Федеральное государственное бюджетное образовательное учреждение
высшего образования «Томский государственный университет систем управления
и радиоэлектроники»

На правах рукописи

Грибанова Екатерина Борисовна

МОДЕЛИ, МЕТОДЫ, АЛГОРИТМЫ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ РЕШЕНИЯ ЗАДАЧ НА ОСНОВЕ ОБРАТНЫХ ВЫЧИСЛЕНИЙ

Специальность 1.2.2 – Математическое моделирование, численные методы и комплексы программ

Диссертация
на соискание ученой степени доктора технических наук

Научный консультант
доктор технических наук, профессор Мицель А.А.

Томск – 2023

ОГЛАВЛЕНИЕ

Введение

Актуальность темы. От решений специалистов в области управления зависит эффективность функционирования системы. При этом возникает необходимость анализа большого количества информации, что затруднительно осуществить без соответствующего математического аппарата и программных средств. В связи с этим разработка инструментов для повышения качества управленческих решений и скорости их формирования путем автоматизации обработки данных является актуальной задачей.

Задачи, возникающие перед специалистами, по направлению причинно-следственной связи можно разделить на прямые и обратные. Обратные задачи являются более сложными по сравнению с прямыми и заключаются в определении значений изменяемых входных переменных для достижения заданного значения показателя деятельности объекта [1]. Таким образом, значимость такого рода задач очень высокая, а их решение позволяет ответить на вопрос "как сделать так чтобы?" и осуществить выбор относительно значений входных управляемых переменных, что играет важную роль в формировании решений. Кроме значимости обратных задач можно также отметить их широкое распространение, в настоящее время такого рода задачи встречаются в таких областях как экономика, физика [2], астрономия, обработка изображений [3–4] и т.д.

Актуальность данного направления исследований подтверждается также государственной политикой, в частности, программа «Цифровая экономика Российская Федерации» (утверждена распоряжением Правительства Российской Федерации от 28 июля 2017 г. N 1632-р.) ставит такие задачи как ускорение технологического развития российских компаний, обеспечение ускоренного внедрения цифровых технологий в экономике и социальной сфере. Сокращению сроков разработки систем принятия решений, а также их распространению (тиражируемости) может способствовать разработка простых в компьютерной реализации алгоритмов решения задач, создание моделей на основе выявленных типовых задач и шаблонов разработки программного обеспечения. Таким образом, могут быть созданы научные основы для разработки систем принятия управленческих решений, позволяющих передать часть интеллектуального труда специалистов ЭВМ. Это повысит эффективность и качество работы организаций, обеспечив их конкурентоспособность.

Степень проработанности проблемы. Интерес к решению обратных задач возник достаточно давно, так отдельные результаты приводятся в трудах античных философов, например, решение Аристотелем задачи восстановления формы Земли по её тени на Луне [1, 5]. При этом актуальность решения обратных задач в различных областях исследования сохраняется и по сей день. Это обусловлено в том числе сложностью решения такого рода задач из-за их некорректности. Определение корректности задачи было сформулировано Ж. Адамаром, в частности, был обозначен такой признак как наличие единственного решения задачи на некотором множестве, которое должно непрерывно зависеть от входных данных.

Значительный вклад в исследование обратных задач принадлежит А. Н. Тихонову [6,7], который показал, что определение дополнительных условий на решение позволяет получить устойчивую задачу. Он предложил способ регуляризации некорректной задачи, т.е. сведение исходной задачи решения некоторого операторного уравнения к проблеме поиска минимума некоторого функционала. Также результаты исследования обратных задач отражены в трудах ученых М. М. Лаврентьева и Л. Я. Савельева [8–9], В.К. Иванова, В. В. Васина, В. П. Тананы [10], Р. Латтеса и Ж. Л. Лионса [11], М.М. Лаврентьева, В. Г. Романова, С. П. Шишатского [12], В. А. Морозова [13], О. А. Лисковца [14], В. В. Васина, А. Л. Агеева [15], А. М. Федотова [16], Г. М. Вайникко [17], В. К. Иванова, И. В. Мельниковой и А. И. Филинкова [18], В. П. Тананы и А. И. Сидиковой [19], Ю. Е. Воскобойникова [20], Кабанихина С. И. [21], A. Tarantola [22], R. L. Parker [23], C. W. Groetsch [24]. Отдельным направлением в решении некорректных задач является разработка простых в компьютерной реализации итерационных алгоритмов. Результаты исследований в этом направлении приводятся в работах А.

А. Самарского и П. Н. Вабищевича [25], А. М. Денисова [26], С. Р. Вогеля [27], Ю. Е. Воскобойникова [28–29], И. В. Емелина [30], С. Ф. Гилязова и Н. Л. Гольдмана [31], С. И. Кабанихина [32], О. В. Матысик [33], В. Н. Страхова [34], М. А. Красносельского, Г. М. Вайникко, П. П. Забрейко, Я. Б. Рутицкого, и В. Я. Стеценко [35] и др.

Одной из областей, где обратные задачи широко востребованы, является прикладная экономика [36], которая в отличие от теоретической, рассматривающей закономерности и принципы на основе идеальных условий, изучает конкретные ситуационные задачи, опираясь на опыт и конкретные показатели хозяйственной деятельности. Отдельные результаты решения такого рода задач в различных постановках при исследовании экономических объектов приводятся в работах Семенчина Е. А. и Невечеря А. П. [37], Урусовой А. С [38], Ekeland I. и Djitte N. [39] и других.

В 1996 году Одинцовым Б.Е. был разработан аппарат обратных вычислений, в следующей постановке [40]: определение изменений аргументов функции на основе начальных значений аргументов, заданного значения функции и экспертной информации, в качестве которой выступает направление изменения показателей и коэффициенты относительной важности. В учебном пособии [40] рассмотрено использование данного аппарата для формирования экономических решений, а также для инженерных расчётов: определение параметров геометрических фигур; расчёт характеристик инженерных конструкций, например, вычисление площади оснований каменного быка при изменении высоты моста, в том числе с использованием дифференциальных уравнений первого порядка, логарифмических, показательных, степенных функций. В литературе наибольшее распространение получило использование аппарата для формирования экономических решений

Одинцовым Б.Е. исследуются обратные задачи, которые в отличие от содержания задач, рассмотренных Тихоновым А.Н. [6,7] и другими авторами [8–13], характеризуются целью управления и наличием прямой функции, отражающей зависимость следствия от причины. Такие задачи являются трудоемкими и капризными [40], что выражается в непредсказуемости поведения функции и сложностью определения диапазонов входных данных, при которых задача имеет смысл и решение. Решение данного класса задач может быть точечными, т.е. базироваться на некоторой точке области решения. В качестве метода решения Одинцовым Б.Е. был предложен аппарат обратных вычислений, применение которого требует указания лицом, принимающим решения, прямой зависимости и целевой установки. Таким образом, под решением обратных задач с помощью обратных вычислений понимается получение точечных значений приростов аргументов прямой функции (аргументы определяются только для одной заданной точки функции) на основании её задаваемого значения и дополнительной экспертной информации, в качестве которой выступает направление изменения показателей и коэффициенты относительной важности. При этом прямые вычисления здесь позволяют получить начальное приближение решения задачи.

Для решения задач с ограничениями был разработан итерационный алгоритм, многократно выполняющий решение обратной задачи с изменением результирующего показателя на некоторую малую величину до тех пор, пока ограничения не будут нарушены [41, 42].

Существуют работы, направленные на модификацию данного аппарата и его применение для ряда частных случаев. Так, в статье Цветкова М.А. [43] рассмотрен случай, когда один и тот же показатель используется в двух разных обратных задачах. Приводится поиск компромиссного варианта такого аргумента с помощью указания коэффициентов распределения приростов. Збарским А.М. [44] исследуется использование аппарата обратных вычислений для приведения предприятия в равновесное состояние, характеризующееся равенством целевых индикаторов, формирующихся из внешних и внутренних факторов предприятия. Одинцовым Б.Е. [45] рассмотрено нахождение решения обратной задачи с учетом «золотых пропорций» показателей предприятия. Для этого осуществляется последовательная корректировка решений с заданными целевыми установками и с учетом «золотых пропорций» в зависимости от приоритетности способов достижения цели.

Ряд исследований посвящен применению обратных вычислений для решения социально-экономических задач. Так в статье Виштак О. В. и Штыровой И. А. [46] рассмотрено формирование интегрального показателя качества дополнительного образования в ВУЗе, определяемого набором групп показателей более низкого уровня иерархической структуры. В работе Барминой Е. А. и Квятковской И. Ю. [47] формирование интегрального показателя качества работы коммерческой организации также осуществляется с помощью иерархичной структуры, а также представлены методики определения интегрированных показателей качества и множества измеряемых показателей качества на основе когнитивного моделирования. Механизм обратных вычислений в данной работе использован для решения задачи формирования рекомендаций по способам повышения качества работы организации. Блюмин С. Л. и Боровкова Г. С. [48] рассматривают применение обратных вычислений для поиска значений показателей результативности деятельности сотрудников кафедры высшего учебного заведения для достижения желаемого значения рейтинга кафедры. Последующий анализ конечных изменений позволил определить сотрудников, способных внести наибольших вклад в формирование рейтинга кафедры. Методика совместного использования системы сбалансированных показателей предприятия и метода обратных вычислений рассмотрена авторами Силкина Г. Ю. и Переверзева А. А. [49].

Таким образом, в существующих работах рассматривается применение аппарата обратных вычислений для решения отдельных задач с учётом особенностей исследуемого объекта [50]. Диссертация посвящена развитию существующего аппарата и разработке комплекса моделей, методов и алгоритмов, являющихся более простыми в компьютерной реализации по сравнению с известными, и предназначенных для решения более широкого круга задач экономики, в том числе не требующих использования экспертной информации, либо требующих меньшего её объема. Кроме того, в диссертационной работе рассмотрена разработка программных систем решения обратных задач.

Объектом исследования является процесс решения задач на основе обратных вычислений.

Предметом исследования являются модели, методы, алгоритмы и программное обеспечение решения задач на основе обратных вычислений.

Цель диссертационной работы – повышение эффективности процесса решения обратных задач за счёт применения математического и программно-алгоритмического инструментария на основе обратных вычислений.

Для достижения поставленной цели необходимо было решить следующие **задачи:**

1) Провести анализ исследований в области решения задач на основе обратных вычислений.

2) Разработать комплекс моделей, методов и алгоритмов решения задач на основе обратных вычислений.

3) Провести вычислительные эксперименты по решению задач с использованием предложенных моделей, методов и алгоритмов, а также сравнение результатов с полученными с помощью известных методов и стандартных математических пакетов.

4) Разработать структуру программы решения обратных задач на основе предложенных методов и алгоритмов.

5) Разработать алгоритм поэтапного формирования аргументов исследуемой функции, образующих дерево цели, с учетом ограничений.

Методы исследования: системный анализ, метод обратных вычислений, методы оптимизации, экономический анализ, экспертные методы, статистические методы, имитационное моделирование, объектно-ориентированный анализ.

Результаты работы, характеризующиеся **научной новизной**:

1) Предложен метод решения задач с помощью обратных вычислений, отличающийся от известного формированием уравнения зависимости между аргументами функции.

2) Предложен стохастический метод решения задач на основе обратных вычислений с ограничениями, отличающийся от известных использованием

процедуры выбора аргументов для достижения цели с помощью моделирования полной группы несовместных событий.

3) Разработаны оптимизационные модели для решения задач на основе обратных вычислений при максимизации соответствия экспертным целеполаганиям и при минимизации отклонений аргументов от исходных значений.

4) Разработаны методы и алгоритмы решения задач на основе обратных вычислений, представленных в виде оптимизационных моделей, а также задач нелинейного программирования, отличающиеся от известных использованием двухшаговой процедурой, включающей оптимизацию целевой функции и переход к значениям аргументов, удовлетворяющим ограничению задачи.

5) Разработан комплекс проблемно-ориентированных программ для решения задач на основе обратных вычислений, отличающийся от существующих возможностью решения иерархических задач с ограничениями.

Теоретическая значимость исследования заключается в том, что применительно к проблематике диссертации результативно использован комплекс существующих базовых методов, в частности аппарат обратных вычислений, эконометрическое моделирование, имитационное моделирование, методы оптимизации. Изложены аргументы в пользу создания комплекса моделей, методов, алгоритмов и концепций разработки программного обеспечения для решения задач на основе обратных вычислений. Раскрыты недостатки и ограничения существующих методов решения обратных и оптимизационных задач. Изучены характеристики разработанных алгоритмов, показано соответствие результатов их работы и результатов, полученных с помощью известных алгоритмов, а также математических пакетов. Проведена модернизация аппарата обратных вычислений, методов нелинейной оптимизации, что позволило получить комплекс моделей, методов и алгоритмов для решения обратных задач, уменьшить объем используемой экспертной информации, снизить временные затраты на решение задач.

Практическая значимость работы заключается в разработке и внедрении комплекса моделей, методов, алгоритмов и программного обеспечения решения задач на основе обратных вычислений. Определены пределы и перспективы практического использования результатов исследования, которые могут быть применены предприятиями и организациями различных сфер деятельности при создании систем поддержки принятия решений для эффективного стратегического и оперативного управления, а также разработанные алгоритмы могут быть применены в других областях исследования, в частности, при создании программных систем, включающих решение оптимизационных задач. Используемый подход позволит обеспечить экономию ресурсов при создании программного обеспечения и быстродействие реализованных программных систем.

Реализация и внедрение результатов диссертационной работы. Разработанные модели, методы, алгоритмы, а также программы ЭВМ внедрены в организациях: ООО «Томская нефть», АО «Разрез «Степановский», ООО «Гамарджоба», г. Томск, ООО «ФОРС», г. Реутов, ООО «Сибмед», г. Томск, ООО «Вокифудтомск», г. Томск, ООО «Титан», г.Томск, ООО «Интенс-строй», г. Томск, ООО «Дельта», г.Юрга, МАОУ ДО Дворец творчества детей и молодежи, г. Томск.

Результаты исследований внедрены в учебный процесс Томского государственного университета систем управления и радиоэлектроники в виде учебных пособий, методических указаний для выполнения практических, лабораторных работ по дисциплинам «Эконометрика», «Математическое и митационное моделирование экономических процессов», «Технико-экономический анализ деятельности предприятия», «Исследование операций и методы оптимизации». Теоретические положения использовались для постановки задач научно-исследовательской работы студентов, 25 выпускных квалификационных работ, в том числе 11 магистерских диссертации.

Результаты диссертации использованы в ФГБОУ ВО «ТУСУР» при выполнении государственного задания Министерства науки и высшего образования РФ, проект FEWM-2020-0036 «Методическое и инструментальное обеспечение принятия решений в задачах управления социально-экономическими системами и процессами в гетерогенной информационной среде».

Также автор является исполнителем гранта Российского научного фонда № 22-28-01795, https://rscf.ru/project/22-28-01795.

Основные положения, выносимые на защиту:

1) Метод на основе формирования уравнения зависимости между аргументами функции позволяет определить решение задачи на основе обратных вычислений без необходимости проверки согласованности экспертной информации: соответствия поставленной цели коэффициентам относительной важности и направлениям их изменения. Это позволяет снизить объем экспертной информации и предотвратить ошибки, связанные с определением входных данных. Соответствует п. 2 паспорта специальности 1.2.2.

2) Стохастический метод позволяет находить решение многоаргументных задач на основе обратных вычислений при отсутствии необходимости указания направлений изменения аргументов, что приводит к уменьшению времени определения входных данных задачи. Соответствует п. 2, 9 паспорта специальности 1.2.2.

3) Оптимизационные модели решения задач на основе обратных вычислений позволяют осуществлять формирование значения ключевого показателя при максимизации соответствия экспертным целеполаганиям, при использовании показателей в нескольких функциях расчёта, а также при минимальном отклонении значений входных переменных от исходных величин. Соответствует п. 8 паспорта специальности 1.2.2.

4) Оптимизационные методы и алгоритмы на основе обратных вычислений позволяют эффективно решать задачи условной оптимизации. В качестве основных критериев эффективности рассматривается качество (точность, достоверность, сходимость) и время решения задачи. Соответствует п. 2 паспорта специальности 1.2.2.

5) Проблемно-ориентированный комплекс программ позволяет осуществлять формирование ключевого показателя путем поэтапного решения

иерархических задач с ограничениями на основе обратных вычислений и снизить временные затраты на обработку данных в среднем на 25%. Соответствует п. 3 паспорта специальности 1.2.2.

Достоверность результатов исследования.

– теория построена на общепризнанных законах, известных проверяемых фактах и согласуется с опубликованными экспериментальными данными;

– идея базируется на проверенных методах и алгоритмах, представленных в литературе, а также реализованных в математических пакетах, обобщении опыта в области решения обратных задач с помощью обратных вычислений;

– использованы данные литературных источников, опубликованные результаты российских и зарубежных исследований, а также данные реальных социально-экономических объектов (предприятия, социальные сети);

– установлено соответствие между результатами проведенных численных экспериментов и решениями, полученными с использованием классических методов и стандартных математических пакетов;

– использованы классические методы сбора и обработки данных.

Апробация работы. Основные научные положения диссертационной работы докладывались и обсуждались на научных конференциях: «Научная сессия ТУСУР», 2005, 2006, 2007, 2008, 2019, 2022 г. (Томск); «Студент и научно-технический прогресс», 2005 г. (Новосибирск); «Молодежь и современные информационные технологии», 2007, 2014 г. (Томск); «Современное образование: традиции и новации», 2006, 2022 г. (Томск); «Электронные средства и системы управления», 2007 (Томск); «Инновационные технологии и экономика в машиностроении», 2008 г. (Юрга); «Современные техника и технологии», 2008 (Томск); «Современные технологии принятия решений в цифровой экономике», 2018 (Юрга); «Информационные технологии в науке, управлении, социальной сфере, медицине», 2018 (Томск); «Инноватика-2019», 2019 (Томск); «Перспективы развития фундаментальных наук», 2019 (Томск); «Математическое и информационное моделирование», 2022 г. (Тюмень); «Информационные системы и технологии в образовании, науке и бизнесе», 2022 г. (Улан-Удэ).

Публикации. По теме диссертации опубликовано 96 работ, в том числе: 27 статей в изданиях, входящих в перечень ВАК; 3 монографии; 10 статей в журналах, индексируемых Scopus/Web of Science; 6 статей в сборниках конференций, индексируемых Scopus/Web of Science; 10 учебных пособий (из них 2 с грифом СибРумц, 3 с грифом УМО). Получено 9 свидетельств о регистрации программ ЭВМ в Федеральной службе по интеллектуальной собственности, патентам и товарным знакам. Без соавторов опубликовано 32 работы, в т.ч. 1 научная монография, 1 коллективная монография, 13 статей в журналах из перечня ВАК, 6 статей в журналах, индексируемых Scopus/Web of Science; 3 статьи в сборниках конференций, индексируемых Scopus/Web of Science; 1 учебное пособие с грифом УМО.

Личный вклад автора. Все основные результаты диссертационной работы получены лично автором. Программные продукты созданы автором либо на основе моделей, методов и алгоритмов автора, под его руководством и при непосредственном участии.

Структура и объем диссертации. Диссертационная работа состоит из введения, семи глав, заключения, списка литературы и приложений. Объем диссертации составляет 331 страницу (включая 139 рисунок и 55 таблиц). Список литературы состоит из 244 наименований.

Автор выражает благодарность и глубокую признательность за помощь в работе над диссертацией и ценные замечания научному консультанту, доктору технических наук, профессору Мицелю Артуру Александровичу.

Глава 1. Методы решения обратных задач

1.1 Понятие обратной задачи

Для оценки деятельности экономического объекта и выбора способов достижения поставленной цели специалисты сталкиваются с необходимостью решения задач, которые в зависимости от их ориентации относительно причинно-следственной связи можно разделить на прямые и обратные [51].

Прямая задача заключается в определении результирующего показателя по имеющимся значениям исходных величин и виду зависимости. Так на рисунке 1.1 представлено решение прямой задачи по определению выходной величины y на основе входных данных x. Здесь f – функция преобразования входных значений в выходную величину:

$$y = f(x).$$

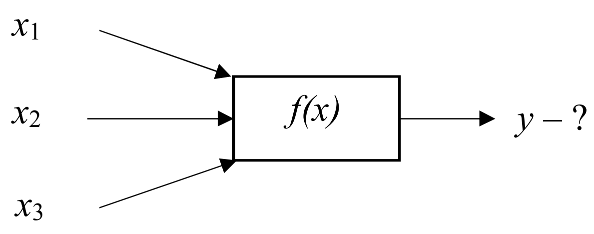


Рисунок 1.1 – Решение прямой задачи

Решение обратных задач в экономике позволяет получить информацию, которая является полезной для специалистов, осуществляющих деятельность в области управления. В результате решения задачи специалист получает сведения о том, каким образом может быть достигнуто заданное состояние экономического объекта. Таким образом, на основе этой информации могут быть проведены мероприятия по изменению управляемых характеристик (цена, себестоимость и т.д.) для достижения целевых показателей экономического объекта. Так, в литературе рассматривается решение обратных задач для управления эффективностью компании, формирования себестоимости промышленного предприятия, совершенствования структуры кредитно-депозитной базы коммерческих банков, управления образовательной организацией с помощью рейтинговой системы.

Решение обратных задач позволяет определить характеристики объекта, требуемые для достижения заданных значений стратегических и оперативных показателей, что в свою очередь обеспечивает повышение качества управленческих решений и эффективности функционирования экономического объекта. При этом возникает необходимость анализа и обработки большого количества информации, что затруднительно осуществить без соответствующего математического аппарата и программных средств. В связи с этим исследования, посвященные разработке математического и алгоритмического инструментария для решения обратных задач и повышения скорости формирования решений путем автоматизации обработки данных, являются актуальными.

Существует условная классификация обратных задач математической физики (ретроспективные, коэффициентные, граничные, геометрические) [52], которая основана на условиях, не заданных в модели (начальные значения, коэффициенты и т.д.) (рисунок 1.2).

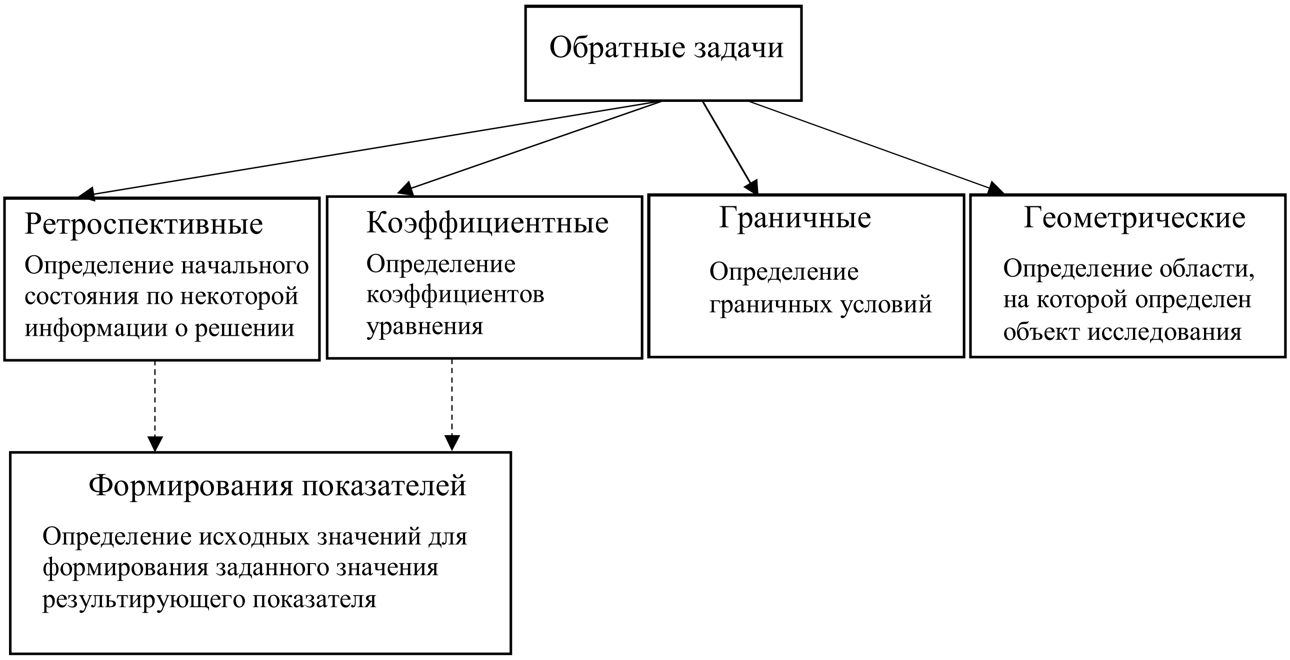


Рисунок 1.2 – Классификация обратных задач

В качестве ретроспективной задачи в работе [52] рассматривается восстановление неизвестных причин по известному следствию. Такие задачи возникают, например, в случае, когда объект исследования недоступен для наблюдения, либо проведение исследования требует высоких затрат. В качестве примера может быть приведено определение свойств поверхности земли по измеренному отраженному солнечному излучению. В работе [1] приводится целый ряд подобных примеров обратных задач. Так, Ератосфен не мог напрямую измерить длину окружности Земли, но получил достаточно точные результаты путем использования параметра угла дуги; Генри Кавендиш в 1798 году для определения массы Земли провел лабораторное исследование, в котором было использовано гравитационное притяжение свинцовых шариков в тонко сбалансированном крутильном маятнике; Эдмунд Галлей был первым, кто предпринял попытку оценить возраст Земли, изучая соленость моря.

В качестве примера коэффициентной обратной задачи рассматривается определение параметров линейной регрессии [53] для подгонки функции к наблюдаемому набору данных (прямая задача будет заключается в прогнозе/оценке модельного значения *y* исходя из уравнения регрессии и заданного значения *x*) (рисунок 1.3).

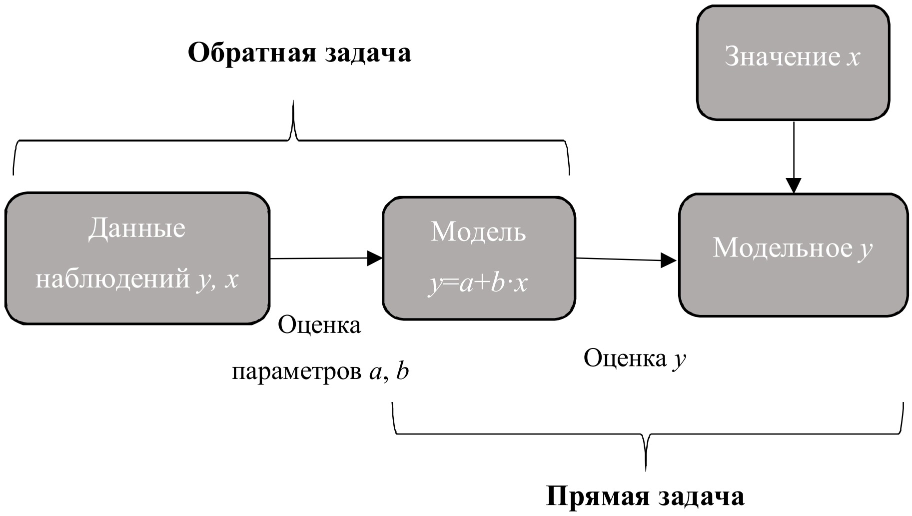


Рисунок 1.3 – Прямая и обратная задача линейной регрессии

При постановке задачи также рассматриваются случаи, когда необходимо выполнить оптимизацию некоторой функции, так, в работе [54] рассматриваются различные варианты обратной задачи по определению значений входных параметров таким образом, чтобы функция, вычисленная на основе вектора стоимости, принимала минимальное значение. Ghobadi и др. [55] определяет обратную задачу как нахождение значений параметров целевой функции, обеспечивающих оптимальное решение прямой задачи и в качестве примера рассматривает решение задачи о диете с помощью линейного программирования.

Представленная классификация (рисунок 1.2) также находит отражение в задачах экономики. Так, в случае необходимости определения аргументов функции расчёта экономического показателя рассматривается задача формирования характеристики. Обратная задача формирования показателя заключается в определении набора исходных величин *x* или их изменений, обеспечивающих заданное значение результирующей переменной y (рисунок 1.4).

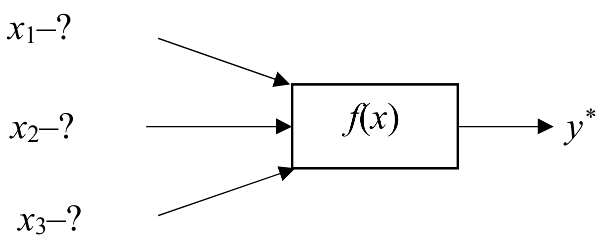


Рисунок 1.4 – Обратная задача: x – входные параметры; y^* – заданное значение выходной величины

Решение задачи формирования показателя может быть выполнено с помощью регуляризации [7] и обратных вычислений (рисунок 1.5) [8].

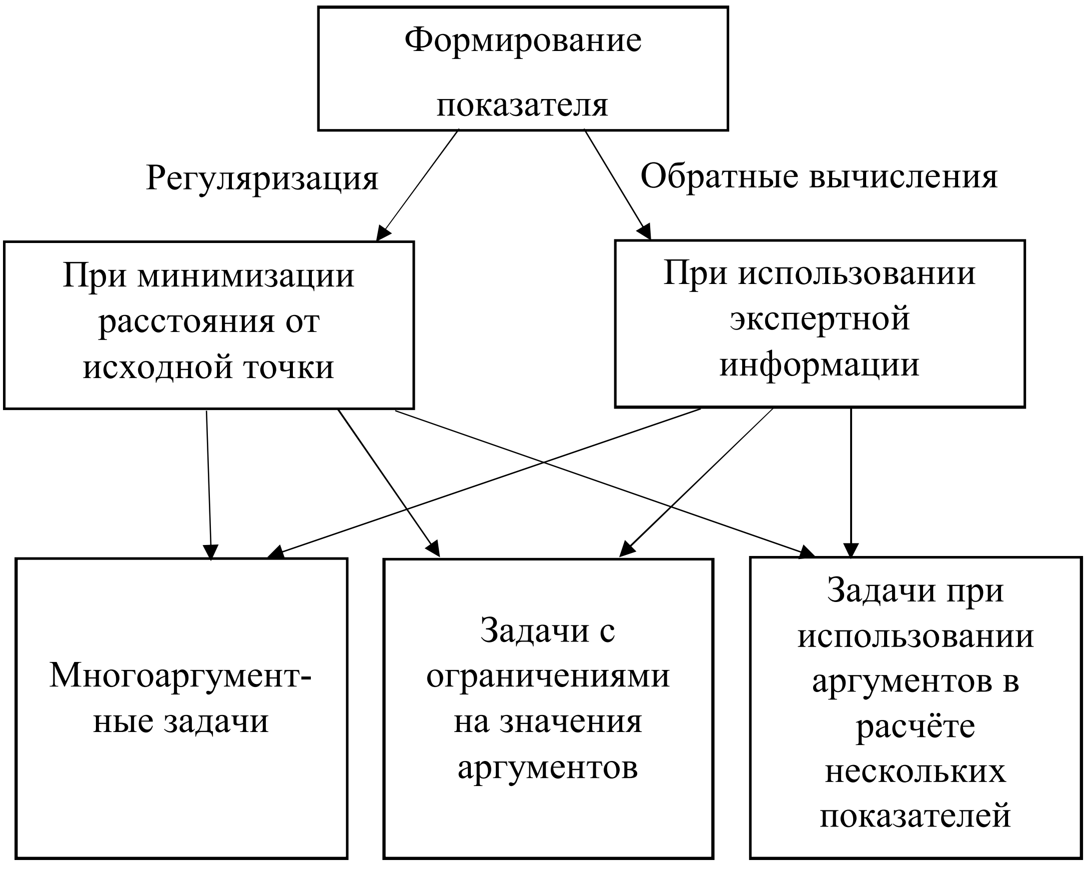


Рисунок 1.5 – Классификация задач формирования показателя

1.2 Решение обратной задачи с использованием регуляризации

Наиболее известными видами регуляризации, основанными на отклонении полученного решения от исходного, являются регуляризация Тихонова [56–58] и регуляризация через манхэттенское расстояние [59, 60].

Обратная задача, показанная на рисунке 1.2, может быть представлена в виде операторного уравнения [61]

$$f(x) = y^*. \tag{1.1}$$

Решение уравнения (1.1) может быть определено с помощью минимизации невязки [62]

$$\left\| f(x) - y^* \right\| \to \min. \tag{1.2}$$

Поскольку в рассматриваемой задаче (рисунок 1.2) y^* – число, а x – n-мерный вектор входных значений, то задача (1.2) имеет множество решений.

В случае использования регуляризации по Тихонову обратная задача (рисунок 1.2) может быть представлена в следующем виде (μ – параметр регуляризации) [63]

$$Q(x) = (f(x) - y^*)^2 + \mu \sum_{i=1}^{n} x_i^2 \to \min. \tag{1.3}$$

где Q – минимизируемый функционал.

Здесь минимизируемая функция включает две составляющие: первая компонента отражает соответствие результирующего показателя заданному значению, а вторая – величину аргументов. Решение задачи (1.3) требует нахождение параметра регуляризации μ, что является отдельной задачей, требующей выбора способа поиска μ [63–65], определяющего полученный результат. При решении обратной задачи (1.3) входные параметры x определяются таким образом, чтобы их значения были как можно ближе к нулю.

При применении регуляризации через манхэттенское расстояние вместо суммы квадратов аргументов в формуле (1.3) используется сумма модулей аргументов [62]

$$Q(x) = (f(x) - y^*)^2 + \mu \sum_{i=1}^{n} \left| x_i \right| \to \min. \tag{1.4}$$

В результате решения задачи (1.4) происходит обнуление некоторых аргументов, таким образом, может быть осуществлен отбор признаков.

Для решения задачи (1.1) также может быть рассмотрен способ, основанный на минимизации нормы входных параметров с учётом ограничения [62]. При этом целевая функция отражает вид регуляризации: минимизация суммы модулей изменений аргументов, минимизация суммы квадратов изменений аргументов.

Так, при минимизации суммы квадратов аргументов задача имеет вид:

$$\begin{gathered} g(x) = \sum_{i=1}^{n} x_i^2 \to \min, \\ f(x) = y^*. \end{gathered} \tag{1.5}$$

где $g(x)$ – целевая функция, отражающая регуляризацию в виде суммы квадратов аргументов.

В случае минимизации суммы модулей аргументов задача имеет вид

$$\begin{gathered} g(x) = \sum_{i=1}^{n} \left| x_i \right| \to \min, \\ f(x) = y^*. \end{gathered} \tag{1.6}$$

Таким образом, обратная задача при использовании регуляризации может быть представлена в виде задачи условной либо безусловной оптимизации. Рассмотрим существующие методы решения подобного рода задач.

1.2.1 Методы решения задач безусловной оптимизации

Методы, используемые для решения задач безусловной оптимизации (1.3, 1.4) могут быть разделены на следующие группы:

– по порядку используемой производной: методы нулевого, первого и второго порядка.

– по применению случайных чисел: детерминированные и стохастические методы.

– по виду экстремума: методы поиска локального и глобального минимума.

Детерминированные методы основаны на поиске решения на основе некоторого правила изменения текущих значений аргументов без использования случайным чисел. В зависимости от порядка используемой производной можно выделить следующие детерминированные методы:

– Методы нулевого порядка: последовательный поиск, Хука-Дживса, симплексный (Нелдера-Мида), поиск с помощью шаблонов [66–67] и др. Данные методы используют для нахождения решения только значения функции и аргументов и не требуют вычисления частных производных функции. В качестве недостатков таких методов можно отметить необходимость выполнения большого числа итераций, а их достоинствами является простота и гибкость.

– Методы первого порядка: градиентный спуск, Коши, Флэтчера – Ривза и др. [66, 68–69]. Методы требуют вычисление первой частной производной оптимизируемой функции в исследуемых точках. Наиболее простым градиентным методом поиска минимума функции является градиентный спуск с постоянным шагом: на каждом шаге происходит корректировка значений аргументов на величину градиента в точке умноженного на некоторый параметр ϕ. Однако недостатком его является зависимость сходимости от выбранного значения параметра ϕ: при малом значении потребуется большое числа шагов для нахождения решения; при большом значении параметра алгоритм может расходиться. Тем не менее простота и гибкость этого метода обеспечили его применение в разных прикладных задачах. В методе Коши параметр ϕ определяется на каждом шаге итерации с помощью процедуры одномерной оптимизации. Метод Флэтчера-Ривза использует поиск вдоль взаимно сопряженных направлений, при расчете параметра ϕ используется отношение нормы градиентов на предыдущем и текущем шаге. Алгоритмы Коши и Флэтчера-Ривза имеют более высокую скорость сходимости (если функция квадратична, то число шагов равно числу переменных) однако обладают высокой вычислительной сложностью.

– Методы второго порядка (ньютоновские методы) [66, 69] требуют существования первой и второй частной производной оптимизируемой функции, а величина шага настраивается в зависимости от направления спуска к минимуму по антиградиенту и выпуклости функции. К недостаткам данного методам можно отнести необходимость построения и решения нелинейной системы уравнений, содержащего элементы матрицы Гессе.

Для решения задач безусловной оптимизации также применяются методы, основанные на использовании случайных величин [70].

Появление идеи использования случайных величин при поиске решения связывают с именем У. Р. Эшби. В нашей стране исследования алгоритмов случайного поиска берут начало в работах Л.А. Растригина [71–72]. Алгоритмы поиска подразделяют на ненаправленные (все случайные испытания строят независимо друг от друга) и направленные (испытания связаны между собой).

Наиболее простым методом решения задач оптимизации является метод ненаправленного случайного поиска. Он заключается в получении случайных значений аргументов из заданного интервала, расчете целевой функции и сравнении её величины с наилучшим из вычисленных. Если новое рассчитанное значение результата оказалось меньше, то осуществляется запоминание полученного решения. Шаги повторяются в течение заданного числа реализаций либо до получения решения с указанной точностью. Такой способ нахождения решения является реализацией метода проб и ошибок.

Данный алгоритм может быть совмещен с локальным поиском, когда из случайно выбранных точек осуществляется локальный спуск в ближайший минимум. Из найденных локальных минимумов выбирается точка с наименьшим значением [71].

Благодаря своей простоте и гибкости данный метод получил широкое распространение при решении различных задач. Например, в статье [73] рассматривается использование метода ненаправленного случайного поиска для решения комбинаторной задачи выбора оптимального портфеля биржевых опционов, что позволило получить целочисленные значения искомых величин.

К алгоритмам направленного поиска относят алгоритм парной пробы, с возвратом при неудачном шаге, наилучшей пробы и т.д. [72].

В алгоритме парной пробы по обе стороны от исходной точки делаются два поисковых шага случайной величины. После этого осуществляется переход в новую точку в направлении наилучшего значения функции. В алгоритме с возвратом при неудачном шаге задается начальная точка x и случайным образом осуществляется моделирование приращения dx. Если значение функции в новой точке $x+dx$ лучше, чем в точке x, то осуществляется переход в эту точку. В некоторых работах предлагается исключать то направление, которое не приводит к улучшению значения функции. В статье [74] описывается простой алгоритм оптимизации (SOPT) в котором случайная величина приращения имеет не равномерное распределение, а распределение Гаусса. Также предлагается многократно вызывать алгоритм для выбора наилучшего решения из полученных.

Недостатком приведенных алгоритмов направленного поиска является то, что они в качестве решения могут определить локальный минимум, а не глобальный (рисунок 1.6). В связи с этим разрабатываются их различные модификации. К ним, в частности, можно отнести адаптивные алгоритмы. Например, таким алгоритмом является случайный поиск ARSET (Adaptive Random Search Technique) [75] и динамический случайный поиск DRASET (Dynamic Random Search Technique) [76].

В адаптивном случайном поиске в зависимости от значения целевой функции пространство поиска сужается (когда происходит поиск наилучшего значения) или расширяется (когда найдено решение с приемлемой точностью), таким образом, уменьшается вероятность нахождения локального минимума вместо глобального из-за недостаточного исследования отдельных участков. В алгоритме DRASET после нахождения решения дополнительно осуществляется локальный поиск вокруг найденной точки для получения более точного значения.

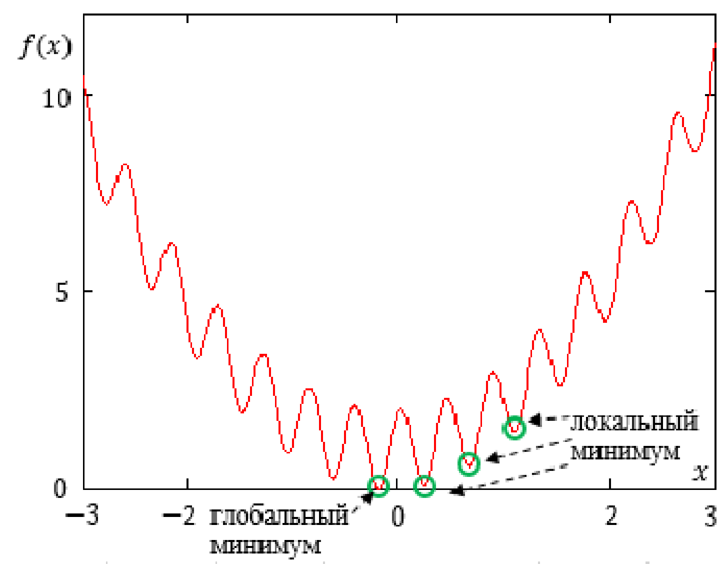


Рисунок 1.6 – Локальный и глобальный минимумы функции

Также существуют алгоритмы поиска глобального минимума, которые используют процедуру случайного блуждания [71], например, метод «зашумления» градиента, и метод сглаживания, который может быть использован в случае, если минимизируемая функция образована путем наложения на унимодальную функцию мелких отклонений; методы, которые воспроизводят поведение живой и неживой природы [77–78] и эволюционные механизмы [79]; методы, основаные на интервальных подходах [80–81].

Стохастические методы позволяют получить некоторое решение за заданное пользователем время, что делает возможным их применение в задачах большой размерности, когда использование классических методов может привести к недопустимо большому времени решения задачи. Однако полученное решение будет субоптимальным и изменяться в различных запусках программной реализации. Кроме того, реализация самих алгоритмов может представлять сложность в связи с множеством правил корректировки решений, полученных на каждой итерации, а необходимость выполнения большого числа итераций для получения решения с заданной точностью, требует затрат вычислительных ресурсов.

1.2.2 Методы решения задач условной оптимизации

В зависимости от вида целевой функции и ограничения задачу условной оптимизации (1.5, 1.6) можно разделить на задачи линейного и нелинейного программирования.

Так в работе [53] задача (1.6) представлена в виде задачи линейного программирования. Основополагающая роль в исследовании подобного рода задач принадлежит опубликованной в 1939 году Канторовичем Л.В. работе «Математические методы организации и планирования производства», в которой были описаны некоторые оптимизационные задачи экономики (рациональное распределение топлива, максимальное уменьшение отходов, наиболее полное использование механизмов и т.д.) и предложен метод разрешающих множителей для их решения. Позже Данцигом Дж. был разработан симплекс-метод [82], который сейчас и является основным методом решения задач линейного программирования. При единственном линейном ограничении и равенстве числовых значений при аргументах в целевой функции единице задача может быть решена более простым методом. Её решение сводится к нахождению элемента с наибольшим абсолютным числовым значением при аргументе в ограничении и решения уравнения относительно этого аргумента [83].

Классическими методами решения задачи нелинейной оптимизации (1.5) являются метод штрафов и множителей Лагранжа. В методе множителей Лагранжа формируется модифицированная функция Z, которая включает неизвестные параметры – множители Лагранжа λ [64, 84]

$$Z(x,\lambda) = \lambda(f(x) - y^*) + \sum_{i=1}^{n} x_i^2 \to \min. \tag{1.7}$$

Для оптимизации функции (1.7) происходит формирование системы уравнений, в которой частные производные приравниваются к нулю, а также включаются условия дополняющей нежесткости. Из-за определения дополнительных переменных λ происходит увеличение размерности задачи, что является недостатком этого метода.

В методе штрафов осуществляется многократная оптимизация модифицированной функции L при последовательном изменении штрафного параметра R

$$L(x) = R(f(x) - y^*)^2 + \sum_{i=1}^{n} x_i^2 \to \min. \tag{1.8}$$

Данная классическая схема решения может быть модифицирована с учётом специфики решаемой задачи. Так, в работе [85] рассматривается решение задачи многокритериальной оптимизации. Авторами [86] представлено решение задачи двухуровневой оптимизации с помощью метода штрафов.

Основным недостатком метода штрафов является необходимость выполнения многократной безусловной оптимизации функции. В силу того, что модифицированная функция (1.8) включает две составляющие (сумму квадратов приращений и соответствие функции f заданному значению результирующего показателя), её оптимизация может занять продолжительное время, а градиентные методы оказаться неэффективными.

В качестве способа преодоления данной трудности в литературе предлагаются алгоритмы решения задачи без применения штрафного параметра, основанные на условиях Куна-Таккера. В результате решение задачи сводится к решению систем уравнений. Так, в работе [87] на каждой итерации осуществляется решение трех систем линейных уравнений для поиска направления изменений аргументов, после чего выполняется линейный поиск в заданном направлении. В статье [88] решение задачи нелинейного программирования сводится к решению задачи линейного программирования симплекс-методом, однако предложенный метод может быть использован только при линейном ограничении. Также для решения задач нелинейной оптимизации с ограничениями-неравенствами применяется метод Зойтендейка [89], который включает решение задачи линейного программирования для определения направления поиска с последующей оптимизацией функции путем движения вдоль выбранного направления.

В работе [90] рассматривается применение рекуррентных нейронных сетей для решения задачи нелинейной оптимизации. Применение нейронных сетей требует реализации алгоритмов обучения сети. В силу этого разработка алгоритма, реализация многократной оптимизаций функции также может занять значительное количество временных и вычислительных ресурсов.

Некоторыми авторами также рассматривается использование комбинации двух методов, например, в работе [91] метод Зодендейка был использован совместно с эвристическим методом.

1.2.3 Методы решения задач целочисленного программирования

Задача целочисленного программирования формулируется следующим образом: найти такие целые значения аргументов, которые обеспечили бы наилучшее значение целевой функции при заданных ограничениях. К отдельной группе относят задачи, в которых аргументы могут принимать только два значения: 0 и 1. Такие задачи называются задачами псевдобулевой оптимизации и подробно описаны в статье А.Н. Антамошкина и И.С. Масича [92].

При небольшом числе переменных задача целочисленного программирования может быть решена простым перебором. При этом количество возможных вариантов без учета ограничений составит 2^n (n–число аргументов целевой функции). Однако с увеличением числа аргументов такой вариант решения задачи становится неприемлемым в связи с необходимостью выполнения большого числа вычислений.

Одним из распространенных методов решения задачи целочисленного программирования является метод ветвей и границ [93]. Метод подразумевает выполнение процедуры ветвления и нахождение оценок. На каждом шаге выполняется анализ элементов разбиения, в ходе которого определяется, является ли полученное решение оптимальным. В зависимости от результатов анализа происходит либо остановка алгоритма, либо запоминание текущего решения, либо новое ветвление.

