Vol. 51.- Issue 3.– P. 281–293.

23. Baldi, P. DNA Microarrays and Gene Expression/ P. Baldi., G. Hatfield.// [s.l.]: Cambridge University Press.- 2002.- P.208.

24. Андреев, В.Л. Классификационные построения в экологии и систематике/ В.Л. Андреев// М.:Наука.- 1980.- С. 142.

25. Berry, M.J.A. Data Mining techniques: for marketing, sales, and customer relationship management, 2nd ed./ M.J.A. Berry, G.S. Linoff// [s.l.]: Wiley.– 2004.– P. 464.

26. Галямов, А.Ф. Управление взаимодействием с клиентами коммерческой организации на основе методов сегментации и кластеризации клиентской базы/ А.Ф. Галямов, С.В. Тархов// Вестник УГАТУ.– 2014.– Т. 18.- № 4(65).– С.149-156.

27. Drezner, Z. Facility location: applications and theory/ Z. Drezner, H. Hamacher.// Berlin: Springer-Verlag.– 2004.– P. 460.

28. Farahani, R. Facility location: Concepts, models, algorithms and case studies/ R.Z. Farahani and M. Hekmatfar (eds.)// Berlin Heidelberg: Springer-Verlag.– 2009.– P. 549.

29. Бельц, Е.А. Оптимизация размещения предприятий с учетом минимально допустимых расстояний/ Е.А. Бельц, А.А. Колоколов// Вестн. Ом. ун-та.– 2012.– No 4.– С. 13-16.

30. Кочетов, Ю.А. Сравнение метаэвристик для решения двухуровневой задачи размещения предприятий и фабричного ценообразования / Ю.А.Кочетов, А.А.Панин, А.В.Плясунов // Дискретный анализ и исследование операций.- 2015.- Т. 22. № 3 (123). С. 36-54.

31. Hansen, P. Cluster analysis and mathematical programming/ P. Hansen, B. Jaumard// Mathematical Progralnming.- 1997.- Vol. 79.- P. 191-215.

32. Hansen, P. Variable neighborhood search for the p-median/ P. Hansen, N. Mladenovic// Location Science.- 1997.- Vol. 5.- No. 4.- P. 207-226.

33. Rosing, R.E. Towards the solution of the (generalized) Weber problem/ R.E. Rosing// Environment and Planning B: Environment and Design.- 1991.- Vol. 18.- P. 347-360.

34. Hall, R.W. Median mean and optimum as facility locations/ R.W. Hall// Journal of Regional Science.-1988.- Vol. 28.- P. 65-81.

35. Ottaviano, G.I.P. New economic geography: what about the N?/ G.I.P. Ottaviano, J.-F. Thisse// Environment and Planning A.- 2005.- Vol. 37,- Issue 10.- P. 1707–1725.

36. Boltyanski, Y. Geometric Methods and Optimization Problems (Combinatorial Optimization)/ Y. Boltyanski, H. Martini, V. Soltan// Dordrecht: Kluwer Academic Publishers.- 1999.- Vol. 4.- P. 432.

37. Volek, J. Location analysis - Possibilities of use in public administration/ J. Volek// Verejna sprava.- Pardubice: Univerzita Pardubice.- 2006.- P. 84-85.

38. Teodorovic, D. Transportne mreze, Poglavlje 9: Lokacijski problem/ D. Teodorovic// Beograd: Saobranajni fakultet.- 2009.- P. 389-399.

39. Watanabe, D. Generalized Weber Model for Hub Location of Air Cargo/ D. Watanabe, T. Majima, K. Takadama, M. Katuhara// The Eighth International Symposium on Operations Research and Its Applications (ISORA’09).- Zhangjiajie.- 2009.- P. 124–131.

40. Береснев, В.Л. Экстремальные задачи стандартизации /В.Л. Береснев, Э.Х. Гимади, В.Т. Дементьев// Новосибирск: Наука.- 1978.- С. 333.

41. Гимади, Э.Х. Задача стандартизации с данными произвольного знака и связными, квазивыпуклыми и почти квазивыпуклыми матрицами/ Э.Х. Гимади// Управляемые системы. Сб. науч. тр. Вып. 27. — Новосибирск: Ин-т математики СО АН СССР.- 1987.- С. 3-11.

42. Гимади, Э.Х. Обоснование априорных оценок качества приближенного решения задачи стандартизации/ Э.Х. Гимади// Управляемые системы: Сб. науч. тр.- Новосибирск: Ин-т математики СО АН СССР.- 1987.- Вып. 27.- С. 12-27.

43. Кочетов, Ю.А. Методы локального поиска для дискретных задач размещения: дис… доктора физ.-мат. Наук: 05.13.18: защищена 19.01.2010/ Ю.А. Кочетов// Новосибирск: Институт математики им. Соболева.- 2010.- С. 259.

44. Васильев, И.Л. Новые нижние оценки для задачи размещения с предпочтениями клиентов/ И.Л. Васильев, К.Б. Климентова, Ю.А. Кочетов// Журнал вычислительной математики и математической физики.- 2009.- Т. 49,- вып. 6.- С. 1055-1066.

45. Гончаров, Е.Н. Поведение вероятностных жадных алгоритмов для многостадийной задачи размещения/ Е.Н. Гончаров, Ю.А. Кочетов// Дискретный анализ и исследование операций. Серия 2.- 1999.- Т. 6. № 1.- С. 12-32.

46. Pfeiffer, B. A unified model for Weber problem with continuous and network distance/ B. Pfeiffer, K. Klamroth// Computers and OR. – 2008.– Vol. 35.- No. 2.– P. 312-326.

47. Cooper, L. The transportation-location problem/ L. Cooper //Oper. Res.– 1972.– Vol. 20,- No. 1.– P. 94-108.

48. Lloyd, S.P. Least Squares Quantization in PCM/ S.P. Lloyd// IEEE Transactions on Information Theory.- 1982.- Vol. 28.- P. 129-137.

49. Fermat, P. de Oeuvres/ Fermat P. de (1643), Ed. H.Tannery, ed.// Paris 1891, Supplement: Paris.- 1922.- Vol. 1.- P. 153

50. Torricelli, E. Opere de Evangelista Torricelli/ E. Torricelli, G. Loria, G. Vassura// English edition.– Part 2.– Faenza.– 1919. –Vol I.– P. 90-97.

51. Kirszenblat, D. Dubins networks: Thesis/ D. Kirszenblat// Melbourne: Department of Mathematics and Statistics of the University of Melbourne.– 2011.– P. 56.

52. Региональная экономика и управление. Учебное пособие в 2 – х частях/ Под ред. А.И. Гаврилова// Н. Новгород: Изд-во ВВАГС.- 2005.- С. 260.

53. Hale, T.S. Location science research: a review/ T.S. Hale, C.R. Moberg// Annals of Operations Research.- 2003.- Vol. 123.- P. 21-35.

54. Weiszfeld, E. Sur le point sur lequel la somme des distances de n points donnes est minimum/ E. Weiszfeld// Tohoku Mathematical Journal.– 1937.– Vol. 43.- No. 1.– P.335–386.

55. Sturm, R. Ueber den Punkt kleinster Entfernungssumme von gegebenen Punkten/ R. Sturm// J. Rein. Angew. Math.– 1884.– Vol. 97.– P. 49–61.

56. Beck, A. Weiszfeld's Method: Old and New Results/ A. Beck, S. Sabach// J. Optim. Theory Appl.– 2015.– Vol. 164,- Iss. 1.– P. 1-40 DOI 10.1007/s10957-014-0586-7.

57. Drezner, Z. The fortified Weiszfeld algorithm for solving the Weber problem/ Z. Drezner// IMA Journal of Management Mathematics.- 2013.- Vol. 26.- P. 1-9. DOI: 10.1093/imaman/dpt019.

58. Hakimi, S.L. Optimum locations of switching centers and the absolute centers and medians of a graph/ L. Hakimi. S.// Operations Research.– 1964.– Vol. 12,- Issue 3.– P. 450–459.

59. Hakimi, S.L. Optimum distribution of switching centers in a communication network and some related graph theoretic problems/ S.L. Hakimi // Operations Research.– 1965.– Vol. 13.- No. 3.– P. 462–475.

60. Сергиенко, И.В. Математические модели и методы решения задач целочисленной оптимизации/ И.В. Сергиенко// 2-е изд., доп. и перераб.- Киев: Наукова думка.- 1988.- С. 472.

61. Гимади, Э.Х. Эффективные алгоритмы для решения многоэтапной задачи размещения на цепи/ Э.Х.Гимади// Дискретн. анализ и исслед. Опер..- 1995.- том 2. № 4.- С. 13–31.

62. Алексеев, О.Г. Некоторые алгоритмы решения задачи о покрытии и их экспериментальная проверка на ЭВМ/ О.Г. Алексеев, В.Ф. Григорьев// Журнал вычислительной математики и математической физики.- 1984.- Т. 24,- № 10.- С. 1565-1570.

63. Агеев, А.А. Полиномиальный алгоритм решения задачи размещения на цепи с одинаковыми производственными мощностями предприятий/ А.А. Агеев, Э.Х. Гимади, А.А. Курочкин// Дискретный анализ и исследование операций.- 2009.- Т. 16.- № 5.- С. 3–18.

64. Браверман, Э.М. Структурные методы в обработке эмпирических данных/ Э.М. Браверман, И.Б. Мучник// М.: Наука.- 1983.- С. 464.

65. Загоруйко, Н.Г. Конкурентное сходство как универсальный базовый инструмент когнитивного анализа данных / Н.Г.Загоруйко, И.А.Борисова, О.А.Кутненко, В.В.Дюбанов, Д.А.Леванов // Онтология проектирования. 2015.- Т. 5. № 1 (15). С. 7-18.

66. Черенин, В.П. Решение методом последовательных расчетов одного класса задач о размещении производства/ В.П. Черенин, В.Р. Хачатуров// В кн.: Экономико-математические методы, вып. 2. - М.: Наука.- 1965.- С. 279-290.

67. Khachaturov, V.R. The Stability of Optimal Values in Problems of Discrete Programming/ V.R. Khachaturov// Optimization Techniques IFIP Technical Conference. Novosibirsk. July 1-7. 1974. Edited by G.I.Marchuk, Springer-Verlag, Berlin, Heidelberg, New York.- 1975.- P. 372-376.

68. Черенин, В.П. Решение некоторых комбинаторных задач оптимального планирования методом последовательных расчетов/ В.П. Черенин// В кн.: Научно-методические материалы экономико-математического семинара ЛЭММ АН СССР. вып. 2. - М.: Гипромез.- 1962. – С. 44.

69. Хачатуров, В.Р. Алгоритмы определения оптимальной совокупности отраслевых вариантов размещения предприятий с учетом эффекта агломерации / В.Р. Хачатуров, Н.Д. Астахов , В.В. Григорьев // М. ВЦ АН СССР.- 1984. – С. 22.

70. Колоколов, А.А. Алгоритмы декомпозиции и перебора L-классов для решения некоторых задач размещения/ А.А. Колоколов, Т.В. Леванова// Вестник Омского университета,- 1996.- № 1,- С. 21-23.

71. Леванова, Т.В. Локальный поиск с чередующимися окрестностями для двухстадийной задачи размещения/ Т.В. Леванова, А.С. Федоренко// Дискретный анализ и исследование операций. 15:3.- 2008.- С. 43–57.

72. Kochetov, Y. Large Neighborhood Local Search for the p-Median Problem/ Y. Kochetov E. Alekseeva, T. Levanova, M. Loresh// Yugoslav Journal of Operations Research, 15:1.- 2005. 53–63http://yujor.fon.rs/index.php/journal/ article/view/579/322.

73. Vidyasagar, M. Statistical learning theory and randomized algorithms for control/ M. Vidyasagar// IEEE Control Systems. -1998.- No. 12.- P. 69-85.

74. Граничин, О.Н. Рандомизированные алгоритмы оценивания и оптимизации при почти произвольных помехах/ О.Н. Граничин, Б.Т. Поляк// М. Наука. -2003.-С. 291.

75. Goldberg, D.E. Genetic algorithm in search, optimization and machine learning/ D.E. Goldberg// MA: Addison-Wesley.- 1989.- P. 432.

76. Kuehn, A.A. A heuristic program for locating warehouses/ A.A. Kuehn, M.J. Hamburger// Management Science.- 1963.- 9(4).- P. 643-666.

77. Kohonen, T. Self-Organization and Associative Memory, 3rd ed./ T. Kohonen// Springer information sciences series.-New York: Springer-Verlag.- 1989.- P. 312.

78. Kirkpatrick, S. Optimization by simulated annealing/ S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi// Science.- 1983.- Vol. 220(4598).- P. 671–680.

79. Alp, O. An Efficient Genetic Algorithm for the p-Median Problem/ O. Alp, E. Erkut, Z. Drezner// Annals of Operations Research.- 122 (1-4).- 2003.- P. 21–42. doi 10.1023/A:1026130003508.

80. Chiou, Y. Genetic clustering algorithms/ Y. Chiou, L.W. Lan// European Journal of Operational Research.- 2001.- Vol. 135.- P. 413-427.

81. Bozkaya, B.A. Genetic Algorithm for the p-Median Problem/ B. Bozkaya, J. Zhang, E. Erkut// Facility Location: Applications and Theory/ Z. Drezner, H. Hamacher [eds.].-New York: Springer.- 2002.- P. 179-205.

82. Krishna, K. Genetic K-means algorithm/ K. Krishna, M. Murty// IEEE Transaction on System, Man and Cybernetics - Part B.- 1999.- Vol. 29.- P. 433-439.

83. Holland, J.H. Adaptation in Natural and Artificial System: University of Michigan Press.- 1975.- P. 18–25.

84. Reeves, C.R. Genetic algorithms for the operations researcher/ C.R. Reeves// INFORMS Journal of Computing.-1997.- Vol. 9,- Issue 3.- P. 231-250.

85. Agarwal, C.C. Optimized crossover for the independent set problem/ C.C. Agarwal, J.B. Orlin, R.P. Tai// Operations research.- 1997.- Vol. 45,- Issue 2.- P. 226-234.

86. Eremeev, A.V. Optimal recombination in genetic algorithms for combinatorial optimization problems, part 1/ A.V. Eremeev, J.V. Kovalenko// Yugoslav Journal of Operations Research.- 2014.- Vol. 24.- Issue 1.- P. 1-20, DOI:10.2298/YJOR130731040E.

87. Steinhaus, H. Sur la division des corps materiels en parties/ H. Steinhaus// Bull. Acad. Polon. Sci.– 1956.– Cl. III,- Vol. IV.– P. 801-804.

88. MacQueen, J.B. Some Methods of Classification and Analysis of Multivariate Observations/ J.B. MacQueen// Proceedings of the 5th Berkley Symposium on Mathematical Statistics and Probability.- 1967.- Vol. 1.- P. 281-297.

89. Alsabti, K. An efficient k-means clustering algorithm/ K. Alsabti, S. Ranka, V. Singh// Proceedings of IPPS/SPDP Workshop on High Performance Data Mining.- 1998.

90. Nigam, K. Text Classification from Labeled and Unlabeled Documents using EM/ K. Nigam, A.K. Mccallum, S. Thrun, T. Mitchell// ACM journal of Machine Learning-Special issue on information retrieval.- 1999.

91. Kanungo, T. An efficient k-means clustering algorithm: analysis and implementation/ T. Kanungo, D.M. Mount, N.S. Netanyahu, C.D. Piatko, R. Silverman,

A.Y. Wu// IEEE Transactions on Pattern Analysis and Machine Intelligence.- 2002.- Vol. 24.- No. 7.- P. 881-892.

92. Cheung, Y.M. K-Means: A new generalized k-means clustering algorithm/ Y.M. Cheung // Pattern Recognition Letters.- 2003.- Vol. 24,- Issue 15.- P. 2883-2893.

93. Xiaoli, C. Optimized big data K-means clustering using Map Reduce/ C. Xiaoli [et al.]// Springer Science + Business Media New York.- 2014.

94. Xiong, H. K-Means Clustering Versus Validation Measures: A Data-Distribution Perspective/ H. Xiong, J. Wu, J. Chen// IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics.- 2009.- Vol. 39.- No. 2.- P. 318-331.

95. Zhang, L. Application of k-means clustering algorithm for classification of NBA guards/ L. Zhang, F. Lu, A. Liu, P. Guo, C. Liu// International Journal of Science and Engineering Applications.- 2016.- Vol. 5.- Issue 1. ISSN- 2319-7560 (online).

96. Wang, J. An Improved K-means Clustering Algorithm/ J. Wang, X. Su// Communication Software and Networks (ICCSN). IEEE 3rd International Conference.- 2011.- P. 44-46.

97. Singh, R.V. Data Clustering with Modified K-means Algorithm/ R.V. Singh, M.P.S. Bhatia// Recent Trends in Information Technology. 2011 IEEE International Conference. -2011.- P. 717-721.

98. Shi, Na Research on K-means Clustering Algorithm: An Improved K-means Clustering Algorithm/ Shi Na, Liu Xumin, Guan Yong// Intelligent Information Technology and Security Informatics. 2010 IEEE Third International Symposium on 2-4 April,- 2010.- P. 63-67.

99. Sharmila Rani, D. Modified K-means Algorithm for Initial Centroid Detection/ D. Sharmila Rani, V.T. Shenbagamuthu// International Journal of Innovative Research in Computer and Communication Engineering.- 2017.- Vol. 2, Special Issue 1.

100. Bhusare, B.B. Initialization for K-Means Clustering using Improved Pillar Algorithm/ B.B. Bhusare, S.M. Bansode Centroids// International Journal of Advanced Research in Computer Engineering & Technology (IJARCET).- 2014.- Vol. 3.- Issue 4.

101. Kaur, K. Statistically Refining the Initial Points for K-Means Clustering Algorithm/ K. Kaur, D. Singh Dhaliwal, R. Kumar Vohra// International Journal of

Advanced Research in Computer Engineering & Technology (IJARCET).- 2013.- Vol. 2.- Issue 11.