Детерминированные методы решения задач целочисленного программирования: динамическое программирование, метод ветвей и границ, метод отсечений, также при большом числе аргументов часто оказываются не способными найти решение за отведенное время (в работе А.В. Овчинникова [94] приводятся характеристики подобных алгоритмов). Кроме того, данные алгоритмы в ряде случаев не позволяются учесть сложных, алгоритмически вычисляемых ограничений и целевой функции. В работе А.Н. Антамошкина, И.С. Масич [92] отмечается, что в этом случае оправданным является использование стохастических методов, в частности, методов случайного поиска [70, 95]. Однако простой случайный поиск также может оказаться малоэффективным для отдельных задач, поэтому разрабатываются его различные модификации, в том числе основанные на идее эволюции, описанные в работах F. Glover [96], J.P. Pedroso [97], T. Jansen, I.Wegener [98], П.В. Галушкина, О.Э. Семенкиной [99], А.Н. Антамошкина, И.С. Масич [92].

1.2.4 Итерационные алгоритмы решения обратных задач

Классические методы решения обратной задачи при регуляризации Тихонова основаны на оптимизации модифицированной функции, формируемой из целевой функции и ограничения, что требует высоких затрат вычислительных ресурсов. В связи с этим для решения обратных задач разрабатываются более простые итерационные методы. Наиболее распространенным является алгоритм Ландвебера [25], согласно которому изменение аргумента x функции осуществляется по формуле

$$x_i = x_i + \alpha \cdot \frac{\partial f(x)}{\partial x_i}\left(y^* - y\right),$$

где α – малое число, определяющее шаг изменения x.

Модификации данного алгоритма [26–28] разрабатываются с целью уменьшения вычислительной сложности [28], дополнительной регуляризации исходя из обучающих данных [26] и повышения точности вычислений с учётом специфики задачи [27]. Также в литературе рассматривается использование метода Гаусса-Ньютона [29], наискорейшего спуска [30].

1.3 Решение обратных задач экономики с помощью обратных вычислений

Развитие экономической системы определяется эффективностью решений специалистов в области управления, которые в своей деятельности используют ряд инструментов. Одним из таких инструментов является дерево цели, представляющее собой иерархическую структуру, включающую генеральную цель, расположенную в корне дерева, и подчиненные подцели первого, второго и последующих уровней [100–101]. В качестве стратегической цели может быть рассмотрено обеспечение финансовой безопасности предприятия [102], развитие «зеленой» экономики региона, оптимизация структуры организации [103] и укрепление её экономической позиции [104] и т.д. Существуют различные модификации дерева цели, в частности, в работе [40] рассматривается целеполагание на основе модели будущего, при котором цели и подцели состоят из конкретных количественных и качественных показателей, связанных между собой функциональной зависимостью. Направление поиска формирует прямую либо обратную цепочку, каждая из которых включает группу подзадач для решения глобальной задачи. Так, при обходе снизу-вверх дерева показателей происходит расчёт новых характеристик на основе существующих с использованием заданных функций (прямая задача). Такой принцип обхода позволяет выполнять оценку и прогнозирование целевой характеристики при известных данных. В случае обхода дерева показателей сверху-вниз происходит определение значений характеристик, формирующих целевую величину.

Данное исследование посвящено решению обратных задач формирования показателя: определению изменений показателей первого и последующих уровней с учётом заданных ограничений и выбранного способа решения таким образом, чтобы целевой показатель принял установленное значение. Так, для задачи на рисунке 1.7 заданы исходные значения показателей подцели первого (x_1, x_2, x_3) и второго (x_{11}, x_{21}, x_{22}) уровня, необходимо определить их изменения (Δx_1, Δx_2, Δx_3, Δx_{11}, Δx_{21}, Δx_{22}) таким образом, чтобы целевой показатель был равен y^*.

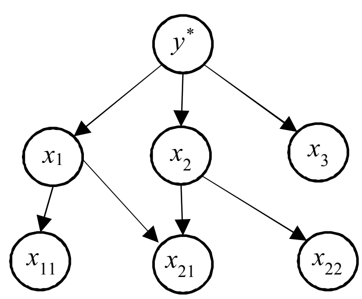


Рисунок 1.7 – Дерево решения

Для решения поставленной задачи необходимо выполнить два шага:

– решение отдельных подзадач;

– решение общей задачи формирования результирующего показателя дерева цели.

Под отдельной подзадачей понимается решение обратной задачи по определению значений характеристик, формирующих показатель более высокого уровня. Так, на рисунке 1.7 решение общей задачи будет включать такие подзадачи как определение значений Δx_1, Δx_2, Δx_3 для формирования y^*; вычисление Δx_{11}, Δx_{21} для формирования $x_1 + \Delta x_1$; расчёт Δx_{21}, Δx_{22} для формирования $x_2 + \Delta x_2$.

В 1990-х годах Одинцовым Б.Е. был разработан аппарат обратных вычислений, предназначенный для решения обратных задач экономики в рассмотренной выше постановке [40]. В качестве дополнительных условий, позволяющих получить единственное решение задачи (рисунок 1.2), выступают ограничения, определяемые экспертной информацией. При этом под решением обратных задач с помощью обратных вычислений понимается получение точечных значений приростов аргументов функции на основании ее задаваемого значения и дополнительной информации, поступающей от лица, формирующего решение. В частности, в качестве такой информации могут быть указаны коэффициенты относительной важности целей, индивидуальные коэффициенты прироста аргументов, единый коэффициент прироста аргументов. Такая задача будет в данной работе в дальнейшем именоваться классической обратной задачей.

Рассмотрим применение обратных вычислений в случае использования коэффициентов относительной важности. Для решения задачи двух аргументов определяется следующая информация:

– начальные значения аргументов ($x_1,\ x_2$) и функции (y);

– новое значение функции ($y^* = y \pm \Delta y$);

– коэффициенты относительной важности аргументов (β_1, β_2), определяющие пропорции изменения аргументов;

– направление изменений аргументов (+,–), которые устанавливают, будет ли значение аргумента уменьшено либо увеличено.

Для графического представления прямых и обратных задач Одинцовым Б.Е. было использовано дерево показателей и дерево целей соответственно. На рисунке 1.8 представлено дерево показателей. Стрелками обозначено направление расчётов и связь показателей: величина показателя более высокого уровня вычисляется на основе значений показателей, расположенных уровнем ниже. На рисунке 1.9 приведено дерево целей. На нём кроме исходных значений показателей отражены также их изменения (за исключением показателя самого верхнего уровня, для которого изменение задано), которые необходимо вычислить. Связи показателей характеризуются коэффициентами относительной важности и направлением изменения показателей. В случае, если один показатель используется при расчёте разных величин, вместо дерева будет получен граф.

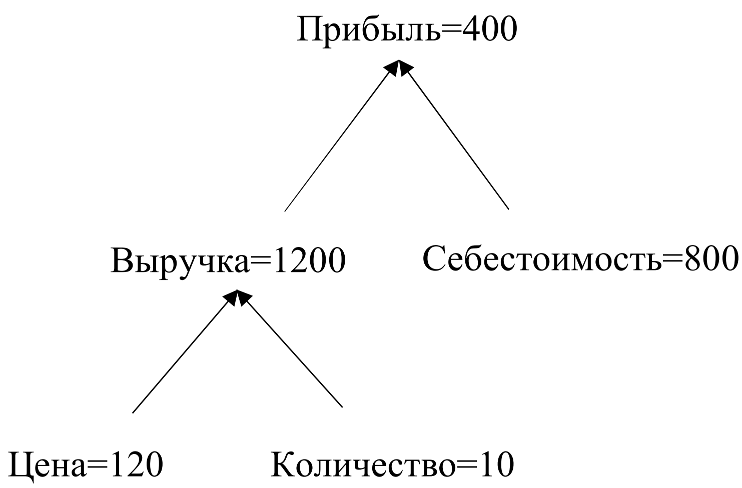


Рисунок 1.8 – Дерево показателей при формировании прибыли

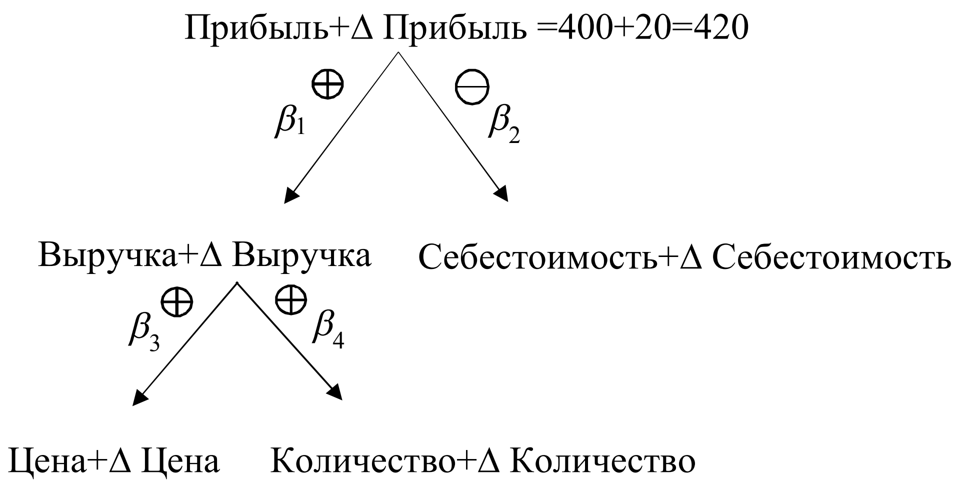


Рисунок 1.9 – Дерево целей

Для решения обратной задачи с двумя аргументами формируется система уравнений

$$\begin{cases} y \pm \Delta y = f(x_1 \pm \Delta x_1(\beta_1), x_2 \pm \Delta x_2(\beta_2)); \\ \dfrac{\Delta x_1}{\Delta x_2} = \dfrac{\beta_1}{\beta_2}; \\ \beta_1 + \beta_2 = 1. \end{cases}$$

Решением системы будут величины Δx_1 и Δx_2, обеспечивающее значение результирующего показателя, равное $y \pm \Delta y$.

Уравнения системы отражают соответствие изменений аргументов коэффициентам относительной важности и направлению изменения показателей и равенство результирующего показателя заданному значению.

С помощью экспертной информации задается вектор, по направлению которого должно происходить изменение аргументов до достижения результирующим показателем заданного значения. Данная информация может быть определена исходя из возможностей организации, например, производственных мощностей, существующих зависимостей, прогнозных изменений показателей. Также коэффициенты относительной важности могут отражать приоритет для достижения цели, определяемый сложностью и стоимостью привлечения ресурсов для изменения аргумента, таким образом, параметр с наибольшим приоритетом будет увеличиваться в большей степени. В таблице 1.1 представлены варианты достижения цели для различных моделей (мультипликативной, аддитивной, кратной), включающих два аргумента. Знак «+» в таблице рядом с аргументом означает, что для достижения целевой установки (положительного или отрицательного изменения результирующего показателя) значение конкретного аргумента необходимо увеличить. Аналогично знак «–» говорит о необходимости уменьшения значения аргумента. Согласно представленным данным достижение заданного значения результирующего показателя при фиксированных коэффициентах относительной важности может быть выполнено двумя способами: изменение аргументов осуществляется в одном направлении (++ или – –), изменение аргументов осуществляется в разных направлениях (+– или –+).

1.3.1 Модели формирования показателей

1.3.1.1 Модели формирования финансовых показателей

Оценке деятельности объектов экономики с помощью стандартных моделей, совершенствованию существующих и разработке новых методик расчета показателей деятельности уделяется большое внимание в литературе. Выбор модели осуществляется специалистом и обуславливается объектом исследования, изучаемыми процессами, целью исследования. Так, например, в работе [49] рассматривается формирование показателей деятельности предприятия ЗАО «НПО Флейм»: рентабельность затрат и уровень компетенции персонала, которые входят в финансовую составляющую и составляющую развития и обучения системы сбалансированных показателей. Структура показателей представляется в виде многоуровневого дерева.

Таблица 1.1 – Варианты достижения целей для различных моделей

Вид зависимости	Условия	Изменение результирующего показателя			
		+		–	
Мультипликативная	$x_1(\beta_1)\cdot x_2(\beta_2)$, $\beta_1>\beta_2$	x_1^+, x_2^+	x_1^+, x_2^-	x_1^-, x_2^-	x_1^-, x_2^+
	$x_1(\beta_1)\cdot x_2(\beta_2)$, $\beta_1<\beta_2$	x_1^+, x_2^+	x_1^-, x_2^+	x_1^-, x_2^-	x_1^+, x_2^-
Аддитивная	$x_1(\beta_1)+x_2(\beta_2)$, $\beta_1>\beta_2$	x_1^+, x_2^+	x_1^+, x_2^-	x_1^-, x_2^-	x_1^-, x_2^+
	$x_1(\beta_1)+x_2(\beta_2)$, $\beta_1<\beta_2$	x_1^+, x_2^+	x_1^-, x_2^+	x_1^-, x_2^-	x_1^+, x_2^-
Кратная	$\dfrac{x_1(\beta_1)}{x_2(\beta_2)}$, $\beta_1>\beta_2$	x_1^+, x_2^+	x_1^+, x_2^-	x_1^-, x_2^-	x_1^-, x_2^+
	$\dfrac{x_1(\beta_1)}{x_2(\beta_2)}$, $\beta_1<\beta_2$	x_1^-, x_2^-	x_1^+, x_2^-	x_1^+, x_2^+	x_1^-, x_2^+

В монографии [40] представлено дерево показателей, формирующих рентабельность, и приведено решение обратных задач для отдельных фрагментов этого дерева. Рассмотрим некоторые из показателей, которые будут далее использованы в работе для апробации моделей, методов и алгоритмов решения обратных задач. Наиболее важным показателем деятельности любой экономической системы является прибыль [105–107].

Для тестирования разработанных моделей, методов и алгоритмов были выбраны следующие функциональные зависимости, наиболее часто используемые в экономическом анализе:

– аддитивная

$$y = x_1 + x_2 + x_3 + x_4 + x_5, \tag{1.9}$$

– мультипликативная (модель выпуска продукции)

$$y = x_1 \cdot x_2 \cdot x_3 \cdot x_4, \tag{1.10}$$

– нелинейная аддитивная (модель затрат в системе управления запасами)

$$y = a_1 / x_1 + b_1 \cdot x_1 + a_2 / x_2 + b_2 \cdot x_2 + a_3 / x_3 + b_3 \cdot x_3\,, \qquad (1.11)$$

– нелинейная мультипликативная (модель Кобба-Дугласа)

$$y = a \cdot x_1^{b_1} \cdot x_2^{b_2}\,, \qquad (1.12)$$

– мультипликативно-аддитивная (модель прибыли)

$$y = x_1 \cdot x_2 - x_3 - x_4. \qquad (1.13)$$

Исходные данные моделей представлены в таблице 1.2–1.4. Направления изменения аргументов при определении экспертной информации – положительные.

Таблица 1.2 – Начальные значения аргументов

Вид модели	x_1	x_2	x_3	x_4	x_5	y^*
Аддитивная	100	200	250	150	130	1000
Мультипликативная	10	26	10	8	–	25000
Нелинейная аддитивная	7	5	4	–	–	17
Нелинейная мультипликативная	2	1,15	–	–	–	17
Мультипликативно-аддитивная	100	50	500	1000	–	4500

Таблица 1.3 – Коэффициенты относительной важности аргументов

Вид модели	x_1	x_2	x_3	x_4	x_5
Аддитивная	0,4	0,2	0,1	0,2	0,1
Мультипликативная	0,4	0,3	0,2	0,1	–
Нелинейная аддитивная	0,3	0,5	0,2	–	–
Нелинейная мультипликативная	0,4	0,6	–	–	–
Мультипликативно-аддитивная	0,4	0,3	0,1	0,2	–

Таблица 1.4 – Дополнительные значения, участвующие в расчётах

Вид модели	a_1	a_2	a_3	b_1	b_2	b_3
Нелинейная аддитивная	20	20	25	0,15	0,05	0,05
Нелинейная мультипликативная	7	–	–	0,5	0,3	–

Мультипликативно-аддитивная модель соответствует маржинальной методике расчёта прибыли, которая подразумевает разделение затрат на постоянные и переменные (рисунок 1.10). В этом случае формулы расчета будут иметь вид:

– прибыль от продаж (*П*) равна разности маржинальной прибыли/маржинального дохода (*МП*) и постоянных затрат (*ПОСТ*)

$$П = МП – ПОСТ. \tag{1.14}$$

– маржинальная прибыль (*МП*) равна разности выручки (*В*) и переменных затрат (*ПЕР*)

$$МП = В – ПЕР. \tag{1.15}$$

– общие переменные затраты (*ПЕР*) равны произведению объема выпуска (*К*) и переменных затрат на единицу продукции ($ПЕР_е$)

$$ПЕР = К \cdot ПЕР_е\,. \tag{1.16}$$

Подставив формулы для вычисления затрат, а также представив выручку как произведение цены (Ц) и объема выпуска (К), получим следующую модель расчёта прибыли

$$П = К \cdot Ц – ПЕР – ПОСТ. \tag{1.17}$$

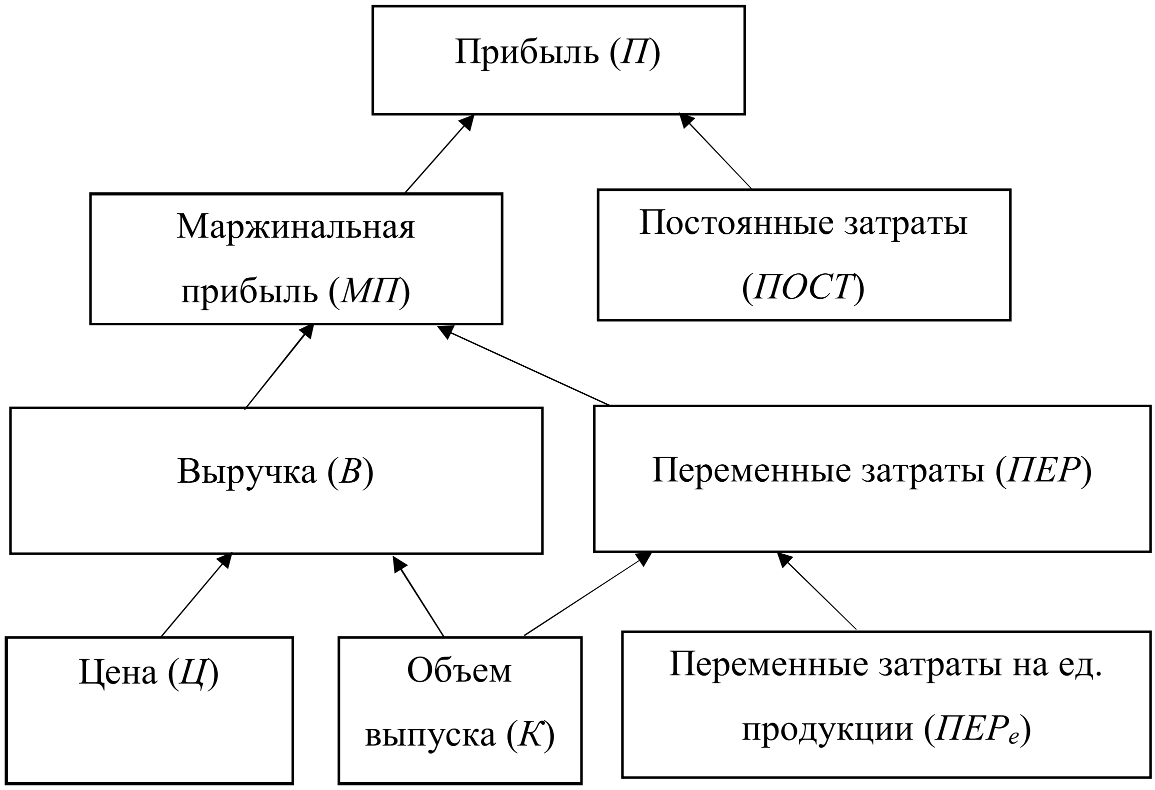


Рисунок 1.10 – Граф формирования прибыли предприятия по маржинальной методике

Аддитивная модель является распространенной в литературе для расчёта обобщающих показателей. Так в работе [34] рассмотрено вычисление следующих показателей:

– стоимость оборотного капитала (О) равна сумме производственных запасов (З), незавершенного производства (Н), прочих элементов оборотного капитала (ПР)

$$O = З + Н + ПР. \tag{1.18}$$

– производственные переменные затраты (*ПРП*) равны сумме прямых материальных затрат (*ПМЗ*), прямых затрат на оплату труда (*ПЗТ*), прочих производственных переменных затрат (*ПР1*)

$$ПРП = ПМЗ + ПЗТ + ПР1. \tag{1.19}$$

В качестве примера мультипликативной модели может быть рассмотрен такой показатель как валовая продукция [108]. В работе [109] подробно рассмотрен факторный анализ данного показателя. Граф показателей представлен на рисунке 1.11. При расчёте показателей используется мультипликативная модель: показатель, расположенный уровнем выше, получается путем перемножения формирующих его величин, расположенных уровнем ниже. Так, например, валовая продукция (*ВП*) равна произведению среднесписочной численности рабочих (*ЧР*) и среднегодовой выработки одного рабочего (*ГВ*)

$$ВП = ЧР \cdot ГВ. \tag{1.20}$$

Рассмотренные показатели (1.9–1.21) являются наиболее распространенными при анализе деятельности экономических объектов, кроме того, они являются примерами различных моделей зависимости (рисунок 1.12). Следовательно, разработанные для данных моделей методы и алгоритмы решения обратных задач могут быть использованы для других показателей, вычисляемых по аналогичной схеме.

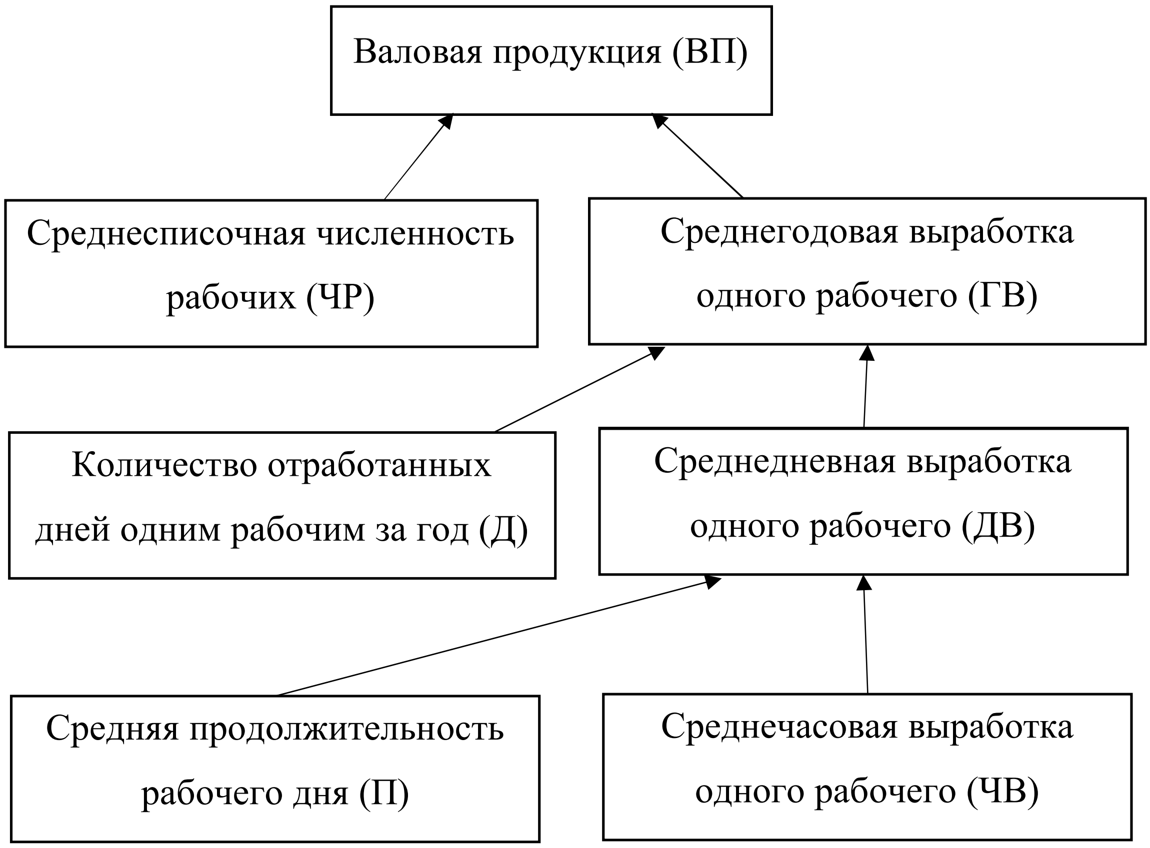


Рисунок 1.11 – Граф формирования валовой продукции

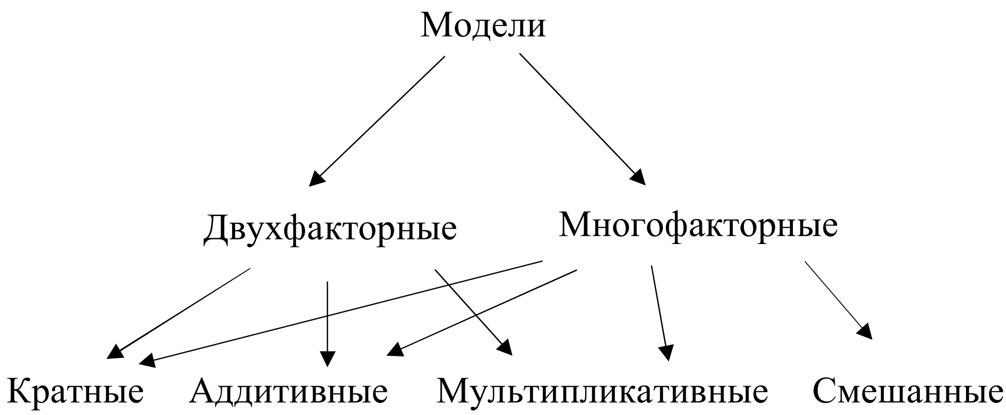


Рисунок 1.12 – Классификация используемых моделей

1.3.1.2 Модель вычисления интегрального показателя

Анализируя существующие исследования, можно сделать вывод, что аппарат обратных вычислений достаточно часто используется для формирования интегральных показателей [46–48]. Метод интегральных критериев является одним из методов решения задач многокритериальной оптимизации, в которой необходимо осуществить поиск оптимального варианта по широкому спектру показателей. Формирование интегрального показателя и построение рейтинга позволяет изучить различия между исследуемыми объектами, оценить положение объекта относительно общей массы, проанализировать динамику его развития.

Область применения интегральных показателей и рейтингов довольно обширна [110]: банковская, страховая сфера, финансовые инструменты, качество управления и услуг, деятельность предприятий, региональное развитие и т.д. Широкий спектр рейтингов и исследований публикуют такие международные агентства как Moody’s, Standard&Poor’s, Fitch Rating.

Можно выделить следующие основные элементы рейтинговой системы [111] (рисунок 1.13):

– участник системы – это исследуемый объект, для которого выполняется расчет рейтинга.

– индикатор – показатель, характеризующий определенное свойство участника.

– целевая функция – правило преобразования индикаторов в интегральную характеристику с целью её сравнения с другими показателями.

– рейтинг – число, полученное путем преобразования индикаторов в единый показатель.

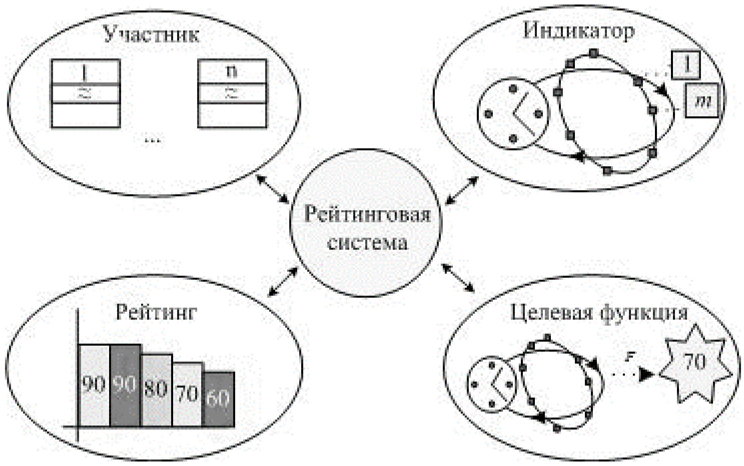


Рисунок 1.13 – Основные элементы рейтинговой системы

Часто также рассматривается иерархическая структура показателей, подразумевающая объединение индикаторов в группы (рисунок 1.14) [112].

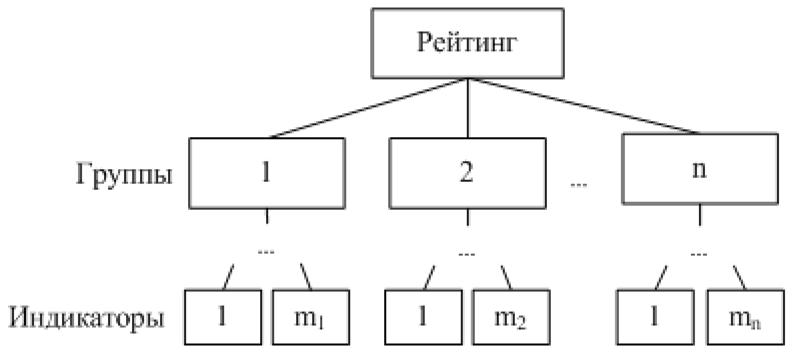


Рисунок 1.14 – Иерархическая структура показателей

Выделение групп может быть выполнено с помощью экспертного анализа, а также с использованием статистических методов. Так, например, в работе [113] рассматривается выделение групп для объединения индикаторов с применением кластерного анализа.

Для преобразования индикаторов в интегральный показатель используются различные методики, включающие два основных этапа: нормирование индикаторов и расчёт интегрального показателя. Наиболее популярным способом расчета стандартизированных оценок [111, 115–117], который ещё называется методом эталонного предприятия, является деление всех индикаторов на максимальное значение

$$x_{ij} = \frac{a_{ij}}{\max a_{ij}},$$

где x_{ij} – i-й стандартизованный коэффициент j-го участника;

a_{ij} – i-й индикатор j-го участника.

Если с экономической точки зрения лучшим является минимальное значение, то осуществляется деление минимальной величины на значения показателей

$$x_{ij} = \frac{\min a_{ij}}{a_{ij}}.$$

Все полученные в результате данного преобразования коэффициенты в случае, если все величины имеют один знак (положительный или отрицательный), принимают значения от 0 до 1, причем, чем выше значение стандартизованных оценок, тем лучше показатель деятельности экономического объекта.

В случае, если среди показателей a_{ij} есть отрицательные значения, то применяется следующая формула нормирования:

$$x_{ij} = \frac{a_{ij} - \min a_{ij}}{\max a_{ij} - \min a_{ij}}.$$

При этом если наилучшим значением является минимальное, то формула будет иметь вид

$$x_{ij} = \frac{\max a_{ij} - a_{ij}}{\max a_{ij} - \min a_{ij}}.$$

Нормирование индикаторов может быть также выполнено путем их деления на среднее значение. В частности, такой способ расчета приводится в официальной методике расчета интегрального показателя отклонения уровня социально-экономического развития субъектов Российской Федерации от среднероссийского, приведенной в приложении 6 к федеральной целевой программе «Сокращение различий в социально-экономическом развитии регионов Российской Федерации (2002-2010 годы и до 2015)». Интегральный показатель рассчитывается как среднее значение полученных величин.

Для расчета стандартизированных коэффициентов могут быть использованы и другие способы преобразования. Так, в работе [114] рассматривается синтетический индекс, построенный на основе нормированных значений исходных показателей, полученных путем вычитания среднего значения и деления на среднее квадратическое отклонение.

Большинство методик расчета рейтинговой оценки предполагает использование весовых коэффициентов, определяющих вклад индикаторов и групп показателей в общий рейтинг. Данные коэффициенты могут быть определены как экспертным путем, так и с использованием математических методов. В работе [118] предлагается использование эконометрического моделирования для определения весовых коэффициентов. Для этого один из индикаторов используется в качестве результативной величины при построении линейной регрессии, после чего на основе полученного индекса детерминации, корреляции и стандартных ошибок осуществляется расчет коэффициентов, показывающих вклад каждой характеристики в величину результативного показателя. Также определение весовых коэффициентов может быть выполнено с помощью анализа иерархий [119], который позволяет придать количественные выражения качественным оценкам. В статье [120] для определения значений весовых коэффициентов предлагается процедура Саймона. Она заключается в расположении экспертом карточек с названиями показателей, которые размещаются снизу-вверх в соответствии с важностью каждого критерия. Веса рассчитываются путем деления ранга характеристики на сумму рангов. В статье [121] приводится методика построение рейтинга с определением весовых коэффициентов на основе метода главных компонент факторного анализа.

Вычисление интегрального показателя R с использованием весовых коэффициентов γ будет иметь вид

$$R_j = \gamma_1 \cdot x_{1j} + ... + \gamma_n \cdot x_{nj}. \tag{1.21}$$

Существуют также модификации этой формулы. В работе [118] значения x возводятся в квадрат и умножаются на величину весовых коэффициентов, в [122] определяется квадратный корень из суммы квадратов отклонений стандартизованных коэффициентов.

На основе вычисленного значения интегрального показателя R может быть составлен рейтинг объектов, сделан вывод о динамике изменения их состояний. Обратная задача заключается в определении стандартизованных значений x для достижения заданного уровня интегрального показателя.

1.3.2 Решение обратных задач для отдельных видов зависимостей

1.3.2.1 Решение обратной задачи с аддитивной моделью

Рассмотрим процесс решения обратной задачи с помощью аппарата обратных вычислений для различных видов зависимостей.

В случае аддитивной модели с двумя переменными система будет иметь вид [40]

$$\begin{cases} y \pm \Delta y = x_1 \pm \Delta x_1(\beta_1) \pm x_2 \pm \Delta x_2(\beta_2)); \\ \dfrac{\Delta x_1}{\Delta x_2} = \dfrac{\beta_1}{\beta_2}. \end{cases}$$

В таблице 1.5 представлены аналитические формулы расчета изменений аргументов, полученные путем решения данной системы уравнений для каждого отдельного случая.

При этом в случае, когда изменения аргументов осуществляется в одном направлении ($x_1^+, x_2^+, x_1^-, x_2^-$), могут быть использованы более простые формулы

$$\Delta x_1 = \Delta y \cdot \beta_1,$$
$$\Delta x_2 = \Delta y \cdot \beta_2.$$

Таблица 1.5 – Аналитические формулы расчета изменений аргументов

Изменение результирующего показателя	Изменения аргументов	Формулы расчёта изменений аргументов	Формулы расчёта новых значений аргументов
+	x_1^+, x_2^+	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{\Delta y}{\dfrac{\beta_1}{\beta_2} + 1}$	$x_1^* = x_1 + \Delta x_1$, $x_2^* = x_2 + \Delta x_2$
	x_1^+, x_2^-	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{\Delta y}{\dfrac{\beta_1}{\beta_2} - 1}$	$x_1^* = x_1 + \Delta x_1$, $x_2^* = x_2 - \Delta x_2$
–	x_1^-, x_2^-	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{\Delta y}{\dfrac{\beta_1}{\beta_2} + 1}$	$x_1^* = x_1 - \Delta x_1$, $x_2^* = x_2 - \Delta x_2$
	x_1^+, x_2^-	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{\Delta y}{\dfrac{\beta_1}{\beta_2} - 1}$	$x_1^* = x_1 + \Delta x_1$, $x_2^* = x_2 - \Delta x_2$

1.3.2.2 Решение обратной задачи с мультипликативной моделью

В случае мультипликативной модели с двумя переменными система имеет вид

$$\begin{cases} y \pm \Delta y = \left(x_1 \pm \Delta x_1(\beta_1)\right)\left(x_2 \pm \Delta x_2(\beta_2)\right); \\ \dfrac{\Delta x_1}{\Delta x_2} = \dfrac{\beta_1}{\beta_2}. \end{cases}$$

В таблице 1.6 представлены аналитические формулы расчета изменений аргументов. При этом из двух вариантов расчёта Δx_2 выбирается наименьшее положительное число.

Таблица 1.6 – Аналитические формулы расчета изменений аргументов при мультипликативной зависимости

Изменение результирующего показателя	Изменения аргументов	Формулы расчёта изменений аргументов	Формулы расчёта новых значений аргументов
+	x_1^{+}, x_2^{+}	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{-\left(x_1 + x_2 \dfrac{\beta_1}{\beta_2}\right) + \sqrt{\left(x_1 + x_2 \dfrac{\beta_1}{\beta_2}\right)^2 + 4 \cdot \dfrac{\beta_1}{\beta_2} \Delta y}}{2 \cdot \dfrac{\beta_1}{\beta_2}}$	$x_1^{*} = x_1 + \Delta x_1$, $x_2^{*} = x_2 + \Delta x_2$
	x_1^{+}, x_2^{-}	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{-\left(-x_1 + x_2 \dfrac{\beta_1}{\beta_2}\right) + \sqrt{\left(-x_1 + x_2 \dfrac{\beta_1}{\beta_2}\right)^2 - 4 \cdot \dfrac{\beta_1}{\beta_2} \Delta y}}{-2 \cdot \dfrac{\beta_1}{\beta_2}}$	$x_1^{*} = x_1 + \Delta x_1$, $x_2^{*} = x_2 - \Delta x_2$
–	x_1^{-}, x_2^{-}	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{-\left(-x_1 - x_2 \dfrac{\beta_1}{\beta_2}\right) - \sqrt{\left(-x_1 - x_2 \dfrac{\beta_1}{\beta_2}\right)^2 - 4 \cdot \dfrac{\beta_1}{\beta_2} \Delta y}}{2 \cdot \dfrac{\beta_1}{\beta_2}}$	$x_1^{*} = x_1 - \Delta x_1$, $x_2^{*} = x_2 - \Delta x_2$

Изменение результирующего показателя	Изменения аргументов	Формулы расчёта изменений аргументов	Формулы расчёта новых значений аргументов
	x_1^+, x_2^-	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{-\left(-x_1 + x_2\dfrac{\beta_1}{\beta_2}\right) - \sqrt{\left(-x_1 + x_2\dfrac{\beta_1}{\beta_2}\right)^2 + 4\cdot\dfrac{\beta_1}{\beta_2}\Delta y}}{-2\cdot\dfrac{\beta_1}{\beta_2}}$	$x_1^* = x_1 + \Delta x_1$, $x_2^* = x_2 - \Delta x_2$

1.3.2.3 Решение обратной задачи с кратной моделью

В случае кратной модели система имеет вид

$$\begin{cases} y \pm \Delta y = \dfrac{\left(x_1 \pm \Delta x_1(\beta_1)\right)}{\left(x_2 \pm \Delta x_2(\beta_2))\right)}; \\ \dfrac{\Delta x_1}{\Delta x_2} = \dfrac{\beta_1}{\beta_2}. \end{cases}$$

В таблице 1.7 представлены аналитические формулы расчета изменений аргументов в случае кратной модели.

Таблица 1.7 – Аналитические формулы расчета изменений аргументов при кратной модели

Изменение результирующего показателя	Изменения аргументов	Формулы расчёта изменений аргументов	Формулы расчёта новых значений аргументов
+	x_1^+, x_2^+	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{-x_2\left(y + \Delta y\right) + x_1}{y + \Delta y - \dfrac{\beta_1}{\beta_2}}$	$x_1^* = x_1 + \Delta x_1$, $x_2^* = x_2 + \Delta x_2$

Изменение результирующего показателя	Изменения аргументов	Формулы расчёта изменений аргументов	Формулы расчёта новых значений аргументов
	x_1^+, x_2^-	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{-x_2\left(y + \Delta y\right) + x_1}{-y - \Delta y - \dfrac{\beta_1}{\beta_2}}$	$x_1^* = x_1 + \Delta x_1$, $x_2^* = x_2 - \Delta x_2$
	x_1^-, x_2^-	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{-x_2\left(y + \Delta y\right) + x_1}{-y - \Delta y + \dfrac{\beta_1}{\beta_2}}$	$x_1^* = x_1 - \Delta x_1$, $x_2^* = x_2 - \Delta x_2$
–	x_1^-, x_2^-	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{x_2\left(-y + \Delta y\right) + x_1}{-y + \Delta y + \dfrac{\beta_1}{\beta_2}}$	$x_1^* = x_1 - \Delta x_1$, $x_2^* = x_2 - \Delta x_2$
	x_1^-, x_2^+	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{x_2\left(-y + \Delta y\right) + x_1}{y - \Delta y + \dfrac{\beta_1}{\beta_2}}$	$x_1^* = x_1 - \Delta x_1$, $x_2^* = x_2 + \Delta x_2$
	x_1^+, x_2^+	$\Delta x_1 = \dfrac{\Delta x_2 \cdot \beta_1}{\beta_2}$, $\Delta x_2 = \dfrac{x_2\left(-y + \Delta y\right) + x_1}{y - \Delta y - \dfrac{\beta_1}{\beta_2}}$	$x_1^* = x_1 + \Delta x_1$, $x_2^* = x_2 + \Delta x_2$

1.3.3 Решение обратной задачи с многоаргументной функцией

В случае, если число аргументов больше двух, может быть выполнена свёртка показателей [40], позволяющая осуществить решение задачи путем последовательного решения задач двух аргументов. Процедура свертки включает следующие этапы.

1) Преобразование задачи в двухфакторную, которая включает результирующий показатель $y + \Delta y$ и два аргумента: первый фактор x_1 и новый показатель x_2', формируемый из оставшихся $rem=n-1$ аргументов (n – число аргументов функции).

2) Коэффициент относительной важности второго аргумента определяется по формуле

$$\beta_2^* = \sum_{i=2}^{n} \beta_i .$$

3) Решение обратной задачи по определению изменений показателей x_1 и x_2'.

После этого происходит решение задачи для функции двух аргументов, где в качестве результирующего показателя рассматривается полученное значение x_2':

4) Преобразование задачи в двухфакторную, которая включает результирующий показатель x_2' и два аргумента: второй фактор x_2 и новый показатель x_3', формируемый из оставшихся $rem=n-2$ аргументов.

5) Коэффициенты важности аргументов нормируются по формуле

$$\beta_2' = \frac{\beta_2}{\sum_{j=2}^{n} \beta_j},$$

$$\beta_3' = \frac{\sum_{i=3}^{n} \beta_i}{\sum_{j=2}^{n} \beta_j}.$$

6) Решение обратной задачи по определению изменений показателей x_2 и x_3'.

Далее шаги 4–6 повторяются для новой задачи до тех пор, пока величина rem не будет равна 1.

Рассмотрев механизм работы свертки, можно сделать вывод, что данная процедура является достаточно трудоемкой из-за необходимости определения новых переменных и формирования новых подзадач. Такой метод может быть использован для решения задачи «вручную» при небольшом числе аргументов. Для более сложных вариантов могут быть рассмотрены другие методы решения задачи.

Одним из таких методов решения обратной задачи при многоаргументной функции является формирование и решение системы уравнений

$$\begin{cases} y \pm \Delta y = f(x_1 \pm \Delta x_1(\beta_1), x_2 \pm \Delta x_2(\beta_2), ..., x_n \pm \Delta x_n(\beta_n)); \\ \dfrac{\pm \Delta x_1}{\pm \Delta x_2} = \dfrac{\beta_1}{\beta_2}; \\ \dfrac{\pm \Delta x_1}{\pm \Delta x_3} = \dfrac{\beta_1}{\beta_3}; \\ ... \\ \dfrac{\pm \Delta x_1}{\pm \Delta x_n} = \dfrac{\beta_1}{\beta_n}; \\ \beta_1 + \beta_2 + ... + \beta_n = 1. \end{cases} \tag{1.22}$$

Элементы данной системы определяют соответствие изменений аргументам коэффициентам относительной важности и равенство результирующего показателя заданному значению.

Данный способ является наиболее подходящим для решения обратных задач с использованием математических пакетов, которые обладают стандартными средствами решения систем уравнений. Однако при реализации алгоритмов при создании программных систем в случае его применения требуется использование сложных вычислительных методов, в частности для решения систем уравнений (1.22).

В связи с этим для компьютерной реализации может быть использован алгоритм решения задачи с помощью пошагового изменения начальных значений аргументов с заданными коэффициентами относительной приоритетности [40].

Исходные данные алгоритма: точность ε, значения коэффициентов относительной приоритетности β, направление изменение показателя t (принимает два возможных значения: –1, +1), заданное значение функции y^* ($y^* = y \pm \Delta y$), начальное значение функции y ($y_0 = y$), величина разницы между заданным значением результирующего показателя и его начальным значением: $\delta_0 = \left| y^* - y_0 \right|$, исходные значения аргументов x, начальное значение приращения s (малое число), изменение приращения Δs, номер шага k ($k = 1$). Реализация алгоритма включает следующие шаги.

Шаг 1. Определение результирующего показателя

$$y_k = f(x_1 + s \cdot t_1 \cdot \beta_1, x_2 + s \cdot t_2 \cdot \beta_2, ..., x_n + s \cdot t_n \cdot \beta_n).$$

Шаг 2. Расчёт разницы между заданным значением результирующего показателя и его текущим значением

$$\delta_k = \left| y^* - y_k \right|.$$

Шаг 3. Проверка условия окончания алгоритма:

Если $\delta_k > \delta_{k-1}$ или $\left| y^* - y_k \right| < \varepsilon$, то работа алгоритма завершается.

Иначе – изменение приращения: $s = s + \Delta s$, переход на шаг 1.

Достоинством данного алгоритма является простота его компьютерной реализации. Однако недостатком является необходимость выполнения многократных итераций для получения результата, т.к. на каждой итерации происходит изменение аргумента на малую величину (в противном случае может быть получена большая разница с заданным значением результирующего показателя).

1.3.4 Решение обратной задачи при наличии ограничений

Задачи экономики могут предполагать наличие ограничений на величину показателей. Так, например, объем выпуска может быть ограничен производственными ресурсами предприятия, а себестоимость – закупочной стоимостью материалов. В этом случае при решении обратной задачи дефицит/излишек одного ресурса может быть скомпенсирован другим. В работе [41] рассматривается понятие взаимозаменяемости ресурсов, под которой понимается процесс вычисления эквивалентного объема одного или нескольких недостающих ресурсов различной природы. С этой целью введено также понятие условного ресурса – объекта абстрактной природы (цена, банковский процент, объем продаж), с помощью которого могут быть выполнены специальные расчёты. Например, если для увеличения выручки недостаточно материальных ресурсов для увеличения объема продаж, то достижение цели может быть выполнено за счёт условного ресурса: увеличения продажной цены. Таким образом, в данной интерпретации цена также выступает в качестве ресурса наравне с материальными. Данная операция по взаимозаменяемости реализуема за счёт того, что количество и цена связаны формулой, по которой происходит пересчёт выручки.

В статье [42] рассматривается следующий способ решения такой задачи при использовании единого коэффициента прироста аргументов и двухаргументной функции:

1) Вычисление единого коэффициента ω прироста аргументов путем его выражения из функции

$$y + \Delta y = f(x_1 + \omega \cdot \beta_1, x_2 + \omega \cdot \beta_2).$$

2) Определение нового значения коэффициента важности для аргумента, на значение которого наложено ограничение. Так, в случае, если установлено максимальное значение для первого аргумента, новая величина коэффициента относительной важности β_1' будет определена из формулы

$$x_1 + \omega \cdot \beta_1' = \max(x_1 + \Delta x_1).$$

3) Определение нового значения второго коэффициента относительной важности исходя из условия равенства суммы коэффициентов единице $\beta_2' = 1 - \beta_1'$.