102. Wang, S. An Improved K-means Clustering Algorithm Based on Dissimilarity/ S. Wang// International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC). Shenyang. China IEEE.- 2013.

103. Mahmud, S. Improvement of K-means clustering algorithm with better initial centroids based on weighted average (англ.)/ S. Mahmud, M. Rahman, N. Akhtar// 7th International Conference on Electrical and Computer Engineering.– IEEE.- 2012-12.– ISBN 9781467314367.– DOI:10.1109/icece.2012.6471633.

104. Abdul Nazeer, K.A. Improving the Accuracy and Efficiency of the k-means Clustering Algorithm/ K.A. Abdul Nazeer, M.P. Sebastian// Proceedings of the World Congress on Engineering.- 2009.- Vol. I. WCE 2009. July 1 – 3.- 2009. London. U.K.

105. Hansen, P. J-Means: a new local search heuristic for minimum sum of squares clustering/ P. Hansen, N. Mladenović// Pattern Recognition.- 2001-02.- Vol. 34.- Issue. 2.- P. 405–413. DOI:10.1016/s0031-3203(99)00216-2.

106. Kaufman, L. Clustering by means of Medoids. Statistical Data Analysis Based on the L1-Norm and Related Methods/ L. Kaufman, P.J. Rousseeuw// Springer US.- 1987.- P. 405–416.

107. Королёв, В.Ю. ЕМ-алгоритм, его модификации и их применение к задаче разделения смесей вероятностных распределений. Теоретический обзор/ В.Ю. Королёв// ИПИ РАН. М.- 2007.- С. 94.

108. Celeux, G. A classification EM algorithm for clustering and two stochastic versions/ G. Celeux, G. Govaert// Computational Statistics and Data Analysis,- 1992.- Vol. 14.- P. 315-332.

109. Казаковцев, Л.А. Эвристические алгоритмы разделения смеси распределений: монография /Л.А. Казаковцев, Д.В Сташков, В.И. Орлов // под общ.ред.В.И.Орлова; СибГУ им.М.Ф.Решетнева. - Красноярск, 2018. - 164 с.

110. Celeux, G. Classification EM Algorithm for Clustering and Two Stochastic Versions/ G. Celeux, A. Govaert// Rapport de Recherche de l’INRIA RR-1364. Centrede Rocquencourt.- 1991.

111. Broniatowski, M. Reconnaissance de m´elanges de densit´es par un algorithme d’apprentisage probabiliste/ M. Broniatowski, G. Celeux, J. Diebolt // in: E. Diday, M. Jambu, L. Lebart, J.-P. Pag`es, R. Tomasone (Eds.) Data Analysis and Informatics, III, North Holland, Amsterdam.- 1983.- P. 359-373.

112. Celeux, G. Reconnaissance de m´elanges de densit´e et classification. Un algorithme d’apprentisage probabiliste: l’algorithme SEM/ G. Celeux, J. Diebolt// Rapport de Recherche de l’INRIA RR-0349. Centre de Rocquencourt.- 1984.

113. Celeux, G. The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem/ G. Celeux, J. Diebolt// Computational Statistics Quarterly.- 1985.- Vol. 2.- No. 1.- P. 73-82.

114. Likas, A. The global k-means clustering algorithm/ A. Likas, M. Vlassis, J. Verbeek// Pattern Recognition.- 2003.- Vol. 36.- P. 451-461.

115. Wu, L.-Y. Capacitated facility location problem with general setup cost/ L.-Y. Wu., X.-S. Zhang, J.-L. Zhang// Computers and Operations Research.- 2006.- Vol. 33.- P. 1226–1241.

116. Franca, P.M. An adaptive tabu search algorithm for the capacitated clustering problem/ P.M. Franca, N.M. Sosa, V. Pureza// International Transactions in operational Research.- 1999.- Vol. 6.- P. 665–678.

117. Орлов, В.И. О непараметрической диагностике и управлении процессом изготовления электрорадиоизделий/ В.И. Орлов, Н.А. Сергеева.// Вестник СибГАУ.- 2013.- Вып. 2(48).- С. 70-75.

118. Куклин, В.И. Результаты работ по обеспечению качества электрорадиоизделий отечественного производства для комплектования бортовой аппаратуры космических аппаратов за период 01.2008 г.–06.2009 г./ В.И. Куклин, В.И. Орлов, В.В. Федосов// VIII Российская научно-техническая конференция. Электронная компонентная база космических систем. М.- 2009.- С. 64-66.

119. Федосов, В.В. Технический отчет. Космический аппарат «SESAT» со сроком активного функционирования 10 лет. Принципы, методы и результаты комплектации аппаратуры электрорадиоизделиями/ В.В. Федосов, В.И. Куклин,

В.И. Орлов, Ш.Н. Исляев и др.// ФГУП «НПО ПМ им. академика Решетнева» .– 1999.– С. 408.

120. Перечень ЦК-1/96. Изделия электронной техники, допускаемые для применения в аппаратуре космического аппарата «Ямал» с 10-летним сроком активного существования// АО ИТЦ «Циклон».– 1997.– С. 90.

121. Решение № SST-TP-97006 о квалификации электрорадиоизделий на соответствие требованиям космического аппарата с 10-летним сроком активного существования (Редакция 1-97)// АО ИТЦ «Циклон».– 1997.– С. 108.

122. Вернова, С.Н. Модель околоземного космического пространства: В 3-х т. Т. 3 Под ред. академика. Издание седьмое/ С.Н. Вернова// М.: МГУ.– 1983.– С. 133.

123. Стойкость изделий электронной техники к воздействию факторов космического пространства и электрических импульсных перегрузок: Справочник. Т. XII. 4-е изд. Книга 2. Термовакуумные и электрические воздействия// ВНИИ «Электронстандарт».– 1990.– С. 162.

124. Пиз, Р.Л. Радиационные испытания полупроводниковых приборов для космической электроники/ Р.Л. Пиз, А.Х. Джонстон, Дж.Л. Азаревич// ТИИЭР.– 1988.– Т. 76.- № 11.– С. 126-145.

125. Радиационная стойкость бортовой аппаратуры и элементов космических аппаратов// I Всесоюзная научно-техническая конференция. Материалы конференции. Томск.– 1991.– С. 257.

126. Радиационная стойкость материалов радиотехнических конструкций: Справочник. Под ред. Н.А Сидорова, В.К. Князева// М.: Советское радио.– 1976.- С. 567.

127. Малышев, М.М. Методология оценки радиационной надежности ИЭТ в условиях низкоинтенсивных ионизирующих излучений/ М.М. Малышев, В.Г. Малинин, И.В. Куликов, Ю.Н. Торгашов, М.В. Ужегов// В сб. Радиационно-надежностные характеристики изделий электронной техники в экстремальных условиях эксплуатации. Под редакцией Ю.Н. Торгашова СПб.: Издательство РНИИ «Электронстандарт».– 1994.– С. 96.

128. Мырова, Л.О. Обеспечение стойкости аппаратуры связи к ионизирующим и электромагнитным излучениям. 2-е изд., перераб. и доп./ Л.О. Мырова, А.З. Чепиженко// М.: Радио и связь.– 1988.– С. 296.

129. Кононов, В.К. Отбраковка потенциально-ненадежных интегральных микросхем с использованием радиационно-стимулирующего метода/ В.К. Кононов, В.Г. Малинин, Д.А. Оспищев, В.Д. Попов// В сб. Радиационно-надежностные характеристики изделий электронной техники в экстремальных условиях эксплуатации. Под редакцией Ю.Н. Торгашова СПб.: Издательство РНИИ «Электронстандарт».– 1994.- С. 96.

130. Орлов, В.И. Усовершенствованная методика формирования партий электронной компонентной базы с особыми требованиями качества/ В.И. Орлов, Д.В. Сташков, Л.А. Казаковцев, И.П. Рожнов, О.Б. Казаковцева, И.Р. Насыров// Современные наукоемкие технологии.- 2018.- № 1.- С. 37-42.

131. Osman, I.H. Metaheuristics: a bibliography/ I.H. Osman, G. Laporte// Ann. Oper. Res.- 1996.- Vol. 63.- P. 513-628.

132. Hansen, P. Variable Neighborhood Search/ P. Hansen, N. Mladenovic// Search Methodology/ E.K.Bruke, G.Kendall [eds.].-Springer US.- 2005.- P. 211-238, doi: 10.1007/0-387-28356-0_8.

133. Mladenovic, N. Variable neighborhood search/ N. Mladenovic, P. Hansen// Comput. Oper. Res.- 1997.- Vol. 24.- P. 1097-1100.

134. Hansen, P. Variable neighborhood search: principles and applications/ P. Hansen, N. Mladenovic// Eur. J. Oper. Res.- 2001.- Vol. 130.- P. 449–467.

135. Brimberg, J. A variable neighborhood algorithm for solving the continuous location-allocation problem/ J. Brimberg, N. Mladenovic// Stud. Locat. Anal.- 1996.- Vol. 10.- P. 1-12.

136. Hansen, P. Variable neighborhood decomposition search/ P. Hansen, N. Mladenovic, D. Perez-Brito// J. Heuristics.- 2001.- Vol. 7.- № 4.- P. 335-350.