4) Решение обратной задачи с новыми значениями коэффициентов относительной важности

$$x_1^* = x_1 + \omega \cdot \beta_1',$$
$$x_2^* = x_2 + \omega \cdot \beta_2'.$$

Обобщенно данный алгоритм может быть представлен следующим образом:

1) Рассчитать значения искомых величин x_1, x_2, обеспечивающих заданное значение функции $y + \Delta y$ с помощью обратных вычислений. Если полученные значения соответствуют ограничениям, то они являются решением задачи, иначе происходит переход к следующему шагу.

2) Присвоить показателю граничное значение. Если для величины установлена и верхняя и нижняя граница, то используется ближайшее к полученному решению значение.

3) Подставить значение в исходную функцию и определить величину второго показателя путем решения полученного уравнения.

В качестве недостатка предложенного способа отмечено отсутствие возможности получения решения для определенного соотношения коэффициентов относительной важности.

В статье [41] описана итерационная процедура оптимизации, которая заключается в последовательном изменении функционала и определении величин изменений аргументов, что позволяет получить результат с учетом заданных ограничений. Данный метод может быть использован для многоаргументных функций. Каждая итерация алгоритма состоит из двух шагов:

1) Изменение значения функции на некоторое число, определение приростов ресурсов с помощью обратных вычислений;

2) Перерасчёт изменений аргументов в случае обнаружения дефицита ресурса и необходимости его восполнения за счёт других. Если есть ресурсы, которые не исчерпаны, то происходит возврат на шаг 1.

При перерасчёте изменений аргументов происходит присвоение аргументам, значения которых не соответствуют ограничениям, граничных значений и изменение коэффициентов относительной важности для ресурсов, которые могут быть использованы в качестве заменителя. При этом новые величины коэффициентов важности определяются по формуле нормирования

$$\beta_i' = \frac{\beta_i}{\sum_{j=1}^{k} \beta_j},$$

где β_i' – новая величина коэффициента относительной важности i-го ресурса;

β_i – исходная величина коэффициента относительной важности i-го ресурса;

k – количество ресурсов, которые могут быть использованы в качестве заменителя.

Шаги 1–2 выполняются до исчерпания ресурсов либо до достижения функцией заданного значения.

В качестве недостатка данного метода решения обратной задачи с ограничениями можно отметить трудоемкость вычислений, поскольку на каждой итерации необходимо выполнять проверку на соответствие ограничениям, корректировать значения коэффициентов относительной важности и решать обратную задачу, задействовав возможные ресурсы.

1.3.5 Решение обратной задачи при использовании аргументов в нескольких подзадачах

Один или несколько показателей могут участвовать в нескольких функциях расчёта. Так случай, когда один и тот же аргумент используется при расчете двух показателей, рассмотрен в статье [43]. Основная суть предложенного метода решения такой задачи заключается в последовательном решении двух обратных задач с целью определения компромиссного варианта. Для этого происходит изменение показателя, участвующего в двух задачах, пропорционально результирующим величинам. Так, для задачи на рисунке 1.15 после определения изменения x_1 в задаче по формированию величины y_1^* (обозначим x_{1,y_1}^*) и величины y_2^* (обозначим x_{1,y_2}^*) происходит определение коэффициентов распределения приростов

$$k_{y_1} = \frac{y_1}{y_1 + y_2};$$

$$k_{y_2} = \frac{y_2}{y_1 + y_2}.$$

где k_{y_1} – коэффициент распределения прироста для первой задачи;

k_{y_2} – коэффициент распределения прироста для второй задачи.

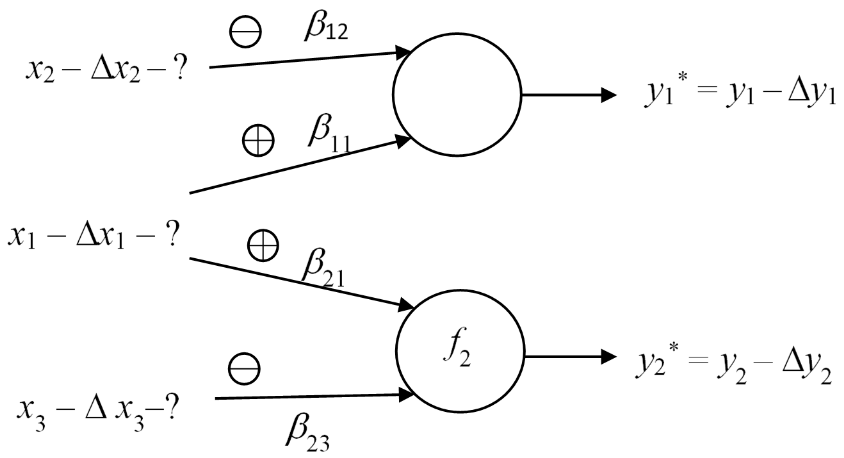


Рисунок 1.15 – Использование аргумента в двух подзадачах

Новые значения изменений аргументов будут равны

$$\Delta\tilde{x}^*_{1,y_1} = (\Delta x^*_{1,y_1} - \Delta x^*_{1,y_2}) \cdot k_{y_1},$$

$$\Delta\tilde{x}^*_{1,y_2} = (\Delta x^*_{1,y_1} - \Delta x^*_{1,y_2}) \cdot k_{y_2}.$$

Поиск компромиссных значений аргументов продолжается до тех пор, пока аналитиком не будет принято решение об остановке расчётов и выборе полученных значений в качестве решения задачи.

Таким образом, данная процедура требует привлечения дополнительной экспертной информации в части поиска компромиссного варианта. Кроме того, она значительно усложняется в случае, когда число задач больше двух, а участвующих одновременно в разных задачах переменных – больше единицы.

1.3.6 Алгоритм решения задачи специалистом с помощью обратных вычислений

Процесс решения обратной задачи специалистом можно представить в виде итерационного алгоритма на рисунке 1.16. Итерационный характер процесса обусловлен следующими причинами:

– при указании входной информации могут быть допущены ошибки (в том числе по причине несогласованности экспертной информации);

– может отсутствовать решение при заданной входной информации;

– присутствует необходимость сравнения нескольких вариантов решения и выбора из них наиболее соответствующего возможностям изменения параметров системы и обеспечивающего минимальный риск недостижения поставленной цели в существующих условиях.

Таким образом, на эксперта возлагается необходимость корректировки входных данных для поиска решения. При большом числе аргументов этот процесс может занять продолжительное время. Решение обратной задачи с помощью обратных вычислений является трудоемким процессом и актуальной является задача по минимизации усилий эксперта и ошибок определения входных данных.

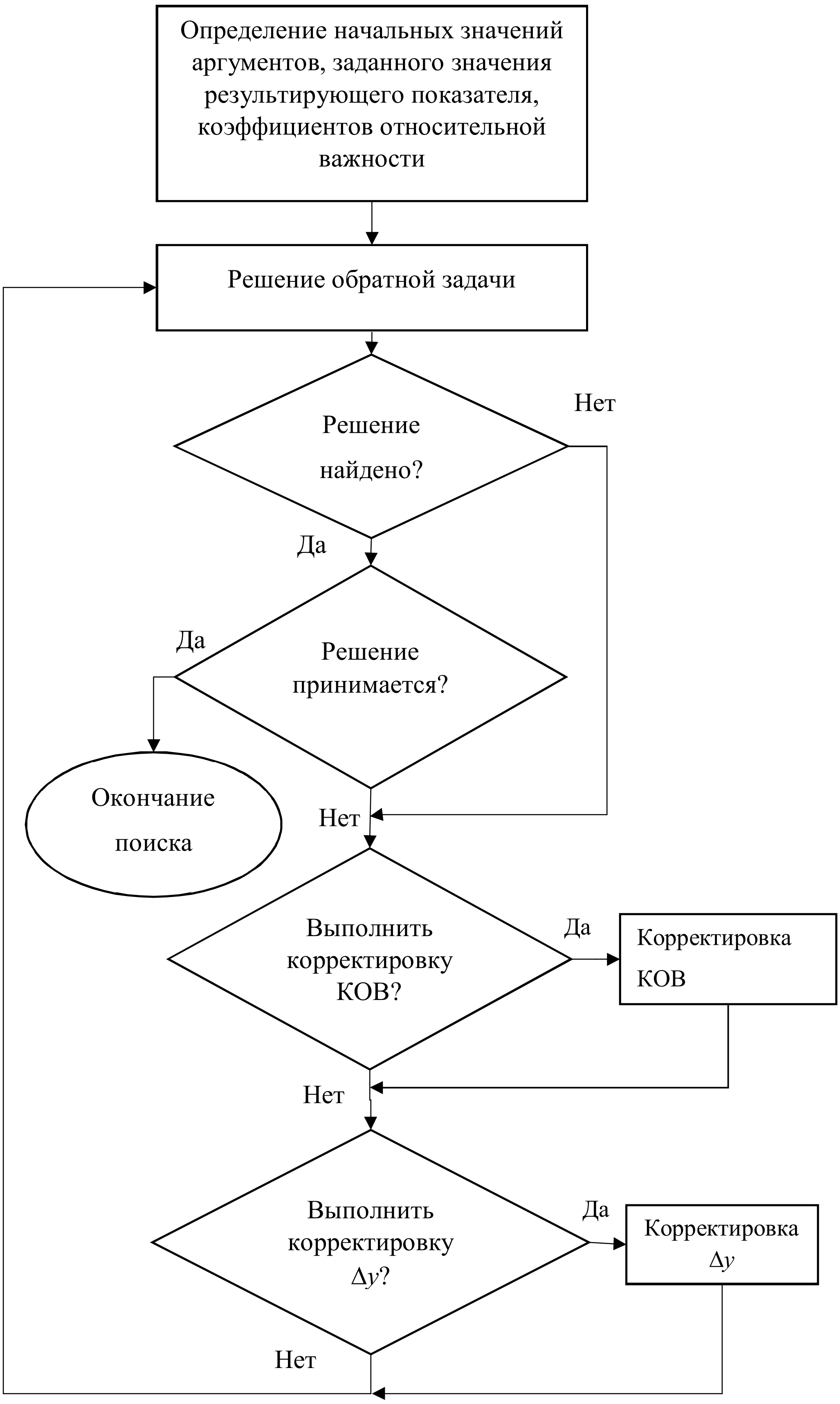


Рисунок 1.16 – Алгоритм поиска решения экспертом

1.4 Выводы по главе 1

1) Обратные задачи играют важную роль в различных областях исследования, однако их решение сопряжено с необходимостью регуляризации.

2) Аппарат обратных вычислений является перспективным и востребованным инструментом решения обратных задач экономики, тем не менее проведенные вычислительные расчёты и анализ текущего состояния исследований позволили сделать вывод о наличии ряда ограничений его использования по следующим причинам:

– получение экспертной информации может быть затруднено и сопряжено с затратами ресурсов;

– определение входной экспертной информации при большом числе аргументов является трудоемким процессом, что может привести к ошибкам установки входных значений;

– отсутствует возможность определения решения, при котором изменения аргументов будут минимальны;

– затруднено использование аппарата в случае рассмотрения более сложных задач: наличие дополнительных ограничений на значения аргументов, участие показателей одновременно в нескольких обратных задачах;

– в случае отсутствия решения задачи требуется корректировка входных значений параметров;

– итерационный характер решения задачи экспертом может привести к высоким затратам вычислительных и временных ресурсов.

3) Выявленные недостатки существующих исследований позволяют сделать вывод о том, что актуальной является задача разработки моделей, методов, позволяющих минимизировать усилия эксперта и ошибки определения входных данных. Кроме того, актуальной является задача разработки алгоритмов решения обозначенных обратных задач, более простых в компьютерной реализации и не

требующих применения сложных вычислительных процедур, что позволит сделать разработку программ более быстрой.

Глава 2. Разработка методов и алгоритмов для решения задач на основе обратных вычислений

2.1 Метод решения обратных задач с помощью формирования уравнения зависимости между аргументами функции

Одним из недостатков классического аппарата обратных вычислений является необходимость определения согласованной экспертной информации, что может привести к ошибочным входным данным. Поскольку специалисту необходимо указать коэффициенты относительной важности и направления изменения показателей, то, следовательно, при решении задачи необходимо выполнять проверку возможности достижения результата с использованием установленных специалистом направлений изменений показателей и значений коэффициентов относительной приоритетности.

Для того, чтобы исключить данные проверки может быть рассмотрен вариант решения задачи, при котором в качестве входных данных указывается один из двух вариантов достижения цели: изменение аргументов в одном направлении, изменение аргументов в разных направлениях. Для этого был разработан метод решения задачи с помощью формирования уравнения зависимости между аргументами.

Метод обратных вычислений с помощью формирования уравнения зависимости между аргументами заключается в определении аргументов функции на основании её указанного значения, начальных значений аргументов, коэффициентов относительной важности и виду зависимости между аргументами [122–123]. Он предполагает построение линейного уравнения связи между аргументами x вида $x_1 = a \pm bx_2$ (где a, b – числовые параметры, определяемые на основе значений коэффициентов относительной важности и исходных значений аргументов) и подстановку полученного уравнения в исходное соотношение.

Для создания уравнения связи между аргументами может быть использован минимаксный метод. Суть его заключается в построении уравнения диагонали прямоугольника, образованного минимальными и максимальными значениями величин (рисунок 2.1), в качестве углового коэффициента используется отношение сторон Lx_1, Lx_2 , величины которых вычисляются по формуле

$$Lx_1 = x_{1\max} - x_{1\min},$$
$$Lx_2 = x_{2\max} - x_{2\min}.$$

При этом необходимо указать направление изменения двух показателей:

– изменение аргументов осуществляется в одном направлении (прямая зависимость): происходит либо уменьшение каждого из аргументов, либо их увеличение.

– изменение аргументов осуществляется в разных направлениях (обратная зависимость), т.е. один из аргументов уменьшается, а второй – увеличивается.

Так, для построения функции обратной зависимости (рисунок 2.1а) используются формулы

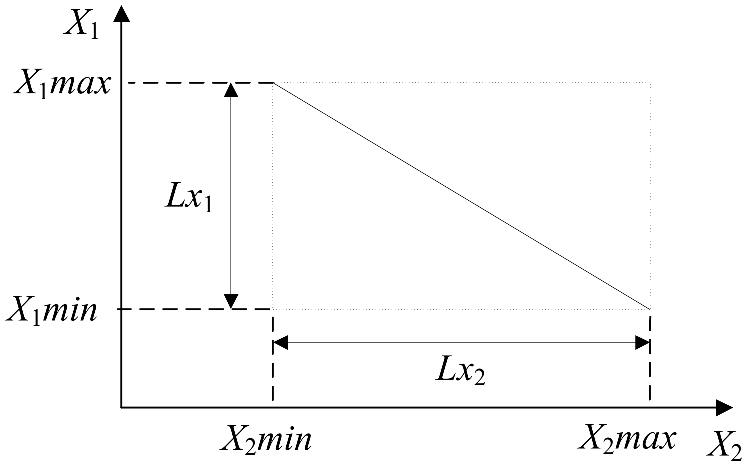
$$b = \frac{Lx_1}{Lx_2},$$

$$a = x_{1\min} + b \cdot x_{2\min}.$$

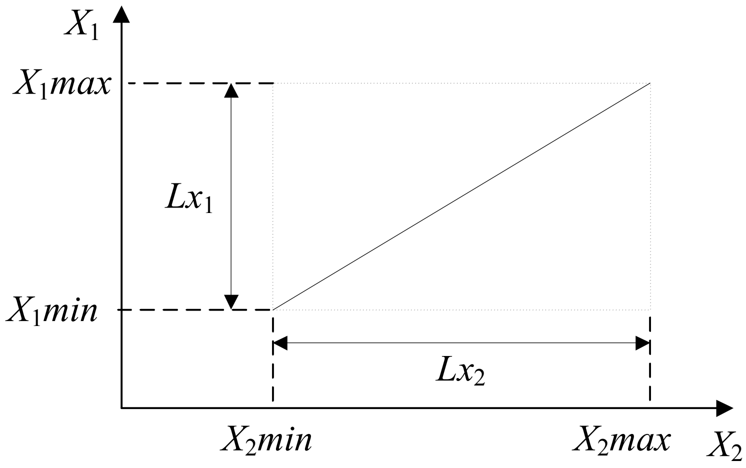
В случае прямой зависимости (рисунок 2.1 б) формулы расчёта параметров имеют вид

$$b = \frac{Lx_1}{Lx_2},$$

$$a = x_{1\min} - b \cdot x_{2\min}.$$



а)



б)

Рисунок 2.1 – Зависимость между аргументами: а) обратная; б) прямая

В предложенном методе решения обратной задачи используется отношение коэффициентов относительной приоритетности в качестве углового коэффициента и исходные данные вместо минимальных значений.

2.1.1 Алгоритм метода решения обратных задач с помощью формирования уравнения зависимости

Таким образом, алгоритм решения обратной задачи следующий:

Шаг 1. Вычислить коэффициенты *a*, *b* линейного уравнения зависимости между аргументами

$$b = \frac{\beta_1}{\beta_2};$$

$$a = x_1 \pm b \cdot x_2;$$

При этом $a = x_1 - b \cdot x_2$, если изменение аргументов происходит в одном направлении, $a = x_1 + b \cdot x_2$, если изменение аргументов осуществляется в разных направлениях.

Шаг 2. Решить систему уравнений

$$\begin{cases} y \pm \Delta y = f(x_1^*, x_2^*); \\ x_1^* = a \pm b x_2^*. \end{cases} \tag{2.1}$$

При этом $x_1^* = a + b x_2^*$, если изменение аргументов происходит в одном направлении, $x_1^* = a - b x_2^*$, если изменение аргументов осуществляется в разных направлениях.

Здесь x_1^*, x_2^* – искомые значения первого и второго аргумента:

$$x_2^* = x_2 + \Delta x_2,$$

$$x_1^* = x_1 + \Delta x_1 = a + b \cdot x_2^*.$$

Решение уравнения осуществляется путем подстановки в функцию выражения для x_1^*. Таким образом также могут быть определены аналитические формулы расчета изменений аргументов.

В таблице 2.1 представлены аналитические формулы для разных видов зависимостей. При этом в случае, если результирующий показатель необходимо увеличить в формуле используется величина $y + \Delta y$, если уменьшить – $y - \Delta y$. Из двух вариантов расчёта Δx_2 выбирается положительное число, наиболее близкое к исходным значениям.

Для оценки точности метода было выполнено сравнение результатов решения задачи с помощью предложенного метода и с помощью обратных вычислений.

Таблица 2.1 – Аналитические формулы расчет изменений аргументов при использовании модифицированного метода обратных вычислений

Вид зависимости	Изменение аргументов	Аналитические формулы расчёта изменений аргументов
Аддитивная	В одном направлении	$x_1^* = x_1 - \frac{\beta_1}{\beta_2} \cdot x_2 + \frac{\beta_1}{\beta_2} \cdot x_2^*, \quad x_2^* = \frac{y \pm \Delta y - x_1 + \frac{\beta_1}{\beta_2} \cdot x_2}{\frac{\beta_1}{\beta_2} + 1}$
	В разных направлениях	$x_1^* = x_1 + \frac{\beta_1}{\beta_2} \cdot x_2 - \frac{\beta_1}{\beta_2} \cdot x_2^*, \quad x_2^* = \frac{y \pm \Delta y - x_1 - \frac{\beta_1}{\beta_2} \cdot x_2}{1 - \frac{\beta_1}{\beta_2}}$
Мультипликативная	В одном направлении	$x_1^* = x_1 - \frac{\beta_1}{\beta_2} \cdot x_2 + \frac{\beta_1}{\beta_2} \cdot x_2^*,$ $x_2^* = \frac{-\left(x_1 - \frac{\beta_1}{\beta_2} \cdot x_2\right) \pm \sqrt{\left(x_1 - \frac{\beta_1}{\beta_2} x_2\right)^2 - 4 \cdot \frac{\beta_1}{\beta_2} \cdot -(y \pm \Delta y)}}{2 \cdot \frac{\beta_1}{\beta_2}}$
	В разных направлениях	$x_1^* = x_1 - \frac{\beta_1}{\beta_2} \cdot x_2 + \frac{\beta_1}{\beta_2} \cdot x_2^*,$ $x_2^* = \frac{-\left(x_1 - \frac{\beta_1}{\beta_2} \cdot x_2\right) \pm \sqrt{\left(x_1 - \frac{\beta_1}{\beta_2} x_2\right)^2 - 4 \cdot \frac{\beta_1}{\beta_2} \cdot -(y \pm \Delta y)}}{2 \cdot \frac{\beta_1}{\beta_2}}$
Кратная	В одном направлении	$x_1^* = x_1 - \frac{\beta_1}{\beta_2} \cdot x_2 + \frac{\beta_1}{\beta_2} \cdot x_2^*,$ $x_2^* = \frac{\left(x_1 - \frac{\beta_1}{\beta_2} \cdot x_2\right)}{y + \Delta y - \frac{\beta_1}{\beta_2}}$
	В разных направлениях	$x_1^* = x_1 - \frac{\beta_1}{\beta_2} \cdot x_2 + \frac{\beta_1}{\beta_2} \cdot x_2^*,$ $x_2^* = \frac{\left(x_1 - \frac{\beta_1}{\beta_2} \cdot x_2\right)}{y + \Delta y - \frac{\beta_1}{\beta_2}}$

В случае решения обратной задачи при многоаргументной функции необходимо применить процедуру свертки.

2.1.2 Применение алгоритма для решения задачи с использованием тестового набора моделей

Рассмотрим случай формирования прибыли (1.17) (рисунок 2.2). Введем новую переменную

$$МП = Ц \cdot К - ПЕР.$$

Полученная двухаргументная задача представлена на рисунке 2.3. Решение задачи будет выполнено следующим образом (использованы формулы для аддитивной модели, представленные в таблице 2.1)

$$МП^* = x_1 - \frac{\beta^*{}_1}{\beta_2} \cdot x_2 + \frac{\beta^*{}_1}{\beta_2} \cdot ПОСТ^* =$$

$$= 4500 - \frac{0,8}{0,2} \cdot 1000 + \frac{0,8}{0,2} \cdot ПОСТ^* = 500 + 4 \cdot ПОСТ^*,$$

$$ПОСТ^* = \frac{y + \Delta y - x_1 + \dfrac{\beta^*{}_1}{\beta_2} \cdot x_2}{\dfrac{\beta^*{}_1}{\beta_2} - 1} = \frac{4500 - 4500 + \dfrac{0,8}{0,2} \cdot 1000}{\dfrac{0,8}{0,2} - 1} = 1333,$$

$$МП^* = 500 + 4 \cdot ПОСТ^* = 500 + 4 \cdot 1333,33 = 5833.$$

Данное решение можно также получить, подставив зависимость $МП^*$ от $ПОСТ^*$ в систему уравнений (2.1)

$$\begin{cases} МП^* = 500 + 4 \cdot ПОСТ^*, \\ МП^* - ПОСТ^* = 4500. \end{cases}$$

В таблице 2.2 приведены этапы решения задачи, в которых последовательно были определены такие переменные как маржинальная прибыль и выручка.

Решение задачи также соответствуют решению с помощью класического аппарата обратных вычислений.

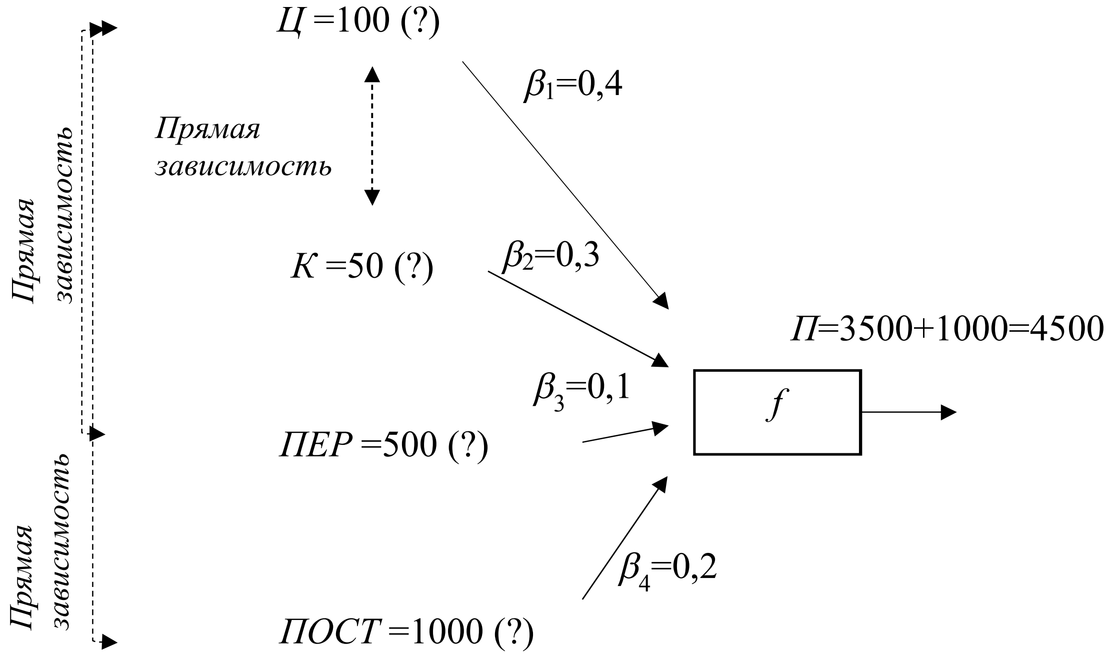


Рисунок 2.2 – Задача формирования прибыли

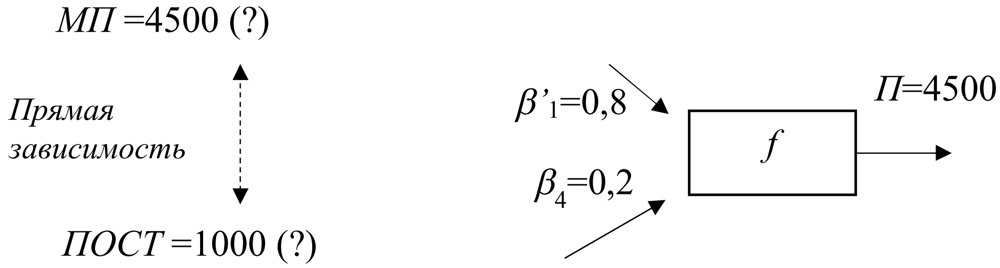


Рисунок 2.3 – Двухаргументная задача формирования прибыли

Таблица 2.2 – Решение задачи в случае многоаргументной функции

Название аргумента 1	Название аргумента 2	КОВ1	КОВ2	Значение аргумента 1	Значение аргумента 2
Маржинальная прибыль	Постоянные затраты	0,8	0,2	5833	1333
Выручка	Переменные затраты	0,875	0,125	6555	722
Цена	Объем продаж	0,571	0,429	111,62	58,73

В таблице 2.3 представлено решение задач по формированию показателей с использованием других тестовых функций (1.9)–(1.13) (данные таблиц 1.2 – 1.4). Полученные числовые значения соответствуют решению задач с помощью классического аппарата обратных вычислений. В случае нелинейной аддитивной модели решение с помощью рассмотренных методов не может быть найдено, т.к. в данной модели показатель формируется с помощью двух слагаемых. В остальных моделях было полное соответствие результатов, полученных с помощью предложенного метода, и с помощью известного метода решения обратных задач с применением свертки.

Таблица 2.3 – Решение задач по формированию показателей

Модель	Показатель	Решение задачи с помощью предложенного метода	Решение задачи с помощью обратных вычислений
Аддитивная	Δx_1	68	68
	Δx_2	34	34
	Δx_3	17	17
	Δx_4	34	34
	Δx_5	17	17
Мультипликативная	Δx_1	2,002	2,002
	Δx_2	0,028	0,028
	Δx_3	0,002	0,002
	Δx_4	0,001	0,001
Нелинейная аддитивная	–	–	–
Мультипликативная нелинейная	Δx_1	1,14	1,14
	Δx_2	1,71	1,71

Таким образом, в отличие от классического варианта в предложенном методе определяются не приращения, а искомые величины. При этом рассматривается возможность получения решения только двумя способами: изменение аргументов осуществляется в одном направлении, изменение аргументов осуществляется в разных направлениях. В отличие от классического метода обратных вычислений метод на основе формирования уравнения зависимости между аргументами функции позволяет определить решение обратной задачи без необходимости проверки согласованности экспертной информации: соответствия поставленной цели коэффициентам относительной важности и направлениям их изменения.

2.2 Стохастический метод решения обратных задач с ограничениями

При большом числе аргументов и наличии ограничений решение обратной задачи с помощью обратных вычислений усложняется из-за необходимости многократного выполнения проверок полученного решения на соответствие ограничениям и корректировок аргументов и коэффициентов относительной важности [40–42]. Использование стохастических методов при решении задач подобного рода позволяет избежать сложных вычислений и найти приближенное решение с учетом коэффициентов важности, ограничений аргументов, в том числе, рассматривать ситуации, когда переменные могут принимать только целые значения либо значения из заданного набора.

Ранее было рассмотрено графическое представление обратной задачи в виде дерева/графа целей, где на нулевом уровне расположено значение результирующей функции, а на нижних – аргументы (рисунок 2.4).

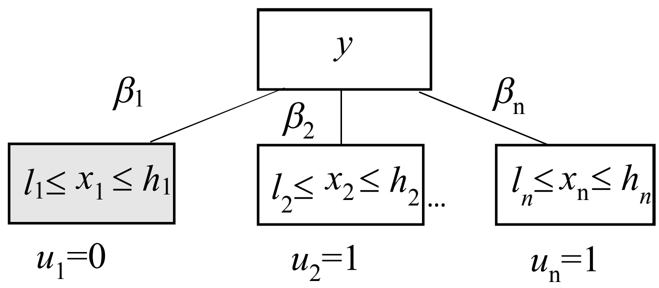


Рисунок 2.4 – Представление задачи в виде дерева

Для использования стохастического метода вместе со стандартными характеристиками (начальные значение *x, y*; заданное/искомое значение y^*, коэффициенты относительной важности β) определим дополнительные элементы дерева целей [124–125] (рисунок 2.4):

– минимальное l и максимальное h значения, которые может принимать данный показатель;

– индикатор u, характеризующий возможность использования данного элемента, и принимающий два значения: 1 (использование возможно) и 0 (использованием невозможно).

Коэффициент относительной важности β указывает степень изменения результирующего показателя за счет данного аргумента.

Значение индикатора u становится равным нулю в случае, если изменение аргумента не может быть выполнено из-за существующего ограничения или отсутствия положительного изменения целевой функции. Также данный индикатор устанавливается равным нулю для величин – констант.

Суть предложенного метода заключается в последовательном изменении результирующего показателя и случайном выборе аргумента для корректировки с помощью моделирования полной группы невовместных событий. При этом коэффициенты важности рассматриваются в качестве вероятности.

2.2.1 Алгоритм стохастического метода решения обратных задач с ограничениями

Для решения обратной задачи был разработан алгоритм, основанный на последовательном изменении функции. Для его применения устанавливается шаг Δy^*, на который будет происходить изменение результирующего показателя (s – номер итерации, y – начальное значение результирующего показателя). Выполнение алгоритма включает следующие этапы.

Шаг 1. Установить новое значение результирующего показателя

$$y_s = y_s + \Delta y^*.$$

Шаг 2. С помощью алгоритма моделирования полной группы несовместных событий выбрать узел из вершин-потомков, для которых значения индикатора равно 1, в соответствии с коэффициентами важности $\beta_1, \beta_2, ..., \beta_n$. Для этого выполняется расчет значений вероятностей по формуле

$$p_i^* = \frac{\beta_i}{\sum_{j=1}^{q} \beta_j}, \; i = 1..q$$

где j – номер вершины, для которой значение индикатора равно 1;

q – число вершин, для которых значение индикатора равно 1.

Выбор узла происходит в зависимости от значения q: если q равно нулю, то осуществляется завершение работы алгоритма, иначе – выбор узла осуществляется с помощью алгоритма на рисунке 2.5.

Шаг 3. Определяется значение $x^*_{s,k}$ выбранной на предыдущем шаге вершины k, при фиксированных значениях остальных величин для получения заданного y_s. Другими словами, происходит решение уравнения $f(x^*_{s,k}) = y_s$.

Выполняется проверка соответствия полученного значения ограничению $l_k \le x^*_{s,k} \le h_k$. Если условие выполняется, то $x_{s,k} = x^*_{s,k}$, и всем вершинам, не являющимся константой, присваивается индикатор, равный единице, а иначе – $u_k = 0$, переход к шагу 1.

Шаг 4. Осуществляется проверка условия: $y^* < y_s$ (достижение заданного значения результирующего показателя). Если условие выполняется, происходит завершение работы алгоритма, иначе – переход к шагу 1.

Полученные значения x_s будут решением задачи.

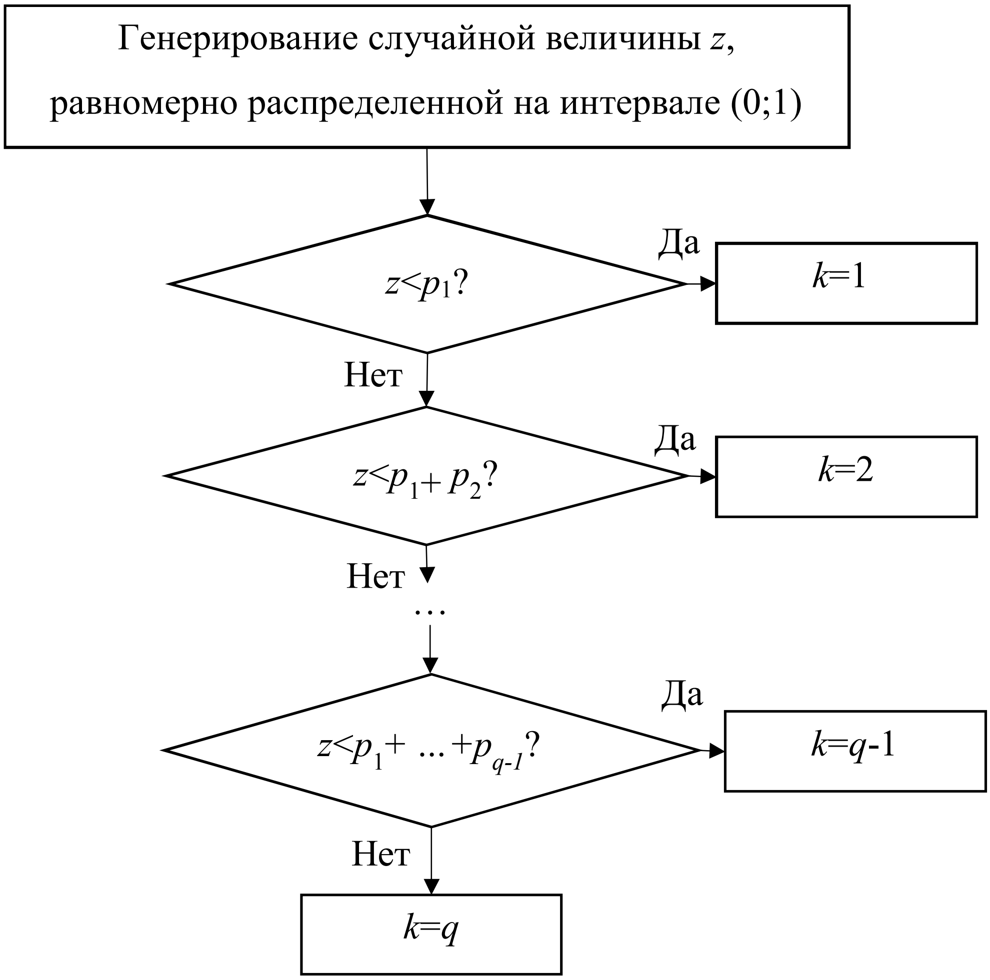


Рисунок 2.5 – Алгоритм выбора номера вершины k

Данный алгоритм может быть использован для аддитивной модели (1.9), в случае иных моделей на третьем шаге при расчёте значения $x^*_{s,k}$ необходимо использовать итерационную формулу

$$x_{s,k} = x_{s-1,k} + \Delta x_{s,k} \cdot \frac{\partial f}{\partial x_{s,k}}, \tag{2.2}$$

где $\Delta x_{s,k}$ – разница между значением аргумента, вычисленным путем решения уравнения относительно него, и предыдущим значением аргумента.

В этом случае алгоритм будет модифицирован следующим образом.

Шаг 1. С помощью алгоритма моделирования полной группы несовместных событий выбрать узел из вершин-потомков, для которых значения индикатора равно 1, в соответствии с коэффициентами важности $\beta_1, \beta_2, ..., \beta_n$.

Шаг 3. Определяется новое значение $x^*_{s,k}$ выбранной на предыдущем шаге вершины k (2.2).

Выполняется проверка соответствия полученного значения ограничению $l_k \le x^*_{s,k} \le h_k$. Если условие выполняется, то $x_{s,k} = x^*_{s,k}$, и всем вершинам, не являющимся константой, присваивается индикатор, равный единице, а иначе – $u_k = 0$, переход к шагу 1.

Шаг 4. Вычислить значение $y_s(x_{s,k})$. Осуществляется проверка условия: $y^* < y_s$ (достижение заданного значения результирующего показателя). Если условие выполняется, происходит завершение работы алгоритма, иначе – переход к шагу 1.

В данных алгоритмах представлена ситуация, когда заданное значение результирующего показателя больше его исходной величины. Если задача заключается в уменьшении его значения, то величина Δy^* будет отрицательной, а на шаге 3 условием окончания будет выполнение ограничения $y^* > y_s$. Описанные в алгоритме шаги могут выполняться многократно с запоминанием решения, наилучшим образом соответствующего заданным значениям коэффициентам относительной важности и результирующего показателя.

2.2.2 Применение стохастического алгоритма для решения задачи с использованием тестового набора моделей

Рассмотрим аддитивную модель (1.9) с пятью аргументами (исходные данные представлены в таблицах 1.2–1.3). Результат работы стохастического алгоритма представлен в таблице 2.4, полученные величины аргументов обозначены x^*. Для каждого значения шага Δy^* было выполнено 5 запусков алгоритма.

Таблица 2.4 – Результаты работы стохастического алгоритма при изменении шага

Номер запуска	Шаг Δy^* =0,1				
	x^*_1	x^*_2	x^*_3	x^*_4	x^*_5
1	165,2	235,4	268,6	183,1	147,7
2	167,7	234,1	265,1	185,9	147,2
3	166,9	233,1	267,1	183,8	149,1
4	166,4	234,6	267,9	184,5	146,6
5	171,1	234	266,8	183,3	144,8
	Шаг Δy^* =0,01				
1	168	233,94	266,79	184,19	147,09
2	168,74	233,28	267,52	183,49	146,98
3	168,23	233,16	267,37	183,87	147,38
4	167,65	233,76	267,24	183,75	147,61
5	169,07	233,15	266,62	184,53	146,64
	Шаг Δy^* =0,001				
1	167,92	234,31	266,94	183,83	146,98
2	167,77	234,13	266,86	184,25	146,99
3	167,69	233,97	267,11	184,21	147,02
4	167,91	233,99	267,00	184,1	146,98
5	167,73	234,2	266,93	184,17	146,97
	Шаг Δy^* =0,0001				
1	168,1	233,91	267	183,98	147,01
2	168,02	234,01	266,97	183,99	147,01
3	168,05	234	267	184	146,94
4	167,87	233,97	267,07	184,07	147,02
5	168	233,99	266,92	184,07	147,03
	Шаг Δy^* =0,00001				
1	167,9	234,02	266,95	184,06	147,06
2	167,99	233,96	267,12	184,01	146,93
3	168,08	233,92	266,99	183,95	147,06
4	167,89	234,07	267,02	183,97	147,05
5	168,03	233,97	266,96	184,06	146,98

Полученные значения аргументов при разных значениях шага Δy^* сведены в графики сходимости (рисунки 2.6–2.10).

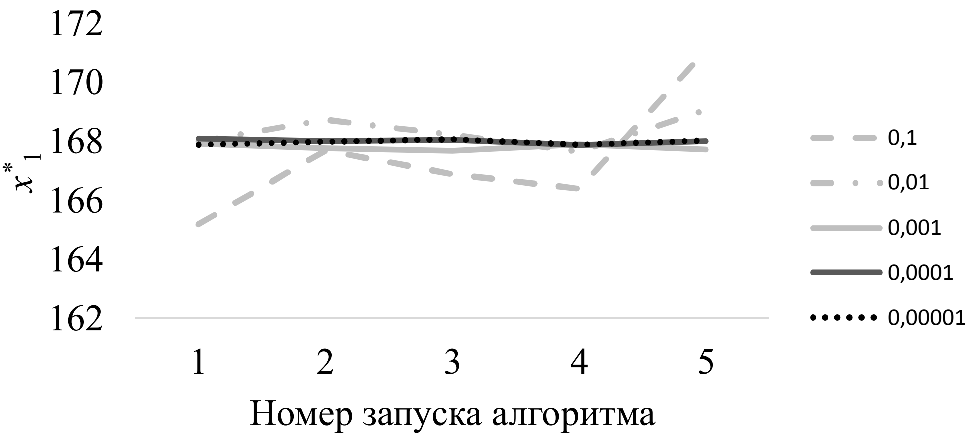


Рисунок 2.6 – График сходимости значения x^*_1 при величине шага Δy^* 0,1, 0,01, 0,001, 0,0001, 0,00001

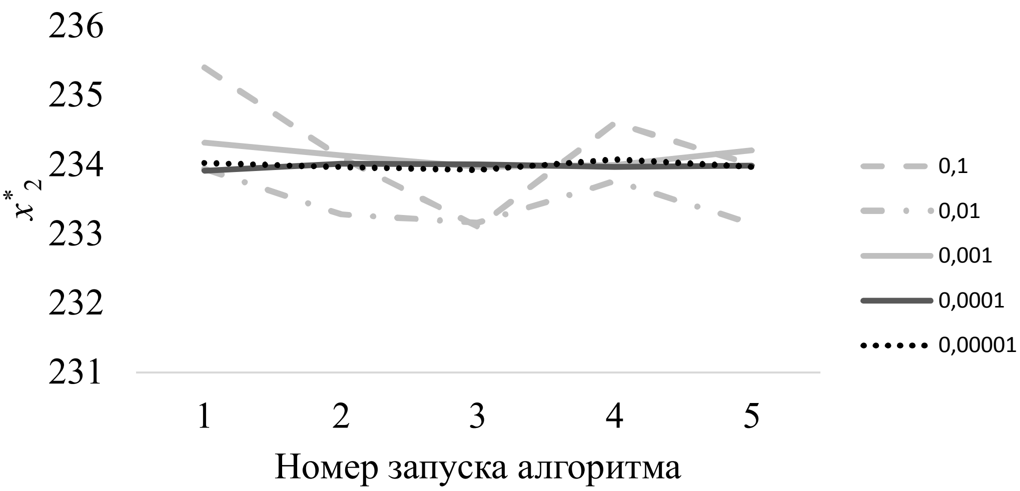


Рисунок 2.7 – График сходимости значения x^*_2 при величине шага Δy^* 0,1, 0,01, 0,001, 0,0001, 0,00001

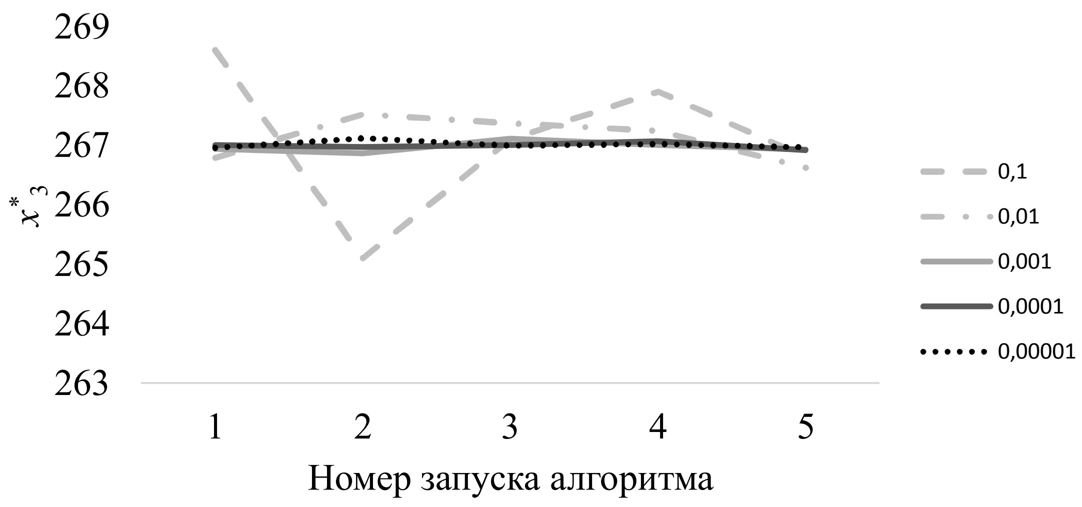


Рисунок 2.8 – График сходимости значения x^*_3 при величине шага Δy^* 0,1, 0,01, 0,001, 0,0001, 0,00001

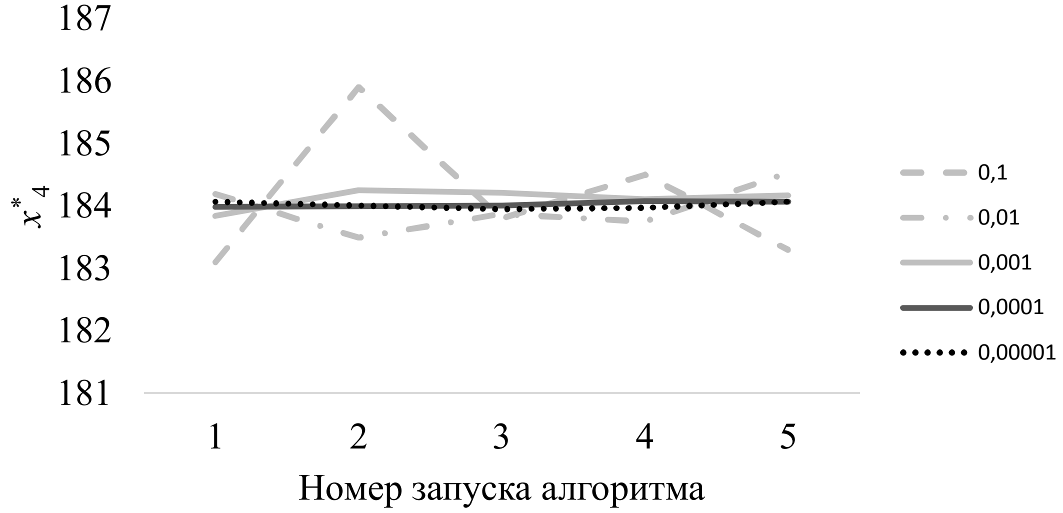


Рисунок 2.9 – График сходимости значения x^*_4 при величине шага Δy^* 0,1, 0,01, 0,001, 0,0001, 0,00001

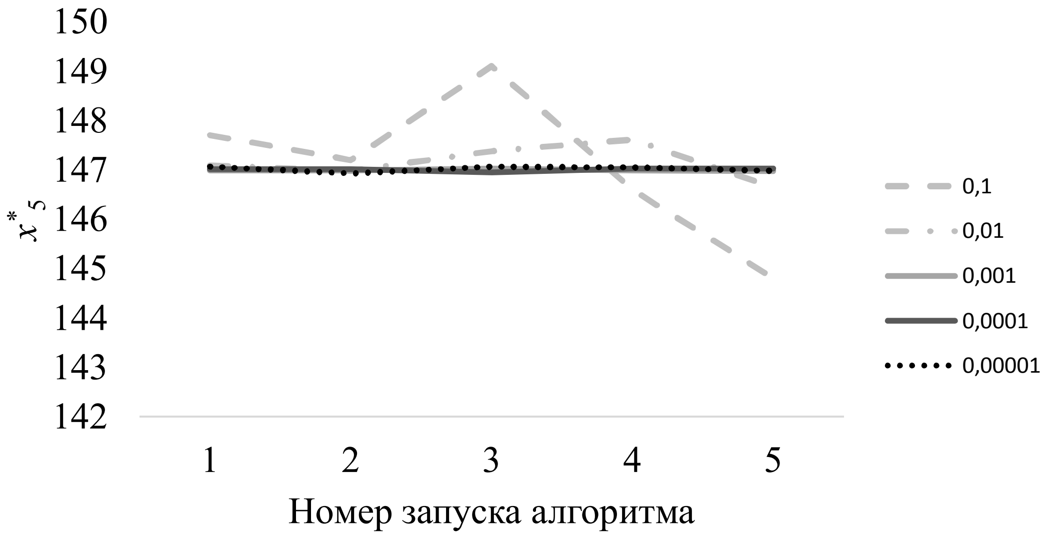


Рисунок 2.10 – График сходимости значения x^*_5 при величине шага Δy^* 0,1, 0,01, 0,001, 0,0001, 0,00001

Из рисунков 2.6–2.10 видно, что с уменьшеним шага Δy^* сходимость улучшается. Так при величине шага 0,1 максимальная дисперсия полученных значений x составляет 4,96, при шаге 0,00001 максимальная дисперсия равна 0,007.