137. Brimberg, J. Improvements and comparison of heuristics for solving the uncapacitated multisource Weber problem/ J. Brimberg, P. Hansen, N. Mladenovic, E. Taillard// Oper. Res.- 2000.- Vol. 48,- № 3.- P. 444-460.

138. Lopez, F.G. The parallel variable neighborhood search for the p-median problem/ F.G. Lopez, B.M. Batista, J. Moreno-Perez, M. Moreno-Vega// Res. Rep. Univ. of La Laguna, Spain.- 2000.

139. Кочетов, Ю.А. Локальный поиск с чередующимися окрестностями/ Ю.А. Кочетов, Н. Младенович, П. Хансен// Дискретн. анализ и исслед. опер. сер. 2.- 2003.- Т. 10- № 1.- С. 11–43.

140. Goldberg, D.E. Genetic algorithms in search, optimization, and machine learning/ D.E. Goldberg// Reading, MA: Addison-Wesley.- 1989.

141. Kazakovtsev, L. Genetic Algorithm with Fast Greedy Heuristic for Clustering and Location Problems/ L.A. Kazakovtsev, A.N. Antamoshkin// Informatica.-2014.- Vol. 38,- No. 3.- P. 229-240.

142. Казаковцев, Л.А. Метод жадных эвристик для систем автоматической группировки объектов: Дис… докт. техн. наук/ Л.А. Казаковцев// Красноярск.- 2016.- С. 429.

143. Семенкин, Е.С. Метод обобщенного адаптивного поиска для оптимизации управления космическими аппаратами: дис... д-ра техн. наук/ Е.С. Семенкин// САА. Красноярск.- 1997.- С. 400.

144. Коробейников, С.П. Методы многокритериальной оптимизации для задач синтеза управления сложными объектами: дис... канд. техн. наук/ С.П. Коробейников// ГХК Красноярск.- 1997.- С. 174.

145. Гарипов, В.Р. Многокритериальная оптимизация систем управления сложными объектами методами эволюционного поиска: дис... канд. техн. наук/ В.Р.Гарипов// САА. Красноярск.- 1999.- С. 138.

146. Семенкин, Е.С. Об эволюционных алгоритмах решения сложных задач оптимизации/ Е.С. Семенкин, А.В. Гуменникова, М.Н. Емельянова, Е.А. Сопов// Вестн. Сиб. гос. аэрокосмич. ун-та им. акад. М.Ф. Решетнева : сб. науч. тр./ под ред. проф. Г.П. Белякова Сиб. гос. аэрокосмич. ун-т. вып. 5. Красноярск.– 2003.- С. 14–23.

147. Казаковцев, Л.А. Модификация генетического алгоритма с жадной эвристикой для непрерывных задач размещения и классификации/

Л.А. Казаковцев, А.А. Ступина, В.И. Орлов// Системы управления и информационные технологии.- 2014.- вып. 2(56).- С. 35-39.

148. Казаковцев, Л.А. Дальнейшее развитие метода жадных эвристик для задач автоматической группировки объектов/ Л.А. Казаковцев, Д.В. Сташков, И.П. Рожнов, О.Б. Казаковцева// Системы управления и информационные технологии.- 2017.- № 4(70).- С. 34-40.

149. Orlov, V.I. Variable neighbourhood search algorithm for k-means clustering/ V.I. Orlov, L.A. Kazakovtsev, I.P. Rozhnov, N.A. Popov, V.V. Fedosov// IOP Conf. Series: Materials Science and Engineering.- 2018.- Vol. 450. Article ID 022035. DOI:10.1088/1757-899X/450/2/022035.

150. Kazakovtsev, L.A. Fast Deterministic Algorithm for EEE Components Classification/ L.A. Kazakovtsev, A.N. Antamoshkin, I.S. Masich// IOP Conf. Series: Materials Science and Engineering.– 2015.– Vol. 94.– article ID 012015.- P. 10. DOI: 10.1088/1757-899X/04/1012015.

151. Hansen, P. Solving large p-median clustering problems by primal dual variable neighborhood search/ P. Hansen, J. Brimberg, D. Urosevic, N. Mladenovic// Data Mining and Knowledge Discovery.- 2009.- 19,- No. 3.- P. 351–375.

152. Hansen, P. Variable neighborhood search for weighted maximum satisfiability problem/ P. Hansen, B. Jaumard, N. Mladenovic, A. Parreira// Les Cahiers du GERAD G-2000-62. Monreal. Canada,- 2000.

153. Рожнов, И.П. Алгоритм для задачи k-средних с рандомизированными чередующимися окрестностями/ И.П. Рожнов, Л.А. Казаковцев, М.Н. Гудыма, В.Л. Казаковцев // Системы управления и информационные технологии.- 2018.- № 3 (73).- С. 46-51.

154. Belacel, N. Fuzzy J-Means: a new heuristic for fuzzy clustering/ N. Belacel, P. Hansen, N. Mladenovic// Pattern Recognition.- 2002.- Vol. 35.- P. 2193–2200.

155. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml].

156. Clustering basic benchmark [http://cs.joensuu.fi/sipu/datasets].

157. https://www.kdd.org/kdd-cup/view/kdd-cup-2004/Tasks.

158. http://www.machinelearning.ru/wiki/index.php?title=Репозиторий_UCI.

159. Орлов, В.И. Алгоритм поиска в чередующихся окрестностях для задачи выделения однородных производственных партий электрорадиоизделий/ В.И. Орлов, И.П. Рожнов, В.Л. Казаковцев, М.Н. Гудыма// Решетневские чтения. Красноярск.- 2018.- Т. 1.- № 22.- С. 315-316.

160. https://developer.nvidia.com/cuda-zone.

161. David, Luebke How gpus work/ David Luebke, Greg Humphreys// Computer.- 40(2).- 2007.- P. 96–100

162. Zechner, M. Accelerating K-Means on the Graphics Processor via CUDA/ M. Zechner, M. Granitzer.

163. Нгуен Виет Хунг Нейросетевые алгоритмы для решения задач кодирования изображений с использованием технологии CUDA: дис… канд. техн. наук/ Нгуен Виет Хунг// Москва.- 2012.- С. 154.

164. Желтов, С.А. Эффективные вычисления в архитектуре CUDA в приложениях информационной безопасности: дис… канд. техн. наук/ С.А. Желтов// Москва.- 2014.- С 141.

165. Lutz Efficient k-Means on GPUs/ Lutz, Breß, Zeuch, Markl, Rabl// DaMoN’18. June 11. Houston. TX. USA.- 2018.

166. Rozhnov, I.P. Parallel implementation of the greedy heuristic clustering algorithms / L.A. Kazakovtsev, I.P. Rozhnov, E.A. Popov, M.V. Karaseva, A.A. Stupina // IOP Conf. Series: MIP: Engineering.- 2019.- Vol. 537.

167. Рожнов, И.П. Реализация жадных эвристических алгоритмов кластеризации для массивно-параллельных систем/ И.П. Рожнов, В.Л. Казаковцев// Системы управления и информационные технологии.- 2019.- № 2 (76).- С. 36-40.

168. Struyf, A. Clustering in an Object-Oriented Environment/ A. Struyf, M. Hubert, P. Rousseeuw// Journal of Statistical Software.- 1997.- Issue 1 (4). P. 1-30.

169. Kaufman, L. Finding groups in data: an introduction to cluster analysis/ L. Kaufman, P.J. Rousseeuw// New York:Wiley.- 1990.- P. 368.

170. Moreno-Perez, J.A. A Parallel Genetic Algorithm for the Discrete p-Median Problem/ J.A. Moreno-Perez, J.L. Roda Garcia, J.M. Moreno-Vega// Studies in Location Analysis.- 1994.- Issue 7.- P. 131-141.

171. Wesolowsky, G. The Weber problem: History and perspectives // Location Science.- 1993.- No. 1.- P. 5-23.

172. Drezner, Z. A Trajectory Method for the Optimization of the Multifacility Location Problem with lp Distances/ Z. Drezner, G.O. Wesolowsky// Management Science.- 1978.- Vol. 24.- P. 1507–1514.

173. Deza, M.M. Metrics on Normed Structures/ M.M. Deza, E. Deza// Encyclopedia of Distances.- Berlin Heidelberg: Springer.- 2013.- P. 89-99. DOI: 10.1007/978-3-642-30958-85.

174. Nicholson, T. A. J. A sequential method for discrete optimization problems and its application to the assignment, traveling salesman and tree scheduling problems/ T. Nicholson// J. Inst. Math. Appl.- 1965.- Vol. 13.- P. 362-375.

175. Page E.S. On Monte Carlo methods in congestion problems. I: Searching for an optimum in discrete situations/ E.S. Page// Oper. Res.- 1965.- Vol. 13,- № 2.- P. 291-299.

176. Kernighan, B.W. An efficient heuristic procedure for partitioning graphs/ B.W. Kernighan, S. Lin// Bell Syst. Tech. J.- 1970.- Vol. 49.- P. 291-307.

177. Гастригин, Л.А. Случайный поиск — специфика, этапы истории и предрассудки/ Л.А. Гастригин// Вопросы кибернетики. М.: Науч. совет по комплексной проблеме «Кибернетика» АН СССР.- 1978.- Вып. 33.- С. 3-16.