Оценка адекватности была выполнена с использованием критерия Стьюдента для сравнения выборочного среднего с заданным значением (с помощью критерия хи-квадрат был сделан вывод о нормальности распределения). Для этого было проведено 50 экспериментов при величине шага, равном 0,00001. В качестве заданного значения использовано решение задачи с помощью обратных вычислений (x_1, x_2, x_3, x_4, x_5 равны 168, 234, 267, 184, 147 соответственно). На рисунке 2.11 для примера представлены значения первого аргумента. В таблице 2.5 приведены средние значения, выборочная дисперсия, значения статистики. Для количества степеней свободы, равного 49, и уровня значимости 0,05 критическое значение равно 2. Все вычисленные значения статистики меньше критического, следовательно, нулевая гипотеза о равенстве выборочного среднего заданному значению не отвергается.

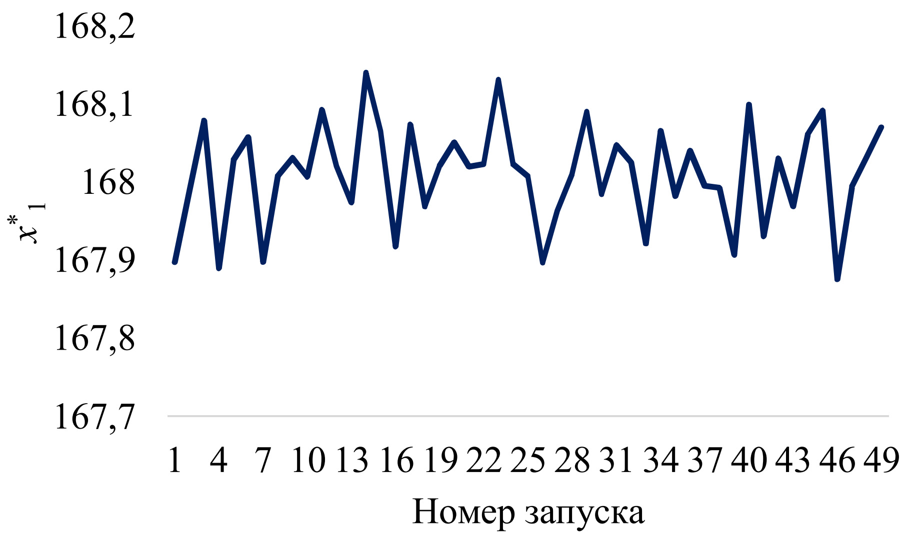


Рисунок 2.11 – График изменения значений x^*_1

Таблица 2.5 – Значения статистики критерия Стьюдента

Показатель	x^*_1	x^*_2	x^*_3	x^*_4	x^*_5
Среднее значение	168,01	234,007	266,992	183,992	146,998
Выборочная дисперсия	0,004	0,003	0,002	0,003	0,002
Значение статистики критерия	1,07	0,97	–1,3	–1,06	–0,3

Стохастический алгоритм имеет временную сложность O($n \times m$), где n – число аргументов, m – число итераций. На рисунке 2.12 приведено изменение времени решения задачи с аддитивной моделью в зависимости от числа аргументов (от 2 до 10).

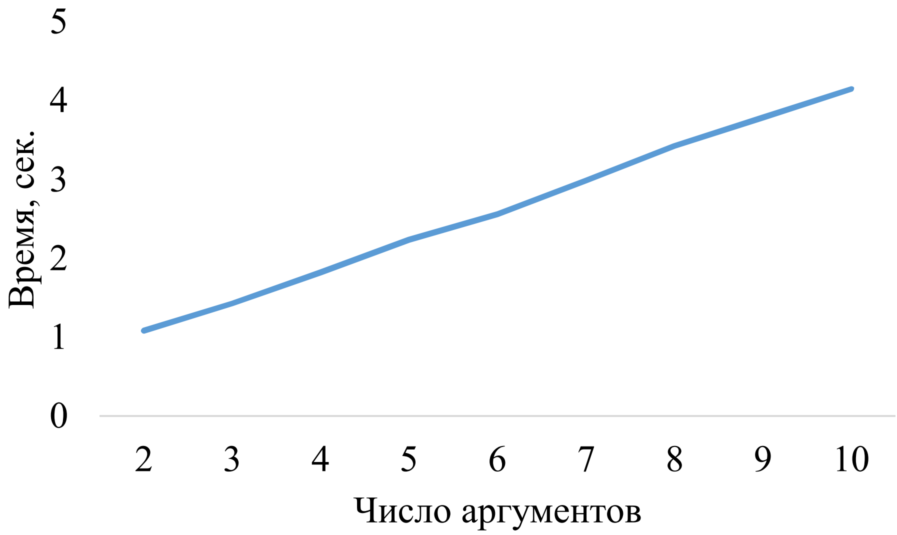


Рисунок 2.12 – График изменения времени решения задачи в зависимости от числа аргументов

На рисунках 2.13–2.14 представлены графики изменения аргументов для аддитивной и нелинейной мультипликативной модели (шаг Δy^* равен 0,0001). Согласно полученной последовательности изменений можно сделать вывод о линейной скорости сходимости.

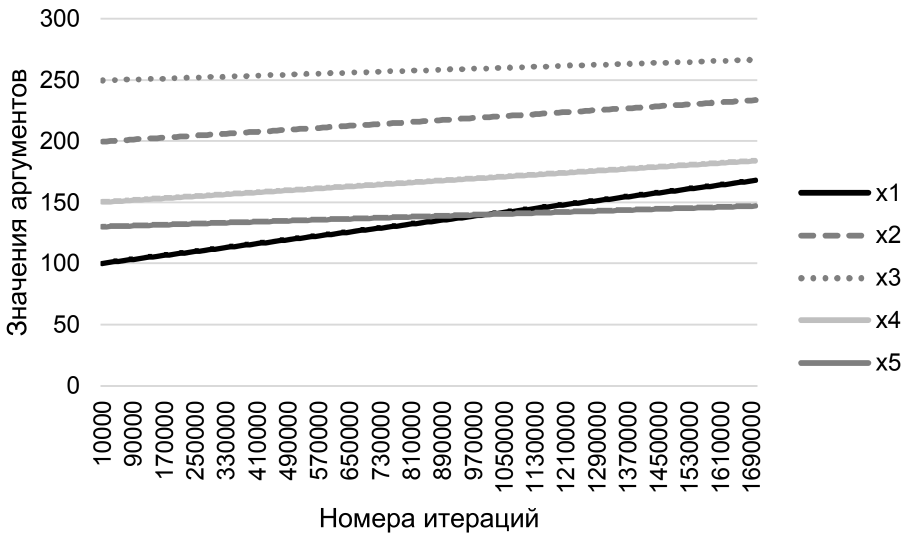


Рисунок 2.13 – График изменения аргументов для аддитивной модели

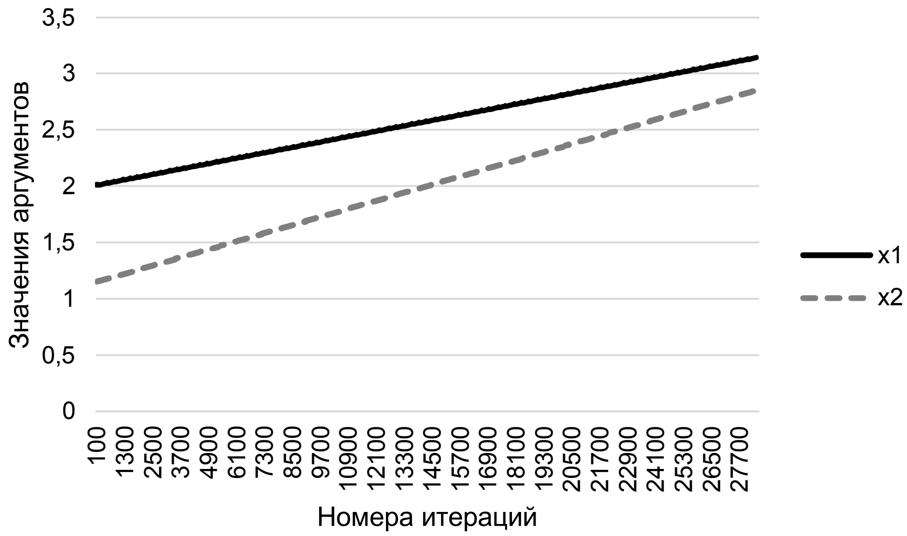


Рисунок 2.14 – График изменения аргументов для нелинейной мультипликативной модели

В некоторых задачах также могут быть наложены ограничения на величины аргументов, в экономических задачах это может быть вызвано ограниченностью ресурсов предприятия. В таблице 2.6 приведены результаты работы алгоритма для пятифакторной аддитивной модели (число запусков равно 50). Для решения задачи с помощью обратных вычислений использована итерационная процедура, представленная в [41]. Полученные величины использованы в качестве заданных значений (x_1, x_2, x_3, x_4, x_5, равных 140, 250, 275, 180, 155 соответственно), с которыми выполнена проверка выборочного среднего с помощью критерия Стьюдента. Поскульку все вычисленные значения статистики (таблица 2.6) меньше критического (равно 2), следовательно, нулевая гипотеза о равенстве выборочного среднего заданному значению не отвергается.

Стоит отметить, что при использовании стохастического алгоритма на основе приращения функции возможно нахождение решения, только для случая, когда сумма абсолютных изменений аргументов минимальна (направления изменения аргументов не задаются в качестве исходных данных). Так, например, в случае двухаргументной функции достижение цели при заданных коэффициентах относительной важности и направления изменения результирующего показателя возможно двумя способами: изменение двух аргументов в одном направлении, изменение аргументов в разных направлениях. Стохастический метод позволяет находить решение многоаргументных задач на основе обратных вычислений при отсутствии необходимости указания направлений изменения аргументов, что приводит к уменьшению времени определения входных данных задачи. Недостаток алгоритма проявляется в отсутствии гибкости, т.к. для некоторых задач может потребоваться поиск решения при наибольшей абсолютной сумме изменения аргументов, а также в непостоянстве полученного решения из-за применения случайных чисел.

Таблица 2.6 – Результаты решения задачи для пятифакторой аддитивной модели с ограничениями

Показатель	x^*_1	x^*_2	x^*_3	x^*_4	x^*_5
Начальное значение	100	200	250	150	130
Коэффициент важности	0,4	0,2	0,1	0,2	0,1
Нижняя граница	0	200	0	150	130
Верхняя граница	140	300	300	180	200
Среднее значение	140	250,01	275	180	154,99
Выборочная дисперсия	0	0,0021	0,0017	0	0,0017
Значение статистики критерия	–	0,87	0,37	–	–1,3

В таблице 2.7 приведены результаты решения обратных задач (для стохастического алгоритма шаг $\Delta y^*=10^{-7}$) с использованием тестовых функций (1.9)–(1.13) (данные таблиц 1.2 – 1.4). Полученные числовые значения соответствуют решению задач с помощью классического аппарата обратных вычислений (итерационный алгоритм). Также в таблице приведены величины эвклидовой номеры отклонения коэффициентов от заданных значений ($e(x)$) и абсолютное отклонение результирующего показателя от заданной величины ($d(x)$).

Таблица 2.7 – Решение задач по формированию показателей

Модель	Показатель	Среднее значение	Дисперсия	Обратные вычисления	$\overline{e}(x)$	$\overline{d}(x)$
Мультипликативная	Δx_1	0,902	$3{,}7\cdot10^{-7}$	0,902	$5\cdot10^{-4}$	10^{-4}
	Δx_2	0,676	$5{,}1\cdot10^{-7}$	0,676		
	Δx_3	0,451	$3{,}8\cdot10^{-7}$	0,451		
	Δx_4	0,225	$1{,}8\cdot10^{-7}$	0,225		
Нелинейная аддитивная	Δx_1	53,731	$2\cdot10^{-5}$	53,743	$2\cdot10^{-4}$	$4\cdot10^{-8}$
	Δx_2	89,584	$1{,}5\cdot10^{-4}$	89,572		
	Δx_3	35,865	$3{,}34\cdot10^{-5}$	35,829		
Мультипликативная нелинейная	Δx_1	1,14	$6{,}85\cdot10^{-6}$	1,14	10^{-3}	10^{-6}
	Δx_2	1,71	$1{,}6\cdot10^{-5}$	1,71		
Мультипликативно-аддитивная	Δx_1	7,691	$5{,}6\cdot10^{-6}$	7,691	$2\cdot10^{-4}$	$5\cdot10^{-5}$
	Δx_2	5,768	$1{,}5\cdot10^{-6}$	5,768		
	Δx_3	1,923	$2\cdot10^{-6}$	1,923		
	Δx_4	3,845	$6{,}7\cdot10^{-6}$	3,846		

На рисунке 2.15 представлены вычисленные значения t-статистики для проверки гипотезы о равенстве среднего значения величине, полученной с помощью обратных вычислений (использовано 50 случайных реализаций). За исключением нелинейной аддитивной модели (где значения статистистики для x_1, x_2, x_3 равны соответственно 19,7, 6,66, 39,83) для всех моделей при $\Delta y^*=10^{-7}$ значение статистики меньше критического.

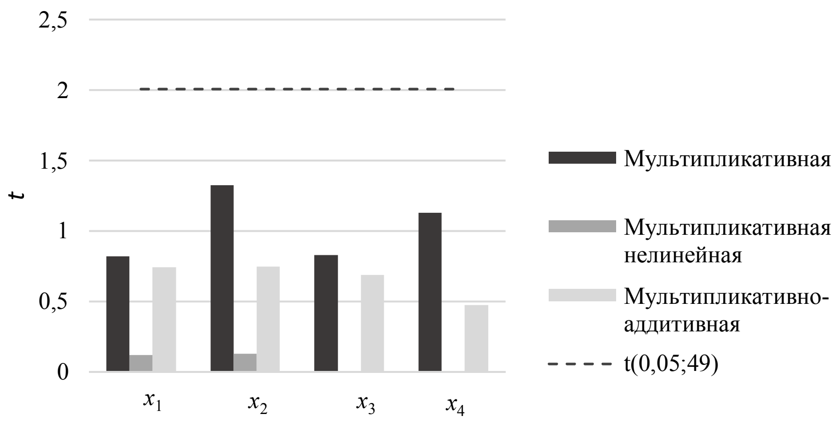


Рисунок 2.15 – Значения t-статистики

2.3 Выводы по главе 2

1) На основе классического аппарата обратных вычислений разработан метод решения обратных задач на основе построения уравнения зависимости между аргументами функции. Преимуществом метода является отсутствие необходимости проверки согласованности дополнительной информации, поступающей от специалиста: соответствия поставленной цели коэффициентам относительной важности и направлениям их изменения. Ограничением метода является отсутствие возможности определения решения, близкого к искомому, в случае недостижимости цели.

2) Для решения обратных задач с ограничениями разработан стохастический метод, основанный на изменении результирующей величины на малое число и выбора аргумента для достижения результата с помощью моделирования полной группы несовместных событий. Его преимуществом является простота компьютерной реализации, в случае его использования отсутствует необходимость решения системы уравнений, применения процедуры свертки и многократного решения задачи с помощью обратных вычислений при многоаргументной функции. Кроме того, алгоритм на основе приращений не требует указания направления изменения аргументов (т.е. требует меньшего объема экспертной информации), решение определяется для варианта, при котором достижение результата достигается при меньшем суммарном абсолютном изменении аргументов. Ограничением алгоритма является невозможность определения решения с заданным направлением изменения коэффициентов относительной важности и изменяемость решения от реализации к реализации из-за использования случайных величин.

3) Проведенные вычислительные эксперименты с использованием тестовых функций показали соответствие результатов заданным значениям результирующего показателя и коэффициентам относительной важности.

4) Выявленные недостатки методов определяют актуальность разработки нового подхода к решению обратных задач на основе использования оптимизационных моделей.

Глава 3. Модели и алгоритмы решения обратной задачи на основе расстояния от исходных значений аргументов

3.1 Оптимизационные модели решения обратных задач на основе минимизации отклонений от исходных значений

Привязка к мнению эксперта имеет свои положительные стороны: может быть рассмотрено несколько возможных вариантов решения задачи, коэффициенты могут быть установлены с учётом реальной возможности направления изменения аргументов и их взаимозависимости. Полученное решение впоследствии может быть скорректировано с учетом дополнительных условий. Однако полученное с использованием коэффициентов относительной приоритетности аргументов решение является субъективным и основано на мнении специалиста, поиск которого затруднён и связан с дополнительными затратами финансовых и временных ресурсов. Это объясняется в том числе трудоемкостью формирования экспертной информации и необходимостью владением специалистом на высоком уровне аппаратом обратных вычислений. Кроме того, постановка задачи может предполагать получение решения без привлечения экспертной информации. К задачам, решение которых может быть найдено без привлечения экспертной информации, в частности, можно отнести задачи поиска решения при минимальном изменении аргументов [126]. Здесь исходные значения характеризуют текущее состояние объекта, следовательно, меньшее их изменение требует меньше усилий для достижения поставленной цели.

В качестве меры удаленности полученного решения от исходного могут быть рассмотрены классические метрики: эвклидова метрика, манхэттенское расстояние.

Пусть x_i – i-й показатель деятельности экономического объекта, y – результирующий показатель деятельности объекта, $f(x_i)$ – функция зависимости между показателями x_i и результирующим показателем y $(y = f(x))$. Задача заключается в определении изменений исходных характеристик Δx_i для достижения заданного значения результирующего показателя $y + \Delta y$.

При минимизации суммы квадратов изменений аргументов задача по определению изменений аргументов может быть представлена в виде (1.5) [127, 128]

$$\begin{aligned} g(\Delta x) &= \sum_{i=1}^{n} \Delta x_i^2 \to \min, \\ f(\Delta x) &= y + \Delta y. \end{aligned} \tag{3.1}$$

Представление задачи в таком виде может быть определено необходимостью достижения заданного значения результирующего показателя таким образом, чтобы изменения входных параметров были как можно ближе к нулю. Данный способ решения отражает стремление минимизировать корректировку входных управляемых показателей, а, следовательно, и снизить затраты ресурсов на мероприятия, сопряженные с изменением показателей по сравнению с их текущим состоянием.

В случае минимизации суммы модулей изменений аргументов задача имеет вид (1.6)

$$\begin{aligned} g(\Delta x) &= \sum_{i=1}^{n} \left|\Delta x_i\right| \to \min, \\ f(\Delta x) &= y + \Delta y. \end{aligned} \tag{3.2}$$

В результате решения такой задачи значения некоторых изменений аргументов получаются равными нулю, таким образом, может быть осуществлен отбор наилучших для изменения признаков.

Такми образом, оптимизационные модели решения обратных задач позволяют осуществлять формирование значения ключевого показателя при минимальном отклонении значений входных переменных от исходных величин.

3.2 Методы решения обратной задачи на основе минимизации суммы квадратов изменений аргументов

Решение задачи (3.1) может быть выполнено с применением методов нелинейной оптимизации, рассмотренных в п.1.2.2.

Выявленные недостатки существующих методов свидетельствуют о целесообразности проведения исследования, посвященного разработке эффективного алгоритма решения представленной оптимизационной задачи, лишенного перечисленных недостатков, связанных с формированием модифицированной функции, требованиями к виду ограничения. Для его разработки рассмотрено использование аппарата обратных вычислений.

Модификация классической схемы решения при этом выражается в изменении величины соотношения значений приращений аргументов, которая будет определяться теперь угловым коэффициентом кривой уровня, определяемого новым значением функции. Далее будет рассмотрено решение задачи в случае аддитивной, кратной и мультипликативной зависимости [127].

3.2.1 Решение задачи с аддитивной зависимостью

Рассмотрим задачу с *аддитивной* зависимостью

$$y = x_1 + x_2 .$$

Пусть начальные условия задачи: $y = 5$, $x_1 = 2$, $x_2 = 3$. Необходимо определить такие значения x_1^*, x_2^*, при которых y^* равно 10. Построим две линии уровня 5 и 10 соответственно (рисунок 3.1), представляющие собой параллельные прямые. Точка *A* соответствует начальному условию задачи. Точки линии $x_2 = 10 - x_1$ обеспечат значение результирующей величины, равное 10. Точки *B* и *C* соответствуют случаям, когда искомое значение функции будет получено только за счет изменения x_2 и x_1 соответственно. Точки линии $x_2 = 10 - x_1$, принадлежащие отрезку *BC*, будут получены при увеличении аргументов функции (x_1^+, x_2^+), расположенные левее точки *B* – уменьшении аргумента x_1 и увеличения x_2 (x_1^-, x_2^+), а точки правее *C* – увеличении x_1 и уменьшения x_2 (x_1^+, x_2^-).

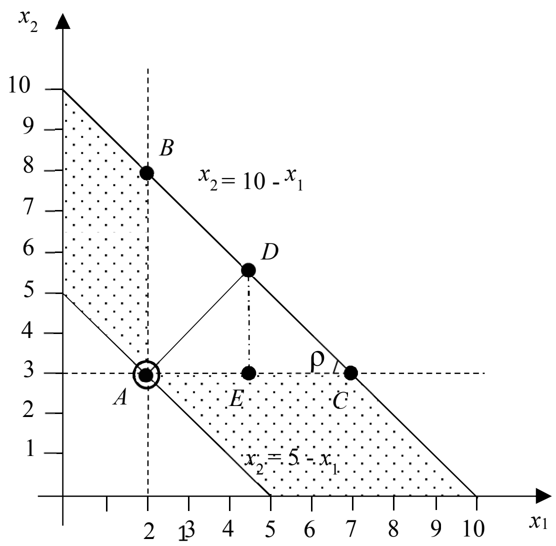


Рисунок 3.1 – Линии уровня 5 и 10

Кратчайшее расстояние из точки *A* до прямой $x_2 = 10 - x_1$ представляет собой длину перпендикуляра *AD*. Таким образом, при переходе из точки *A* в точку *D* изменение аргументов будет наименьшим. Изменение первого аргумента (Δx_1) равно длине отрезка *AE*, изменение второго аргумента (Δx_2) – отрезка *DE* ($AD^2 = AE^2 + DE^2$).

Задача оптимизации при этом может быть представлена в виде задачи квадратичного программирования

$$g(\Delta x_1, \Delta x_2) = \Delta x_1^2 + \Delta x_2^2 \to \min,$$
$$2 + \Delta x_1 + 3 + \Delta x_2 = 10.$$

В результате решения задачи будут получены следующие значения приращений: Δx_1 =2,5, Δx_2 =2,5. Таким образом, новые величины аргументов

$$x_1^* = 2 + 2{,}5 = 4{,}5;$$

$$x_2^* = 3 + 2{,}5 = 5{,}5.$$

Из рисунка 3.1 можно увидеть, что угол *ADE* равен углу *ECD*. Поскольку тангенс угла равен угловому коэффициенту, то

$$\frac{AE}{DE} = -\rho \ ,$$

где ρ – коэффициент угла наклона прямой.

Для рассматриваемого примера $\rho = -1$. Следовательно, для определения значений приращений с наименьшим изменением необходимо решить систему

$$\begin{cases} \dfrac{\Delta x_1}{\Delta x_2} = 1; \\ 2 + \Delta x_1 + 3 + \Delta x_2 = 10. \end{cases}$$

Подставляя во второе уравнение $\Delta x_1 = \Delta x_2$, получим

$$2 + \Delta x_2 + 3 + \Delta x_2 = 10,$$
$$\Delta x_2 = 2{,}5,$$
$$\Delta x_1 = 2{,}5.$$

Рассмотрим случай при многоаргументной функции. Пусть уравнение зависимости имеет вид

$$y = c_0 + c_1 x_1 + c_2 x_2 + \ldots c_n x_n ,$$

где c – некоторые числовые значения.

Тогда определение аргументов с наименьшим приращением будет выполнено путем решения системы

$$\begin{cases} \dfrac{\Delta x_i}{\Delta x_k} = \dfrac{c_i}{c_k}, i = 1..n, i \neq k \\ c_1(x_1 + \Delta x_1) + c_2(x_2 + \Delta x_2) + \ldots c_n(x_n + \Delta x_n) = y^*, \end{cases}$$

где k – номер аргумента, который принимается за базовый.

Выразив приращения аргументов, получим формулы расчета

$$\Delta x_i = \frac{-\left(c_0 + c_1 \cdot x_1 + c_2 \cdot x_2 + \ldots + c_n \cdot x_n - y^*\right)}{\sum\limits_{t=1}^{n} c_t^2} \cdot c_i \ .$$

Тогда новые значения входных переменных вычисляются по формуле

$$x_i^* = x_i + \Delta x_i .$$

3.2.2 Решение задачи с кратной зависимостью

При *кратной* зависимости модель имеет следующий вид

$$y = \frac{x_2}{x_1} .$$

Пусть начальные значения равны: y =2, x_1 =5, x_2 =10. Необходимо определить значения x_1^*, x_2^*, при которых значение функции равно 4.

Задача квадратичного программирования имеет вид

$$g(\Delta x_1, \Delta x_2) = \Delta x_1^2 + \Delta x_2^2 \to \min$$
$$\frac{(10 + \Delta x_2)}{(5 + \Delta x_1)} = 4$$

Решение задачи: Δx_1 =–2,353, Δx_2 =0,588, x_1^* =2,647, x_2^* =10,588.

На рисунке 3.2 представлены линии уровня 2 и 4. Начальные значения аргументов образуют точку *A*. Наименьшее расстояние из этой точки до прямой $x_2 = 4x_1$ – это длина перпендикуляра *AB*. Высота *BC* образует два подобных треугольника. Следовательно, справедливо соотношение

$$\frac{AC}{BC} = -4 .$$

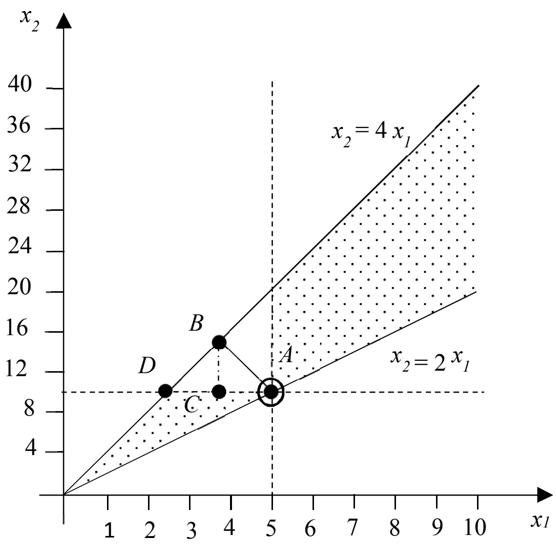


Рисунок 3.2 – Линии уровня 2 и 4

Тогда система будет иметь вид

$$\begin{cases} \dfrac{\Delta x_1}{\Delta x_2} = -4; \\ \dfrac{10 + \Delta x_2}{5 + \Delta x_1} = 4. \end{cases}$$

Решение системы: $\Delta x_1 = -2{,}353$, $\Delta x_2 = 0{,}588$.

Таким образом, для решения обратной задачи при кратной зависимости необходимо решить следующую систему

$$\begin{cases} \dfrac{\Delta x_1}{\Delta x_2} = -y^*; \\ \dfrac{x_1 + \Delta x_2}{x_2 + \Delta x_1} = y^*. \end{cases}$$

Случай многоаргументой функции будет рассмотрен далее при описании обобщенного алгоритма.

3.2.3 Решение задачи с мультипликативной зависимостью

Наконец, рассмотрим *мультипликативную* зависимость

$$y = x_1 \cdot x_2 .$$

Примем начальные условия задачи: y =10, x_1 =5, x_2 =2. Необходимо определить значения x_1^*, x_2^*, при которых y^* равно 20.

Задача квадратичного программирования будет иметь следующий вид:

$$g(\Delta x_1, \Delta x_2) = \Delta x_1^2 + \Delta x_2^2 \to \min,$$
$$(5 + \Delta x_1)(2 + \Delta x_2) = 20.$$

Решение задачи: Δx_1 =0,837, Δx_2 =1,426, x_1^* =5,837, x_2^* =3,426.

Рассмотрим линии уровня 10 и 20 (рисунок 3.3). Точка А соответствует начальным значениям аргументов. Определение точки графика $x_2 = 20/x_1$ таким образом, чтобы изменения аргументов были минимальны может быть выполнено с использованием уравнения касательной

$$k = f'(x_0)(x - x_0) + f(x_0),$$

где x_0 – точка функции $f(x)$, к которой строится касательная.

Уравнение касательной к точке В будет иметь вид

$$x_2 = \frac{-20}{5^2}(x_1 - 5) + \frac{20}{5} = -0,8x_1 + 8 .$$

Поиск наименьших изменений аргументов для перехода в точку на кривой может быть выполнен с помощью решения системы уравнения

$$\begin{cases} \dfrac{\Delta x_1}{\Delta x_2} = \dfrac{y^*}{\left(x_1\right)^2} = 0,8; \\ \left(5 + \Delta x_1\right) \cdot \left(2 + \Delta x_2\right) = 20. \end{cases}$$

Отношение приращений аргументов будет равно угловому коэффициенту в уравнении касательной (со знаком «минус»). Решение системы: Δx_1 =1,046, Δx_2 =1,308, x_1^* =6,046, x_2^* =3,308.

Далее необходимо построить уравнение касательной к новой точке (x_1^* =6,046, x_2^* =3,308) и выполнить поиск новой точки с минимальным изменением аргументов.

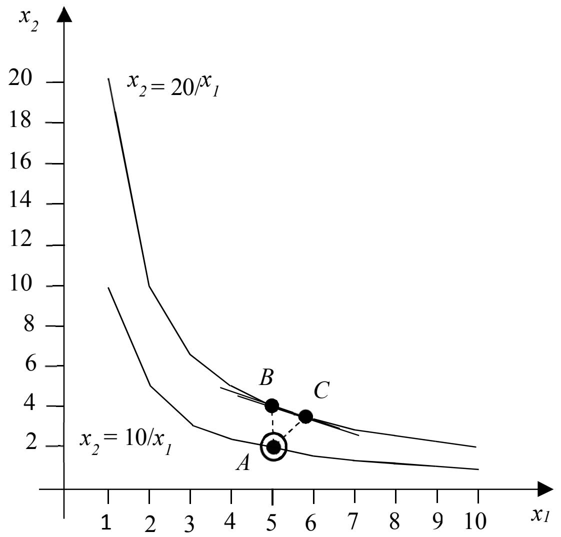


Рисунок 3.3 – Кривые уровня 10 и 20

Таким образом, алгоритм нахождения решения в случае мультипликативной модели включает следующие шаги:

1) Установка начальных значений: номер итерации s=0, $x_1(s) = x_1$, $x_2(s) = x_2$.

2) Построение уравнения касательной к точке $x_1(s)$ функции $x_2 = \dfrac{y^*}{x_1}$.

3) Поиск точки $x_1(s+1)$, $x_2(s+1)$ на кривой, до которой расстояние от исходной точки x_1 , x_2 будет минимальным. Для этого происходит решение следующей системы уравнений

$$\begin{cases} \dfrac{\Delta x_1(s+1)}{\Delta x_2(s+1)} = \dfrac{y^*}{\left(x_1(s)\right)^2}; \\ \left(x_1 + \Delta x_1(s+1)\right)\left(x_2 + \Delta x_2(s+1)\right) = y^*. \end{cases}$$

Далее определяются новые координаты точки

$$x_1(s+1) = x_1 + \Delta x_1(s+1),$$
$$x_2(s+1) = x_2 + \Delta x_2(s+1).$$

4) Проверка выполнения условия останова: если изменение положения точки меньше заданной точности ε ($\delta = \sqrt{\left(x_1(s+1) - x_1(s)\right)^2 + \left(x_2(s+1) - x_2(s)\right)^2} < \varepsilon$), то алгоритм завершается, иначе осуществляется увеличение номера итерации ($s = s + 1$) и переход на шаг 2.

В таблице 3.1 представлены результаты выполнения итераций для рассматриваемого примера (ε =0,002).

Таблица 3.1 – Результаты выполнения итераций

s	$x_1(s)$	$x_2(s)$	$-f'(x_1(s))$	δ
0	5	2	0,8	–
1	6,046	3,308	0,547	1,675
2	5,794	3,452	0,596	0,29
3	5,846	3,421	0,585	0,061
4	5,835	3,427	0,587	0,013
5	5,838	3,426	0,587	0,003
6	5,837	3,426	–	0,001

Таким образом, приведено использование метода для решения двухаргументных обратных задач (за исключение аддитивной зависимости).

Рассмотрим теперь общий алгоритм, который может быть использован для решения многоаргументных задач.

3.2.4 Обобщенные алгоритмы решения обратной задачи на основе минимизации квадратов изменений аргументов

Общий алгоритм для решения обратных задач основан на выражении одного из аргументов через оставшиеся аргументы и определении частных производных полученной функции. Значения частных производных в полученных на каждой из итераций точках используются в качестве отношений приращений аргументов. Графически такая операция заключается в определении кратчайшего расстояния до кривой заданного уровня.

Алгоритм включает следующие шаги (исходное значение переменных $\hat{x}=x$, номер итерации s = 1):

Шаг 1. Выражение k-го аргумента x_k через остальные аргументы

$$x_k(x) = \varphi_k(y^*; x_1, ..., x_{k-1}, x_{k+1}, ... x_n).$$

Шаг 2. Вычисление частных производных по каждому из аргументов функции φ_k.

Шаг 3. Решение системы уравнений

$$\begin{cases} \dfrac{\Delta x_i}{\Delta x_k} = -\dfrac{\partial \varphi_k(\widehat{x})}{\partial x_i}, \ i = 1, ..., n; \ i \neq k \\ f(x_1 + \Delta x_1, x_2 + \Delta x_2, ..., x_n + \Delta x_n) = y^*. \end{cases} \tag{3.3}$$

Шаг 4. Проверка окончания работы алгоритма: если $s \geq 2$ и абсолютное изменение нормы приращений аргументов меньше заданной точности ε

$$\delta = \sqrt{\sum_{i=1}^{n} \left(\Delta x_{is} - \Delta x_{is-1} \right)^2} \leq \varepsilon ,$$

либо квадрат эвклидовой метрики (значение целевой функции) увеличился по сравнению с предыдущим шагом

$$\sum_{i=1}^{n} \Delta x_{is}{}^{2} > \sum_{i=1}^{n} \Delta x_{is-1}{}^{2},$$

то алгоритм завершается. Иначе $\hat{x}_i = x_i + \Delta x_i$, $s = s + 1$, переход на шаг 1.

Следует отметить, что для разных значений k $(k = 1,...n)$ будут получены разные системы вида (3.3), а значит и разные решения, т.е. всего получим n решений $\Delta x^j = (\Delta x_1^j, \Delta x_2^j, ..., \Delta x_n^j)$, $j = 1,...,n$. В качестве оптимального можно принять решение с минимальной нормой $\Delta x_{opt} = \min\limits_{1 \le j \le n} \|\Delta x^j\|$.

Однако представленный алгоритм может быть использован для ограниченного круга задач по причине того, что не всегда можно выразить один аргумент через оставшиеся аргументы. В связи с этим для решения более сложных задач был разработан градиентный метод. Градиент представляет собой вектор частных производных, который показывает направление наибольшего возрастания функции. Соответственно антиградиент показывает направление наибольшего убывания функции. Основная идея заключается в изменении аргументов функции в соответствии со значениями элементов вектора градиента функции ограничения до достижения заданного значения. Так, на рисунке 3.4 точка *A* соответствует исходным значениям прибыли (равна 2 усл.ден.ед.) и затрат (равны 15 усл.ден.ед.). На рисунке 3.4 также представлена линия заданного уровня рентабельности (0,2). Стрелкой обозначен градиент, путем осуществления движения в этом направлении до пересечения с линией заданного уровня рентабельности, получена точка *B*, которая является решением задачи.

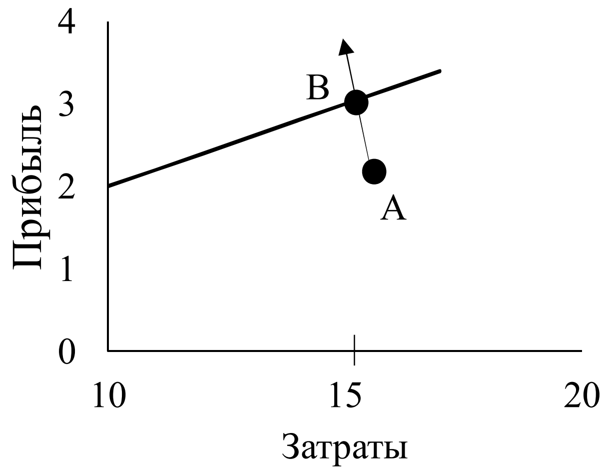


Рисунок 3.4 – Решение задачи формирования рентабельности путем движения вдоль градиента

Данный метод в отличие от метода на основе построения кривой заданного уровня не требует формирования функции зависимости между аргументами и, следовательно, может быть использован в тех случаях, когда невозможно выразить аналитически один аргумент через оставшиеся аргументы.

Решение обратной задачи при использовании вектора-градиента включает следующие шаги:

Шаг 1. Определение направления изменения параметров. Если результирующий показатель необходимо увеличить ($y^* > y$), то используются элементы вектора-градиента ($t = 1$), если уменьшить ($y^* < y$) – антиградиента ($t = -1$).

Шаг 2. Определить необходимые приращения аргументов Δx_i для достижения заданного значения функции y^* путем решения системы уравнений

$$\begin{cases} \dfrac{\Delta x_\eta}{\Delta x_i} = \dfrac{t \cdot \dfrac{\partial f(x)}{\partial x_\eta}}{t \cdot \dfrac{\partial f(x)}{\partial x_i}}; i = 1..n, i \neq \eta \\ f(x + \Delta x) = y^*. \end{cases}$$

Шаг 3. Определить новые значения аргументов функции

$$x_i^* = x_i + \Delta x_i.$$

Однако в случае, если направление вектора градиента изменяется в ходе движения по направлению к кривой заданного уровня, полученное решение может оказаться неудовлетворительным.

По сравнению с решением оптимизационных задач нелинейного программирования представленный алгоритм является более простым в компьютерной реализации: решение задачи сводится к решению системы уравнений.

3.2.5 Применение алгоритма минимизации суммы квадратов изменений аргументов

Рассмотрим применение алгоритмов для решения задач с использованием тестового набора моделей (таблица 1.2).

Задача оптимизации в случае мультипликативно-аддитивной модели будет иметь вид

$$\begin{aligned} &g(\Delta x_1, \Delta x_2, \Delta x_3, \Delta x_4) = \Delta x_1^2 + \Delta x_2^2 + \Delta x_3^2 + \Delta x_4^2 \to \min, \\ &f(\Delta x_1, \Delta x_2, \Delta x_3, \Delta x_4) = (x_1 + \Delta x_1) \cdot \left(x_2 + \Delta x_2\right) - (x_3 + \Delta x_3) - (x_4 + \Delta x_4) = 4500. \end{aligned} \quad (3.4)$$

При использовании градиентного метода получим следующую систему уравнений (отношения изменений аргументов приравниваются к отношению частных производных: $\frac{\partial f}{\partial x_1} = x_2$, $\frac{\partial f}{\partial x_2} = x_1$, $\frac{\partial f}{\partial x_3} = -1$, $\frac{\partial f}{\partial x_4} = -1$)

$$\begin{cases} \dfrac{\Delta x_1}{\Delta x_2} = \dfrac{x_2}{x_1}; \\ \dfrac{\Delta x_1}{\Delta x_3} = \dfrac{x_2}{-1}; \\ \dfrac{\Delta x_1}{\Delta x_4} = \dfrac{x_2}{-1}; \\ (x_1 + \Delta x_1) \cdot (x_2 + \Delta x_2) - (x_3 + \Delta x_3) - (x_4 + \Delta x_4) = y^*. \end{cases}$$

Для использования алгоритма на основе построения кривой заданного уровня выразим x_3:

$$x_3 = x_1 \cdot x_2 - x_4 - y^*. \tag{3.5}$$

Тогда система уравнений будет иметь следующий вид (частные производные по аргументам функции (3.5) будут равны: $\dfrac{\partial x_3}{\partial x_1} = x_2$, $\dfrac{\partial x_3}{\partial x_2} = x_1$, $\dfrac{\partial x_3}{\partial x_4} = -1$)

$$\begin{cases} \dfrac{\Delta x_1}{\Delta x_3} = -x_2; \\ \dfrac{\Delta x_2}{\Delta x_3} = -x_1; \\ \dfrac{\Delta x_4}{\Delta x_3} = 1; \\ (x_1 + \Delta x_1) \cdot (x_2 + \Delta x_2) - (x_3 + \Delta x_3) - (x_4 + \Delta x_4) = y^*. \end{cases}$$

В таблице 3.2 представлены результаты решения задачи с помощью градиентного метода, метода на основе построения кривой заданного уровня (точность $\varepsilon = 10^{-6}$) и математического пакета MathCad.

Таблица 3.2 – Результаты решения задачи при мультипликативно-аддитивной модели

Метод	Δx_1	Δx_2	Δx_3	Δx_4	g	d
Градиентный	3,879	7,758	–0,078	–0,078	75,245	$2 \cdot 10^{-6}$
Построение кривой уровня	4,19188	7,58467	–0,0728	–0,0728	75,10963	$3 \cdot 10^{-4}$
Стандартная функция MathCad	4,19187	7,58467	–0,0728	–0,0728	75,10957	$8{,}9 \cdot 10^{-5}$

В таблице 3.3 представлены результаты итераций при использовании метода на основе формирования кривой заданного уровня. В таблице 3.4 приведены результаты решения задачи с использованием других тестовых моделей.

Таблица 3.3 – Результаты выполнения итераций

r	Δx_1	Δx_2	Δx_3	Δx_4	g	δ
1	3,87901	7,75801	–0,07758	–0,07758	75,24547	–
2	4,21122	7,57398	–0,07291	–0,07291	75,11015	0,38
3	4,19069	7,58532	–0,07279	–0,07279	75,10963	0,023
4	4,19196	7,58462	–0,0728	–0,0728	75,10955	$1{,}4\cdot10^{-3}$
5	4,19188	7,58467	–0,0728	–0,0728	75,10963	$8{,}9\cdot10^{-5}$

Таблица 3.4 –Минимизация суммы квадратов изменений аргументов с использованием тестовых функций

Модель	Показатель	Градиентный метод	Метод с помощью формирования кривой, $\varepsilon=10^{-3}$	Функция Mathcad
Аддитивная	Δx_1	34	34	34
	Δx_2	34	34	34
	Δx_3	34	34	34
	Δx_4	34	34	34
	Δx_5	34	34	34
	g	5780	5780	5780
	d	0	0	0
Мультипликативная	Δx_1	0,51072	0,5151	0,51512
	Δx_2	0,19643	0,20668	0,20672
	Δx_3	0,51072	0,5151	0,51512
	Δx_4	0,6384	0,62783	0,62778
	g	0,9678	0,9675334	0,967533
	d	$8{,}9\cdot10^{-6}$	$3{,}6\cdot10^{-12}$	$9{,}4\cdot10^{-4}$
Нелинейная аддитивная	Δx_1	–0,165	–	–0,1
	Δx_2	–0,48	–	–0,319
	Δx_3	–0,968	–	–1,028
	g	1,195	–	1,168
	d	$2{,}9\cdot10^{-8}$	–	$2{,}7\cdot10^{-7}$
Мультипликативная нелинейная	Δx_1	1,354	1,475	1,472
	Δx_2	1,412	1,264	1,268
	g	3,827	3,77579	3,77574
	d	$5\cdot10^{-7}$	$3{,}5\cdot10^{-15}$	$2{,}3\cdot10^{-6}$

Представленные в таблицах 3.2, 3.4 результаты свидетельствуют о соответствии решений, полученных с помощью трех методов. С помощью градиентного метода решение было получено за одну итерацию, однако при этом значение целевой функции приняло большее значение.

Точность решения в случае использования нелинейных моделей и градиентного метода ниже по сравнению с методом на основе формирования уравнения зависимости. Так в случае нелинейной аддитивной модели и нелинейной мультипликативной точность составила 2,3% и 1,4% соответственно. Для повышения точности решения была выполнена разработка итерационых алгоритмов.

3.2.6 Итерационные алгоритмы минимизации суммы квадратов аргументов

Анализируя подход к решению обратных задач на основе минимизации суммы квадратов изменений аргументов (п. 3.2.4), можно отметить, что при линейной функции ограничения могут быть получены аналитические формулы, которые для двух представленных алгоритмов будут идентичны. При этом обеспечивается высокое соответствие полученного с помощью данных методов решения решению задачи с помощью математических пакетов. Однако при нелинейных ограничениях были выявлены следующие недостатки методов [129]:

1) для некоторых видов функций была значительная разница между полученным решением и оптимальным (например, в задаче формирования маржинальной прибыли разница составила более 170% [129]), либо решение не было найдено (например, направление вектора градиента в исходной точке не позволяет достичь заданного значения функции ограничения).

2) требуется многократное решение системы уравнений, и, соответственно, реализации соответствующих численных методов (например, использование

метода Ньютона), что усложняет процесс поиска решения, а также приводит к увеличению времени решения задачи.

Таким образом, выявленные недостатки обусловили разработку итерационных алгоритмов, отличающихся более простой компьютерной реализацией и более высокой точностью решения задачи, что позволило расширить круг решаемых задач.

Исходные данные алгоритмов: начальные значения аргументов x, заданное значение результирующего показателя $y + \Delta y$, α – некоторое малое положительное число, обеспечивающее движение по направлению к заданному значению ограничения $y + \Delta y$.