178. Рожнов, И.П. Алгоритмы с чередованием жадных эвристических процедур для дискретных задач кластеризации/ И.П. Рожнов// Системы управления и информационные технологии.- 2019.- № 1 (75).- С. 49-55.

179. Sheng, W. A genetic k-medoids clustering algorithm/ W. Sheng, X. Liu// Journal of Heuristics.- 2006.-Vol. 12,- No. 6.- P. 447-466.

180. Черезов, Д.С. Обзор основных методов классификации и кластеризации данных/ Д.С. Черезов, Н.А. Тюкачев // Вестник Воронеж. гос. ун-та. Сер. Системный анализ и информационные технологии.- 2009.- Вып. 2.

181. Kazakovtsev, L. Algorithms with Greedy Heuristic Procedures for Mixture Probability Distribution Separation/ L. Kazakovtsev, D. Stashkov, M. Gudyma, V. Kazakovtsev// Yugoslav Journal of Operations Research.- 2019.- Vol. 29.- P. 51-67.

182. Сташков, Д.В. Алгоритм для серии задач разделения смеси распределений / Д.В. Сташков, М.Н. Гудыма, Л.А. Казаковцев, И.П. Рожнов, В.И. Орлов // Решетневские чтения.– 2017.- Т. 1.- № 21.- С. 327-328.

183. Казаковцев, Л.А. Усовершенствованный СЕМ-алгоритм для данных большой размерности / Л.А. Казаковцев, И.П. Рожнов, П.Ф. Шестаков // Наука и образование: опыт, проблемы, перспективы развития. Красноярск. КГАУ. - 2019.- С. 244-247.

184. Rozhnov, I. Improved Classification EM algorithm for the Problem of Separating Semiconductor Device Production Batches / I. Rozhnov, L. Kazakovtsev, E. Bezhitskaya, S. Bezhitskiy // IOP Conf. Series: MIP: Engineering.- 2019.- Vol. 537.

185. Рожнов, И.П. Усовершенствованный алгоритм разделения смеси распределений для данных большой размерности/ Л.А. Казаковцев, Д.В. Сташков, О.Б. Казаковцева, И.П. Рожнов, А.В. Медведев// Лесной и химический комплексы - проблемы и решения. (7 декабря 2017). Красноярск.- 2017.- С. 502-505.

186. Ghosh, J. Cluster ensembles/ J. Ghosh, A. Acharya// Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery.- 2011.- Vol. 1(4).- P. 305−315.

187. Бериков, В.В. Классификация данных с применением коллектива алгоритмов кластерного анализа/ В.В. Бериков// Знания-Онтологии-Теории (ЗОНТ-2015).- 2015.- С. 29-38.

188. Rozhnov, I. Ensembles of clustering algorithms for problem of detection of homogeneous production batches of semiconductor devices/ I. Rozhnov, V. Orlov, L. Kazakovtsev// В сборнике: CEUR Workshop Proceedings Сер. OPTA-SCL 2018 - Proceedings of the School-Seminar on Optimization Problems and their Applications. CEUR-WS. -2018.- Vol. 2098.- P. 338-348.

189. Rozhnov, I.P. Increase in Accuracy of the Solution of the Problem of Identification of Production Batches of Semiconductor Devices/ I.P. Rozhnov, V.I. Orlov, L.A. Kazakovtsev// 14th International Scientific-Technical Conference on

Actual Problems of Electronic Instrument Engineering. APEIE.- 2018.- P. 363-367. DOI: 10.1109/APEIE.2018.8546294.

190. Рожнов, И.П. Составление оптимальных ансамблей алгоритмов кластеризации/ И.П. Рожнов, В.И. Орлов, М.Н. Гудыма, В.Л. Казаковцев// Системы управления и информационные технологии.- 2018.- № 2 (72).- С. 31-35.

191. Hamiter, L. The History of Space Quality EEE Parts in the United States/ L. Hamiter// ESA Electronic Components Conference.- Noordwijk. The Netherlands: ESTEC.- 1990.- Nov 12-16.- P. 503-508.

192. Kirkconnell, C.S. High Efficiency Digital Cooler Electronics for Aerospace Applications/ C.S. Kirkconnell, T.T. Luong, L.S. et al. Shaw// Proc. SPIE 9070. Infrared Technology and Applications XL.- Baltimore: SPIE.- 2014.- Article 90702Q.- P. 15. [Электронный ресурс] Режим доступа DOI:10.1117/12.2053075 (дата обращения 01.09.2015).

193. Ooi, M.P.-L. Getting more from the semiconductor test: Data mining with defect-cluster extraction/ M.P.-L. Ooi [et al.]// IEEE Trans.Instrum. Meas.- 2011.- Vol. 60,- No. 10.- P. 3300-3317.

194. Kwon, Y. Data mining approaches for modeling complex electronic circuitdesign activities/ Y. Kwon, O.A. Omitaomu, G.-N. Wang// Computer & Industrial Engineering.- 2008.- Vol. 54.- P. 229-241.

195. Khamidullina, N.M. Predictions of integrated circuit serviceability in space radiation fields/ N.M. Khamidullina [et al.]// RadiationMeasurements.-1999.- Vol. 30.- P. 633-638.

196. Zhao, X. Defect Pattern Recognition on Nano/Micro Integrated Circuits Wafer/ X. Zhao, L. Cui// Proceedings of the 3rdIEEE Int. Conf. on Nano/Micro Engineered and Molecular Systems (Sanya, China,January 6-9, 2008). Sanya, China: [s.n.].- 2008.- P. 519-523.

197. Bechow, L. An Improved Method for Automatic Detection and Location of Defects in Electronic Components Using Scanning Ultrasonic Microscopy/ L. Bechow [et al.]// IEEE Transactions on Instrumentation and Measurement.- 2003.- Vol. 52,- No. 1.- P. 135-142.

198. Ooi, M.P.-L. Identifying Systematic Failures on Semiconductor Wafers Using ADCAS/ M.P.-L. Ooi [et al.]// Design &Test. IEEE.- 2013.- Vol.30 (5).- P. 44-53.

199. Анисимов, В.Г. Исследование сложных дефектов упаковки в монокристаллах кремния/ В.Г. Анисимов, Л.Н. Данильчук, Ю.А. Дроздов [и др.] // Поверхность. Рентгеновские, синхротронные и нейтронные исследования.- 2004.– № 11.- С. 74–81.

200. Орлов, В.И. Фирменный стиль: надежность и качество/ В.И. Орлов, В.В. Федосов// Петербургский журнал электроники.- 2010.- вып. 1(62).- С. 55-64.

201. Орлов, В.И. К вопросу о сертификации ЭРИ ИП [Электронный ресурс]/ В.И. Орлов, В.В. Федосов// Научно-технический семинар «Обеспечение предприятий радиоэлектронной промышленности надежной электронной компонентной базой. Вопросы импортозамещения».- М.: ЗАО "Тестприбор".- 2014.- Режим доступа: URL http://www.makd.ru/media/downloads/sections/electro/230714/on_the_question_of_certification_esi_ip.pdf (Дата обращения: 03.05.2015).

202. Федосов, В.В. Вопросы обеспечения работоспособности электронной компонентной базы в аппаратуре космических аппаратов: учеб. Пособие/ В.В. Федосов// Сиб. гос. аэрокосмич. Ун-т. – Красноярск.- 2015. – 68 С.

203. ОСТ В 11 0998-99. Микросхемы интегральные. Общие технические условия.

204. MIL-PRF-38535 – Performance Specification: Integrated Circuits (Micricircuit) Manufacturing, General Specifications for. Department of Defence, United States of America. – 2007.

205. Коплярова, Н.В. О непараметрических моделях в задаче диагностики электрорадиоизделий. Заводская лаборатория/ Н.В. Коплярова, В.И. Орлов, Н.А. Сергеева, В.В. Федосов// Диагностика материалов № 7.- 2014.- Том 80.- С. 73-77.

206. Казаковцев, Л.А. Быстрый детерминированный алгоритм для классификации электронной компонентной базы по критерию равнонадежности/

Л.А. Казаковцев, И.С. Масич, В.И. Орлов, В.В. Федосов// Системы управления и информационные технологии.- 2015. – Вып. 4(62). - С. 39- 44.

207. Данилин, Н. Проблемы применения современной индустриальной электронной компонентной базы иностранного производства в ракетно-космической технике/ Н. Данилин, С. Белослудцев // Современная электроника.– 2007.– вып. 7.– С. 8-12.

208. Калашников, О.А Функциональный контроль микропроцессоров при проведении радиационных испытаний / О.А.Калашников, П.В. Некрасов, А.А.Демидов // Приборы и техника эксперимента.- 2009. - № 2. - С. 48.

209. Qualified manufacturers list of products qualified under performance specification MIL-PRF-19500 Semiconductor Devices, General Specification for. Department of Defense.- 2010.- P. 188.