Итерационный поиск на основе градиентного метода может быть представлен в следующем виде:

Шаг 1. Используя исходные данные, вычислить величину функции ограничения $f(x)$ и сравнить с заданным значением $y + \Delta y$:

Если $f(x) < y + \Delta y$, то изменение аргументов необходимо выполнить в направлении увеличения величины функции ограничения (направлении вектора-градиента): t=1.

Если $f(x) > y + \Delta y$, то изменение аргументов необходимо выполнить в направлении уменьшения величины функции ограничения (направлении вектора-антиградиента): $t = -1$.

Шаг 2. Определить абсолютную разницу между величиной функции ограничения и заданным значением $y + \Delta y$

$$d_0 = \left| f\left(x\right) - y - \Delta y \right|.$$

Шаг 3. Определить новые значения аргументов путем движения в сторону градиента/антиградиента

$$x_i^* = x_i + t \cdot \alpha \cdot \frac{\partial f\left(x_i\right)}{\partial x_i}, \tag{3.6}$$

где i=1..n, n – число аргументов.

Шаг 4. Вычислить значение функции ограничения $f(x_i^*)$ и отклонение d_1 от заданного значения $y + \Delta y$.

Проверка: если $d_1 > d_0$ или d_1 меньше заданной точности, то работа алгоритма завершается. Иначе $d_0 = d_1$, $x = x^*$, переход на шаг 3.

Решением задачи будут значения x.

Алгоритм на основе формирования кривой заданного уровня включает следующие шаги (k – номер выражаемой переменной, ε – заданная точность, s – номер реализации):

Шаг 1. Установить начальные значения переменных: s=0, $\hat{x}$=x.

Из функции ограничения $f(x)$ выразить k-ую переменную

$$x_k = \varphi(x_l), l \neq k.$$

Используя исходные данные, вычислить величину ограничения $f(x)$ и сравнить с заданным значением $y + \Delta y$:

Если $f(x) < y + \Delta y$, то изменение аргументов необходимо выполнить в направлении увеличения величины функции ограничения (направлении вектора-градиента): t=1.

Если $f(x) > y + \Delta y$, то изменение аргументов необходимо выполнить в направлении уменьшения величины функции ограничения (направлении вектора-антиградиента): $t = -1$.

Шаг 2. Определить абсолютную разницу между величиной функции ограничения и заданным значением $y + \Delta y$

$$d_0 = \left| f(x) - y - \Delta y \right|.$$

Шаг 3. Определить значение частных производных функции q

$$r_i = \frac{-\partial \varphi(x_i)}{\partial x_i}.$$

Шаг 4. Определить новые значения аргументов

$$\begin{aligned} x_k^* &= x_k + t \cdot \alpha, \\ x_i^* &= x_i + t \cdot \alpha \cdot r_i, i \neq k. \end{aligned} \qquad (3.7)$$

Шаг 5. Вычислить значение функции ограничения $f(x_i^*)$ и отклонение d_1 от заданного значения $y+\Delta y$.

Проверка: если $d_1 > d_0$ или d_1 меньше заданной точности, то переход на шаг 6. Иначе $d_0 = d_1$, $x = x^*$, переход на шаг 4.

Шаг 6. Вычисление значения целевой функции: $s = s + 1$, $g_s = g(x)$.

Если $s>1$, то выполняется проверка окончания работы алгоритма:

если $|g_s - g_{s-1}|<\varepsilon$, то работа алгоритма завершается.

Шаг 7. Вычисление новых значений частных производных: $r_i = \dfrac{-\partial q\left(x_i^*\right)}{\partial x_i}$,

$x = x^*$, переход на шаг 4.

Решением задачи будут величины x.

Графически такой процесс решения может быть представлен как приближение с некоторым шагом к заданному значению функции ограничения. Так, на рисунке 3.5 (процесс решения задачи для нелинейного ограничения) представлена начальная точка Q, координаты которой изменяются в соответствии со значениями частных производных функции ограничения и вторых производных целевой функции [130]. При этом при большом размере шага решение может значительно отличаться от оптимального, при маленьком шаге для достижения заданного значения функции ограничения потребуется большое число итераций.

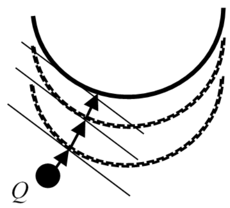


Рисунок 3.5 – Решение задачи путем итерационного приближения к линии заданного уровня

3.2.7 Применение итерационных алгоритмов для решения задач с использованием тестового набора моделей

Рассмотрим применение итерационных алгоритмов для решения задч с использованием тестовых моделей.

Задача оптимизации в случае мультипликативной нелинейной модели (таблица 1.2), соответствующей функции Кобба-Дугласа [131]), будет иметь следующий вид

$$g(\Delta x_1, \Delta x_2) = \Delta x_1{}^2 + \Delta x_2{}^2 \to \min$$
$$a \cdot \left(x_1 + \Delta x_1\right)^{b_1} \cdot \left(x_2 + \Delta x_2\right)^{b_2} = 17.$$

Для алгоритма на основе построения линии заданного уровня первая итерационная формула будет иметь вид ($x_1 = \left(\frac{17}{7x_2{}^{0,3}}\right)^2$, $\frac{\partial x_1}{\partial x_2} = \frac{-3,539}{x_2{}^{1,6}}$)

$$x_1 = x_1 + \alpha,$$
$$x_2 = x_2 + \alpha\frac{3,539}{x_2{}^{1,6}}.$$

Для градиентного алгоритма первая итерационная формула представляется следующим образом ($\frac{\partial f}{\partial x_1} = \frac{3,5x_2{}^{0,3}}{x_1{}^{0,5}}$, $\frac{\partial f}{\partial x_2} = \frac{2,1x_1{}^{0,5}}{x_2{}^{0,7}}$)

$$x_1 = x_1 + \alpha\frac{3,5x_2{}^{0,3}}{x_1{}^{0,5}},$$

$$x_2 = x_2 + \alpha\frac{2,1x_1{}^{0,5}}{x_2{}^{0,7}}.$$

В таблицах 3.5, 3.6 приведены изменения аргументов в процессе решения задачи при α = 0,01 (реализация алгоритма выполнена в Excel с помощью VBA).

Таблица 3.5 – Результаты итераций при градиентном методе

Номер итерации	x_1	x_2	d	$g(x)$
1	2,026	1,177	6,537	0,001
2	2,052	1,204	6,399	0,006
3	2,077	1,231	6,263	0,012
…	…	…	…	…
57	3,449	2,468	0,046	3,835

Таблица 3.6 – Результаты реализаций при использовании метода на основе формирования линии заданного уровня (ε=0,02)

Номер реализации, s	r	x_1	x_2	d	$g(x)$
1	2,83	2,810	3,442	0,002	5,910
2	0,489	3,840	2,051	0,016	4,198
3	1,121	3,310	2,619	0,0003	3,873
4	0,758	3,550	2,326	0,011	3,784
5	0,917	3,430	2,461	0,014	3,765

Согласно полученным результатам, можно сделать вывод, что при использовании метода на основе формировании линии заданного уровня была достигнута меньшая разница с заданным значением ограничения d и меньшее значение целевой функции $g(x)$. Однако число итерационных вычислений было выполнено больше и составило 699. Достижение большего соответствия заданному значению ограничения может быть достигнуто с помощью уменьшения значения параметра α. Так, в таблице 3.7 представлены результаты решения задачи с использованием двух алгоритмов при α=10^{-8} (ε=0,001). В последнем столбце представлена величина u – разница между значением целевой функции $g(x)$, полученной при использовании данного метода, и значением целевой функции при использовании стандартной функции MathCad. Также в таблице 3.7 приведены результаты применения классических методов решения задачи (штрафов и множителей Лагранжа), градиентного метода и метода на основе формирования линии заданного уровня (описание которых приводится в п. 3.2.4). В методе штрафов шаг изменения штрафного параметра равен 10, а точность 10^{-8}.

Наибольшее значение разницы *u* было получено при использовании градиентного метода, а разницы *d* – метода штрафов. Рассматривая параметры *d* и *u* как минимизируемые величины, можно отметить, что эффективными по Парето будут результаты, полученные с помощью метода множителей Лагранжа, штрафов, и стандартной функции MathCad. При этом наилучший среди алгоритмов на основе обратных вычислений результат был получен с использованием итерационного алгоритма на основе формирования линии заданного уровня. Также можно отметить, что с помощью итерационных алгоритмов было найдено значение целевой функции с точностью до третьего знака.

Таблица 3.7 – Решение задачи формирования объема производства ($\alpha = 10^{-8}$)

Метод	x_1	x_2	d	$g(x)$	u
Итерационный градиентный	3,441	2,455	$4{,}1\cdot10^{-7}$	3,779	$3{,}4\cdot10^{-3}$
Итерационный на основе формирования кривой заданного уровня	3,463	2,429	$5{,}6\cdot10^{-8}$	3,776	$2{,}8\cdot10^{-4}$
Градиентный, на основе движения по направлению градиента/антиградиента функции	3,412	2,562	$5\cdot10^{-7}$	3,827	0,051
На основе формирования кривой заданного уровня	3,463	2,429	$2{,}4\cdot10^{-7}$	3,776	$2{,}8\cdot10^{-4}$
Множителей Лагранжа	3,472	2,418	0	3,776	$2{,}8\cdot10^{-6}$
Штрафов	3,472	2,418	$3{,}9\cdot10^{-4}$	3,775	$-4{,}6\cdot10^{-4}$
Использование функции MathCad	3,472	2,418	$2{,}3\cdot10^{-6}$	3,776	–

На рисунках 3.6 и 3.7 представлено изменение аргументов x_1 и x_2 в случае использования итерационного градиентного метода. Исходя из полученных значений можно сделать вывод о линейной скорости сходимости.

Рисунок 3.6 – Изменение аргумента x_1 при итерационном градиентном алгоритме

Рисунок 3.7 – Изменение аргумента x_2 при итерационном градиентном алгоритме

На рисунках 3.8, 3.9 показано изменение приращений аргументов x_1, x_2 в случае использования итерационного алгоритма на основе формирования кривой заданного уровня. На основе конечных значений каждого итерационного цикла по определению кратчайшего расстояния можно сделать вывод, что скорость сходимости является также линейной, однако скорость выше по сравнению с итерационным градиентным алгоритмом (если для итерационного градиентного алгоритма параметр близок к единице, то в случае использования итерационного алгоритма на основе формирования кривой заданного уровня параметр примерно равен 0,6).

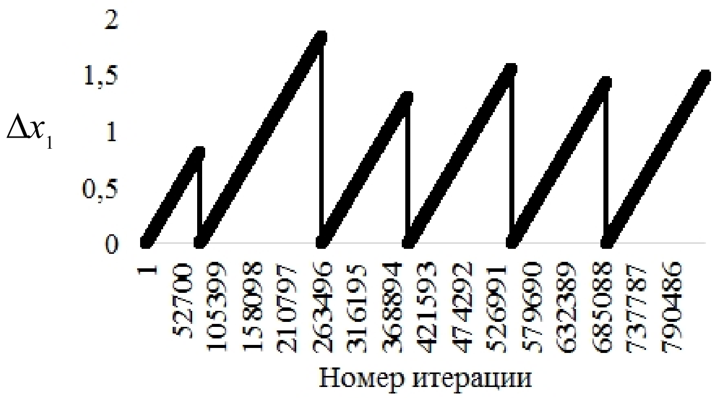


Рисунок 3.8 – Изменение приращения аргумента x_1 при итерационном алгоритме на основе формирования кривой заданного уровня

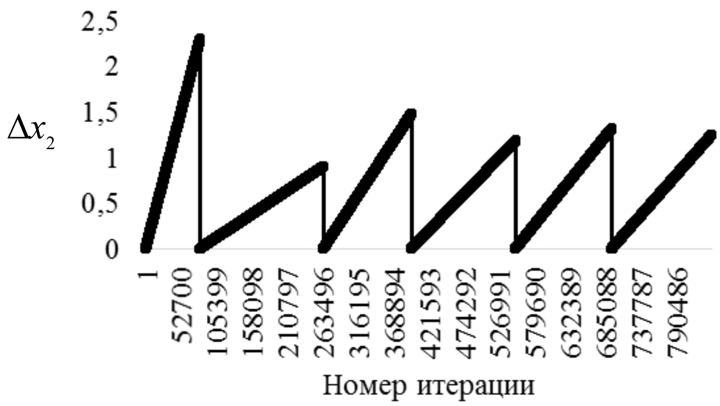


Рисунок 3.9 – Изменение приращения аргумента x_2 при итерационном алгоритме на основе формирования кривой заданного уровня

Для сравнения скорости работы алгоритмов была выполнена реализация на языке VBA метода штрафов (для оптимизации модифицированной функции использован метод градиентного спуска), итерационного градиентного алгоритма, итерационного алгоритма на основе формирования кривой заданного уровня. Была поставлена задача определения решения, при котором абсолютная величина d меньше $4 \cdot 10^{-4}$ при минимальном числе итерации (изменялся параметр α путем деления на 10). Полученное значение величины α для всех алгоритмов получилось равным 10^{-3}. Результаты расчётов представлены в таблице 3.8.

Таблица 3.8 – Сравнение алгоритмов по скорости расчётов (мультипликативная нелинейная модель)

Метод	d	Время решения задачи, сек.	Число арифметических операций
Итерационный градиентный	$3{,}4 \cdot 10^{-4}$	<1	141875
Итерационный на основе формирования кривой заданного уровня	$5{,}7 \cdot 10^{-5}$	1	219495
Штрафов	$3{,}9 \cdot 10^{-4}$	5	93938712

Итерационные алгоритмы имеют временную сложность O($n \times m$), где m – число итераций для решения задачи. На рисунке 3.10 представлено изменение времени решения в зависимости от размера задачи (мультипликативная нелинейная модель). Можно увидеть, что скорость роста времени решения задачи выше для итерационного алгоритма на основе формирования кривой уровня. Однако данный алгоритм обеспечивает меньшее значение абсолютной погрешности (рисунок 3.11).

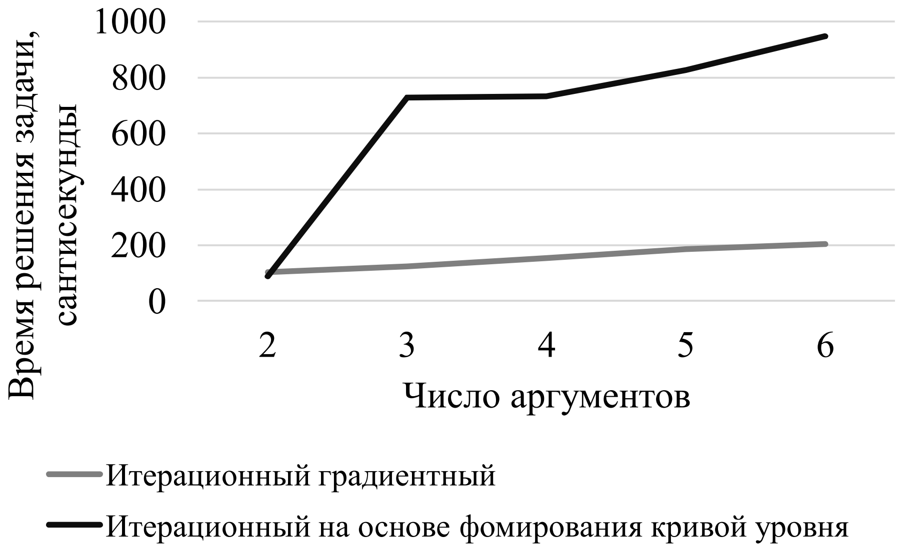


Рисунок 3.10 – График изменения времени решения задачи в зависимости от числа аргументов

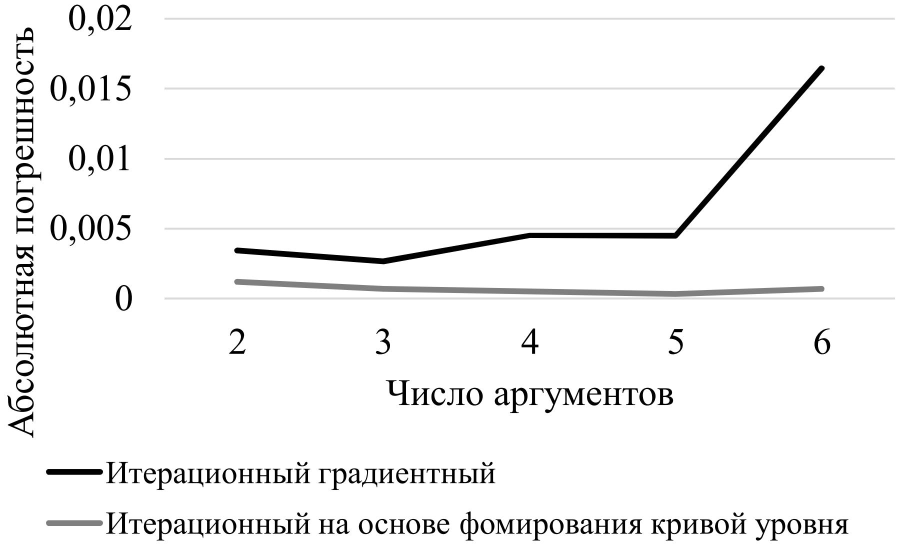


Рисунок 3.11 – График изменения абсолютной погрешности решения задачи в зависимости от числа аргументов

В качестве примера задачи, для которой не может быть применен метод на основе построения кривой заданного уровня, может быть рассмотрено формирование показателя для нелинейной аддитивной модели [132] (таблица 1.2). Результаты решения задачи представлены в таблице 3.9.

Таблица 3.9 – Решение задачи с помощью нелинейной аддитивной модели ($\alpha=10^{-8}$)

Метод	x_1	x_2	x_3	d	$g(x)$
Итерационный градиентный	8,35	7,99	8,23	$1{,}2\cdot 10^{-9}$	28,65
Градиентный	7,85	7,48	9	$9{,}28\cdot 10^{-8}$	31,89
Использование стандартной функции MathCad	8,52	8,1	8,07	$3{,}75\cdot 10^{-6}$	28,51

Полученные результаты также свидетельствуют о том, что использование итерационного алгоритма позволило получить решение с меньшим значением целевой функции по сравнению с градиентным алгоритмом.

Таким образом, на основе проведенных вычислительных экспериментов по формированию экономических показателей с использованием стандартных экономических моделей можно сделать вывод, что итерационный алгоритм является более простым в компьютерной реализации и позволяет получать результаты с высокой степенью точности для нелинейных функциий-ограничениий.

В таблице 3.10 представлены результаты сравнения разработанного алгоритма с методом штрафов по числу арифметических операций и времени решения задачи.

Таблица 3.10 – Сравнение алгоритмов по скорости расчётов (нелинейная аддитивная модель, $d < 10^{-3}$)

Метод	d	Время решения задачи, сек.	Число арифметических операций, млн.
Итерационный градиентный	$9\cdot 10^{-4}$	<1	0,05
Штрафов	10^{-3}	6	219,6

Согласно данным таблицы 3.10 итерационный алгоритм обеспечивает меньшее число арифметических операций (меньше в 4392 раза) и более быстрое время решения задачи.

В таблице 3.11 представлены результаты решения задачи для мультипликативно-аддитивной модели.

Таблица 3.11 – Сравнение алгоритмов по скорости расчётов (мультипликативно-аддитивная модель)

Метод	d	Время решения задачи, сек.	Число арифметических операций
Итерационный градиентный	$2,4\cdot10^{-4}$	1	14 499 005
Итерационный на основе формирования кривой заданного уровня	$1,2\cdot10^{-4}$	<1	2 147 824
Штрафов	$1,4\cdot10^{-4}$	2	101 871 820

Данные таблицы 3.11 свидетельствуют о том, что итерационный алгоритм на основе формирования кривой заданного уровня обеспечивает меньшее число арифметических операций (меньше в 47 раз) и более быстрое время решения задачи.

В таблице 3.12 приведены результаты решения задачи для тестовых моделей.

Таблица 3.12 – Решение задач по формированию показателей на основе минимизации суммы квадратов изменений аргументов

Модель	Показатель	Итерационный градиентный алгоритм	Итерационный алгоритм на основе формирования кривой заданного уровня, ε=10^{-3}	Функция Mathcad
Аддитивная	Δx_1	34	34	34
	Δx_2	34	34	34
	Δx_3	34	34	34
	Δx_4	34	34	34
	Δx_5	34	34	34
	g	5780	5780	5780
	d	0	0	0
Мультипликативная, a=10^{-13}	Δx_1	0,5131	0,51491	0,51512
	Δx_2	0,20162	0,20667	0,20672
	Δx_3	0,5131	0,51512	0,51512
	Δx_4	0,63279	0,62796	0,62778
	g	0,967594	0,967533	0,967533
	d	$3,5\cdot10^{-6}$	$1,7\cdot10^{-5}$	$9,4\cdot10^{-4}$
Мультипликативно-аддитивная	Δx_1	4,045	4,192	4,19187
	Δx_2	7,66592	7,58462	7,58467
	Δx_3	–0,07517	–0,0728	–0,0728
	Δx_4	–0,07517	–0,0728	–0,0728
	g	75,140	75,10959	75,10957
	d	$8,5\cdot10^{-7}$	$2,8\cdot10^{-7}$	$8,9\cdot10^{-5}$

3.3 Метод решения обратной задачи на основе минимизации суммы модулей изменений аргументов

Сумма абсолютных значений изменений аргументов может быть рассмотрена как альтернатива евклидового расстояния и её минимизация предполагает уменьшение влияния больших отклонений.

В случае использования суммы модулей аргументов задача определения приращений аргументов может быть представлена в виде [133] (1.6)

$$\begin{aligned} & g(\Delta x_1, \Delta x_2, ..., \Delta x_n) = \left|\Delta x_1\right| + \left|\Delta x_2\right| + ... + \left|\Delta x_n\right| \to \min, \\ & f(x_1 + \Delta x_1, x_2 + \Delta x_2, ..., x_n + \Delta x_n) = y + \Delta y. \end{aligned} \tag{3.8}$$

В зависимости от вида модели и прироста результата у аргументов целевой функции (3.8) при раскрытии модуля будет либо положительный, либо отрицательный знак.

После раскрытия модуля в случае линейного ограничения ($f(x+\Delta x) = c_1(x_1+\Delta x_1) + c_2(x_2+\Delta x_2) + ... + c_n(x_n+\Delta x_n)$) ($c$ – числовые значения при аргументах в ограничении) задача (3.8) представляет собой задачу линейного программирования [82]. Классическим методом её решения является симплекс-метод. Однако при единственном ограничении и равенстве числовых значений при аргументах в целевой функции единице задача может быть решена более простым методом. Её решение сводится к нахождению элемента с большим абсолютным числовым значением c при приросте аргумента в ограничении и решения уравнения относительно этого аргумента [83].

В случае, если таких максимальных значений несколько, то при решении задачи может быть выполнено либо равномерное изменение этих аргументов для достижения заданного значения результата, либо их изменение в соответствии с коэффициентами относительной важности, заданных экспертом, либо изменение одного из аргументов, выбранного случайным образом. При реализации программных систем решение уравнения выполняется с помощью классических методов нахождения корней (например, методом Ньютона), далее будет рассмотрено аналитическое решения таких задач.

В случае если ограничение имеет нелинейный вид, рассматриваемая задача относится к задачам нелинейного программирования. Для преобразования ограничения в линейный может быть использовано разложение в ряд Тейлора ($f(\Delta x_1, \Delta x_2, ..., \Delta x_n) = \sum_{i=1}^{n} \frac{\partial f}{\partial \Delta x_i} \Delta x_i$).

Следовательно, в качестве числовых значений при приращениях аргументов рассматривается значение частных производных ($c_i = \frac{\partial f}{\partial \Delta x_i}$) в нулевой точке.

3.3.1 Алгоритм решения задач на основе минимизации суммы модулей изменений аргументов

Алгоритм решения задачи при минимизации суммы абсолютных значений изменений аргументов включает следующие шаги:

Шаг 1. Вычислить величины c_i и определить номер аргумента k, для которого величина c_k максимальна ($c_k = \max_i \{c_i\}$).

Шаг 2. Выполнить решение уравнения относительно аргумента с номером k

$$f(x_k + \Delta x_k) = y + \Delta y \quad .$$

Кроме того, в нелинейных моделях может происходить изменение соотношения производных, поэтому задачу необходимо решать итерационно путем разбиения задачи на подзадачи, в каждой из которых происходит изменение аргумента, имеющего большее значение производной. Итерационный алгоритм решения такой задачи может быть представлен в следующем виде:

Шаг 1. Используя исходные данные, вычислить величину функции ограничения $f(x)$ и сравнить с заданным значением $y + \Delta y$:

Если $f(x) < y + \Delta y$, то изменение аргументов необходимо выполнить в направлении увеличения величины функции ограничения (направлении вектора-градиента): t=1.

Если $f(x) > y + \Delta y$, то изменение аргументов необходимо выполнить в направлении уменьшения величины функции ограничения (направлении вектора-антиградиента): $t = -1$.

Шаг 2. Определить абсолютную разницу между величиной функции ограничения и заданным значением $y + \Delta y$

$$d_0 = \left| f\left(x\right) - y - \Delta y \right|.$$

Шаг 3. Вычислить величины c_i и определить номер аргумента k, для которого величина c_k максимальна ($c_k = \max\limits_i \left\{c_i\right\}$). Определить новое значение аргумента путем движения в сторону градиента/антиградиента

$$\Delta x_k^* = \Delta x_k + t \cdot \alpha \cdot c_k.$$

Шаг 4. Вычислить значение функции ограничения $f(x_i^*)$ и отклонение d_1 от заданного значения $y + \Delta y$.

Проверка: если $d_1 > d_0$ или d_1 меньше заданной точности, то работа алгоритма завершается. Иначе $d_0 = d_1$, $\Delta x_k = \Delta x^*_k$, переход на шаг 3.

Решением задачи будут значения Δx.

3.3.2 Применение алгоритма решения задач на основе минимизции суммы модулей изменений аргументов

Рассмотрим случай мультипликативной зависимости между аргументами (таблица 1.2). Задача оптимизации имеет вид

$$|\Delta x_1| + |\Delta x_2| + |\Delta x_3| + |\Delta x_4| \to \min,$$
$$(x_1 + \Delta x_1) \cdot (x_2 + \Delta x_2) \cdot (x_3 + \Delta x_3)(x_4 + \Delta x_4) = 25000.$$

Значения частных производных равны: $\frac{\partial f}{\partial \Delta x_1} = 2080$, $\frac{\partial f}{\partial \Delta x_2} = 800$, $\frac{\partial f}{\partial \Delta x_3} = 2080$, $\frac{\partial f}{\partial \Delta x_4} = 2600$.

Поскольку полученное значение больше для аргумента x_4, то осуществляется решение уравнения относительно этого аргумента:

$x_4 = 25000 / (10 \cdot 26 \cdot 10) = 9{,}615.$

В таблице 3.13 представлены реультаты решения задачи с помощью предложенного алгоритма и с помощью математического пакета. Для мультипликативной нелинейной модели был использован итерационный алгоритм, для остальных моделей решение было определено за одну итерацию.

Представленный метод не требует использования симплекс-метода или методов нелинейной оптимизации и может быть использован при решении задач линейного программирования рассмотренного вида в других областях исследования. К недостаткам предложенного метода можно отнести необходимость вычисления частных производных функции в случае нелинейной зависимости между аргументами функции.

Таким образом, оптимизационные методы и алгоритмы на основе обратных вычислений позволяют эффективно решать задачи условной оптимизации.

Таблица 3.13 – Решение задач по формированию показателей при минимизации суммы абсолютных значений изменений аргументов

Модель	Показатель	Предложенный метод	Функция Mathcad
Аддитивная	Δx_1	34	34
	Δx_2	34	34
	Δx_3	34	34
	Δx_4	34	34
	Δx_5	34	34
	g	170	170
	d	0	0
Мультипликативная	Δx_1	0	0
	Δx_2	0	0
	Δx_3	0	0
	Δx_4	1,615385	1,615013
	g	1,615385	1,615399
	d	0	$5{,}9\cdot10^{-4}$
Нелинейная аддитивная	Δx_1	0	0
	Δx_2	0	0
	Δx_3	1,126161	1,126164
	g	1,126161	1,126164
	d	0	$9{,}4\cdot10^{-7}$
Мультипликативная нелинейная, $\alpha=10^{-6}$	Δx_1	1,672	1,673
	Δx_2	1,053	1,051
	g	2,7248	2,7249
	d	$2\cdot10^{-6}$	$5\cdot10^{-6}$
Аддитивно-мультипликативная	Δx_1	0	0
	Δx_2	10	10
	Δx_3	0	0
	Δx_4	0	0
	g	10	10,0005
	d	0	$2\cdot10^{-4}$

3.4 Выводы по главе 3

1) Предложены оптимизационные модели для решения обратных задач, где в качестве целевой функции рассматривается мера отдаленности от исходных значений входных параметров. Минимизация суммы квадратов изменений аргументов позволяет определять решения задачи таким образом, чтобы значения аргументов были близки к нулю, минимизация суммы абсолютных значений аргументов позволяет осуществлять выбор аргументов для их изменения.

2) Предложены два метода решения обратных задач при минимизации суммы квадратов изменений аргументов. Их отличием от классических методов решения задач нелинейной оптимизации заключается в отсутствии необходимости оптимизации модифицированной функции: решение задачи сводится к решению системы алгебраических уравнений.

3) Выявлены ограничения алгоритмов на основе формирования системы уравнений. В частности, в случае нелинейной аддитивной модели и нелинейной мультипликативной точность решения оказалась наименьшей и составила 2,3% и 1,4% соответственно, что связано с изменением направления вектора-градиента при переходе от текущего решения к искомому. Также в некоторых задачах не может быть получена аналитическая зависимость одного аргумента через другие аргументы, следовательно, метод на основе формирования кривой заданного уровня не может быть использован. Предложены итерационные алгоритмы решения обратной задачи при минимизации суммы квадратов аргументов. Данный аппарат позволяет выполнить переход от исходных значений аргументов к величинам аргументов, которые удовлетворяют ограничению задачи. Используемый подход обеспечивает простоту реализации алгоритмов ввиду отсутствия необходимости реализации методов решения систем уравнений. При градиентном методе итерационное изменение аргументов обеспечило меньшее значение целевой функции по сравнению с решением задачи с помощью системы уравнений. Таким образом, итерационные алгоритмы могут быть использованы для

решения более широкого круга задач. По сравнению с многократной оптимизацией модифицированной функции с помощью метода штрафа, использование предложенного подхода обеспечивает сокращение времени решения задачи (например, для мультипликативной модели сокращение времени более, чем в 4 раза, при этом число выполняемых арифметических операций меньше более, чем в 400 раз).

4) Предложен метод решения обратной задачи с помощью обратных вычислений на основе минимизации суммы абсолютных изменений аргументов. Особенностью метода является отсутствие необходимости выполнения многократных итераций, связанных с применением симплекс-метода или оптимизации модифицированной функции. Это достигается за счёт представления задачи в виде задачи линейного программирования с одним ограничением. Её решение сводится к определению максимальных значений частных производных функции-ограничения и решению уравнения относительно выявленных аргументов. Для случая нелинейного ограничения разработан итерационный алгоритм, в котором поэтапно осуществляется выбор аргументов для решения уравнения.

5) Применение разработанных алгоритмов позволяет получать результаты, согласующиеся с результатами использования математических пакетов и классических методов решения оптимизационных задач. Подтверждение этого приводится в результатах численного решения обратных задач по формированию показателей с использованием стандартных экономических моделей.

Глава 4. Алгоритмы решения задач нелинейного программирования

При рассмотрении коэффициентов относительной важности, будет изменена целевая функция в модели при минимизации суммы квадратов изменений аргументов (п.3.1). Вследствие этого рассмотрим модификацию разработанных методов и алгоритмов для решения задач оптимизации более широкого круга

$$\begin{aligned} &g(x) \to \min, \\ &f_i(x) = 0. \end{aligned} \tag{4.1}$$

где $g(x)$ – целевая оптимизируемая функция;

f_i – функции ограничений-равенств.

В этом случае $f(x)$ рассматривается как функция по формированию показателя, используемая в обратной задаче. Необходимо определить значения x, формирующие заданное значение функции f, при минимизации целевой функции g. Подход на основе решения обратной задачи с использованием функции-ограничения может быть использован для решения оптимизационных задач линейного и нелинейного программирования [134–136].

4.1. Алгоритм решения оптимизационных задач нелинейного программирования с одним ограничением

Рассмотренная в п. 3.1 задача минимизации суммы квадратов изменений аргументов относится к задачам нелинейной оптимизации

$$\begin{aligned} &g(\Delta x_1, \Delta x_2, ..., \Delta x_n) = \Delta x_1^2 + \Delta x_2^2 + ... + \Delta x_n^2 \to \min, \\ &f(x_1 + \Delta x_1, x_2 + \Delta x_2, ..., x_n + \Delta x_n) = y + \Delta y. \end{aligned}$$

Следовательно, можно сделать вывод, что представленный в п. 3.2.6

алгоритм может быть использован для решения более широкого круга оптимизационных задач.

По виду целевой функции разделим задачи на две группы [137]:

1) Изменение аргумента функции на величину ϕ (относительно точки минимума) приведет к такому же изменению целевой функции, как и при изменении другого аргумента на величину ϕ. Это означает выполнение следующего соотношения

$$g(x_1^* + \phi, x_2^*, ..., x_n^*) = g(x_1^*, x_2^* + \phi, ..., x_n^*) = g(x_1^*, x_2^*, ..., x_n^* + \phi), \quad (4.2)$$

где $(x_1^*, x_2^*, \ldots, x_n^*)$ – точка минимума;

ϕ – некоторое число.

В этом случае вторые частные производные функции будут постоянны и равны между собой.

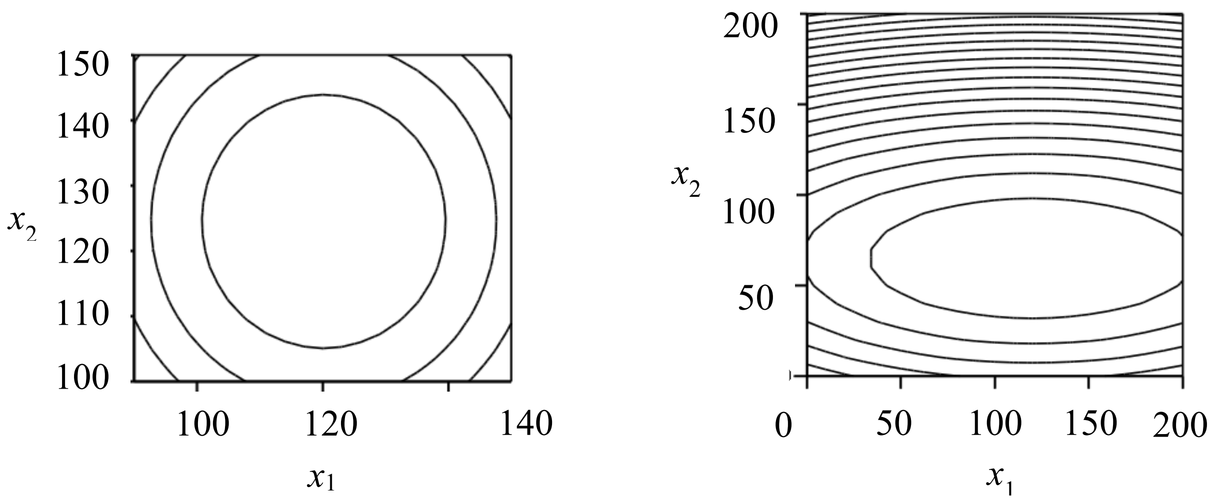
2) Условие для целевой функции (4.2) не выполняется. Частные производные функции – одномерные функции.

На рисунках 4.1а и 4.1б представлены линии уровня для первого и второго случая соответственно (a, b – параметры, a = 148,2, b = –1,15, a_1 = 148,2, b_1 = –1,15, a_2 = 200, b_2 = –3).

По виду ограничения задачи также разделим на две группы:

1) ограничение имеет вид линейного равенства.

2) ограничение имеет нелинейный вид. Частные производные функции ограничения – одномерные линейные функции (вторые частные производные постоянны).



а б

Рисунок 4.1 – Контурный график: *а* – функция $f(x) = \left(a + b \cdot x_1 - 10\right)^2 + \left(a + b \cdot x_2 - 5\right)^2$; *б* – функция $f(x) = \left(a_1 + b_1 \cdot x_1 - 10\right)^2 + \left(a_2 + b_2 \cdot x_2 - 5\right)^2$

Рассмотрим применение разработанного метода для решения представленных задач. Решение задачи будет включать два основных этапа: решение задачи безусловной оптимизации и последующая корректировка полученного решения x^* на величину Δx с учетом ограничения. Величина Δx определяется разностью $\Delta x = x - x^*$, где x^* – значение аргумента, являющееся решением задачи безусловной оптимизации. При этом необходимо учесть влияние отдельных аргументов на изменение целевой функции.

Рассмотрим вариант, когда целевая функция удовлетворяет условию (4.2) и используется градиентный метод решения задачи корректировки значений аргументов x^* (3.2.5). Суть градиентного метода заключается в том, чтобы отношение величин приращений аргументов соответствовало отношению элементов вектора-градиента, т. е. изменение аргументов происходило в направлении наибольшего возрастания/убывания функции-ограничения. Поскольку при движении в направлении градиента/антиградиента наблюдается наибольшее возрастание/убывание функции, то это говорит о том, что можно достигнуть её заданного значения при меньших изменениях аргументов. В свою очередь меньшее изменение аргументов приведет к меньшему отклонению значения целевой функции от величины, полученной путем решения задачи безусловной оптимизации.

Таким образом, решение задачи при использовании вектора-градиента включает следующие шаги:

Шаг 1. Решение задачи безусловной оптимизации целевой функции $g(x)$. Полученное решение включает набор значений x^*. Подстановка полученных величин x^* в ограничение и проверка условия: если неравенство выполняется, то работа алгоритма завершается, иначе – переход на следующий шаг.

Шаг 2. Подстановка полученных значений x^* в ограничение $y^* = f(x^*)$. Проверка направления изменения аргументов: если $y^* > y$, то значение функции ограничения необходимо уменьшить (используются элементы вектора антиградиента) и $t = -1$, в противном случае – увеличить (используются элементы вектора градиента) и $t = 1$.

Шаг 3. Определить необходимые приращения аргументов Δx_i для достижения заданного значения ограничения y^* путем решения системы уравнений (η – номер аргумента, который выбран в качестве базового)

$$\begin{cases} \dfrac{\Delta x_\eta}{\Delta x_i} = \dfrac{t \cdot \dfrac{\partial f(x^*)}{\partial x_\eta}}{t \cdot \dfrac{\partial f(x^*)}{\partial x_i}}; i = 1..n, i \neq \eta \\ f(x^* + \Delta x) = y^*. \end{cases}$$

В результате решения системы получим значения изменений аргументов Δx_i. В случае линейного ограничения полученное соотношение эквивалентно системе, представленной в работе [126], где приращения аргументов определялись исходя из критерия минимизации их суммы квадратов

$$\begin{cases} \dfrac{\Delta x_\eta}{\Delta x_i} = \dfrac{k_\eta}{k_i}; i = 1..l, i \neq \eta \\ f(x^* + \Delta x) = y^*. \end{cases}$$

где k_i – коэффициенты при p_i в линейном уравнении ограничении.

Шаг 4. Изменить значения аргументов функции

$$x_i^* = x_i^* + \Delta x_i.$$

В данном алгоритме выполняется один проход шагов 1-4, значение ключевого показателя формируется на основе значений элементов вектора-градиента, вычисленных в исходной точке.

Рассмотрим теперь случай, когда условие (4.2) для целевой функции не выполняется. Это означает, что изменение аргументов оказывает различное влияние на изменение целевой функции. Пусть частные производные целевой функции представляют собой одномерные функции, чтобы учесть влияние аргументов на изменение целевой функции относительно точки минимума будут использованы значения вторых частных производных

$$\begin{cases} \dfrac{\Delta x_\eta}{\Delta x_i} \dfrac{\dfrac{\partial^2 g\left(x^*\right)}{\partial x_\eta^2}}{\dfrac{\partial^2 g\left(x^*\right)}{\partial x_i^2}} = \dfrac{t \cdot \dfrac{\partial f(x^*)}{\partial x_\eta}}{t \cdot \dfrac{\partial f(x^*)}{\partial x_i}}; i = 1..l, i \neq \eta \\ f(x^* + \Delta x) = y^*. \end{cases}$$

Решение обратной задачи также может осуществляться итерационно с помощью алгоритмов, рассмотренных в п.3.2.6.

Рассмотренный выше алгоритм предполагает, что ограничение задачи имеет вид равенства. Задача с ограничением-неравенством будет иметь вид

$$g(x) \to \min,$$
$$f(x) \leq / \geq y^*.$$

В этом случае после выполнения шага 1 осуществляется проверка: если значение функции $f(x)$ удовлетворяет ограничению задачи $f(x) \leq / \geq y^*$, то работа алгоритма завершается, иначе ограничение-неравенство преобразуется в ограничение равенство и выполняются дальнейшие шаги (начиная со второго) представленного выше алгоритма.

4.1.1 Применение алгоритма для решения задачи оптимизации цены

Рассмотрим применение алгоритма для решения задачи оптимизации цены. Задаче оптимизации цены посвящено множество работ [138–144], поскольку данная характеристика является одной из определяющих для максимизации прибыли экономического объекта. В них приводятся оптимизационные модели с учетом специфики объекта исследования.

Из представленных в литературе [139, 141–146] исследований следует, что задача оптимизации цены часто представляется в виде задачи нелинейного программирования. Рассмотрим задачу оптимизации с одним ограничением в виде равенства [137, 147]. Предполагается линейная зависимость спроса от цены, параметры линейной регрессии для определения прогнозного значения еженедельного спроса определяются на основе имеющихся статистических данных о значениях цен и спроса за предыдущие периоды. Классическим методом оценки параметров регрессии является метод наименьших квадратов.

Определим следующие обозначения:

p_j – искомая цена на изделие в j-м периоде (j =1..n, n– число периодов);

p_i – искомая цена на изделие i-го вида (i=1..m, m – число видов изделий);

q_i – текущая цена на изделие i-го вида;

c_j – планируемый объем выпуска изделия в j-м периоде;

P_1, P_2 – значение выручки, которое необходимо получить;

r – затраты ресурса на изготовление единицы изделия;

R – величина имеющегося запаса материального у предприятия;

h_i – геометрический объем единицы изделия i-го вида;

S – объем доставки;

a_i и b_i – параметры линейной регрессии, используемые для определения спроса v_i на изделие i-го вида

$$v_i = a_i + b_i \cdot p_i.$$

При этом предполагается отрицательная эластичность спроса, т.е. его уменьшение при росте цен, следовательно, на знак параметров накладываются следующие ограничения: $a_i \geq 0$ и $b_i \leq 0$.

Если рассматривается изделие одного вида, то параметры указываются без индексов: a и b.

Можно определить следующие варианты задач оптимизации цены p (в качестве финансового показателя деятельности предприятия используется выручка):

Вариант 1: Минимизация отклонения прогнозируемого спроса в j-м периоде от планируемого объема производства при наличии ограничения на объем используемого ресурса: объем используемого ресурса для производства изделия не превышает имеющийся запас [148]

$$g(p) = \sum_{j=1}^{n} \left(a + b \cdot p_j - c_j\right)^2 \to \min,$$

$$f(p) = \sum_{j=1}^{n} r v_j = \sum_{j=1}^{n} r\left(a + b \cdot p_j\right) \leq R, \;\; p_j \geq 0. \tag{4.3}$$

Вариант 2: Максимизация суммарной выручки (минимизация величины, полученной путем умножения выручки на –1), полученной в результате суммирования выручки от продажи i-го вида изделия, при наличии ограничения на объем доставки всех видов изделий

$$g(p) = -\sum_{i=1}^{m} \left(a_i + b_i \cdot p_i\right) p_i \to \min,$$

$$f(p) = \sum_{i=1}^{m} h_i v_i = \sum_{i=1}^{m} h_i \left(a_i + b_i p_i\right) \leq S, \;\; p_i \geq 0. \tag{4.4}$$

Вариант 3: Минимизация отклонения прогнозируемого спроса в j-м периоде от планируемого объема производства изделия при наличии ограничения на значение суммарной выручки за n периодов, величина которой должна быть не менее установленной

$$g(p)=\sum_{j=1}^{n}\left(a+b\cdot p_j-c_j\right)^2\to\min,$$

$$f(p)=\sum_{j=1}^{n}p_jv_j=\sum_{j=1}^{n}p_j\left(a+b\cdot p_j\right)\geq P_1,\ p_j\geq 0. \tag{4.5}$$

Вариант 4: Минимизация отклонения прогнозируемого спроса на изделие i-го вида от планируемого объема производства изделия данного вида при наличии ограничения на значение суммарной выручки по всем видам изделий, величина которой должна быть не менее установленной

$$g(p)=\sum_{i=1}^{m}\left(a_i+b_i\cdot p_i-c_i\right)^2\to\min,$$

$$f(p)=\sum_{i=1}^{m}p_iv_i=\sum_{i=1}^{m}p_i\left(a_i+b_i\cdot p_i\right)\geq P_1,\ p_i\geq 0. \tag{4.6}$$

Вариант 5: Минимизация отклонения искомой цены от текущей (найти значение наиболее близкое к искомому) при наличии ограничения на суммарную выручку по всем видам изделий, величина которой должна быть не менее заданного значения

$$g(p)=\sum_{i=1}^{m}\left(p_i-q_i\right)^2\to\min,$$

$$f(p)=\sum_{i=1}^{m}p_iv_i=\sum_{i=1}^{m}p_i\left(a_i+b_i\cdot p_i\right)\geq P_2,\ p_i\geq 0. \tag{4.7}$$

Рассмотрим решение представленных задач с помощью классических методов нелинейного программирования: штрафов и множителей Лагранжа, которые основаны на сведении задачи условной оптимизации к задаче безусловной оптимизации.

В методе штрафов формируется функция, включающая целевую функцию и штраф-функцию от ограничения и штрафного параметра. Так, для третьего варианта штрафная функция V будет иметь вид

$$V(p,W)=\sum_{j=1}^{n}\left(a+b\cdot p_j-c_j\right)^2-W\cdot\ln\left(\sum_{j=1}^{n}p_j\left(a+b\cdot p_j\right)-P_1\right).$$

Примем следующие значения исходных данных: $n = 3$, $a = 148{,}2$, $b = -1{,}15$, $c_1 = 10$, $c_2 = 5$, $c_3 = 11$, $P_1 = 3400$ руб. В таблице 4.1 представлены результаты, полученные с помощью метода штрафов (начальные значения аргументов равны 300).

Таблица 4.1 – Результаты, полученные с помощью метода штрафов

Штрафной параметр W	Аргументы функции			Целевая функция $g(p)$
	p_1	p_2	p_3	
100	116,4	120,46	115,59	58,84
10	118,70	122,93	117,85	8,988
1	119,36	123,65	118,51	2,71
0,1	119,5	123,79	118,64	1,9
0,01	119,51	123,81	118,65	1,813
0,001	119,51	123,81	118,65	1,804

В методе множителей Лагранжа формируется функция L, включающая неизвестный параметр – множитель Лагранжа: определяется сумма целевой функции и ограничения, умноженного на множитель Лагранжа λ. Так, для первого варианта задачи (r=30, R=600) функция Лагранжа будет иметь вид

$$L(p,\lambda) = \sum_{j=1}^{n}\left(a + b\cdot p_j - c_j\right)^2 + \lambda\left(\sum_{j=1}^{n} r\left(a + b\cdot p_j\right) - R\right).$$

Для решения задачи необходимо вычислить частные производные функции по переменным p, приравнять их нулю и решить систему уравнений:

$$\begin{cases} 2{,}645p_1 - 34{,}5\lambda - 317{,}86 = 0 \\ 2{,}645p_2 - 34{,}5\lambda - 329{,}36 = 0 \\ 2{,}645p_3 - 34{,}5\lambda - 315{,}56 = 0 \\ \lambda\left(\left(148{,}2 - 1{,}15\cdot p_1\right)\cdot 30 + \left(148{,}2 - 1{,}15\cdot p_2\right)\cdot 30 + \left(148{,}2 - 1{,}15\cdot p_3\right)\cdot 30 - 600\right) = 0 \end{cases}$$

Решением системы будет значение множителя Лагранжа, равное 0,133, значения цен равны: p_1=121,91, p_2=126,26, p_3=121,04.

Для представленных задач (4.3)–(4.7) полученные системы для определения приращений аргументов с помощью обратных вычислений будут соответственно выглядеть следующим образом.

Определение приращений аргументов для решения задачи (4.3) выполняется с помощью системы уравнений

$$\begin{cases} \dfrac{\Delta p_\eta}{\Delta p_j} = \dfrac{b \cdot r}{b \cdot r} = 1; j = 1..n, \eta \neq j \\ f(p^* + \Delta p) = \sum\limits_{j=1}^{n} r\left(a + b \cdot \left(p_j^* + \Delta p_j\right)\right) = R. \end{cases}$$

Выполнив решение системы, получим следующее выражение для расчета приростов аргументов функции

$$\Delta p_j = \frac{\dfrac{\dfrac{R}{r} - n \cdot a}{b} - \sum\limits_{i=1}^{n} p_i^*}{n}, j = 1...n.$$

Система уравнений для решения задачи (4.4) имеет вид:

$$\begin{cases} \dfrac{\Delta p_\eta}{\Delta p_i} = \dfrac{h_\eta}{h_i}; i = 1..m, i \neq \eta \\ f(p^* + \Delta p) = \sum\limits_{i=1}^{m} h_i\left(a_i + b_i\left(p_i^* + \Delta p_i\right)\right) = S. \end{cases}$$

Решая систему, получим выражения для расчета аргументов:

$$\Delta p_\eta = \frac{S - \sum\limits_{j=1}^{n}\left(h_j\left(a_j + b_j p_j^*\right)\right)}{\sum\limits_{j=1}^{n}\left(\dfrac{h_j^2 \cdot b_j}{h_\eta}\right)},$$

$$\Delta p_i = \frac{h_i}{h_\eta} \cdot \Delta p_\eta, i = 1..m, i \neq \eta.$$

Вычисление приращений аргументов для решения задачи (4.5) осуществляется следующим образом:

$$\begin{cases} \dfrac{\Delta p_\eta}{\Delta p_j} = \dfrac{\left(a + 2bp_\eta^*\right)}{\left(a + 2bp_j^*\right)}; j = 1..n, j \neq \eta \\ f(p^* + \Delta p) = \sum\limits_{j=1}^{n}\left(p_j + \Delta p_j\right)\left(a + b \cdot \left(p_j + \Delta p_j\right)\right) = P_1. \end{cases}$$

В данном и последующих вариантах ограничение имеет нелинейный вид, поэтому определение приращения базового аргумента может быть выполнено с помощью стандартных способов решения квадратного уравнения. Так, для текущего варианта уравнение будет иметь вид

$$\Delta p_\eta^2\left(b\sum_{i=1}^{n}\left(\frac{a+2bp_i^*}{a+2bp_\eta^*}\right)^2\right)+\Delta p_\eta\left(a\sum_{i=1}^{n}\left(\frac{a+2bp_i^*}{a+2bp_\eta^*}\right)+2b\sum_{i=1}^{n}\left(p_2^*\frac{a+2bp_i^*}{a+2bp_\eta^*}\right)\right)+$$

$$+a\sum_{i=1}^{n}p_i^*+b\sum_{i=1}^{n}\left(p_i^*\right)^2-P_1=0.$$

Система уравнений для решения задачи (4.6) имеет вид

$$\begin{cases}\dfrac{\Delta p_\eta\dfrac{b_\eta^2}{b_i^2}}{\Delta p_i}=\dfrac{\left(a_\eta+2b_\eta p_\eta^*\right)}{\left(a_i+2b_i p_i^*\right)};i=1..m,i\neq\eta\\ f(p^*+\Delta p)=\sum\limits_{i=1}^{m}\left(p_i^*+\Delta p_i\right)\left(a_i+b_i\cdot\left(p_i^*+\Delta p_i\right)\right)=P_1.\end{cases}$$

Наконец, определение приращений аргументов функции для решения задачи (4.7) осуществляется путем решения системы уравнений

$$\begin{cases}\dfrac{\Delta p_\eta}{\Delta p_i}=\dfrac{\left(a_\eta+2b_\eta q_\eta\right)}{\left(a_i+2b_i q_i\right)};i=1..m,i\neq\eta\\ f(p^*+\Delta p)=\sum\limits_{i=1}^{m}\left(p_i^*+\Delta p_i\right)\left(a_i+b_i\cdot\left(p_i^*+\Delta p_i\right)\right)=P_2.\end{cases}$$

Можно сделать вывод, что в случае линейного ограничения могут быть получены аналитические формулы расчета приращений. При нелинейном виде ограничения (задачи (4.5) – (4.7)) для определения приращений аргументов возникает необходимость решения квадратного уравнения. В этом случае могут быть использованы классические методы нахождения корней (методы Ньютона, дихотомии, использование дискриминанта, и т.д.).

Для решения оптимизационной задачи были использованы данные, представленные в таблице 4.2.

Таблица 4.2 – Исходные данные задачи оптимизации цены

Показатель	Номер продукции i /номер периода j		
	1	2	3
Параметр линейной регрессии a	148,2	152,1	130,5
Параметр линейной регрессии b	–1,15	–1,21	–1,1
Затраты ресурса на единицу продукции r, г.	30	–	–
Планируемый объем производства, c, кг.	10	5	11
Объем единицы изделия, h, куб.м.	0,2	0,4	0,5
Текущая цена продукции, q, руб.	80	75	83

Значения ограничений: R=600 г., S=60 куб.м., P_1= 3400 руб., P_2=12700 руб.

В таблице 4.3 представлено решение задач оптимизации (4.4)–(4.7). В последнем столбце представлена разница полученного решения, с решением задачи с помощью математического пакета

$$u = g(x) - g^*(x),$$

где $g(x)$ – значение целевой функции, полученное путем решения задачи с помощью обратных вычислений;

$g^*(x)$ – значение целевой функции, полученное путем решения задачи с использованием встроенной функции MathCad «Minimize».

Из таблицы 4.3 можно увидеть, что решение третьей задачи также согласуется с решениями, полученными с помощью метода штрафов (таблица 4.1) и множителей Лагранжа. Решение для пятого варианта с помощью метода обратных вычислений имеет наименьшую точность.

Таблица 4.3 – Решение задач оптимизации

Номер варианта	Значение целевой функции, $g(x)$	Цена продукции			u
		p_1	p_2	p_3	
1	12	121,91	126,26	121,04	$-7\cdot 10^{-12}$
2	–12800	71,38	76,74	76,68	$-5\cdot 10^{-6}$
3	1,803	119,51	123,81	118,65	$-1\cdot 10^{-5}$
4	4,434	119,1	120,49	107,64	$6\cdot 10^{-6}$
5	30,245	77,18	72,68	78,89	$2\cdot 10^{-3}$

4.1.2 Применение алгоритма для решения задачи оптимизации закупок

Также рассмотрим решение задачи оптимизации закупок. В процессе своей деятельности предприятие сталкивается с необходимостью использования различных ресурсов, поэтому организация закупок оказывает значительное влияние на эффективность его работы. В условиях конкуренции и возможности выбора из множества предложений возникает необходимость в принятии решения относительно таких характеристик закупок как формируемый ассортимент продукции, перечень поставщиков и т.д. При этом финансовые ресурсы предприятия ограничены, в связи с чем возникает необходимость их рационального использования. Таким образом возникает задача оптимизации закупок, т.е. выбора среди существующих вариантов наилучшего, который бы максимизировал полезность закупок при имеющихся ресурсах. Для решения такого рода задач существует ряд моделей и методов. Так, например, задача оптимизации закупок фирмы, т.е. выбора вида и количества заказываемого у поставщиков товара при ограниченном бюджете, приводится в работе Р.Ф. Фарманова [149].

Рассмотрим применение рассмотренного алгоритма для решения задачи оптимизации закупок [149] на основе данных кондитерской фирмы [130, 150]. Организация формирует прогнозное значение ежедневного спроса на основе имеющихся статистических данных за предыдущие периоды. Необходимо осуществить закупку товаров таким образом, чтобы максимально удовлетворить спрос при ограниченном количестве денежных ресурсов. Исходными данными модели являются:

– b_i – прогнозное значение среднего спроса на товар i ($i = 1,...,n$, n – число наименований товаров);

– a_i – цена закупки i-го товара;

– A – величина бюджета закупок.

Полученная задача представляет собой задачу квадратичного программирования с линейным ограничением в виде равенства

$$g(x) = \sum_{i=1}^{n} \left(x_i - b_i \right)^2 \to \min\,, \tag{4.8}$$

$$f_1(x) = \sum_{i=1}^{n} a_i x_i = A,$$

$$x_i \geq 0$$

Информация о трех кондитерских изделиях представлена в таблице 4.4. Бюджет закупок равен 3000 руб., заданное значение суммарной маржинальной прибыли – 1150 руб.

Таблица 4.4 – Исходные данные задачи оптимизации закупок

Показатель	Номер изделия, *i*			
	1	2	3	4
Прогнозный спрос, кг.	11	16	8	5
Цена закупки, руб. за кг.	125	105	170	160
Маржинальная прибыль, руб.	50	40	40	55

Задачу квадратичного программирования с одним ограничением можно представить в следующем виде

$$\left(x_1 - 11\right)^2 + \left(x_2 - 16\right)^2 + \left(x_3 - 8\right)^2 + \left(x_4 - 5\right)^2 \to \min,$$
$$125x_1 + 105x_2 + 170x_3 + 160x_4 = 3000.$$

Минимумом целевой функции

$$g(x) = \left(x_1 - 11\right)^2 + \left(x_2 - 16\right)^2 + \left(x_3 - 8\right)^2 + \left(x_4 - 5\right)^2$$

будут значения прогнозного спроса *b*: $x_1^* = 11$, $x_2^* = 16$, $x_3^* = 8$, $x_4^* = 5$.

Таким образом, при решении задачи оптимизации закупок в качестве решения, полученного на первом шаге рассмотренного алгоритма, принимаются значения прогнозного спроса, т.е. отсутствует необходимость решения задачи безусловной оптимизации.

Поскольку в данном случае влияния аргументов на изменение функции одинаковы, то отношения вторых частных производных целевой функции будут равны 1.

С помощью обратных вычислений определим изменения значений объема заказа путем решения системы уравнений

$$\begin{cases} 125(11+\Delta x_1)+105(16+\Delta x_2)+170(8+\Delta x_3)+160(5+\Delta x_4)=3000; \\ \dfrac{\Delta x_1}{\Delta x_4}=\dfrac{a_1}{a_4}=0{,}781; \\ \dfrac{\Delta x_2}{\Delta x_4}=\dfrac{a_{21}}{a_4}=0{,}656; \\ \dfrac{\Delta x_3}{\Delta x_4}=\dfrac{a_3}{a_4}=1{,}063. \end{cases}$$

В таблице 4.5 представлено полученное решение задачи.

Таблица 4.5 – Решение задачи оптимизации закупок

Метод решения	Показатель					
	Δx_1	Δx_2	Δx_3	Δx_4	$g(x)$	d
Предложенный алгоритм	–3,411892	–2,865989	–4,640173	–4,367221	60,45871842	$4{,}5\cdot 10^{-13}$
Функция MathCad	–3,411888	–2,865986	–4,640176	–4,367221	60,45871835	$1{,}2\cdot 10^{-6}$

При полученных значениях аргументов суммарная маржинальная прибыль составит 1074 руб. Значение целевой функции соттветствует полученному с помощью математического пакета с точностью до седьмого знака после запятой, при этом было обеспечено большее соответствие функции ограничению.

4.2 Алгоритм решения оптимизационных задач нелинейного программирования при нескольких ограничениях

Задача оптимизации с несколькими ограничениями-равенствами имеет вид

$$
\begin{aligned}
& g(x) \to \min, \\
& f_1(x) = y_1^*, \\
& \ldots \\
& f_r(x) = y_r^*.
\end{aligned}
\tag{4.9}
$$

Для решения задачи (4.9) с использованием разработанного алгоритма необходимо выполнить преобразование ограничений в одно ограничение. Для этого могут быть рассмотрены два способа:

1) замена переменных: последовательное выражение переменной из одного ограничения и подстановка её во второе ограничение и целевую функцию (основное преимущество – уменьшение размерности решаемой задачи);

2) формирование ограничения в виде суммы квадратов разницы между функцией ограничения и её заданным значением

$$
f = \left(f_1(x) - y_1^* \right)^2 + \ldots + \left(f_r(x) - y_r^* \right)^2 . \tag{4.10}
$$

Рассмотрим алгоритм решения оптимизационной задачи при наличии двух ограничений и использовании метода замены переменных.

Шаг 1. Преобразование задачи с двумя ограничениями в задачу с одним ограничением путем подстановки выраженной переменной x_k из ограничения f_1 в целевую функцию и функцию–ограничения f_2

$$
x_k = \psi(x_l), l = 1..n, l \neq k
$$

$$
g(x_1, x_2, \psi(x_l), \ldots, x_n) \to \min,
$$

$$
f_2(x_1, x_2, \psi(x_l), \ldots, x_n) = y^*.
$$

Шаг 2. Решение задачи безусловной оптимизации целевой функции g, сформированной на первом шаге. В результате будет получен набор значений $x_1^*, x_2^*, \ldots, x_n^*$.

Шаг 3. Решение задачи с помощью обратных вычислений для определения приращений $\Delta x_1, \Delta x_2, \ldots, \Delta x_n$ для достижения заданного значения результирующего показателя y^* с учётом влияния аргументов на изменение целевой функции.

Шаг 3.1 Решение системы уравнений (η – номер аргумента, выбранного в качестве базового)

$$\begin{cases} \dfrac{\Delta x_\eta}{\Delta x_i} \dfrac{\dfrac{\partial^2 g\left(x^*\right)}{\partial x_\eta^2}}{\dfrac{\partial^2 g\left(x^*\right)}{\partial x_i^2}} = \dfrac{\dfrac{\partial f_2(x^*)}{\partial x_\eta}}{\dfrac{\partial f_2(x^*)}{\partial x_i}}; i = 1..n, i \neq \eta \\ f_2(x^* + \Delta x) = y^*. \end{cases}$$

Решением системы будут величины аргументов $x_i^* = x_i^* + \Delta x_i$ $(i \neq k)$.

Шаг 3.2 Определение значения *k*-го аргумента путем подстановки в функцию выражения: $x_k^* = \psi\left(x_l^* + \Delta x_l\right), l = 1..n, l \neq k$.

В случае, если вместо замены переменных используется интегральное ограничение, то на шаге 1 осуществляется формирование единого ограничения (4.10), а на шаге 2 выполняется оптимизация исходной целевой функции $g(x)$.

4.3. Итерационные алгоритмы решения оптимизационных задач нелинейного программирования

Итерационные алгоритмы, рассмотренные в п. 3.2.6 также могут быть использованы для решения задач нелинейного программирования с одним ограничением в виде равенства. В этом случае может быть использован градиентный метод, для итерационного алгоритма необходимо выполнить следующую модификацию:

1) Осуществить безусловную оптимизацию целевой функции $g(x)$, в результате последующего использования итерационных алгоритмов происходит корректировка полученных значений аргументов x^*. Т.е. вместо исходных

значений x, применяемых в обратной задаче, используются значения, полученные в результате безусловной оптимизации целевой функции $g(x)$.

2) В итерационных формулах расчёта необходимо выполнить корректировку, которая отражает влияние изменения аргумента на изменение целевой функции (если вторые частные производные не постоянны и не равны между собой). Данная операция выполняется путем деления частных производных функции ограничения первого порядка на частные производные второго порядка целевой функции

$$x_i^* = x_i^* + t \cdot \alpha \cdot \frac{\dfrac{\partial f\left(x_i^*\right)}{\partial x_i}}{\dfrac{\partial^2 g\left(x_i^*\right)}{\partial x_i^2}}.$$

4.3.1 Применение итерационных алгоритмов для решения задачи оптимизации портфеля ценных бумаг

Рассмотрим решение двух классических задач исследования операций с помощью итерационных алгоритмов: оптимизация портфеля ценных бумаг и формирование затрат при заданном суммарном заказе.

Задача оптимизации портфеля ценных бумаг при отсутствии их взаимного влияния и при минимизации риска имеет вид [151]

$$\begin{aligned} &g(x) = \sigma_1 x_1^2 + \sigma_2 x_2^2 + \sigma_3 x_3^2 + \sigma_4 x_4^2 \to \min, \\ &m_1 x_1 + m_2 x_2 + m_3 x_3 + m_4 x_4 = M, \\ &x_1 + x_2 + x_3 + x_4 = 1, \end{aligned} \qquad (4.11)$$

где σ – показатель риска;

m – показатель доходности;

M – ограничение доходности.

Значения показателей риска и доходности: $\sigma_1 = 0{,}0165$, $\sigma_2 = 0{,}0032$, $\sigma_3 = 0{,}0008$, $\sigma_4 = 0{,}0002$, $m_1 = 0{,}291$, $m_2 = 0{,}121$, $m_3 = 0{,}481$, $m_4 = 0{,}381$. Заданное значение доходности M равно 0,37.

Задача (4.11) имеет два ограничения, для использования итерационного алгоритма необходимо выполнить их преобразование в одно ограничение. Рассмотрим формирование ограничения в виде суммы квадратов разницы между функцией ограничения и её заданным значением.

Задача оптимизации в этом случае имеет вид

$$g(x) = \sigma_1 x_1^2 + \sigma_2 x_2^2 + \sigma_3 x_3^2 + \sigma_4 x_4^2 \to \min,$$
$$\left(m_1 x_1 + m_2 x_2 + m_3 x_3 + m_4 x_4 - M\right)^2 + \left(x_1 + x_2 + x_3 + x_4 - 1\right)^2 = 0.$$

Первая итерационная формула для первой переменной будет иметь вид (начальные значения переменных x равны нулю)

$$\frac{\partial^2 g(x_i)}{\partial x_i^2} = 2\sigma_i,$$

$$\frac{\partial f(x_1)}{\partial x_1} = 2x_1 + 2x_2 + 2x_3 + 2x_4 + 2m_1\left(m_1 x_1 + m_2 x_2 + m_3 x_3 + m_4 x_4 - 0{,}37\right) - 2,$$

$$x_1 = 0 - \alpha \frac{-2{,}215}{0{,}033}.$$

Результаты решения задачи представлены в таблице 4.6 (d – значение функции-ограничения). С помощью итерационного градиентного метода были определены значения аргументов с точностью до 4 знака после запятой в сравнении с решением, полученным с помощью функции Mathcad. При этом значение функции-ограничения было определено с большей точностью по сравнению с решением с помощью математического пакета.

На рисунках 4.2–4.5 представлено изменение аргументов x_1, x_2, x_3, x_4 при изменении параметра α (используется итерационный градиентный алгоритм). Можно увидеть, что при уменьшении параметра точность решения увеличивается и значение аргумента сходится к искомому.

Таблица 4.6 – Результаты решения задачи оптимизации портфеля при α=10^{-8}

Метод	x_1	x_2	x_3	x_4	d	$g(x)$
Итерационный градиентный	0,01135	0,08626	0,11998	0,7824	$6\cdot10^{-22}$	$1{,}5\cdot10^{-4}$
Градиентный	0,008	0,04	0,186	0,76	$3\cdot10^{-4}$	$1{,}4\cdot10^{-4}$
Использование функции MathCad	0,01135	0,08626	0,11997	0,78243	10^{-12}	$1{,}5\cdot10^{-4}$

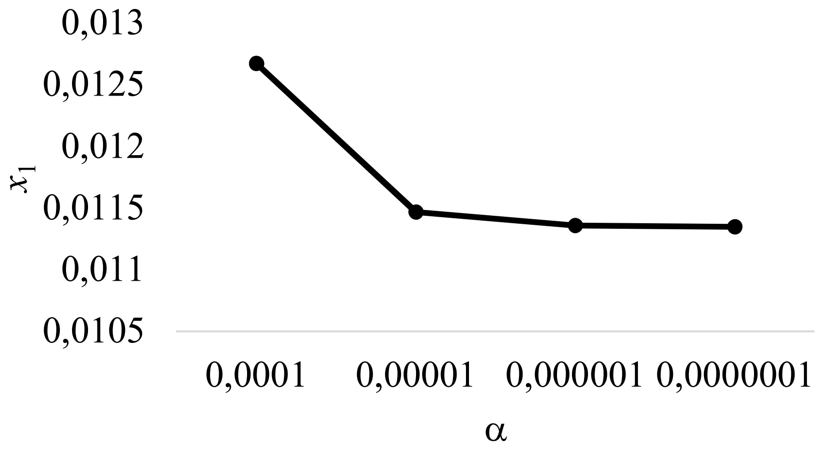


Рисунок 4.2 – Изменение значений аргумента x_1

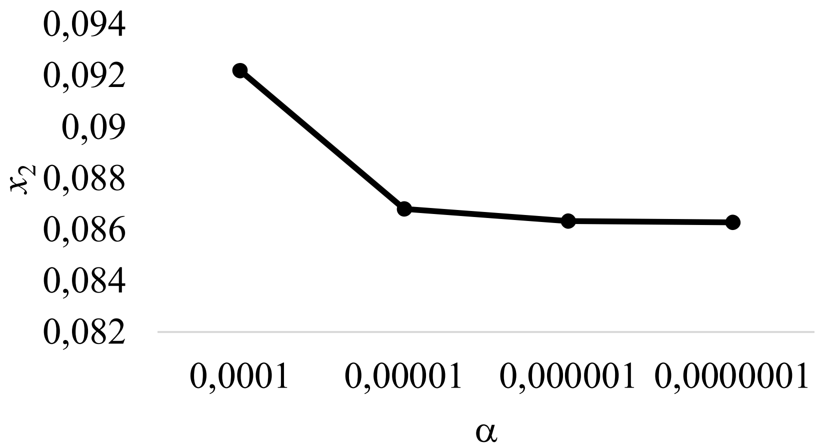


Рисунок 4.3 – Изменение значений аргумента x_2

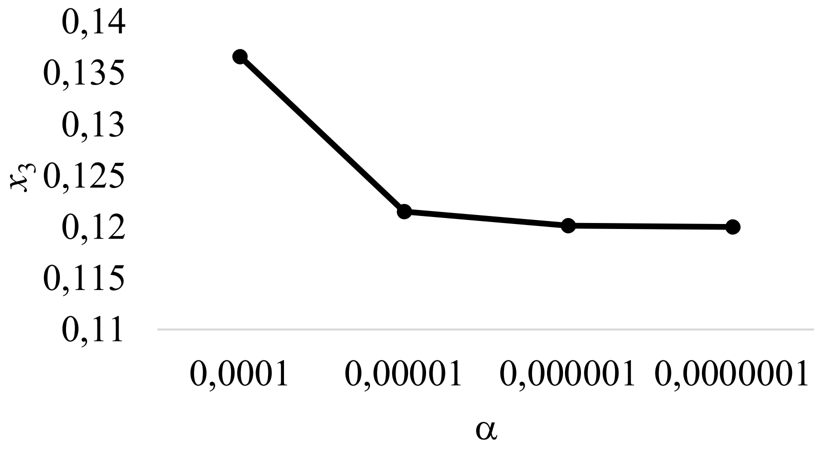


Рисунок 4.4 – Изменение значений аргумента x_3

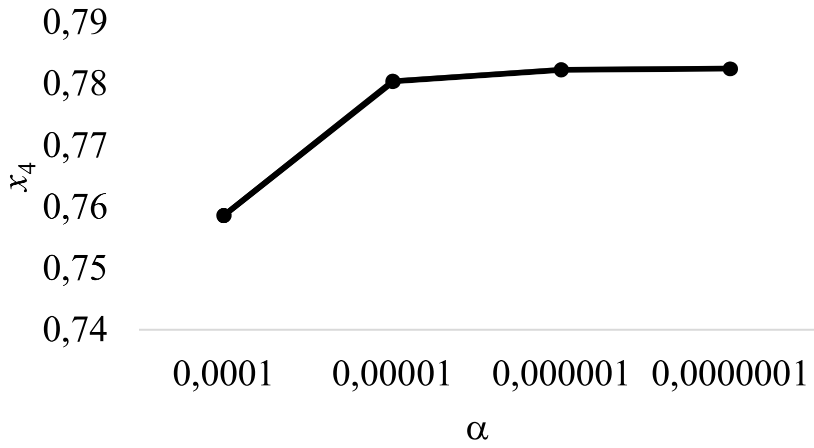


Рисунок 4.5 – Изменение значений аргумента x_4

4.3.2 Применение итерационных алгоритмов для решения задачи оптимизации складских затрат

Также рассмотрим задачу минимизации функции затрат на закупку и хранение запаса (модель управления запасами) при заданном объеме закупок, который должен быть равен 28

$$y = \frac{w_1 q_1}{x_1} + \frac{s_1}{2} x_1 + \frac{w_2 q_2}{x_2} + \frac{s_2}{2} x_2 + \frac{w_3 q_3}{x_3} + \frac{s_3}{2} x_3.$$

$$x_1 + x_2 + x_3 = 28,$$

где x – размер заказа;

s – затраты на хранение единицы товара в единицу времени;

w – затраты на выполнение одного заказа;

q – интенсивность спроса.

Исходные данные представлены в таблице 1.2–1.4 (нелинейная аддитивная модель), при этом исходными значениями аргументов x будут величины, полученные путем безусловной оптимизации целевой функции: x_1=11,547, x_2=19,999, x_3 =22,358.

В таблице 4.7 представлены результаты решения задачи оптимизации затрат. С помощью итерационного градиентного метода были определены значения аргументов с точностью до 3 знака после запятой в сравнении с решением, полученным с помощью функции Mathcad.

Таблица 4.7 – Результаты решения задачи оптимизации затрат при α=10^{-8}

Метод	x_1	x_2	x_3	d	$g(x)$
Итерационный градиентный	7,8844	9,4974	10,6182	$4\cdot10^{-6}$	9,185
Градиентный	9,389	8,786	9,825	$4\cdot10^{-15}$	9,29
Использование функции MathCad	7,8844	9,4973	10,6183	$6\cdot10^{-13}$	9,185

На рисунках 4.6–4.8 представлено изменение аргументов x_1, x_2, x_3 при изменении параметра α и использовании итерационного градиентного алгоритма. При уменьшении параметра значение аргумента определяется с большей точностью.

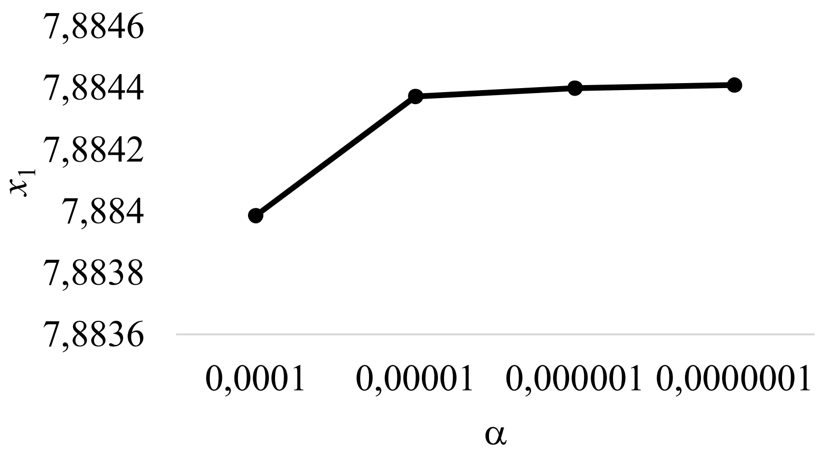


Рисунок 4.6 – Изменение значений аргумента x_1 в задаче оптимизации затрат

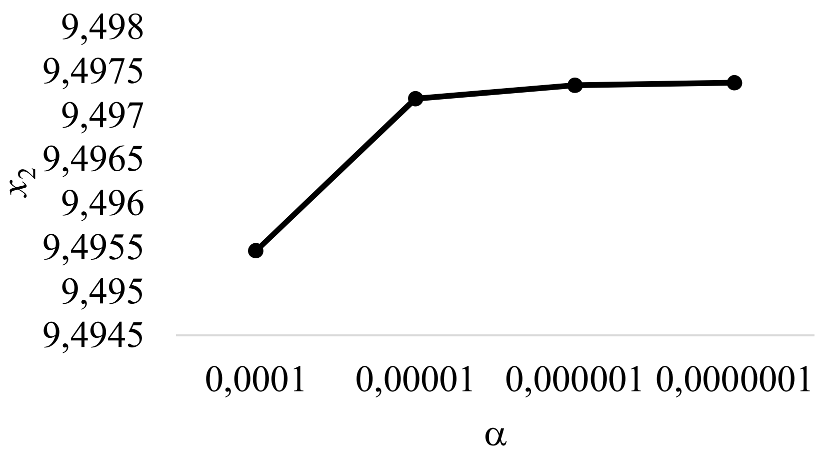


Рисунок 4.7 – Изменение значений аргумента x_2 в задаче оптимизации затрат

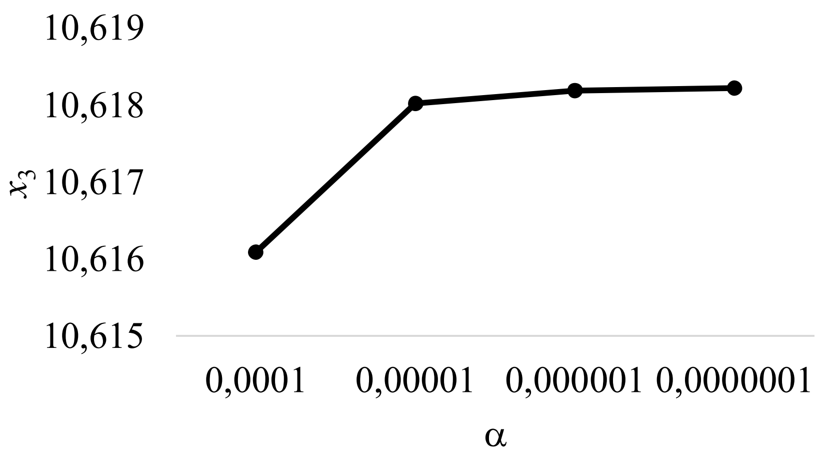


Рисунок 4.8 – Изменение значений аргумента x_3 в задаче оптимизации затрат

Согласно полученным результатам, итерационный градиентный метод обеспечил большее соответствие решению, полученному с помощью математического пакета. В задаче оптимизации затрат его использование обеспечило меньшее значение целевой функции, а в задаче оптимизации портфеля – меньшую разницу между значением функции ограничения и заданным значением.

4.4Выводы по главе 4

1) Выполнена модификация алгоритмов решения обратных задач для решения задач оптимизации более широкого круга, представляющих собой задачу квадратичного программирования с одним ограничением. Особенностью предложенного подхода является отсутствие необходимости формирования модифицированной функции и многократного решения задачи безусловной оптимизации, что обеспечивает упрощение процедуры его компьютерной реализации и ускорение времени решения задачи. Это возможно благодаря использованию в предложенном алгоритме метода решения обратной задачи, позволяющего перейти от значений аргументов, полученных в результате безусловной оптимизации целевой функции к значениям аргументов, удовлетворяющих ограничению задачи. Ограничением алгоритма является требуемый вид целевой функции: она должна быть квадратичной, при этом её частные производные должны быть одномерными функциями.

2) Предложен алгоритм решения оптимизационных задач при наличии нескольких ограничений. Суть алгоритма заключается в переходе от задачи с несколькими ограничениями к задаче с одним ограничением. Рассмотрены два варианта такого перехода: замена переменных и формирование модифицированного ограничения, включающего сумму квадратов ограничений с нулевой правой частью. Также предложена модификация алгоритма для случая наличия в задаче ограничения-неравенства.

3) Выполнена модификация итерационных алгоритмов, предназначенных для решения обратных задач, для решения оптимизационных задач нелинейного программирования представленного вида. В этом случае выполняется безусловная оптимизации целевой функции, а итерационные формулы корректируются с целью учета влияния аргументов на изменение целевой функции. В тестовых примерах значения аргументов были определены с точностью до 3–4 знака после запятой, что является достаточным для решения большинства практических задач.

4) Применение разработанных алгоритмов позволяет получать результаты, согласующиеся с результатами использования классических методов нелинейной оптимизации и математических пакетов. Подтверждение этого приводится в рассмотренных примерах, в качестве которых в том числе рассмотрены классические задачи оптимизации закупок, цены, портфеля ценных бумаг, запасов предприятия.

Глава 5. Оптимизационные модели и алгоритмы решения обратной задачи с использованием коэффициентов относительной важности

5.1 Оптимизационная модель и алгоритм решения классической обратной задачи

При использовании классического аппарата обратных вычислений возможна ситуация, когда решение задачи не будет найдено при заданных значениях входных данных. В связи с этим специалисту необходимо анализировать входные данные и корректировать входную экспертную информацию. Это делает процесс решения трудоемким.

В качестве средства преодоления данной трудности может быть рассмотрен подход, основанный на представлении обратной задачи по определению изменений аргументов с учетом экспертной информации в виде оптимизационной.

5.1.1 Оптимизационная модель решения обратной задачи при максимальном соответствии целеполаганию в виде заданной экспертной информации

Обратная задача по определению изменений аргументов при наличии коэффициентов относительной важности может быть представлена в следующем виде [152]

$$g(\Delta x) = \sum_{i=1, i \neq j}^{n} \left(\Delta x_j \pm \Delta x_i \frac{\beta_j}{\beta_i} \right)^2 \to \min,$$
$$f(\Delta x) = y + \Delta y. \tag{5.1}$$

Целевая функция отражает соответствие полученного решения коэффициентам относительной важности, а ограничение – равенство результирующего показателя заданному значению.

Элементы целевой функции сформированы из уравнений системы, которая формируется в классическом аппарате обратных вычислений (1.23)

$$\frac{\Delta x_i}{\Delta x_j} = \frac{\beta_i}{\beta_j}, i = 1..n, i \neq j$$

Знак в скобке целевой функции g (Δx) (5.1) говорит о направлении изменений аргументов. Если изменения аргументов $\Delta x_j, \Delta x_i$ должны иметь одно направление (положительное или отрицательное), то применяется знак «минус» ($\Delta x_j - \Delta x_i \frac{\alpha_j}{\alpha_i}$), в противном случае – знак «плюс» ($\Delta x_j + \Delta x_i \frac{\alpha_j}{\alpha_i}$).

Таким образом, оптимизационная модель решения обратных задач позволяет осуществлять формирование значения ключевого показателя при максимизации соответствия экспертным целеполаганиям.

5.1.2 Алгоритм решения оптимизационной задачи при использовании коэффициентов относительной важности

Для устранения недостатков классических методов, связанных с формированием и оптимизацией модифицированной функции, с помощью которых может быть выполнено решение задачи (5.1), были разработаны два итерационных алгоритма, в основе которых лежит итерационная формула изменения аргумента (п.3.2.6). Однако данные алгоритмы имеют ограничение, согласно которому частные производные целевой функции должны быть одномерными функциями. Для задачи (5.1) данное условие не выполняется, поэтому необходимо выполнить модификацию алгоритма.

Алгоритм решения задачи (5.1) будет включать следующие шаги:

Шаг 1. Установить начальное значение изменения j-го аргумента Δx_j, номер итерации k=0, шаг h изменения переменной Δx_j.

Шаг 2. Выполнить безусловную оптимизацию целевой функции $g(\Delta x)$ при фиксированном значении Δx_j. Полученные значения Δx используются для дальнейшей корректировки.

Шаг 3. Используя итерационную формулу выполнить изменение значений аргументов ($i \neq j$)

$$\Delta x_i^* = \Delta x_i + t \cdot \alpha \cdot \frac{\dfrac{\partial f(x + \Delta x)}{\partial \Delta x_i}}{\dfrac{\partial^2 g(\Delta x)}{\partial x_i^2}}. \tag{5.2}$$

Шаг 4. Проверка: если $d(\Delta x_i^*) > d(\Delta x_i)$ (где $d(\Delta x)=\left|f(\Delta x)-\left(y+\Delta y\right)\right|$),

то $k = k+1$, $g_k = g(\Delta x)$, переход на шаг 5. Иначе: $\Delta x_i = \Delta x_i^*$, переход на шаг 3.

Шаг 5. Проверка: если $k > 1$ и $g_k > g_{k-1}$, то завершение работы алгоритма, иначе изменение переменной Δx_j на шаг h, переход на шаг 2.

В качестве решения задачи принимаются значения Δx на k–1 шаге.

Величина шага h определяет точность полученного решения, с увеличением величины шага ошибка вычислений также будет увеличиваться.

В случае решения задачи при условии целочисленности одного из аргументов, данный аргумент рассматривается в качестве базового (j-го) в алгоритме, а величина шага является целым числом. Решение задачи, когда условие целочисленности накладывается на два и более аргументов, может быть выполнено с помощью алгоритма на основе метода ветвей и границ [153–155]:

Шаг 1. Решение обратной задачи без учета условия целочисленности аргументов.

Шаг 2. Если в варианте/вариантах решения есть переменные, не удовлетворяющие условию целочисленности, то переход к следующему шагу, иначе – переход на шаг 5.

Шаг 3. Выполнить округление переменной, не удовлетворяющей условию целочисленности, в большую и меньшую сторону и для каждого варианта выполнить решение задачи (5.1).

Шаг 4. Решения, не удовлетворяющие заданному уровню точности вычисления f, исключаются. Возврат на шаг 2.

Шаг 5. Из рассмотренных вариантов решения выбирается тот, который обеспечивает минимум целевой функции $g(\Delta x)$.

5.1.3 Применение алгоритма для решения задач при использовании коэффициентов относительной важности

Рассмотрим решение задачи с мультипликативно-аддитивной моделью (таблица 1.2). Задача оптимизации имеет вид

$$\begin{gathered} g(\Delta x) = \left(\Delta x_1 - 0,4\frac{\Delta x_2}{0,3}\right)^2 + \left(\Delta x_1 - 0,4\frac{\Delta x_3}{0,1}\right)^2 + \left(\Delta x_1 - 0,4\frac{\Delta x_4}{0,2}\right)^2 \\ f(\Delta x) = (100 + \Delta x_1)\cdot(50 + \Delta x_2) - (500 + \Delta x_3) - (1000 + \Delta x_4) = 4500 \\ \Delta x_1 \in \mathbb{Z} \end{gathered} \quad (5.3)$$

Так, решение задачи безусловной оптимизации функции $g(\Delta x)$ (5.3) при изменении значений Δx_1 с шагом, равным 1, представлено в таблице 5.1.

Величина t определяется следующим образом: после решения задачи безусловной оптимизации вычисляется значение функции $f(\Delta x)$. Если полученное значение меньше заданной величины 4500, то t=1 (значение функции нужно увеличить, выполнить движение в направлении градиента), иначе t=–1 (значение функции нужно уменьшить, выполнить движение в направлении антиградиента).

Таблица 5.1 – Решение задачи безусловной оптимизации функции $g(\Delta x)$

Δx_1	Δx_2	Δx_3	Δx_4
1	0,75	0,25	0,5
2	1,5	0,5	1
…	…	…	…
6	4,5	1,5	3
7	5,25	1,75	3,5
8	6	2	4
9	6,75	2,25	4,5

Для формирования итерационных формул необходимо вычислить вторые частные производные целевой функции и частные производные функции ограничения f. Итерационные формулы изменения переменных x_2, x_3, x_4 имеют следующий вид

$$\Delta x_2^* = \Delta x_2 + t \cdot \alpha \frac{(\Delta x_1 + 100)}{3,5556},$$

$$\Delta x_3^* = \Delta x_3 + t \cdot \alpha \frac{-1}{32},$$

$$\Delta x_4^* = \Delta x_4 + t \cdot \alpha \frac{-1}{8}.$$

В таблице 5.2 представлено решение задачи.

Таблица 5.2 – Решение задачи с помощью итерационного алгоритма ($\alpha = 10^{-10}$)

Δx_1	Δx_2	Δx_3	Δx_4	$g(\Delta x)$	d
1	9,413	0,24	0,462	133,422	10^{-6}
2	8,838	0,492	0,968	95,728	$4,5 \cdot 10^{-7}$
…	…	…	…	…	…
6	6,646	1,498	2,991	8,188	$8,3 \cdot 10^{-7}$
7	6,124	1,749	3,496	1,357	$1,2 \cdot 10^{-6}$
8	5,611	2	4,002	0,269	$1,6 \cdot 10^{-6}$
9	5,108	2,252	4,507	4,794	$4,2 \cdot 10^{-7}$

Величина $g(\Delta x)$ принимает минимальное значение при следующих величинах Δx : Δx_1=8, Δx_2=5,611, Δx_3=2, Δx_4=4,002.

В отличие от итерационного алгоритма, основанного на изменении аргументов пропорционально коэффициентам относительной важности, данный алгоритм позволяет найти решение, при котором результирующее значение соответствует заданной величине. При этом пропорции изменения аргументов, определяемые коэффициентами относительной важности, корректируются для достижения цели.

Так в случае, если значение аргумента x_1 в мультипликативно-аддитивной модели необходимо уменьшить, то с помощью классического аппарата обратных вычислений решение не будет найдено. При использовании же рассмотренного алгоритма будет выполнена корректировка коэффициентов для достижения заданного значения функции. На рисунке 5.1 представлены полученные пропорции β изменения аргументов.

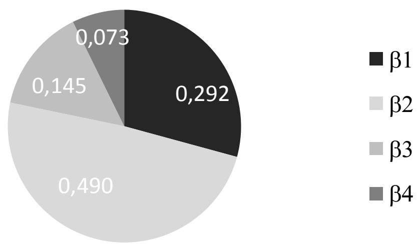


Рисунок 5.1 – Пропорции изменения аргументов

В таблице 5.3 представлено использование разработанного алгоритма для решения задачи с использованием тестового набора моделей (таблица 1.2) при условии, что x_1 является целочисленным.

Таблица 5.3 – Решение задач для тестовых моделей с помощью итерационного алгоритма ($\alpha = 10^{-10}$)

Модель	Показатель	Предложенный алгоритм	Функция MathCad
Аддитивная	Δx_1	68	68
	Δx_2	34	34
	Δx_3	17	17
	Δx_4	34	34
	Δx_5	17	17

Модель	Показатель	Предложенный алгоритм	Функция MathCad
	g	0	0
	d	0	0
Мультипликативная	Δx_1	1	1
	Δx_2	0,647997	0,647993
	Δx_3	0,38408	0,383923
	Δx_4	0,213238	0,213312
	g	0,0938629	0,0938626
	d	$3{,}4\cdot 10^{-5}$	$8\cdot 10^{-6}$
Мультипликативно-аддитивная	Δx_1	10	10
	Δx_2	4,74915	4,74915
	Δx_3	7,466365	7,466364
	Δx_4	14,940084	14,940084
	g	0,1281332	0,1281328
	d	$6{,}5\cdot 10^{-8}$	$3{,}2\cdot 10^{-5}$
Нелинейная аддитивная	Δx_1	–1	–1
	Δx_2	–1,12877	–1,143058
	Δx_3	–0,54409	–0,535052
	g	0,137964	0,137675
	d	$1{,}3\cdot 10^{-8}$	$4{,}3\cdot 10^{-7}$
Мультипликативная нелинейная	Δx_1	1	1
	Δx_2	1,935324	1,935327
	g	0,084226	0,084227
	d	$5{,}4\cdot 10^{-9}$	$4{,}2\cdot 10^{-6}$

5.2 Решение обратной задачи при использовании переменных в расчёте нескольких показателей

5.2.1 Оптимизационная модель решения обратной задачи при использовании показателей в нескольких функциях расчёта

В экономических системах могут встречаться задачи, когда зависимость между показателями носит иерархический характер, а отдельные переменные могут участвовать в расчете разных показателей. Так в работах [48–49] приведена иерархическая структура обратной задачи, которая предполагает разбиение задачи на подзадачи и переход от одного уровня к другому. Таким образом, использование аргументов в расчёте нескольких показателей возникает при рассмотрении многоуровневых задач, решение которых выполняется пошагово путем прямого и обратного хода дерева цели.

Представленный в п.1.3.5 метод является трудоемким и не позволяет выполнять решение задачи в случае, когда несколько переменных используется при расчёте разных показателей.

В связи с этим рассмотрим представление обратной задачи в виде оптимизационной [156–157]

$$\begin{aligned} & g(\Delta x) \to \min, \\ & f_i(\Delta x) = y_i + \Delta y_i, \quad i = 1..r. \end{aligned} \tag{5.4}$$

Здесь целевая функция представляет собой сумму квадратов изменений аргументов ($g(\Delta x) = \sum_{i=1}^{n} \Delta x_i^2$) либо сумму модулей изменений аргументов ($g(\Delta x) = \sum_{i=1}^{n} \left|\Delta x_i\right|$) либо отражает минимизацию отклонений от заданных коэффициентов относительной важности.

В случае использования коэффициентов относительной важности может быть рассмотрено два способа решения задачи.

Первый способ заключается в формировании целевой функции g, характеризующей степень отклонения соотношения полученных изменений аргументов от соотношения коэффициентов относительной важности, установленных экспертом

$$\begin{aligned} & g(\Delta x) = \sum_{k=1}^{r} \sum_{i=2}^{n_r - 1} \left(\Delta x_q \pm \Delta x_i \frac{\beta_{kq}}{\beta_{ki}} \right)^2 \to \min, \\ & f_i(\Delta x) = y_i + \Delta y_i, \quad i = 1..r, \end{aligned} \tag{5.5}$$

где n_r – число аргументов, участвующих в формировании показателя r;

q – номер аргумента, выбранного в качестве базового.

Таким образом, для всех подзадач формируется единая задача нелинейного программирования. Выражение в скобках получено исходя из уравнений системы обратных вычислений. Знак в скобках указывает на направление изменения аргументов: знак минус говорит о том, что изменение должно происходить в одном направлении, знак плюс – в разных. Каждое уравнение-ограничение соответствует формируемому показателю.

Второй способ заключается в отдельном решении каждой подзадачи

$$g_u\left(\Delta x\right)=\sum_{i=2}^{n_{r-1}}\left(\Delta x_q \pm \Delta x_i \frac{\beta_{kq}}{\beta_{ki}}\right)^2 \rightarrow \min,$$
$$f_u\left(\Delta x\right)=y_u+\Delta y_u, \qquad (5.6)$$
$$u=1..r.$$

Далее выполняется последующая корректировка аргументов при минимизации суммы квадратов изменений аргументов.

Таким образом, оптимизационные модели решения обратных задач позволяют осуществлять формирование значения ключевого показателя при использовании показателей в нескольких функциях расчёта.