210. Орлов, В.И. Усовершенствованный метод формирования производственных партий электронной компонентной базы с особыми требованиями качества/ В.И. Орлов, Д.В Сташков., Л.А. Казаковцев, И.П. Рожнов, И.Р. Насыров, О.Б. Казаковцева// Современные наукоемкие технологии.- 2018.- № 1.- С. 37-42.

211. Янко, Э.А. Производство алюминия: Пособие для мастеров и рабочих цехов электролиза алюминиевых заводов / Э.А. Янко - Санкт-Петербург, 2007 г. – 69 с.

212. Савин, А.Н. Качество обожженных анодов, поставляемых на отечественные алюминиевые заводы, их расход в процессе электролиза и оценка эффективности использования / А.Н. Савин // Цв. металлы. 2007. № 4. С. 84-87.

213. Клюшин, А.Ю. Совершенствование и новые технологии в производстве алюминия / А.Ю.Клюшин, Д.Т.Дим, В.А.Павлов // Новое слово в науке: перспективы развития. 2016. № 4-1 (10). С. 215-217.

214. Лайнер, Ю.А. Перспективные способы получения алюминия и соединений на его основе / Ю.А.Лайнер, Г.А.Мильков, Е.Н. Самойлов // Цв. металлы. 2012. № 6. С. 42-47.

215. Орлов, В.И. Алгоритмическое обеспечение поддержки принятия решений по отбору изделий микроэлектроники для космического приборостроения: монография/ В. И. Орлов, Л. А. Казаковцев, И. С. Масич, Д. В. Сташков// Сиб. гос. аэрокосмич. ун-т. – Красноярск.- 2017. – С. 250.

216. Ackermann, M.R. A Clustering Algorithm for Data Streams J/ M.R. Ackermann et al 2012 K.M. Stream// Exp.Algorithmics 17 2.4:2.1-2.30.

217. Kanungo, T. Computing nearest neighbors for moving points and applications to clustering Proc. of the tenthannual ACM-SIAM symp. on Discrete algorithms/ T. Kanungo, D.M. Mount, N.S. Netanyahu, C.D. Piatko, R. Silverman, A.Y. Wu// Society for Industrial and AppliedMathematics.- 1999.- P. 931-932.

218. Kazakovtsev, L.A. Modification of the genetic algorithm withgreedy heuristic for continuous location and classification problems / L.A. Kazakovtsev, A.A. Stupina, V.I. Orlov // Sistemy upravleniya i informatsionnye tekhnologii. - 2(56).- 2014.- P. 31-34.

219. Orlov, V.I. Fuzzy clustering of EEE components for space industry/ V.I. Orlov, D.V. Stashkov, L.A. Kazakovtsev, A.A. Stupina// IOP Conference Series: Materials Science and Engineering.- 2016. -Vol. 155. Article ID 012026. http://iopscience.iop.org/article/10.1088/1757-899X/155/1/012026/pdf.

220. Kazakovtsev, L.A. Improved model for detection of homogeneous production batches of electronic components/ L.A. Kazakovtsev, V.I. Orlov, D.V. Stashkov, A.N. Antamoshkin, I.S. Masich// IOP Conference Series: Materials Science and Engineering.- 2017.

221. Казаковцев, Л.А. Метод жадных эвристик для задач размещения/ Л.А. Казаковцев, А.Н. Антамошкин// Вестник СибГАУ.–2015.–№ 2.–С. 317-325.

222. Шкаберина, Г.Ш. Факторный анализ с использованием матрицы Спирмена в задаче автоматической группировки электрорадиоизделий по производственным партиям / Г.Ш. Шкаберина, В.И. Орлов, Е.М. Товбис, Л.А. Казаковцев// Системы управления и информационные технологии, № 1 (75).- 2019. – С. 91-96.

223. Shkaberina, G.Sh. Estimation of the impact of semiconductor device parameters on the accuracy of separating a mixed production batch / G.Sh. Shkaberina, V.I. Orlov, E.M. Tovbis, E.V. Sugak, L.A. Kazakovtsev // IOP Conf. Series: MIP: Engineering.- 2019.- Vol. 537.

224. Uberla, K. Factorenanalyse / K. Uberla // Berlin: Springer-Verlag, -1977. – P. 399.

225. Calinski, T. A dendrite method for cluster analysis / T. Calinski, J. Harabasz // Communications in Statistics.- 1974.- Vol. 3.- P. 1-27. doi: 10.1080/ 03610927408827101.

226. Davies, D.L. A Cluster Separation Measure / D.L. Davies, D.W. Bouldin // IEEE Transactions on Pattern Analysis and Machine Intelligence.- 1979.- PAMI-1 (2).- P. 224–227.

227. Krzanowski, W. A criterion for determining the number of groups in a dataset using sum of squares clustering/ W. Krzanowski, Y.Lai// Biometrics.- 1985.- No. 44.- P. 23–34.

228. Hartigan, J.A. Clustering Algorithms/ J.A. Hartigan// New York: Wiley.- 1975.- P. 369.

229. Schwarz, G. Estimating the Dimension of a Model/ G. Schwarz// Annals of Statistics. 6.- 1978.- No. 2.- P. 461-464. doi:10.1214/aos/1176344136.

230. Tibshirani, R. Estimating the number of clusters in a data set via the gap statistic/ R. Tibshirani, G. Walther, T. Hastie// Journal of the Royal Statistical Society,- 2001.- Vol. 63.- P. 411–423.

231. Akaike, H. A new look at the statistical model identification/ H. Akaike// IEEE Transactions on Automatic Control,- 1974.- Vol. 19 (6).- P. 716–723. doi:10.1109/TAC.1974.1100705.

232. Rousseeuw, P. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis/ P. Rousseeuw// Journal of Computational and Applied Mathematics.- 1987.- Vol. 20.- P. 53-65.

233. Лосева, Е.Д. Алгоритм автоматизированного формирования ансамблей нейронных сетей для решения сложных задач интеллектуального анализа данных

/ Е.Д. Лосева, А.Н. Антамошкин // Известия ТулГУ. Технические Науки.- 2017.- № 4.- С. 234-242.

234. Rozhnov I.P. Clustering algorithms and their ensembles for homogeneous production batches of semiconductor devices / I.P. Rozhnov // Молодежь. Общество. Современная наука, техника и инновации. Красноярск.- 2018.- № 17.- С. 281-282.

235. Рожнов, И.П. Выделение партий электрорадиоизделий ансамблями алгоритмов кластеризации / И.П. Рожнов, Л.А. Казаковцев, В.И. Орлов // В книге: Проблемы оптимизации и их приложения. Тезисы докладов VII Международной конференции: памяти профессора А.А. Колоколова.- 2018.- С. 90.

236. Орлов, В.И. Анализ алгоритмов кластеризации и их ансамблей для задачи выделения производственных партий электрорадиоизделий/ В.И. Орлов, И.П. Рожнов, О.Б. Казаковцева, Л.А. Казаковцев// Экономика и менеджмент систем управления.- 2017.- № 4.4 (26).- С. 486-492.

237. Бочкарёв, П.В. Разработка ансамбля алгоритмов кластеризации на основе изменяющихся метрик расстояний/ П.В. Бочкарёв, В.С. Киреев // Труды XVIII Международной конференции DAMDID/RCDL’2016 «Аналитика и управление данными в областях с интенсивным использованием данных». Ершово.- 11-14 октября 2016.- С. 32-36.

238. Рожнов, И.П. Формирование электронной компонентной базы с особыми требованиями качества с применением ансамблей алгоритмов кластеризации / И.П. Рожнов, В.И. Орлов, Л.А. Казаковцев // Решетневские чтения. Красноярск.- 2018.- Т. 1.- № 22.- С. 322-324.

239. Koza, J.R. Genetic Programming/ J.R. Koza// On the Programming of Computers by Means of Natural Selection: MIT Press.- 1992.- P. 109 – 120.

240. Huang, J.-J. Two-stage genetic programming (2SGP) for the credit scoring model/ J.-J. Huang, G.-H. Tzeng, Ch.-Sh. Ong// Applied Mathematics and Computation.- 2006.- No.- 174 (2).- P. 1039-1053.

241. Integer Magoulas, G.D. Improving the Convergence of the Backpropagation Algorithm Using Learning Rate Adaptation Methods/ G.D. Integer Magoulas,

M.N. Vrahatis, G.S. Androulaki// Neural Computation.- 1999.- GR-261.10.- P. 1769-1796.

242. Ashish, G. Evolutionary Algorithm for MultiCriterion Optimization: A Survey/ G. Ashish, D. Satchidanada// International Journal of Computing & Information Science.- 2004.- Vol. 2.- No. 1.- P. 43 – 45.

243. Крутиков, В.Н. Исследование субградиентных методов обучения нейронных сетей / В.Н.Крутиков, Д.В.Арышев // Вестник Кемеровского государственного университета. 2004. № 1 (17). С. 119-123.

244. Berkhin, P. A survey of clustering data mining techniques/ P. Berkhin// Grouping multidimensional data.- Springer 2006.- P. 25-71.

245. MacQueen, J. Some methods for classification and analysis of multivariate observations/ J. MacQueen// Proc. 5th Berkeley Symp. on Math. Statistics and Probability.- 1967.- P. 281—297.