5.2.2 Алгоритм решения обратной задачи при использовании показателей в нескольких функциях расчёта

Для минимизации суммы квадратов изменений аргументов (5.4) и использования разработанного ранее алгоритма решения обратной задачи необходимо привести задачу (5.4) к задаче с одним ограничением. Рассмотрим также случай, когда в задаче имеется одно ограничение неравенство: $h\left(\Delta x\right)\geq/\leq y_j+\Delta y_j$.

Алгоритм решения задачи при минимизации суммы квадратов изменений аргументов включает следующие шаги:

Шаг 1. Преобразование ограничений-равенств f в одно ограничение $f^*(x)$

$$f^*\left(\Delta x\right)=\sum_{i=1}^{r}\left(f_i\left(\Delta x\right)-y_i-\Delta y_i\right)^2. \qquad (5.7)$$

Шаг 2. Решение задачи с одним ограничением (5.8):

$$g(\Delta x) \to \min,$$
$$f^*(\Delta x) = \sum_{i=1}^{r} \left(f_i(\Delta x) - y_i - \Delta y_i \right)^2 = 0.$$

Для решения задачи используются итерационные формулы (α – малое число, обеспечивающее постепенное изменение, t – показатель направления изменения, принимающий значения –1 или 1)

$$\Delta x_i = \Delta x_i + t \cdot \alpha \cdot \frac{\partial f^*(\Delta x_i)}{\partial \Delta x_i}. \tag{5.8}$$

Шаг 3. Если полученное решение Δx удовлетворяет ограничению-неравенству f, то работа алгоритма завершается.

Шаг 4. Выполнить замену знака неравенства на знак равенства и сформировать одно ограничение-равенство f^*

$$f^*(\Delta x) = \sum_{i=1}^{r} \left(f_i(\Delta x) - y_i - \Delta y_i \right)^2 + \left(h(\Delta x) - y_j - \Delta y_j \right)^2.$$

Решение задачи с одним ограничением

$$g(\Delta x) \to \min,$$
$$f^*(\Delta x) = \sum_{i=1}^{r} \left(f_i(\Delta x) - y_i - \Delta y_i \right)^2 + \left(h(\Delta x) - y_j - \Delta y_j \right)^2.$$

Решением задачи будут полученные значения Δx.

При минимизации суммы модулей изменений аргументов аналогичным образом происходит формирование одного ограничения и использование итерационных формул (п.3.3.1).

Алгоритм решения задачи при использовании коэффициентов важности (5.5) и формировании единой целевой функции также включает её преобразование в задачу с одним ограничением и использование итерационных формул. При этом происходит изменение Δx_q с некоторым шагом, для каждого из вариантов осуществляется решение по итерационным формулам. При этом в качестве исходных значений Δx используются величины, полученные путем безусловной оптимизации целевой функции, которые могут быть вычислены по формулам

$$\Delta x_i = \mp\Delta x_q \cdot \frac{\beta_{ki}}{\beta_{kq}}.$$

В итерационных формулах (5.8) необходимо учесть влияние изменения аргумента на изменение целевой функции

$$\Delta x_i = \Delta x_i + t \cdot \alpha \cdot \frac{\dfrac{\partial f\left(\Delta x_i\right)}{\partial \Delta x_i}}{\dfrac{\partial g^2\left(\Delta x_i\right)}{\partial \Delta x_i^2}}.$$

Алгоритм решения задачи при использовании коэффициентов относительной важности и отдельном формировании каждого показателя (5.6) будет включать следующие шаги:

Шаг 1. Решение обратной задачи для каждого формируемого показателя по определению x^* (задача с одним ограничением) (5.6).

Шаг 2. Решение задачи по определению изменений аргументов

$$g\left(\Delta x^*\right) = \sum_{j=1}^{n}\left(\Delta x_j^*\right)^2 \to \min,$$

$$f_i\left(\Delta x^*\right) = y_i^*, \quad i = 1..r.$$

где n – общее число характеристик, участвующих в формировании показателей.

В качестве исходных значений x выступают величины, полученные на шаге 1, а для величин, которые были вычислены в нескольких подзадачах, определяется среднее значение.

5.2.3 Применение алгоритма для решения задачи формирования маржинальной прибыли

В качестве примера использования разработанного алгоритма рассмотрим решение задачи формирования маржинальной прибыли

$$L = \sum_{j=1}^{m} p_j,$$

где L – суммарная маржинальная прибыль;

p_j – маржинальная прибыль j-й торговой точки.

Маржинальная прибыль отдельной торговой точки вычисляется по формуле

$$p_j = \sum_{i=1}^{n} s_{ji} \cdot w_i \cdot r_i,$$

где s_{ji} – доля распределения i-й продукции для j-й торговой точки;

w_i – объем закупки i-й продукции;

r_i – маржинальная прибыль от продажи единицы i-го изделия.

Задача заключается в определении объема w_i закупки каждого вида товара для достижения заданного значения суммарной маржинальной прибыли. При этом существует ограничение на бюджет закупок Q

$$Q = \sum_{i=1}^{n} w_i \cdot q_i,$$

где q_i – стоимость закупки изделия i-го вида.

Рассмотрим решение задачи для двух торговых точек и трех видов продукции. Дерево цели представлено на рисунке 5.2.

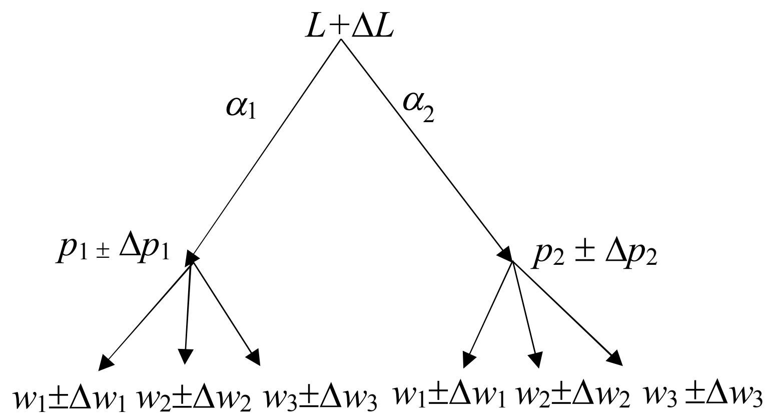


Рисунок 5.2 – Дерево цели задачи формирования маржинальной прибыли

Таким образом, решение задачи включает два этапа:

1) Определение изменение прибыли Δp_1, Δp_2 в двух точках для достижения заданного значения суммарной прибыли $L+\Delta L$.

2) Определение изменения объема закупок Δw_1, Δw_2, Δw_3 продукции первого, второго и третьего вида для достижения полученных на предыдущем этапе величин прибыли первой и второй торговых точек.

Исходные данные о продукциях представлены в таблицах 5.4, 5.5, исходное значение маржинальной прибыли равно 665 д.е., заданное значение маржинальной прибыли равно 720 д.е., ограничение на бюджет закупок составляет 1600 д.е.

Таблица 5.4 – Исходные данные о продукциях

Показатель	Продукция 1	Продукция 2	Продукция 3
Маржинальная прибыль, r	2	3	2,5
Объем закупки, w	100	80	90
Стоимость закупки, q	6	5	5,5

Исходные значения маржинальной прибыли для первой и второй торговой точки составляют соответственно 196,5 д.е. и 468,5 д.е.:

$100 \cdot 0{,}3 \cdot 2 + 80 \cdot 0{,}1 \cdot 3 + 90 \cdot 0{,}5 \cdot 2{,}5 = 196{,}5$,

$100 \cdot 0{,}7 \cdot 2 + 80 \cdot 0{,}9 \cdot 3 + 90 \cdot 0{,}5 \cdot 2{,}5 = 468{,}5$.

Решение задачи первого этапа выполним при коэффициентах относительной важности 0,427 и 0,573 и положительном направлении изменения показателей. Для первой и второй точки соответственно значения изменений прибыли Δp будут равны $(720–665) \cdot 0{,}427 = 23{,}5$ и $(720 – 665) \cdot 0{,}573 = 31{,}5$. Соответственно новые значения маржинальной прибыли для первой и второй точки: 220 д.е. и 500 д.е.

На втором этапе задача представляется в виде оптимизационной

$$g(\Delta x) = \sum_{j=1}^{n} \Delta w_j^2 \to \min,$$

$$f_1(\Delta w) = (100 + \Delta w_1) \cdot 0{,}3 \cdot 2 + (80 + \Delta w_2) \cdot 0{,}1 \cdot 3 + (90 + \Delta w_3) \cdot 0{,}5 \cdot 2{,}5 = 220,$$

$$f_2(\Delta w) = (100 + \Delta w_1) \cdot 0{,}7 \cdot 2 + (80 + \Delta w_2) \cdot 0{,}9 \cdot 3 + (90 + \Delta w_3) \cdot 0{,}5 \cdot 2{,}5 = 500,$$

$$h(\Delta w) = (100 + \Delta w_1) \cdot 6 + (80 + \Delta w_2) \cdot 5 + (90 + \Delta w_3) \cdot 5{,}5 \le 1600.$$

Таблица 5.5 – Исходные данные о схеме распределения продукции по торговым точкам

Номер торговой точки	Доля распределения продукции 1	Доля распределения продукции 2	Доля распределения продукции 3
1	0,3	0,1	0,5
2	0,7	0,9	0,5

Рассмотрим использование представленного выше алгоритма для решения задачи. На начальном этапе происходит формирование задачи с одним ограничением

$$g(\Delta x) = \sum_{j=1}^{n} \Delta w_j^2 \to \min,$$

$$f(\Delta w) = \left((100 + \Delta w_1)\cdot 0,3\cdot 2 + (80 + \Delta w_2)\cdot 0,1\cdot 3 + (90 + \Delta w_3)\cdot 0,5\cdot 2,5 - 220\right)^2 +$$
$$\left((100 + \Delta w_1)\cdot 0,7\cdot 2 + (80 + \Delta w_2)\cdot 0,9\cdot 3 + (90 + \Delta w_3)\cdot 0,5\cdot 2,5 - 500\right)^2.$$

Решение задачи выполняется с использованием итерационного алгоритма.

Решение задачи ($\alpha = 10^{-8}$): Δw_1=6,538, Δw_2=1,154, Δw_3=15,385.

Ограничение h не выполняется, поэтому происходит формирование новой функции-ограничения

$$f(\Delta w) = \left((100+\Delta w_1)\cdot 0,3\cdot 2 + (80+\Delta w_2)\cdot 0,1\cdot 3 + (90+\Delta w_3)\cdot 0,5\cdot 2,5 - 220\right)^2 +$$
$$+\left((100+\Delta w_1)\cdot 0,7\cdot 2 + (80+\Delta w_2)\cdot 0,9\cdot 3 + (90+\Delta w_3)\cdot 0,5\cdot 2,5 - 500\right)^2 +$$
$$+\left((100+\Delta w_1)\cdot 6 + (80+\Delta w_2)\cdot 5 + (90+\Delta w_3)\cdot 5,5 - 1600\right)^2.$$

Решение задачи с использованием итерационного алгоритма: Δw_1= – 5, Δw_2=5, Δw_3=20.

Значение целевой функции f равно 449,9994. Значение целевой функции при решении задачи с помощью математического пакета MathCad оставило 450,002.

Рассмотрим теперь случай формирования прибыли с учетом коэффициентов относительной важности. Исходные данные задачи приведены в таблице 5.6. Направление изменения аргументов: положительное для объема выпуска, и отрицательное для себестоимости и цены.

Таблица 5.6 – Исходные данные задачи формирования прибыли с использованием коэффициентов важности

Показатель	Исходное значение	Формируемое значение
Себестоимость единицы продукции, д.е., x_2	2	–
Объем выпуска, усл.ед., x_1	10	–
Цена, д.е., x_3	5	–
Себестоимость продаж, д.е.	20	15
Выручка, д.е.	50	47
Коэффициент важности α_{12}	0,2	–
Коэффициент важности α_{11}	0,8	–
Коэффициент важности α_{21}	0,7	–
Коэффициент важности α_{23}	0,3	–

Для решения задачи первым способом необходимо сформировать целевую функцию, задача оптимизации будет иметь вид

$$\begin{aligned} & g(\Delta x) = \left(\Delta x_1 \quad + \Delta x_2 \frac{0,8}{0,2} \right)^2 + \left(\Delta x_1 \quad + \Delta x_3 \frac{0,7}{0,3} \right)^2 \to \min, \\ & (x_1 + \Delta x_1) \cdot (x_2 + \Delta x_2) = 15, \\ & (x_1 + \Delta x_1) \cdot (x_3 + \Delta x_3) = 47. \end{aligned} \tag{5.9}$$

Решение задачи путем преобразования ограничений и изменения аргумента Δx_1 с шагом 0,1: $\Delta x_{1=}2,6$; $\Delta x_{2=}– 0,81$; $\Delta x_{3=}– 1,26$.

На рисунке 5.3 представлен график изменения целевой функции при увеличении Δx_1.

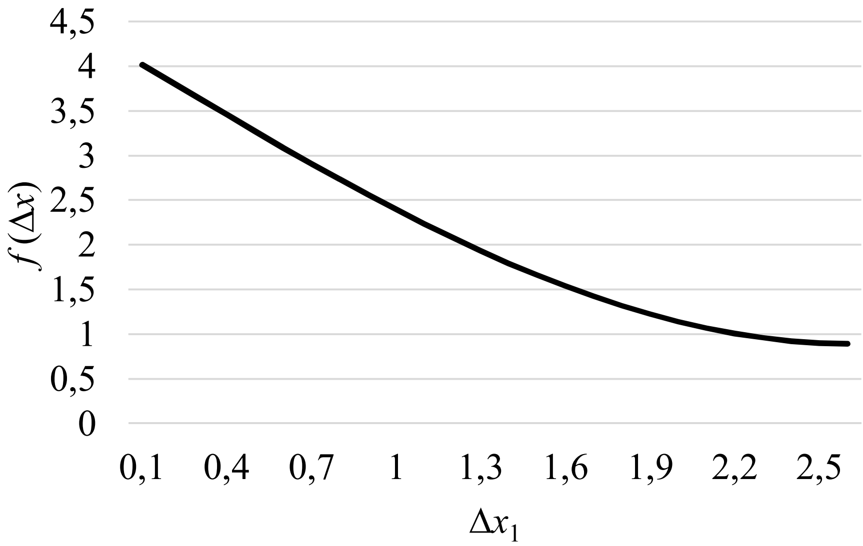


Рисунок 5.3 – Изменение целевой функции

При использовании второго способа на первом шаге получаются следующие значения:

– задача формирования себестоимости: $\Delta x_{1,1}$=3,583; Δx_2=– 0,896.

– задача формирования выручки: $\Delta x_{1,2}$=1,212; Δx_3=– 0,81.

Далее решается задача оптимизации при минимизации суммы изменений аргументов

$$g(\Delta x) = \left(\Delta x_1^*\right)^2 + \left(\Delta x_2^*\right)^2 + \left(\Delta x_3^*\right)^2 \to \min,$$
$$\left(x_1 + \frac{\Delta x_{1,1} + \Delta x_{1,2}}{2} + \Delta x_1^*\right) \cdot \left(x_2 + \Delta x_2 + \Delta x_2^*\right) = 15, \quad (5.10)$$
$$\left(x_1 + \frac{\Delta x_{1,1} + \Delta x_{1,2}}{2} + \Delta x_1^*\right) \cdot \left(x_3 + \Delta x_3 + \Delta x_3^*\right) = 47.$$

Решение задачи: Δx_1^*=– 0,103; Δx_2^*=0,116; Δx_3^*=– 0,36.

В таблице 5.7 представлено решение задачи с использованием двух алгоритмов и при использовании математического пакета MathCad. Также представлено решение, полученное с помощью имитационного алгоритма, воспроизводящего действия специалиста согласно алгоритму, приведенному в [43].

Таблица 5.7 – Результаты решения задачи формирования прибыли

Алгоритм	x_1	x_2	x_3
Формирование единой задачи для двух подзадач, $\alpha=10^{-7}$	12,6	1,19	3,73
Минимизация отклонения от решения каждой задачи, $\alpha=10^{-7}$	12,29	1,22	3,82
Имитационный алгоритм	11,21	1,1	4,19
Использование MathCad для решения задачи (5.9)	12,61	1,19	3,73
Использование MathCad для решения задачи (5.10)	12,29	1,22	3,82

На рисунке 5.4 представлено значение евклидовой нормы для отклонений полученного соотношения изменений аргументов от установленных коэффициентов относительной важности для трех алгоритмов.

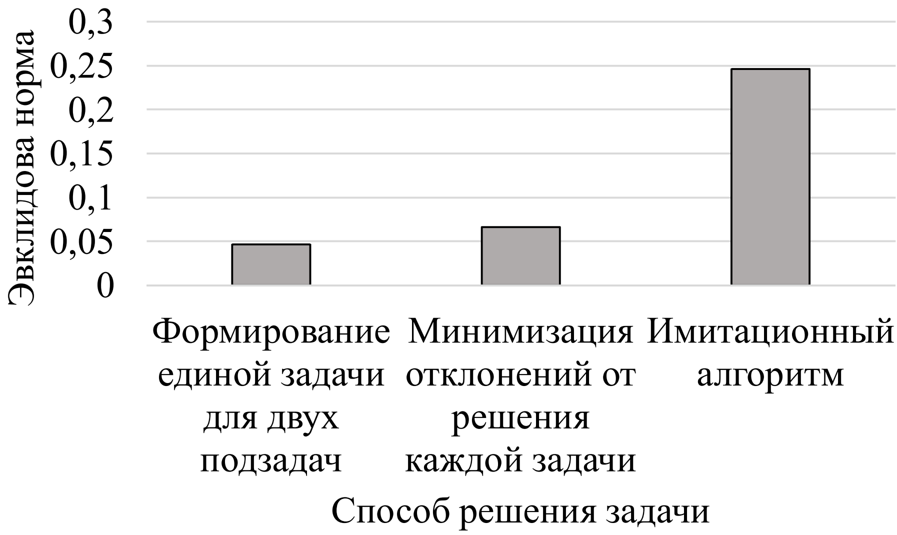


Рисунок 5.4 – Значение евклидовой нормы для задач

Согласно представленным на рисунке 5.4 значениям алгоритм на основе формирования единой задачи обеспечивает наилучшее соответствие заданным коэффициентам относительной важности.

5.2.4 Применение алгоритма для решения задачи формирования риска

Устойчивое функционирование промышленных предприятий во многом зависит от обеспечения ресурсом (финансовые и материальные средства, время, персонал и т.д.), направляемого для поддержания и развития системы комплексной безопасности (СКБ) созданной на предприятии, имеющего предел ограничений в удовлетворении полного объема запросов от руководителей служб (отделов) – кураторов направлений комплексной безопасности. При реализации задач в таких условиях, необходима научная проработка в адресном обеспечении ресурсом тех направлений комплексной безопасности, которые имеют высокий рисковый показатель уязвимости [156].

Сегодня разработка и развитие методологии безопасного управления СКБ, созданной на промышленном предприятии (человеческий фактор), становится актуальной и востребованной. Более 90% аварий на объектах атомной энергетики, более 80 % аварий на объектах химической и нефтехимической промышленности, более 75% аварий при шельфовой нефтедобыче и более 70% авиационных аварий связаны с человеческим фактором.

С точки зрения реализации опасностей на промышленных предприятиях рассматривались пожары. На основании запроса проводился анализ возникновения пожаров на предприятиях электроэнергетики за период 3 лет (2017 – 2019 гг.). Обработка статистических данных о пожарах на электроэнергетических предприятиях, позволила установить перечень причин возникновения пожаров (обозначены соответствующими причинными кодами, их расшифровка представлена в таблице 5.8).

Таблица 5.8 – Причины возникновения пожаров

Код причины возникновения пожаров	Причина возникновения пожара
8	Прочие причины, связанные с неисправностью производственного оборудования
9	Недостаток конструкции и изготовления электрооборудования

Код причины возникновения пожаров	Причина возникновения пожара
10	Нарушение правил монтажа электрооборудования
11	Нарушение правил технической эксплуатации (ПТЭ) электрооборудования
12	Нарушение правил пожарной безопасности (ППБ) при эксплуатации бытовых электроприборов
13	Нарушение ППБ при проведении электрогазосварочных работ
15	Самовозгорание веществ и материалов
17	Нарушение ППБ при эксплуатации печей
20	Нарушение ППБ при эксплуатации теплогенерирующих агрегатов и устройств
25	Прочие причины, связанные с неосторожным обращением с огнем
28	Прочие причины, не относящиеся ни к одной из групп
32	Нарушение ПТЭ и выбора аппаратов защиты электрических сетей
33	Прочие причины, связанные с нарушением правил устройства и эксплуатации электрооборудования

Рассмотрим пример, связанный с оценкой практического воздействия служб на СКБ созданную в крупной электроэнергетической компании ПАО «Мосэнерго». Приглашались эксперты для определения факторных причин воздействий служб (электроэнергетической, промышленной безопасности, пожарной безопасности, охраны труда) (далее – «службы») на отраслевую подсистему пожарной безопасности.

Были определены следующие оценки показателей воздействия «служб» (прямое воздействие – 0,43; непосредственное воздействие – 0,28; опосредованное воздействие – 0,19; косвенное воздействие – 0,1) (таблица 5.9).

Таблица 5.9 – Оценки влияния «служб» промышленного предприятия на подсистемы безопасности

Код причины возникновения пожаров	Электротехническая служба	Служба промышленной безопасности	Служба пожарной безопасности	Служба охраны труда
8	0,1	0,1	0,1	0,1
9	0,19	0,19		
10	0,1	0,1		
11	0,43	0,1		
12	0,1		0,43	
13	0,1		0,43	
15	0,19		0,1	0,1
17	0,19		0,1	0,1
20	0,19		0,1	0,1
25	0,1	0,1	0,1	0,1
28	0,1	0,1	0,1	0,1
32	0,43	0,1		

Код причины возникновения пожаров	Электротехническая служба	Служба промышленной безопасности	Служба пожарной безопасности	Служба охраны труда
33	0,43			0,1

Оценка риска возникновения пожаров по каждой из факторных групп вычисляется по формуле:

$$r_i = \sqrt{\frac{\sum_{j=1}^{n}(q_{ji})^2}{n}},$$

где q_{ji} – количество реализованных опасностей в -м году, вызванных причиной их возникновения - i;

r_i– риск возникновения пожара по -й причине;

n– число рассматриваемых лет.

Далее, при умножении полученных весовых коэффициентов на полученные значения риска возникновения пожаров r_i, будет получена оценка деятельности каждой из «служб» (электроэнергетической, промышленной безопасности, пожарной безопасности, охраны труда) по формуле:

$$I_k = \sum_{i-1}^{m} r_i \cdot \beta_i,$$

где I_k – показатель деятельности k-й *«службы»*;

m – число причин возникновения пожаров.

Суммарный показатель деятельности всех *«служб»* вычислялся по формуле:

$$I = \sum_{k=1}^{S} I_k,$$

где S– количество служб.

На рисунке 5.5 представлен в динамике показатель работы всех «служб», где просматривается их положительная динамика (за исключением службы пожарной безопасности), при этом наибольшее изменение отнесено к работе электротехнической службы.

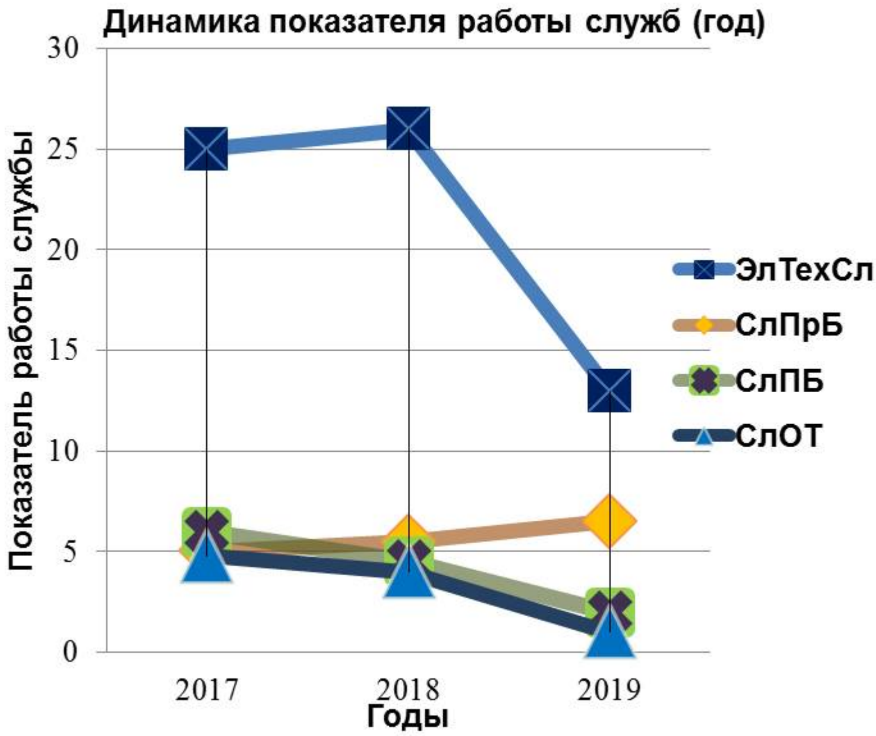


Рисунок 5.5 – Динамика показателя работы служб

Для определения необходимых воздействий для достижения желаемого состояния изучаемого объекта решались задачи на основе обратных вычислений.

При решении первой задачи определялся показатель работы «служб» для уменьшения общего риска. Была сформирована система уравнений:

$$\begin{cases} (r_1 - \Delta r_1) + (r_2 - \Delta r_2) + \ldots (r_{13} - \Delta r_{13}) = R - \Delta R; \\ \dfrac{\Delta r_1}{\Delta r_2} = \dfrac{\alpha_1}{\alpha_2}; \\ \dfrac{\Delta r_1}{\Delta r_3} = \dfrac{\alpha_1}{\alpha_3}; \\ \ldots \\ \dfrac{\Delta r_1}{\Delta r_{13}} = \dfrac{\alpha_1}{\alpha_{13}}, \end{cases}$$

где α_i – коэффициент относительной важности i-й факторной группы;

Δr_i – изменение риска по i-й группе;

ΔR – изменение суммарного риска.

Показатели для изменения рисков по каждой из факторных групп, отнесенных к «службам», представлены в таблице 5.10. На рисунке 5.6 представленны полученные изменения показателей работы служб.

Таблица 5.10 – Решение задачи по определению изменения величин риска

Код причины возникновения пожара	Исходные значения риска	Доля риска в общем сумме	Изменение риска пропорционально его доле в общей сумме	Равномерное изменение риска
8	3,46	0,04	-0,40	-0,77
9	19,07	0,22	-2,22	-0,77
10	4,62	0,05	-0,54	-0,77
11	8,72	0,10	-1,02	-0,77
12	2,08	0,02	-0,24	-0,77
13	3,11	0,04	-0,36	-0,77
15	2,58	0,03	-0,30	-0,77
17	2,45	0,03	-0,29	-0,77
20	2,38	0,03	-0,28	-0,77
25	4,51	0,05	-0,53	-0,77
28	4,69	0,05	-0,55	-0,77
32	2,08	0,02	-0,24	-0,77
33	26,13	0,30	-3,04	-0,77
Итого	85,89	1	-10,00	-10

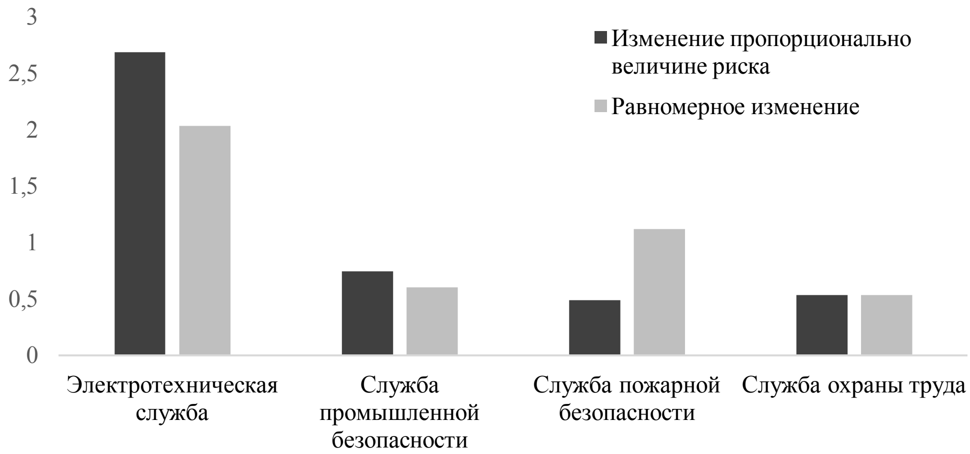


Рисунок 5.6 – Изменение показателей работы служб

На основе полученных значений разрабатываются меры по снижению риска.

5.3 Выводы по главе 5

1) Предложены оптимизационные модели для решения обратной задачи с использованием коэффициентов относительной важности. В отличие от существующих работ использование оптимизационной модели позволяет найти решение, соответствующее заданному значению результирующего показателя в случае, когда коэффициенты относительной важности не позволяют достичь поставленной цели и требуется их корректировка. Кроме того, использование представленной модели позволяет упростить решение задачи экспертом ввиду отсутствия необходимости указания направления изменения каждого показателя, вместо этого необходимо обозначить, будут ли показатели изменяться в одном направлении или в разных направлениях. Это позволяет уменьшить ошибки по определению входных данных и трудоемкость предварительного анализа.

2) Разработан алгоритм решения обратной задачи с использованием коэффициентов относительной важности. Для этого выполнена модификация предложенного ранее итерационного алгоритма решения задач нелинейного программирования, включающего безусловную оптимизацию целевой функции и итерационную корректировку изменений аргументов до достижения заданного значения результирующего показателя.

3) Предложен алгоритм решения обратной задачи при участии переменных в расчете нескольких показателей и дополнительных ограничениях на их значения, основанный на её преобразовании к задаче нелинейного программирования с одним ограничением в виде равенства и дальнейшем использовании разработанных ранее методов и итерационных алгоритмов. При использовании коэффициентов относительной важности рассмотрено два способа решения задачи: формирование единой оптимизационной задачи для подзадач и корректировка решения подзадач при минимизации суммы квадратов изменений аргументов. Способ на основе формирования единой оптимизационной задачи обеспечил меньшее значение эвклидовой метрики, характеризующей отклонение

изменений аргументов от установленных значений коэффициентов важности. Это обуславливается выражением целевой функции, минимизацией отклонения от заданной экспертной информации. Алгоритм решения задачи с помощью корректировки решения подзадач не требует формирования целевой функции на основе данных подзадач и поэтому является более простым в реализации. Предложенные алгоритмы обеспечили меньшее значение эвклидовой метрики отклонений изменений аргументов от установленных значений коэффициентов важности по сравнению с существующим методом решения задачи. Эвклидова норма при использовании метода, представленного в [43], основанного на последовательном решении двух обратных задач с целью определения компромиссного варианта составила 0,25. Это значение в 5,25 и 3,7 раза больше по сравнению с эвклидовой метрикой при использовании алгоритма на основе формирования единой задачи и алгоритма на основе изолированного решения подзадач соответственно. Кроме того, предложенные алгоритмы не требуют движения по сети от одного показателя к другому и многократного решения обратной задачи на основе информации, поступающей от специалиста. Также в отличие от существующего метода алгоритмы позволяют определять решение задачи при использовании более одного аргумента в расчёте нескольких показателей. К недостаткам алгоритмов можно отнести необходимость вычисления градиента и вторых частных производных, что повышает сложность реализации. Выполнено решение с помощью разработанных алгоритмов двух обратных задач формирования прибыли: при минимизации суммы квадратов изменений аргументов и при использовании коэффициентов относительной важности. Результаты численного решения задач согласуются с результатами использования стандартной функции математического пакета MathCad, абсолютная разница в значениях целевой функции составила от $2{,}6\cdot 10^{-3}$ до $2{,}8\cdot 10^{-5}$.

Глава 6. Алгоритмы и структура программы решения обратных задач

Кроме разработки математического и алгоритмического обеспечения актуальной является задача разработки программного обеспечения, применение которого способно обеспечить автоматизацию вычислений и скрыть от пользователя сложные расчёты.

Для реализации методов и алгоритмов могут быть использованы стандартные пакеты (MathCad, Excel и т.д.) и языки программирования. Преимуществом использования стандартных пакетов является возможность хранения данных, просмотра всех формул расчёта, что повышает доверие к результатам вычислений, наличие встроенных функций для решения задач (решение систем уравнений, решение оптимизационных задач), а также для построения графиков. Кроме того, к преимуществам использования пакета Excel можно отнести то, что это программное обеспечение является распространенным и установлено на компьютере практически каждого специалиста. Однако при увеличении размерности задачи определение формул является трудоемкой работой, в которой сложно избежать ошибок, а, также, от пользователя требуется наличие знаний в области использования соответствующих пакетов и математического аппарата. Кроме того, реализация циклических алгоритмов представляет сложность и невозможна без применения встроенного языка VBA.

Разработка программы позволяет создать удобный интерфейс для пользователя и обеспечить реализацию сложных многоэтапных алгоритмов. Представленные в литературе программы решения обратных задач можно разделить на две группы: универсальные и специфические.

В монографии [40] приводится общая концепция разработки универсальной системы формирования решений. В частности, отмечено, что такие системы целесообразно создавать на основе программных оболочек, позволяющих без применения программирования настраивать процесс расчета. В качестве основных элементов такой системы указаны:

– база данных – хранилище экономической информации предприятия (данные о производственных показателях, финансах, основных фондах, оборотных средств, кадрах), используемой для принятия решений;

– база знаний, включающая модели формирования решений: дерево целей, дерево вероятностей и т.д.;

– модули приобретения значений и ввода данных, осуществляющие создание/редактирование деревьев целей и записей в базе данных;

– модуль обработки данных, предназначенный для обработки информации, хранящейся в базе данных, и выполнения предварительных расчетов;

– модуль обработки знаний, предназначенный для обработки каждого вида знаний;

– пользовательский интерфейс, обеспечивающий взаимодействие пользователя с системой.

Формирование решений с помощью программных оболочек будет включать следующие шаги: формализация проблемы, цели, критерии оценки альтернатив; постановка задачи и выбор модели знаний; наполнение системы данными и знаниями; анализ предложенного варианта решений. При этом для эксперта предлагается определить такие функции как правила, ввод, редактирование, а для пользователя ввод, выполнение, печать.

Разработка подобного рода систем является не всегда оправданной, т.к. требует значительных затрат ресурсов, а готовых решений на рынке не представлено. Кроме того, такие системы могут включать избыточные функции и не всегда позволяют рассматривать особенности исследуемой области, всех возможных правил системы, а их использование специалистами может быть затруднено из-за недостаточно понятного интерфейса.

Реализация же программного обеспечения под конкретную задачу позволит сделать интерфейс, более соответствующий предметной области и потребностям пользователя. Однако основным недостатком таких систем является их специфичность и сложность повторного использования.

Так, в качестве примера на рисунке 6.1 представлен интерфейс программы решения обратных задач с помощью обратных вычислений, разработанной автором совместно со студентом кафедры АСУ Крюковым А.С. [158]. Работа с программой включает выбор модели (мультипликативной, аддитивной, кратной, смешанной) и ввод входных данных, после чего будет отображен результат решения задачи. Такого рода программа обладает определенной универсальностью, т.к., например, мультипликативная модель может быть применена для расчёта целого ряда показателей: выручка, валовая продукция и т.д. Однако необходимо владеть математическим аппаратом, выявить соответствие расчёта показателя конкретной математической модели.

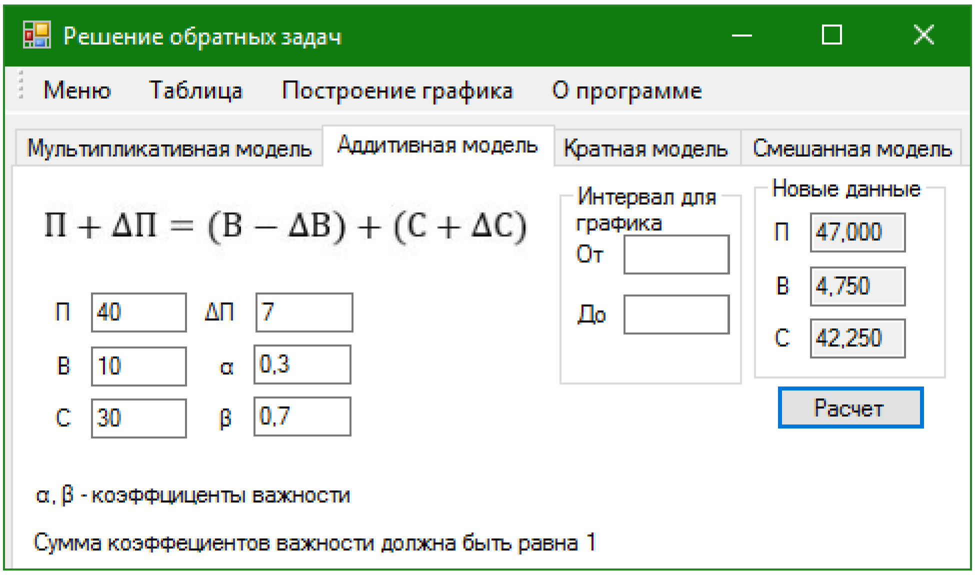


Рисунок 6.1 – Интерфейс реализованной программы решения обратных задач

На рисунке 6.2 представлен интерфейс разработанного web-сайта для решения задачи формирования прибыли с помощью обратных вычислений [159]. Работа с такими программами является более понятной для специалиста, однако использование их ограничено одной задачей, программа не может быть использована для формирования показателей с другими моделями.

Рисунок 6.2 – Интерфейс web-сайта для решения обратной задачи по формированию прибыли с помощью обратных вычислений

Использование объектно-ориентированного подхода при разработке программы позволяет создать инструмент, который может обеспечить с одной стороны хорошую поддержку специфичных обратных задач, а с другой – расширяемость, интеграцию моделей различных типов. К преимуществам данного подхода, появление которого связано с возникновением языка SIMULA в начале 60-х годов прошлого века, относят более простую модификацию полученных решений, возможность использования существующего программного кода для создания новых разработок, а применяемые концепции (объект, класс, наследование, сообщение) нередко соответствуют нашему способу видения вещей в исследуемой области реального мира.

6.1 Объектно-ориентированный подход

При реализации своих решений на универсальном языке программирования разработчик может использовать различные стили [160–161]:

– процедурно-ориентированный;
– объектно-ориентированный;
– логико-ориентированный;
– ориентированный на правила;
– ориентированный на ограничения.

Из перечисленных парадигм наиболее распространенными являются первые две. При этом более ранним считается процедурно-ориентированный стиль, в котором основным строительным блоком выступает процедура или функция, а объектно-ориентированный рассматривается как его развитие.

Согласно определению Буча Г. [160] под объектно-ориентированным подходом (ООП) понимается технология создания программного обеспечения (ПО), основанная на представлении программы в виде совокупности программных объектов, каждый из которых является экземпляром определенного типа (класса), а классы образуют иерархию с наследованием свойств. Как правило, полученные решения являются простыми для модификации и обслуживания, пригодны для использования при создании новых разработок, поэтому данный подход используется для реализации сложных систем. Работа Буча [160] является классической монографией по объектно-ориентированному подходу к анализу и проектированию программного обеспечения, в которой, в частности, рассматривается процесс разработки программ и объектные модели: логические, физические, статические и динамические.

Разработка программных систем с использованием ООП, включает следующие этапы: анализ предметной области, проектирование, реализация, модификация [161].

Существуют различные методологии объектно-ориентированного проектирования, например, OMT (Object Modeling Technique), OOSE (Object-Oriented Software Engineering) и др. В результате слияния некоторых из них появился стандарт объектно-ориентированной разработки приложений, имеющий набор графических нотаций и семантик для предоставления возможности отображения отдельной области объектно-ориентированной модели, называемый UML. UML может быть использован в различных организованных процессах изготовления программного продукта. Компанией Rational Corp. создан способ организации, называемый Рациональным Унифицированным Процессом (Rational Unified Process), который систематизирует процесс создания ПО на основе UML. RUP – проекты включают четыре фазы: начало (первичная оценка проекта, его экономическое обоснование), уточнение (анализ предметной области и построение исполняемой архитектуры), построение (реализация большей части функциональности продукта), внедрение (создание и передача финальной версии заказчику). При этом каждая фаза может быть разбита на итерации [162].

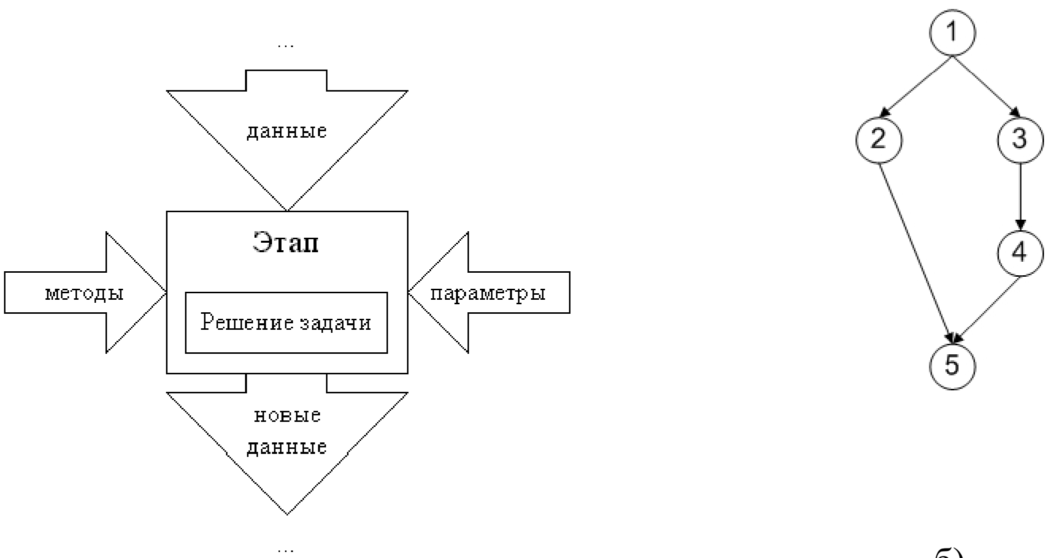
6.2 Структура системы поэтапного решения задач

В настоящее время объектно-ориентированная технология является популярной для разработки систем, используемых для решения различных экономических задач.

В данной работе рассматривается создание такого инструмента решения обратных задач, который с одной стороны обеспечивал бы решение специфических задач в предметной области, а с другой – гибкую модификацию, расширяемость программы, т.е. наращивание ее функциональных возможностей. Кроме того, система должна обеспечивать возможность поэтапного решения задач. Данное требование означает, что алгоритм моделирования должен быть разбит на некоторые составляющие, в результате чего решение выполняется поэтапно: после очередного этапа система переходит в режим ожидания действий пользователя.

Для создания систем, удовлетворяющих перечисленным требованиям, в данной работе предлагается использование подхода, основанного на декомпозиции на основе дерева целей.

При выполнении диссертационной работы автором был использован подход, предложенный в диссертационной работе [163], согласно которому выполняется анализ и декомпозиция на этапы алгоритма решения задачи в предметной области. Этапом называется та часть алгоритма, в которой полностью рассчитывается какая–либо величина, часто встречаемая в языке предметной области (рисунок 6.3 а). В результате последовательность решения задачи представляется в виде графа (либо дерева в зависимости от конкретной задачи) (рисунок 6.3 б). Так, запись, представленная на рисунке 6.3 б означает, что подзадача с номером 1 является начальной и этапы 2, 3 не могут быть выполнены прежде нее. Аналогично, подзадача 4 не может быть решена прежде, чем будет выполнен этап 3. Пятая вершина называется конечной.



б)

а)

Рисунок 6.3 – Построение графа решения задачи. а) этап схемы системы; б) пример графа: 1 – решение первой задачи; 2 – решение второй задачи и т.д.

При выделении этапов графа «с одной стороны не следует выделять слишком много этапов, на которых рассчитываются малоинформативные сами по себе величины, с другой стороны не следует выделять слишком крупные этапы, так как в этом случае будет создано мало точек контроля процесса расчета» [163]. Кроме того, критерием разделения может быть принадлежность к какому-либо одному подпроцессу.

Полученная таким образом структура позволяет осуществлять следующие действия:

– контролировать результаты, полученные на каждом этапе;
– автоматически пересчитывать этапы при изменении параметров;
– применять различные методы решения локальных задач каждого этапа.

Для ее программной реализации было использовано представление в виде многосвязного списка (рисунок 6.4). Каждый элемент этого списка является объектом класса `Узел`, который может ссылаться на любое количество предков и потомков (рисунок 6.4). `Узел` для решения своей задачи содержит ссылку на абстрактный класс `Расчет` (его метод `решение` осуществляет решение задачи узла) (рисунок 6.5–6.6). Функция метода `вызовПотомка` заключается в решении задачи вызванного пользователем узла, рекурсивном вызове и пересчете уже рассчитанных потомков. Для проверки расчета предыдущих этапов используется метод `вызовПредка` (в том случае, если хотя бы один из предков не был рассчитан, решение задачи текущего узла не может быть выполнено). Также структура классов включает класс `Хранилище`, который служит для хранения данных, используемых всеми этапами, и классы для реализации графического интерфейса.

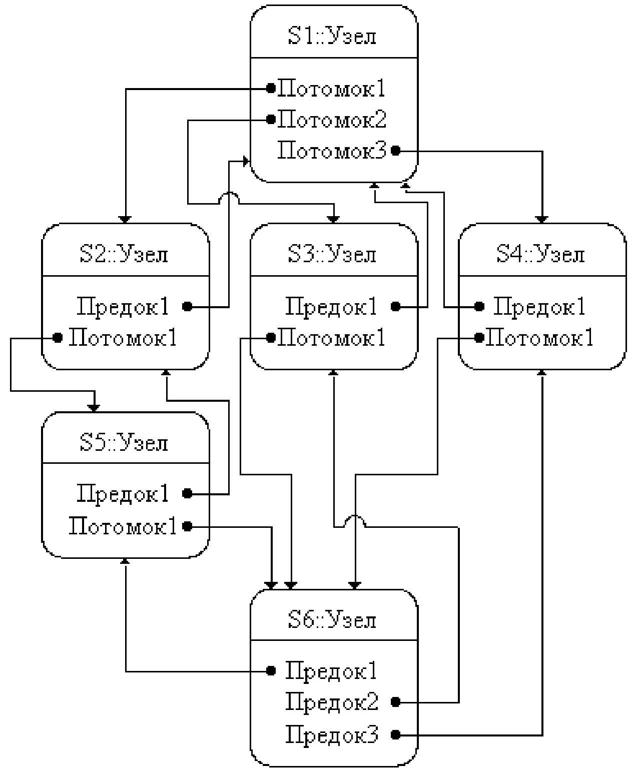


Рисунок 6.4 – Представление схемы в виде многосвязного списка

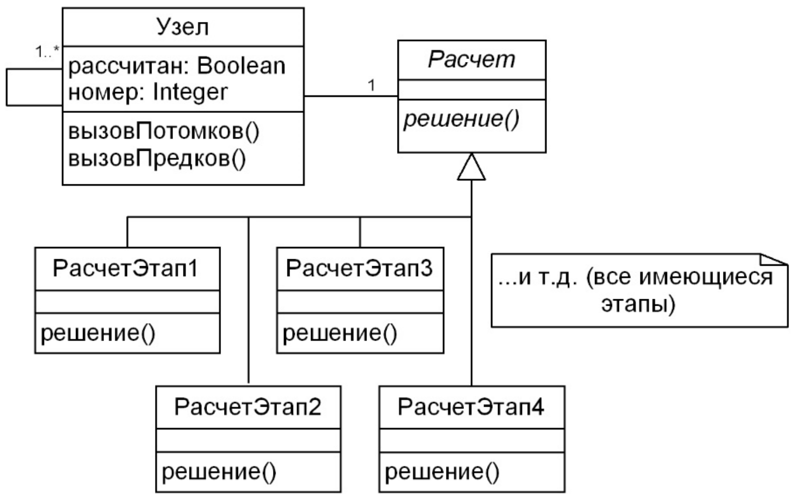


Рисунок 6.5 – Структура классов

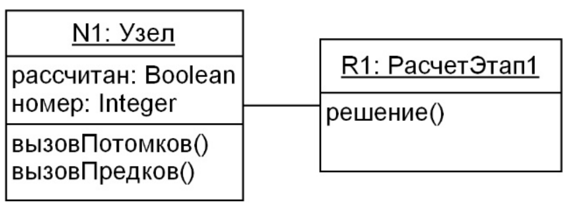


Рисунок 6.6 – Связь объектов

Похожую структуру имеет известный шаблон поведения объектов «Стратегия» [164], который определяет семейство алгоритмов, инкапсулирует каждый из них и делает взаимозаменяемыми (рисунок 6.7). Выделенное семейство алгоритмов, обладающих общей функциональностью вследствие наследования класса Strategy, можно использовать в разных контекстах.