246. Bhattacharya, A. A tight lower bound instance for k-means++ in constant dimension/ A. Bhattacharya, R. Jaiswal, N. Ailon// Theory and Applications of Models of Computation. Springer.- 2014.- P. 7-22.

247. Arthur, D. How slow is the k-means method? / D. Arthur, S. Vassilvitskii // Proceedings of the twenty-second annual symposium on Computational geometry. ACM.- 2006.- P. 144-153.

248. Hamerly, G. Accelerating Lloyd's algorithm for k-means clustering/ G. Hamerly, J. Drake// Partitional Clustering Algorithms. Springer.- 2014.- P .41-78.

249. Dhillon, I.S. Kernel k-means: spectral clustering and normalized cuts/ I.S. Dhillon, Y. Guan, B. Kulis// Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '04). ACM.- 2004. New York. USA. P. 551-556. DOI=http://dx.doi.org/10.1145/1014052.1014118.

250. Dempster, A. Maximum likelihood estimation from incomplete data / A. Dempster, N. Laird, D. Rubin // Journal of the Royal Statistical Society, Series B.- 1977. Vol. 39.- P. 1-38.

251. Антамошкин, А.Н. Алгоритм размещения с метрикой Москвы-Карлсруэ/ А.Н. Антамошкин, Л.А. Казаковцев// Системы управления и информационные технологии.- 2012.- Т. 49.- № 3.1.- С. 111-115.

252. Казаковцев, Л.А. Алгоритм для задачи размещения, основанной на угловом расстоянии/ Л.А. Казаковцев // Фундаментальные исследования.- 2012.- № 9-4.- С. 918-923.

253. Khozeimeh, F. An expert system for selecting wart treatment method/ F. Khozeimeh, R. Alizadehsani, M. Roshanzamir, A. Khosravi, P. Layegh, S. Nahavandi// Computers in Biology and Medicine.- 2017.- Vol. 81. 2/1/.- P. 167-175.

254. Khozeimeh, F. Intralesional immunotherapy compared to cryotherapy in the treatment of warts/ F. Khozeimeh, F. Jabbari Azad, Y. Mahboubi Oskouei, M. Jafari, S. Tehranian, R. Alizadehsani, et al.// International Journal of Dermatology.- 2017. DOI: 10.1111/ijd.13535.

255. Орлов, В.И. Система составления оптимальных ансамблей алгоритмов кластеризации для задачи выделения производственных партий электрорадиоизделий/ В.И. Орлов, И.П. Рожнов, Л.А. Казаковцев, С.М. Голованов// М.: РОСПАТЕНТ.- 2019. Свидетельство о государственной регистрации программы для ЭВМ № 2019610095 от 09.01.2019.

ПРИЛОЖЕНИЕ А Сравнительный анализ вычислительных экспериментов различных алгоритмов

В Таблицах А.1-А.3 приведены сравнительные результаты вычислительных экспериментов полученных нами и проведенных ранее вычислительных экспериментов над наборами данных электрорадиоизделий различными модификациями генетического алгоритма. Выполнено сравнение результатов работы новых (k-GH-VNS1, k-GH-VNS2, k-GH-VNS3, k-GH-VNS1-RND, k-GH-VNS2-RND, k-GH-VNS3-RND, j-means-GH-VNS1, j-means-GH-VNS2), известных алгоритмов (k-средних, j-means) и различных модификаций генетического алгоритма по значению целевой функции.

Для расчетов были использованы сборные партии электрорадиоизделий:

- 1526TL1 - 3 партии (1234 векторов данных, каждый размерностью 157);
- 2Д522Б - 5 партий (3711 векторов данных, каждый размерностью 10);
- Н5503ХМ1 - 5 партий (3711 векторов данных, каждый размерностью 229).

В Таблицах А.1-А.3 были использованы следующие аббревиатуры и сокращения [142]: ГА - генетический алгоритм, ЖЭ - жадная эвристика, ГАЖЭ – генетический с жадной эвристикой с вещественным алфавитом, ЛП - локальный поиск, ГА ФП – генетический алгоритм с рекомбинацией подножеств фиксированной длины [179], IBC – Information Bottleneck Clustering, ЖЛ – мультистарт жадной эвристики с включенным локальным поиском, ALA мультистарт – мультистарт ALA-процедуры.

Лучшие значения целевой функции (минимальное значение, среднее значение и среднеквадратичное отклонение) выделены полужирным курсивом.

Таблица А1 - Результаты вычислительных экспериментов над производственными партиями электрорадиоизделий 1526TL1 (10 кластеров, 1 минута, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Средне-квадратичное отклонение
j-means	***43 841,97***	43 843,51	43 842,59	0,4487
k-средних	43 842,10	43 844,66	43 843,38	0,8346
k-GH-VNS1	***43 841,97***	43 844,18	43 842,34	0,9000
k-GH-VNS2	***43 841,97***	43 844,18	43 843,46	1,0817
k-GH-VNS3	***43 841,97***	43 842,10	***43 841,99***	0,0424
k-GH-VNS1-RND	нет резуль	тата		
k-GH-VNS2-RND	нет резуль	тата		
k-GH-VNS3-RND	нет резуль	тата		
j-means-GH-VNS1	***43 841,97***	43 841,97	***43 841,97***	***0,0000***
j-means-GH-VNS2	***43 841,97***	43 844,18	43 842,19	0,6971
ГАЖЭ+ЛП	43 842,10	43 845,73	43 843,72	1,3199
ГАЖЭ вещ., σ e =0.25	***43 841,98***	43 844,18	43 842,6	0,6762
ГАЖЭ вещ.частичн., σ e =0.25	***43 841,98***	43 841,98	***43 841,98***	1,53E-11
ГА ФП	***43 841,98***	43 842,34	43 842,10	0,0945
ГА классич.	43 842,10	43 842,88	43 842,44	0,2349
IBC, σ e =0.25	нет резуль	тата		
Детерм. ЖЭ, σ e =0.25	45 113,56	45 113,56	45 113,56	***0,0000***
Детерм. ЖЭ, σ e =0.001	45 021,21	45 021,21	45 021,21	***0,0000***
IBC, σ e =0.001	нет резуль	тата		
ЖЭ адапт. σ e =0.25	***43 841,98***	43 842,88	43 842,40	0,4508
ЖЭ адапт. σ e =0.001	43 842,75	43 844,18	43 843,92	0,5366
ЖЭ, σ e =0.25, β=0.5	***43 841,98***	43 842,74	43 842,21	0,2903
ЖЭ, σ e =0.25, β=1	***43 841,98***	43 843,78	43 842,49	0,6596
ЖЭ, σ e =0.25, β=3	43 842,75	43 843,52	43 843,32	0,3452
ЖЭ, σ e =0.001, β=0.5	***43 841,98***	43 842,59	43 842,12	0,2180
ЖЭ, σ e =0.001, β=1	43 842,10	43 844,18	43 843,32	0,9767
ЖЭ, σ e =0.001, β=3	43 844,18	43 844,18	43 844,18	***0,0000***
ЖЛ, σ e =0.25, β=0.5	43 842,74	43 843,52	43 843,09	0,3987
ЖЛ, σ e =0.25, β=1	***43 841,98***	43 843,52	43 842,58	0,5319
ЖЛ, σ e =0.25, β=3	43 842,74	43 845,40	43 843,29	0,9700
ЖЛ, σ e =0.001, β=0.5	***43 841,98***	43 842,94	43 842,51	0,4630
ЖЛ, σ e =0.001, β=1	43 842,34	43 844,18	43 842,92	0,5839
ЖЛ, σ e =0.001, β=3	43 842,74	45 118,74	44 191,73	596,6553
ALA мультистарт	***43 841,98***	43 842,74	43 842,36	0,3165

Таблица А2 - Результаты вычислительных экспериментов над производственными партиями электрорадиоизделий 2Д522Б (10 кластеров, 1 минута, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Средне-квадратичное отклонение
j-means	7 719,98	7 720,74	7 720,36	1,0174
k-средних	7 718,57	7 722,91	7 720,74	2,8714
k-GH-VNS1	7 716,88	7 717,18	7 717,03	0,0738
k-GH-VNS2	7 722,32	7 726,42	7 724,37	1,8752
k-GH-VNS3	7 722,81	7 725,22	7 724,51	1,3946
k-GH-VNS1-RND	нет резуль	тата		
k-GH-VNS2-RND	нет резуль	тата		
k-GH-VNS3-RND	нет резуль	тата		
j-means-GH-VNS1	7 717,22	7 721,40	7 719,81	1,7851
j-means-GH-VNS2	7 717,90	7 720,14	7 719,92	1,4016
ГАЖЭ+ЛП	***7 714,13***	7 715,50	***7 714,61***	0,3837
ГАЖЭ вещ., σ e =0.25	***7 714,15***	7 714,77	***7 714,66***	0,1954
ГАЖЭ вещ.частичн., σ e =0.25	***7 714,15***	7 714,41	***7 714,29***	0,0899
ГА ФП	***7 714,14***	7 714,29	***7 714,22***	***0,0612***
ГА классич.	***7 714,14***	7 714,30	***7 714,21***	***0,0678***
IBC, σ e =0.25	нет резуль	тата		
Детерм. ЖЭ, σ e =0.25	7 902,21	7 902,21	7 902,21	***0,0000***
Детерм. ЖЭ, σ e =0.001	нет резуль	тата		
IBC, σ e =0.001	нет резуль	тата		
ЖЭ адапт. σ e =0.25	***7 714,24***	7 714,81	***7 714,61***	0,2481
ЖЭ адапт. σ e =0.001	***7 714,78***	7 725,76	7 717,91	5,3185
ЖЭ, σ e =0.25, β=0.5	***7 714,22***	7 714,83	***7 714,64***	0,2521
ЖЭ, σ e =0.25, β=1	***7 714,26***	7 714,79	***7 714,61***	0,2266
ЖЭ, σ e =0.25, β=3	***7 714,21***	7 714,48	***7 714,29***	0,0851
ЖЭ, σ e =0.001, β=0.5	***7 714,34***	7 725,18	7 716,23	3,9519
ЖЭ, σ e =0.001, β=1	***7 714,77***	7 715,56	***7 714,90***	0,2918
ЖЭ, σ e =0.001, β=3	***7 714,55***	7 714,77	***7 714,73***	0,0849
ЖЛ, σ e =0.25, β=0.5	***7 714,26***	7 727,78	7 716,28	5,0695
ЖЛ, σ e =0.25, β=1	***7 714,29***	7 725,63	7 716,01	4,2413
ЖЛ, σ e =0.25, β=3	***7 714,29***	7 727,55	7 716,49	4,8981
ЖЛ, σ e =0.001, β=0.5	***7 714,14***	7 714,51	***7 714,36***	0,1316
ЖЛ, σ e =0.001, β=1	***7 714,28***	7 725,63	7 715,99	4,2505
ЖЛ, σ e =0.001, β=3	***7 714,49***	7 731,48	7 722,03	7,2419
ALA мультистарт	***7 714,14***	7 714,48	***7 714,26***	0,0982

Таблица А3 - Результаты вычислительных экспериментов над производственными партиями электрорадиоизделий Н5503ХМ1 (10 кластеров, 1 минута, 30 попыток)

Алгоритм	Значение целевой функции			
	Min (рекорд)	Max	Среднее	Средне-квадратичное отклонение
j-means	43 675,96	43 681,52	43 678,74	1,4126
k-средних	43 675,90	43 684,88	43 679,77	2,8062
k-GH-VNS1	***43 671,89***	43 671,89	***43 671,89***	***0,0000***
k-GH-VNS2	43 672,24	43 674,44	43 673,34	1,0476
k-GH-VNS3	43 672,84	43 675,76	43 674,30	1,5916
k-GH-VNS1-RND	нет резуль	тата		
k-GH-VNS2-RND	нет резуль	тата		
k-GH-VNS3-RND	нет резуль	тата		
j-means-GH-VNS1	***43 671,89***	43 671,89	***43 671,89***	***0,0000***
j-means-GH-VNS2	43 673,14	43 675,56	43 674,35	0,9162
ГАЖЭ+ЛП	43 702,28	43 766,87	43 739,69	20,3107
ГАЖЭ вещ., σ e =0.25	43 678,79	43 693,63	43 687,01	4,5961
ГАЖЭ вещ.частичн., σ e =0.25	43 675,79	43 686,87	43 680,82	3,3026
ГА ФП	43 708,14	43 736,26	43 716,26	8,4025
ГА классич.	43 703,31	43 724,42	43 715,80	6,1660
IBC, σ e =0.25	нет резуль	тата		
Детерм. ЖЭ, σ e =0.25	43 830,25	43 830,25	43 830,25	***0,0000***
Детерм. ЖЭ, σ e =0.001	44 573,13	44 573,13	44 573,13	***0,0000***
IBC, σ e =0.001	нет резуль	тата		
ЖЭ адапт. σ e =0.25	нет резуль	тата		
ЖЭ адапт. σ e =0.001	43 684,45	43 693,51	43 691,02	3,087926
ЖЭ, σ e =0.25, β=0.5	43 692,04	43 711,26	43 699,21	6,032778
ЖЭ, σ e =0.25, β=1	43 684,45	43 703,25	43 691,72	6,898894
ЖЭ, σ e =0.25, β=3	43 680,28	43 700,12	43 688,81	6,230507
ЖЭ, σ e =0.001, β=0.5	43 694,11	43 719,47	43 704,39	7,696556
ЖЭ, σ e =0.001, β=1	43 684,19	43 703,13	43 691,67	7,368125
ЖЭ, σ e =0.001, β=3	43 683,36	43 690,45	43 686,30	2,600117
ЖЛ, σ e =0.25, β=0.5	43 705,63	43 733,45	43 717,25	11,08307
ЖЛ, σ e =0.25, β=1	43 702,32	43 734,84	43 714,63	12,79796
ЖЛ, σ e =0.25, β=3	43 692,50	43 738,93	43 720,10	17,90737
ЖЛ, σ e =0.001, β=0.5	43 707,93	43 740,98	43 720,43	10,20816
ЖЛ, σ e =0.001, β=1	43 695,14	43 727,59	43 713,57	11,27041
ЖЛ, σ e =0.001, β=3	43 703,31	43 760,96	43 723,19	20,38476
ALA мультистарт	43 701,35	43 753,06	43 735,46	18,04498

ПРИЛОЖЕНИЕ Б Акты об использовании результатов исследования

РУСАЛ

САЯНОГОРСКИЙ АЛЮМИНИЕВЫЙ ЗАВОД

« 16 » мая 2019 г.
№_____

АКТ
о внедрении результатов диссертационного исследования
Рожнова Ивана Павловича

Настоящим актом подтверждается, что в составе "Автоматизированной системы управления технологическим процессом производства анодов", используемой на АО «РУСАЛ Саяногорск» был внедрен в опытную эксплуатацию новый подход к разработке алгоритмов автоматической группировки, основанных на параметрических оптимизационных моделях, с комбинированным применением алгоритмов поиска в чередующихся рандомизированных окрестностях и жадных агломеративных эвристических процедур.

Применение новых алгоритмов поиска в чередующихся рандомизированных окрестностях с использованием вышеуказанного подхода (в особенности параллельных алгоритмов с жадной агломеративной эвристической процедурой для больших задач автоматической группировки, адаптированных к архитектуре CUDA), разработанных в рамках диссертационного исследования соискателя учёной степени кандидата технических наук Рожнова Ивана Павловича, позволили повысить стабильность результатов оценки партий «зеленых» анодов, одновременно снизив временные затраты и требования к вычислительным ресурсам.

И.О. Управляющего директора
АО РУСАЛ Саяногорск Д.В. Муравьев

Акционерное общество «РУСАЛ Саяногорский Алюминиевый Завод»
(АО «РУСАЛ Саяногорск»)
территория Промплощадка, г.Саяногорск, Республика Хакасия, 655603
Телефон: (39042) 2-11-01, Факс: (39042) 7-39-05

АКТ

о внедрении результатов диссертационного исследования

Рожнова Ивана Павловича

Настоящим актом подтверждается, что, процедура составления оптимальных ансамблей алгоритмов автоматической группировки с совместным применением генетического алгоритма метода жадных эвристик и согласованной матрицы бинарных разбиений, а также новый подход к разработке алгоритмов автоматической группировки, основанных на параметрических оптимизационных моделях, с совместным применением алгоритмов поиска в чередующихся рандомизированных окрестностях и жадных агломеративных эвристических процедур, были использованы при разработке системы составления оптимальных ансамблей моделей кластеризации для задачи выделения производственных партий электрорадиоизделий и успешно используются в деятельности АО "Испытательный технический центр - НПО ПМ" (г. Железногорск).

Применение новых алгоритмов поиска в чередующихся рандомизированных окрестностях с использованием вышеуказанного подхода и внедрение системы составления оптимальных ансамблей моделей кластеризации для задачи выделения производственных партий электрорадиоизделий, разработанных в рамках диссертационного исследования соискателя учёной степени кандидата технических наук Рожнова Ивана Павловича, позволили повысить точность решения задачи разделения сборной партии электрорадиоизделий космического применения на однородные партии, что, в свою очередь повышает эффективность проводимого разрушающего физического анализа (РФА), гарантируя отбор экземпляров для РФА из каждой однородной партии.

Директор АО "ИТЦ - НПО ПМ", к.т.н.

В.И. Орлов

02.04.2019

ПРИЛОЖЕНИЕ В Свидетельство о государственной регистрации программы для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2019610095

Система составления оптимальных ансамблей алгоритмов кластеризации для задачи выделения производственных партий электрорадиоизделий

Правообладатель: ***Акционерное общество "Испытательный технический центр-НПО ПМ" (RU)***

Авторы: ***Орлов Виктор Иванович (RU), Рожнов Иван Павлович (RU), Казаковцев Лев Александрович (RU), Голованов Сергей Михайлович (RU)***

Заявка № **2018664960**
Дата поступления **24 декабря 2018 г.**
Дата государственной регистрации
в Реестре программ для ЭВМ ***09 января 2019 г.***

Руководитель Федеральной службы по интеллектуальной собственности

Г.П. Ивлиев