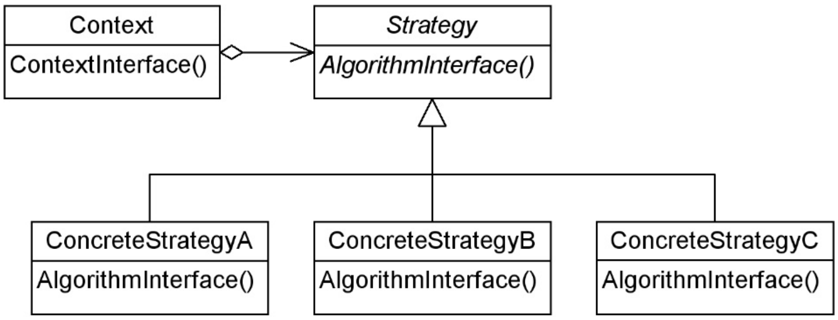


Рисунок 6.7 – Структура шаблона «Стратегия»

Узлы графа могут быть динамически присоединяемыми (объекты данного типа формируются в процессе работы пользователя с программной реализацией имитационной модели и присоединяются в качестве потомков к одному из этапов) либо статическими (создаются при инициализации структуры, например, при ее загрузке из файла). Динамические узлы используются в том случае, если они могут быть присоединены к любому блоку и неизвестно заранее их количество (например, графики).

Впоследствии данный подход был использован автором для решения задач имитационного моделирования (рисунок 6.8), полученные результаты представлены в диссертационной работе [165], а также публикациях [166–191]. Необходимость модификации была обусловлена следующими функциональными возможностями систем имитационного моделирования:

– имитация рассматриваемой системы во времени;
– осуществление многократного прогона модели;
– проведение экспериментов с целью исследования влияния какого-либо параметра на результат моделирования.

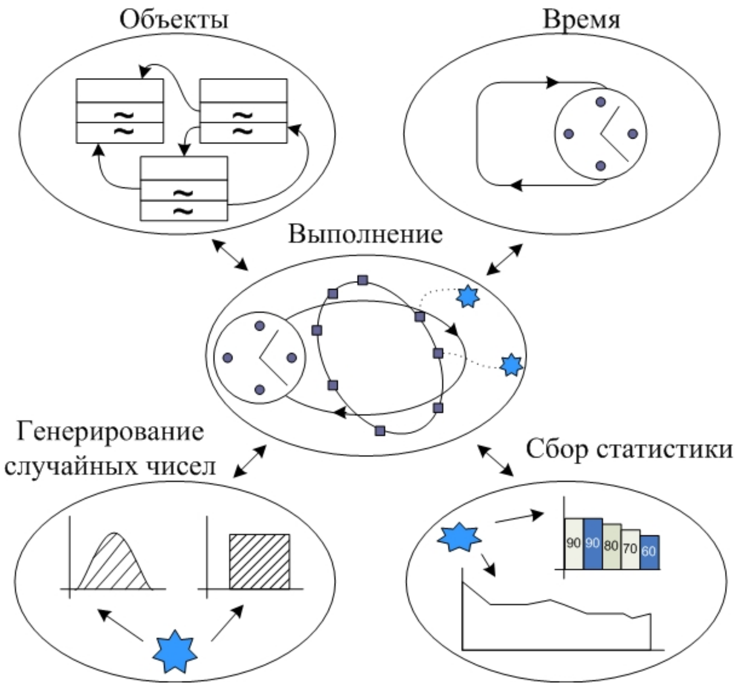


Рисунок 6.8 – Структура системы имитационного моделирования

Для реализации перечисленных функций в структуру были включены дополнительные классы и введены различные виды блоков, составляющих граф (или дерево) решения задачи.

Разработанный подход был использован для создания трех программных систем: «Имитатор» [191], «Запас» [175], «Аукцион» [173] (приложение А). Графы решения задач представлены на рисунке 6.9–6.11. На рисунке 6.9 и 6.11 декомпозиции был подвержен моделирующий поведение системы имитационный алгоритм, на рисунке 6.11 – комплекс моделей и операций по обработке результатов.

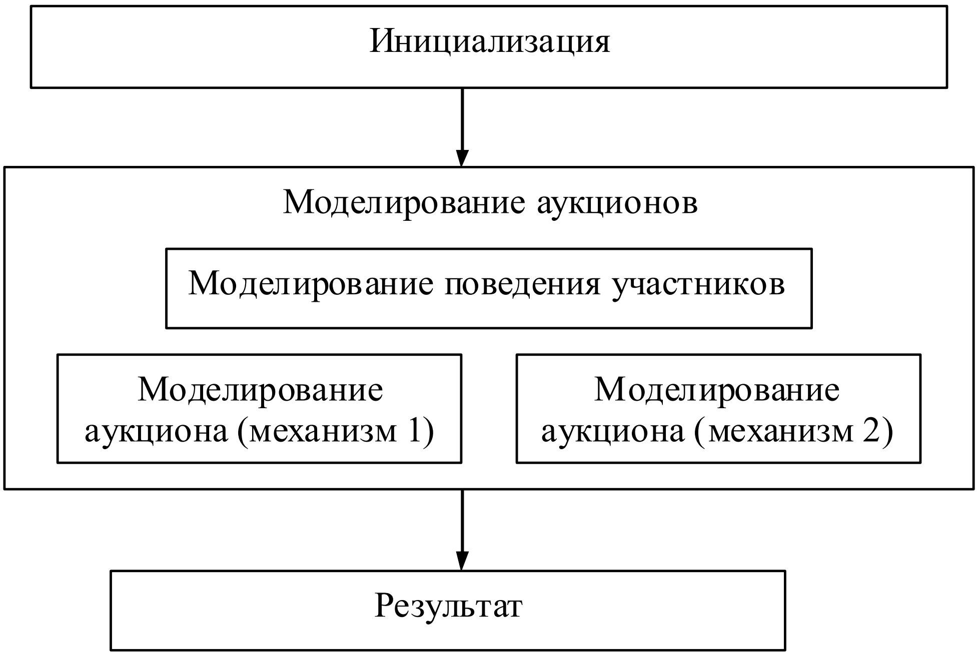


Рисунок 6.9 – Граф решения задачи моделирования торгов

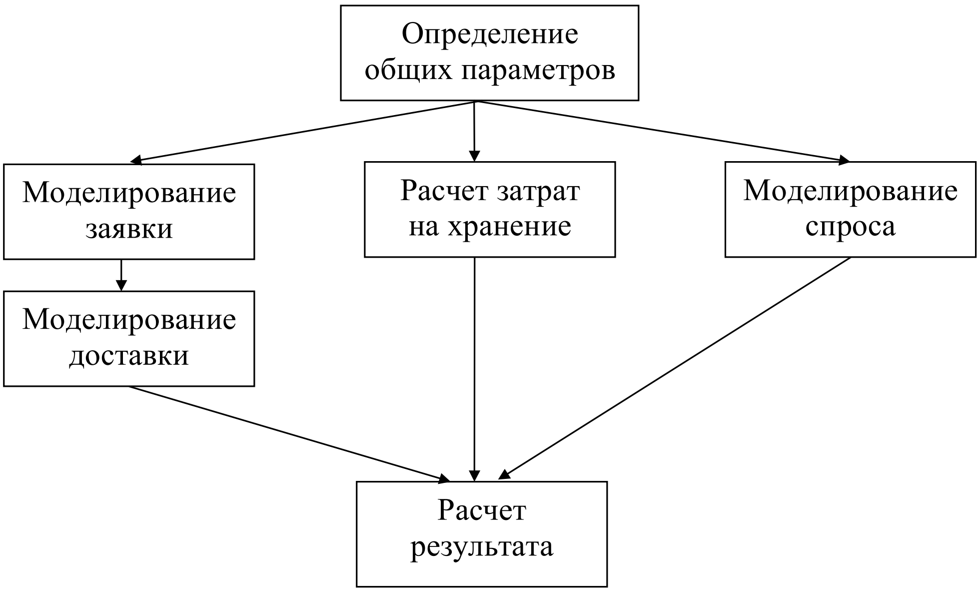


Рисунок 6.10 – Граф решения задачи моделирования управления запасами

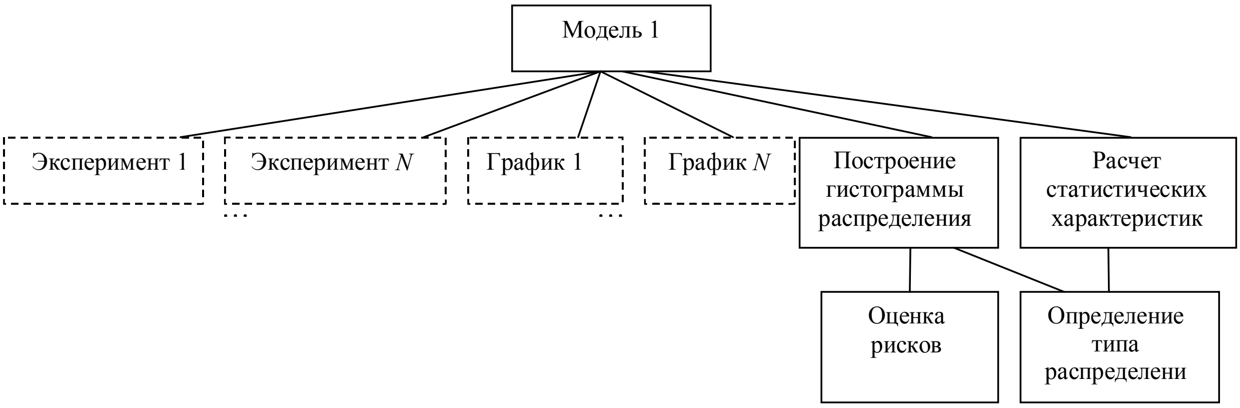


Рисунок 6.11 – Граф решения задачи моделирования экономических объектов

Созданные программные системы были внедрены в учебный процесс и производство. Так программа «Имитатор» была использована в учебном процессе факультета дистанционного обучения ТУСУР при изучении студентами дисциплин «Имитационное моделирование» и «Математическое и имитационное моделирование экономических процессов» (направление подготовки – 09.03.03 Прикладная информатика) более 10 лет. Описание данной программы и имитационных моделей приводится в учебных пособиях [192–199]. Также на основе имитационных моделей программы были разработаны табличные имитационные модели [200–208] и использованы в учебном процессе дисциплин «Основы бизнеса» и «Эконометрика» и в производстве.

6.3 Разработка программы решения обратных задач на основе дерева цели

Для применения описанного подхода при создании системы решения обратных задач на основе обратных вычислений необходимо также выполнить модификацию. Декомпозиции в этом случае подвергается дерево целей, а этап определяет переменную, формирующую модель (рисунок 6.12).

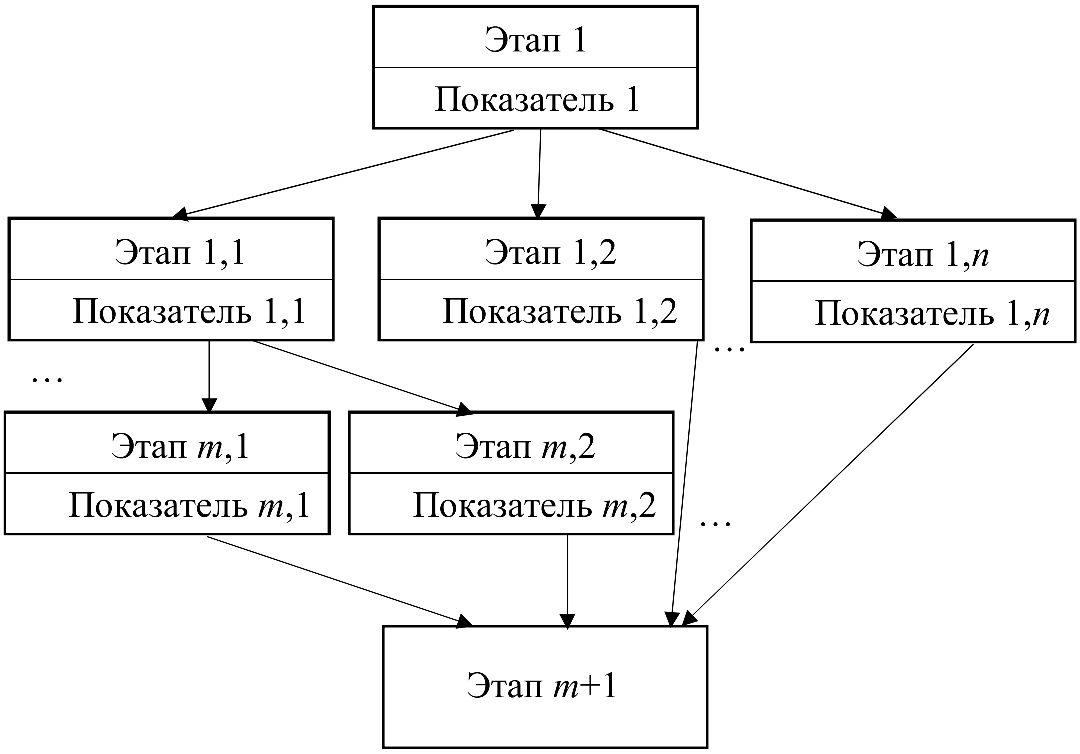


Рисунок 6.12 – Этапы системы решения обратных задач

Также необходимо обеспечить возможность использования связей вершин для решения обратных задач. Такую возможность обеспечивают соответствующие методы, которые реализуют доступ объекта списка к данным других объектов.

6.3.1 Разработка алгоритма для решения общей задачи формирования результирующего показателя дерева цели

Для реализации программы необходимо разработать алгоритм поэтапного формирования характеристик, образующих дерево цели и отличающегося от существующих возможностью решения иерархических обратных задач с ограничениями и использованием аргументов в нескольких подзадачах. Для этого необходимо решить такие задачи как [209]:

– разработка алгоритмов для решения отдельных подзадач;

– разработка алгоритма для решения общей задачи формирования результирующего показателя дерева цели;

– разработка структуры классов;

– реализация на основе разработанных алгоритмов программного обеспечения.

Под отдельной подзадачей понимается решение обратной задачи по определению значений характеристик, формирующих показатель более высокого уровня. Так, на рисунке 6.13 решение общей задачи будет включать такие подзадачи как определение значений Δx_1, Δx_2, Δx_3 для формирования y^*; вычисление Δx_{11}, Δx_{21} для формирования $x_1 + \Delta x_1$; расчёт Δx_{21}, Δx_{22} для формирования $x_2 + \Delta x_2$. Для решения отдельных подзадач могут быть использованы методы, расссмотренные в п.3.2.6, 3.3.1, 5.1.2, 5.2.2.

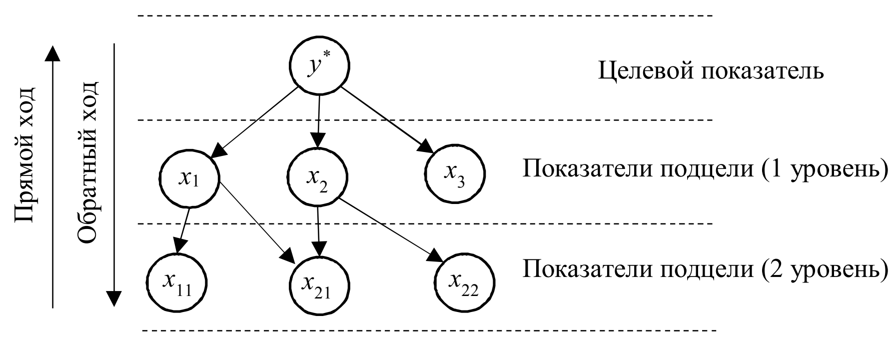


Рисунок 6.13 – Дерево цели/показателей

Рассмотрим решение общей задачи по определению результирующего показателя с учетом ограничений. В этом случае необходимо для каждого показателя определить дополнительные характеристики: минимальное l и максимальное h значение; признак использования в расчётах u. Для обхода дерева цели используется рекурсивная процедура (рисунок 6.14), в которой показатели представлены в виде узлов (узлы-потомки – характеристики, которые формируют показатель более высокого уровня). Вычисленные с помощью процедуры решения обратной задачи новые значения узлов-потомков становятся целевыми для определения значений узлов-потомков следующей уровня.

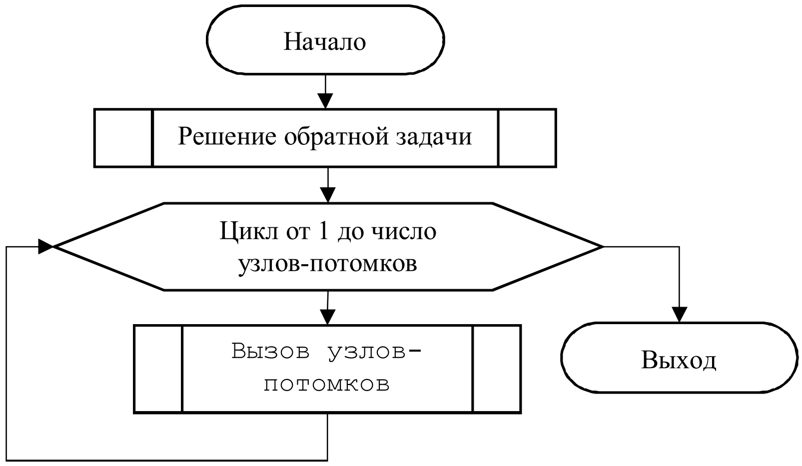


Рисунок 6.14 – Процедура обхода дерева «Вызов узлов-потомков»

На рисунке 6.15 представлен общий алгоритм для решения обратной задачи. Условием останова алгоритма является увеличение разницы между целевым значением результирующего показателя и текущей его величиной. Вычисление нормированных значений коэффициентов важности осуществляется с учётом исключения из вычислений тех узлов, значения которых достигли граничных. Расчёт нового значения узла выполняется по итерационным формулам, представленным в п.3.2.6, 5.1.2, а также в работе [40]. Под «позитивным» изменением здесь понимается случай, когда значение в текущей итерации стало ближе к допустимой области по сравнению с предыдущей итерацией (изначально величина не удовлетворяла заданным ограничениям). Признак возможности использования в расчётах устанавливается равным 1, если дальнейшее изменение данного показателя возможно, или 0, если данный показатель не может быть изменён. После завершения работы алгоритма происходит проверка на соответствие ограничениям. Если значение не удовлетворяет ограничениям, то ему присваивается ближайшее граничное значение и работа алгоритма начинается заново. При этом изменение данного аргумента в ходе выполнения алгоритма не происходит. Также выполняется проверка на отсутствие решения в случае использования коэффициентов относительной важности, когда заданные соотношения и направления изменения не позволяют достичь цели.

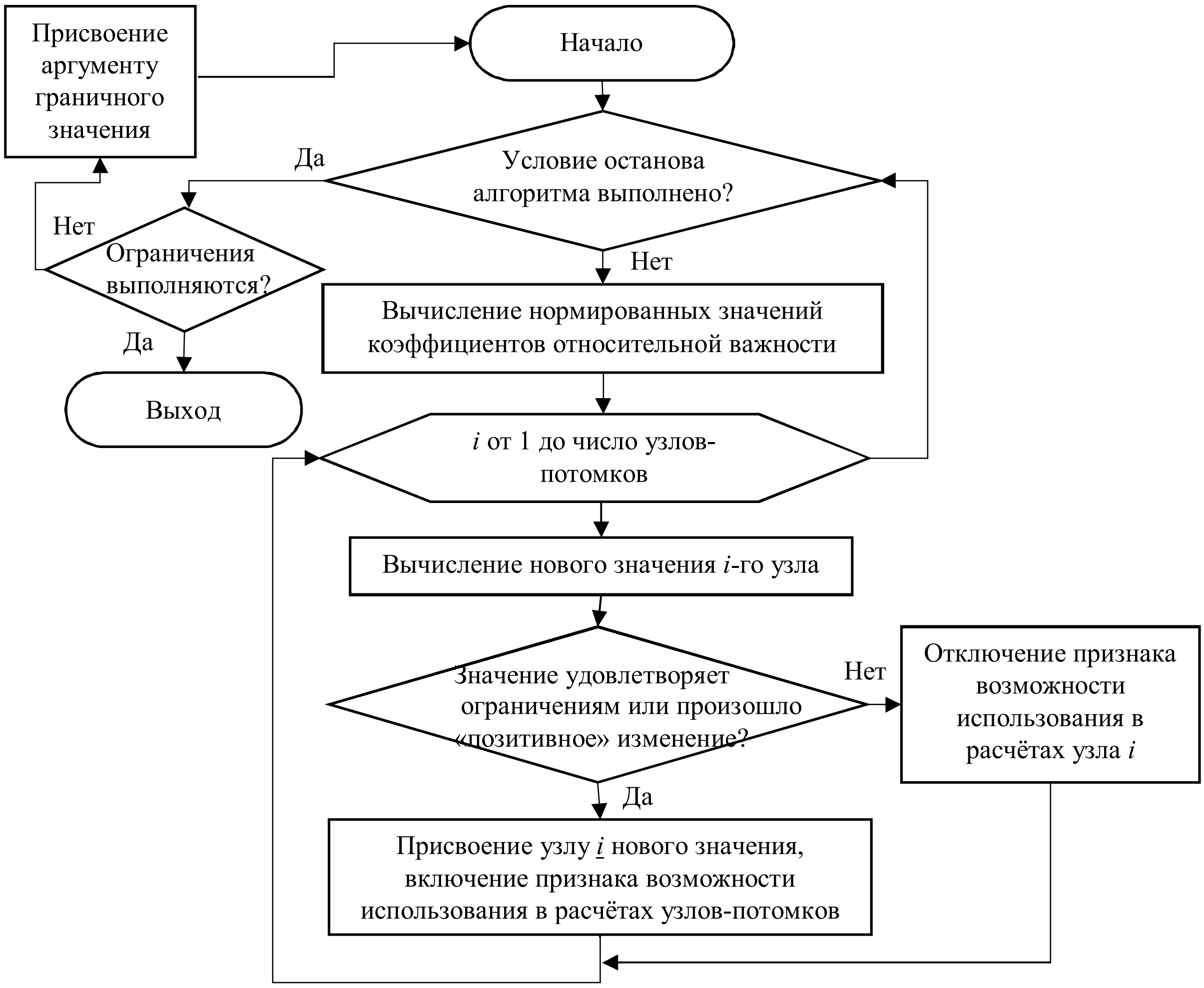


Рисунок 6.15 – Алгоритм решения обратной задачи при одновременном изменении аргументов

Алгоритм решения задачи, основанный на выборе аргументов для изменения, представлен на рисунке 6.16. Здесь узел с номером j – это вершина дерева, представляющая собой формируемый показатель в текущей подзадаче. В стохастическом алгоритме вычисление данных для выбора узла включает нормирование коэффициентов важности, которые выступают в роли вероятностей, а выбор k-го узла осуществляется с помощью моделирования полной группы несовместных событий. При минимизации суммы абсолютных значений аргументов вычисление данных включает расчёт частных производных и выбор k-го узла осуществляется путем определения максимального значения частной производной. Значение k-го узла вычисляется по итерационным формулам, представленным в п.2.2.1, 3.3.1.

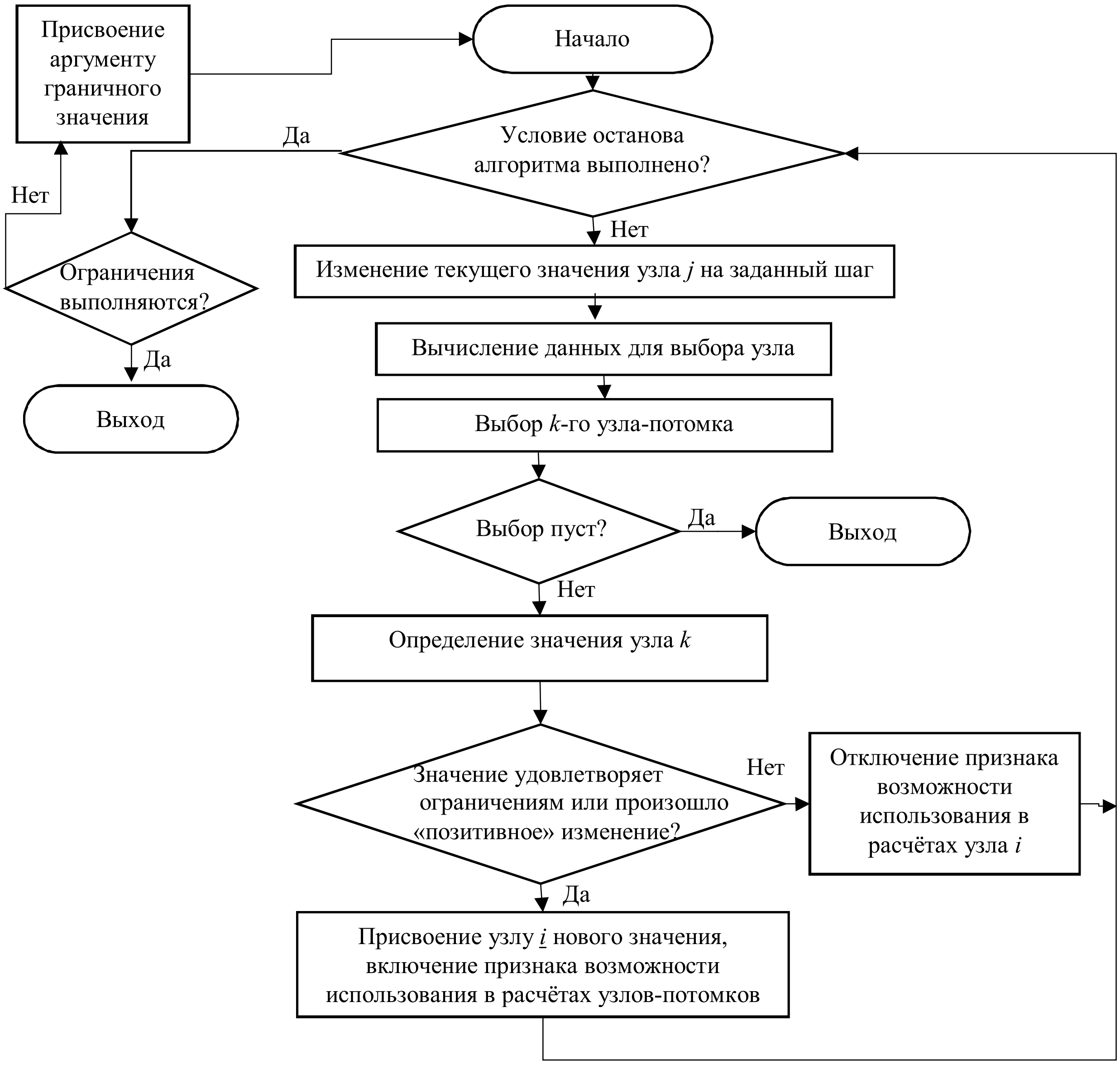


Рисунок 6.16 – Алгоритм решения обратной задачи при выборе узла для изменения

Если одна и та же характеристика используется при расчёте разных показателей (так, на рисунке 6.13 величина x_{21} используется при расчёте x_1 и x_2), то после нахождения решения выполняется корректировка. Например, такая ситуация возникает в случае, когда происходит параллельное формирование себестоимости разных изделий, которая в свою очередь определяется ценой одного и того же продукта. В этом случае происходит формирование дерева корректировки, включающего показатели дерева цели, и решение задачи при минимизации суммы квадратов изменений аргументов. Дерево корректировки создается следующим образом (рисунок 6.17):

– на нулевом уровне расположена стартовая вершина для запуска процедуры обхода дерева;

– на первом уровне располагаются корректировки: число узлов на этом уровне соответствует количеству корректировок, которые необходимо выполнить;

– на втором и третьем уровне располагаются показатели и формирующие их характеристики соответственно.

Решение задачи выполняется итерационно: определяются изменения характеристик таким образом, чтобы сумма квадратов изменений была минимальна, а значения формируемых показателей были равны величинам, определенным с использованием дерева цели

$$g(\Delta x) = \sum_{j=1}^{n} \Delta x_{1j}^2 \to \min,$$

$$f_i\left(\Delta x_{1j}\right) = x_i^*, i = 1..m.$$

где n – число характеристик, относящихся к первой корректировке, значения которых подлежат изменению;

m – число формируемых показателей;

x_i^* – целевое значение x_i, определенное в ходе решения обратной задачи с использованием дерева цели.

После решения основной задачи и корректировки также может возникнуть задача обработки результатов: вычисление новых значений на основе вычисленных, а также проверка достижимости полученных значений. Например, определение суммарного объема сырья, участвующего в формировании себестоимости разных видов продукций. Для этого из соответствующих узлов создается граф обработки результатов (рисунок 6.18). Поскольку данная задача является прямой, то для её решения используется обход графа снизу-вверх. В этом случае вычисление узлов осуществляется последовательно, конечный узел является фиктивным и предназначен для запуска рекурсивной процедуры обхода графа, которая подразумевает вычисление текущей вершины только в том случае, если рассчитаны все её потомки. Для реализации процедуры обхода каждый узел имеет атрибут «рассчитан». На начальном этапе данному атрибуту присваивается значение false. В процессе рекурсивного обхода графа происходит изменение значения атрибута на true. Алгоритм метода «вызов узлов-предков», вызываемого для фиктивного конечного узла представлен на рисунке 6.19. Также в данном алгоритме используется переменная flag, принимающая значение false, если вычислены все узлы-потомки текущего узла (только в этом случае происходит расчёт текущего узла).

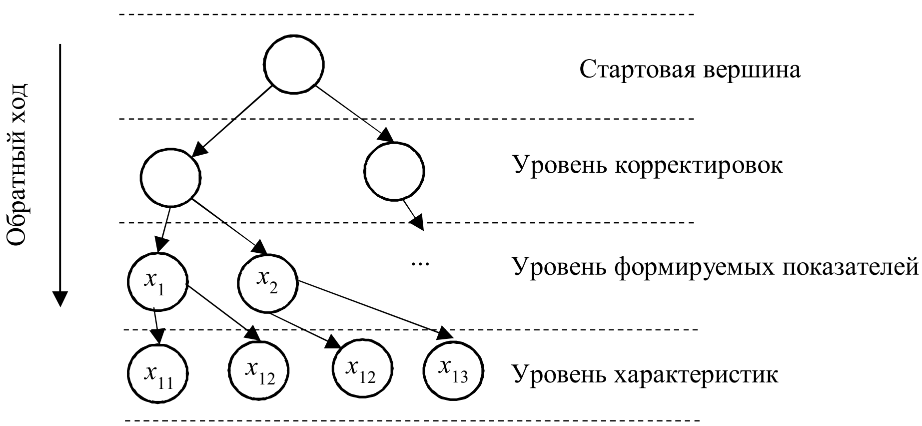


Рисунок 6.17 – Дерево корректировки

Прямой обход может быть использован и для обхода дерева показателей, когда, например, решение обратной задачи осуществляется на основе решения прямой задачи.

Для проверки достижимости результата могут быть использованы методы регрессионного анализа, в этом случае на основе существующих зависимостей определяется, принадлежит ли вычисленное значение доверительному интервалу.

В случае, если значение не является достижимым, то выполняется корректировка входных данных с последующим решением новой обратной задачи.

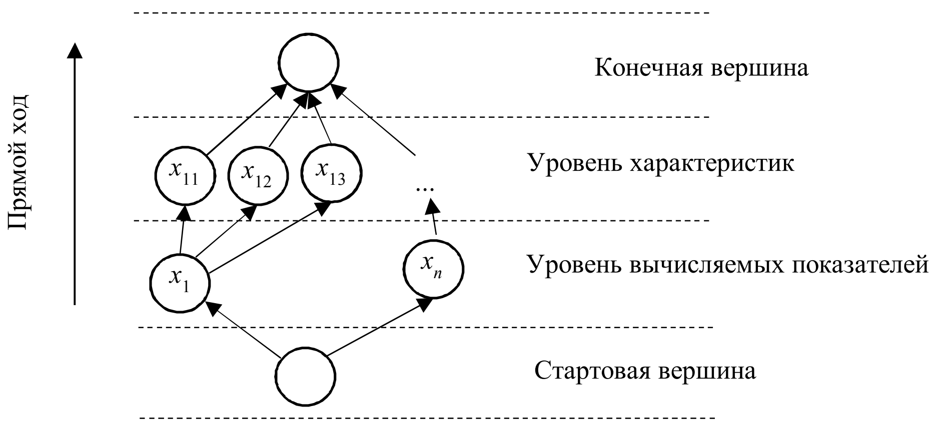


Рисунок 6.18 – Граф обработки результатов

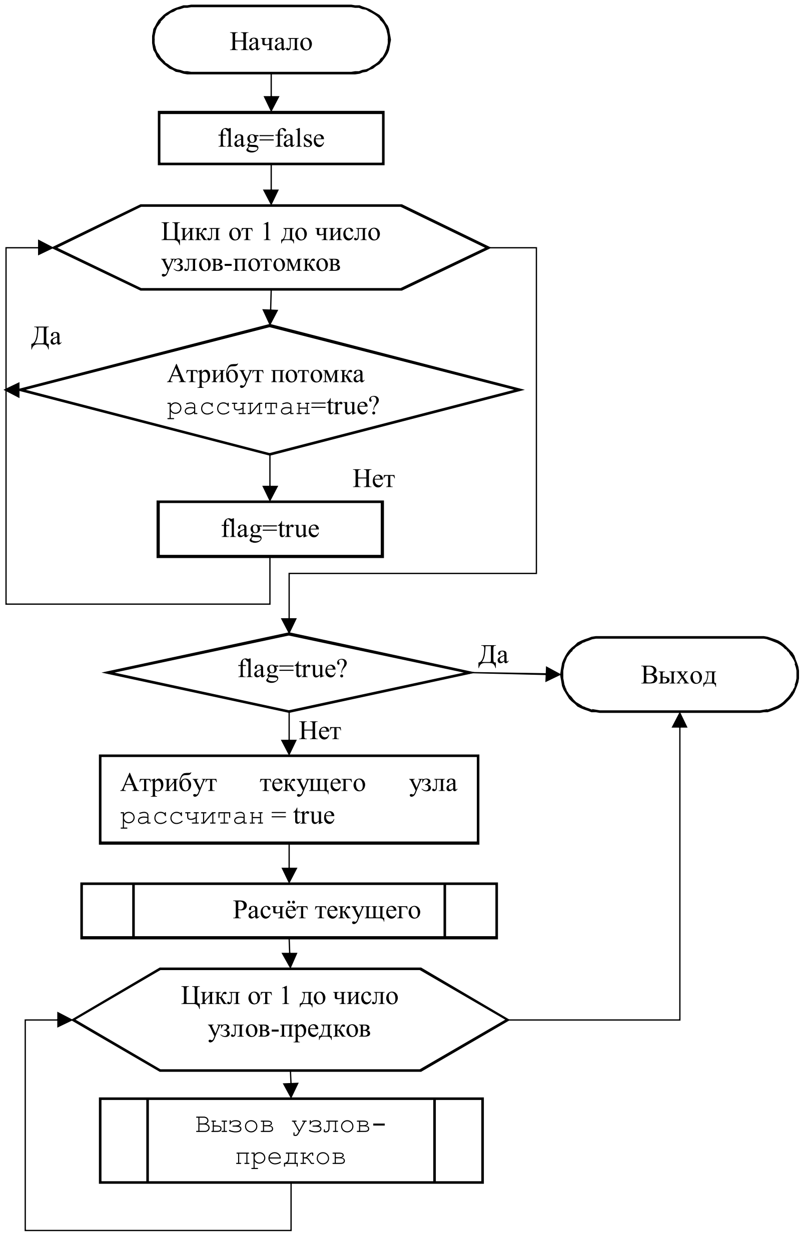


Рисунок 6.19 – Алгоритм прямого обхода графа

Решение обратной задачи может включать прохождение трех уровней, на каждом из которых формируется своё дерево/граф (рисунок 6.20).

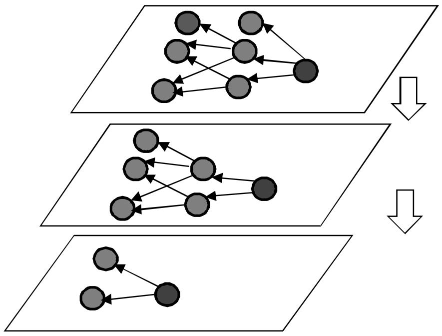


Рисунок 6.20 – Уровни решения задачи

Таким образом, алгоритм поэтапного формирования характеристик, образующих дерево цели, позволяет осуществлять решение иерархических обратных задач с ограничениями и использованием аргументов в нескольких подзадачах.

6.3.2 Разработка структуры системы решения обратной задачи

Для программной реализации графа/дерева было использовано представление в виде многосвязного списка, рассмотренное в п.6.2. Каждый элемент этого списка является объектом класса `Узел`, который может ссылаться на любое количество узлов-предков и узлов-потомков. Класс `Узел` для решения своей задачи содержит ссылку на абстрактный класс `Расчет`. Реализация алгоритмов решения обратных и прямых задач выполняется в классах наследниках `Расчёт`. Каждый объект класса Узел содержит ссылку на объект наследник класса Расчёт, таким образом, для вершины устанавливается способ решения прямой и обратной задачи. В частности, в методах «`расчётОбр`» реализованы итерационные формулы расчёта. Методы «`вызовПотомков`» и «`вызовПредков`» предназначены для рекурсивного обхода графа/дерева. Алгоритм решения обратной задачи с учётом ограничений (рисунки 6.15, 6.16) реализован в методе «решение», который вызывается при реализации рекурсивного обхода. На рисунке

6.21 также представлены основные атрибуты класса `Узел`. Применение объектно-ориентированного подхода предоставляет возможность гибкой модификации и расширения системы: добавления новых показателей, методов решения обратных задач и т.д.

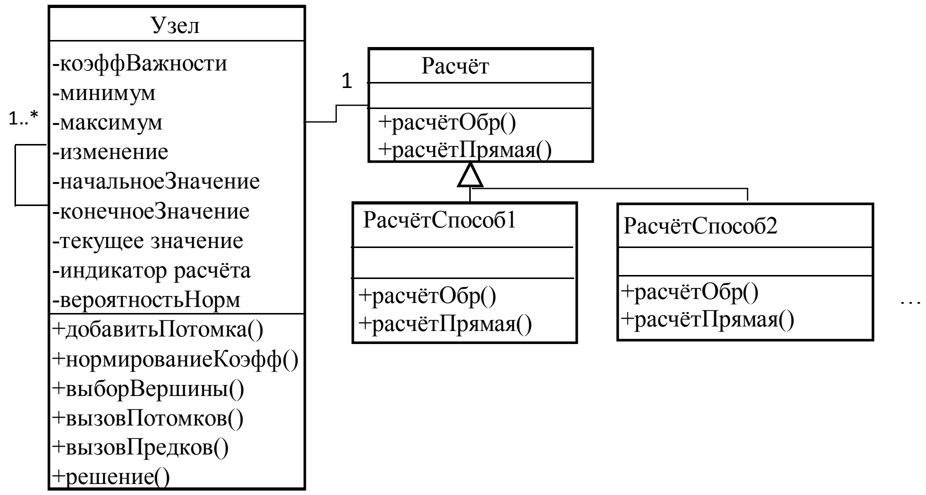


Рисунок 6.21 – Структура классов

Как уже было сказано выше, используемый подход к разработке системы позволяет вносить в существующую структуру изменения. Они могут быть следствием:

– появления новой задачи в предметной области;

– необходимости использовать еще один метод решения задачи.

Первый случай соответствует появлению нового этапа в существующей схеме. Последовательность действий при этом следующая:

1) создать еще один экземпляр класса `Узел` с соответствующим номером;

2) в зависимости от положения на графе нового узла добавить ссылки на его предков, потомков;

3) изменить при необходимости ссылки узлов, которые оказались связанными с новым;

4) создать новый подкласс класса `Расчет` и реализовать в нем метод `решение`;

5) создать экземпляр подкласса из п.4 и установить в объекте класса `Узел` ссылку на него.

На рисунке 6.22 представлены основные модули системы. Передача данных программе пользователем осуществляется посредством интерейса, после этого с помощью модуля ввода данных выполняется их проверка на корректность. С помощью модуля построения дерева цели осуществляется формирование дерева цели с использованием наборов объектов в соответствии с описанной ранее структурой классов. Модуль решения задачи включает рекурсивный обход дерева цели и решение прямых и обратных задач с помощью алгоритмов, представленных в п. 6.3.1. Исходные данные и результаты работы программы могут быть выгружены в хранилище.

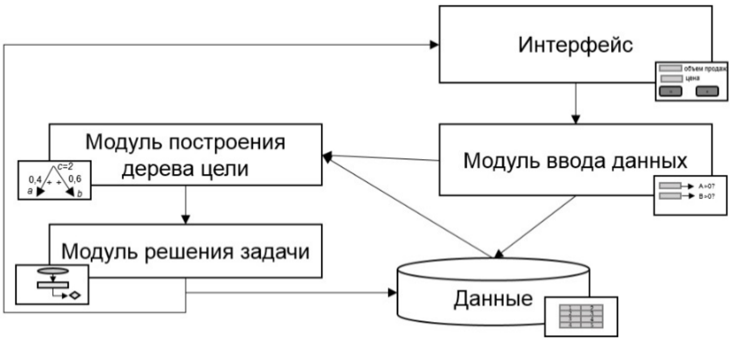


Рисунок 6.22 – Основные модули системы

6.3.3 Алгоритмы решения задачи на основе статистических данных

6.3.3.1 Алгоритм решения задачи при стохастической зависимости между аргументами

При наличии детерминированной зависимости между результирующим показателей и входными значениями, отдельные аргументы этой функции могут быть связаны стохастической зависимостью. Одним из наиболее распространенных аппаратов для исследования стохастической зависимости между аргументами является регрессионный анализ [210]. В качестве примеров задач, решение которых рассматривалось с помощью уравнения регрессии можно привести следующие: оценка дохода торгового агента сетевого маркетинга в зависимости от затраченного времени, возраста, пола, опыта работы [211], прогноз выручки ресторана быстрого питания в зависимости от дня недели и праздничных дней [212–214], оценка числа проданных онлайн-купонов на скидку в зависимости от группы товара, размера скидки и отзывах о товаре/услуге [215–216], оценка стоимости квартиры города Томска в зависимости от числа комнат, площади, этажа, типа постройки, вида отделки, рейтинга дома, определяющего его близость к объектам инфраструктуры, района, года постройки [217], оценка удовлетворенности потребителей на основе онлайн-отзывов [218], оценка технической эффективности [219] и цифрового капитала организаций [220–221].

Статистические данные при наличии стохастической зависимости могут быть использованы для определения дополнительных условий задачи, а также для проверки достижимости результата (на рисунке 6.20 слой обработки результатов).

Так, решение обратной задачи может быть выполнено путем построения уравнения связи между аргументами и включения его в обратную задачу

$$f(x_1+\Delta x_1, x_2+\Delta x_2, \dots, x_n+\Delta x_n) = y+\Delta y,$$
$$x_k+\Delta x_k = h\left(x_l+\Delta x_l\right),$$

где k, l – индексы аргументов, между которыми существует зависимость.

В п.4.2 представлен алгоритм решения оптимизационной задачи при наличии нескольких ограничений: замена переменных и формирование одного модифицированного ограничения. Однако в большинстве случаев нахождение решения при соответствии точечной оценки является трудоемким процессом, т.к. возникает необходимость корректировки коэффициентов относительной важности.

Для решения задачи, в которой аргументы связаны стохастической зависимостью, может быть также использован способ, основанный на построении предиктивного интервала. В ходе реализации данного способа исследуется область вокруг решения, полученного без учета стохастической зависимости, и выбирается решение наиболее близкое к исходному и наиболее соответствующее зависимости.

Для решения задачи выполняется построение уравнения связи между аргументами: собираются статистические данные за предыдущие периоды и осуществляется определение параметров методом наименьших квадратов. При заданном значении объясняющей переменной величина объясняемой переменной может принадлежать некоторому интервалу. Построение такого предиктивного интервала осуществляется по формуле

$$l_t = \hat{y} + se \cdot t_\alpha ,$$

$$l_b = \hat{y} - se \cdot t_\alpha ,$$

где l_t – верхняя граница интервала;

l_b – нижняя граница интервала;

se – среднее квадратическое отклонение ошибки прогноза индивидуального значения зависимой переменной;

$\hat{y}$ – точечная оценка прогноза зависимой переменной;

t_α– табличное значение t-критерия при уровне значимости α.

Рассморим алгоритм решения задачи (аргумент x_2 зависит от аргументов x_{1j}, j=1..v, v – число объясняющих переменных) [222–223]. Исходные данные: функция зависимости $f(x)$, статистические данные x_{1k}, x_2, начальные значения аргументов $\tilde{x}$, заданное значение функции y^*, экспертная информация для решения обратной задачи (либо вид регуляризации).

Шаг 1. Используя исходные данные $\tilde{x}$ выполнить нахождение $\hat{x}$ для достижения заданного значения функции $f(x)$ путем решения обратной задачи. Решение задачи в зависимости от постановки может быть выполнено как с помощью указания коэффициентов относительной важности, так и с использованием регуляризации на основе расстояния от исходных значений.

Шаг 2. Используя статистические данные x_{1j}, x_2 выполнить построение предиктивного интервала для x_2 при определенных на первом шаге величинах $\hat{x}_{1j}$. Если $\hat{x}_2$ принадлежит полученному интервалу, то решение считается найденным и работа алгоритма завершается, иначе осуществляется переход на шаг 3.

Шаг 3. Изменять $\widehat{x}_{1j}$ на величину заданного шага h на интервале $[x_{\min};x_{\max}]$. Для каждого полученного значения $\widehat{x}_{1j}^*$ ($\widehat{x}_{1j}^* = \widehat{x}_{1j} + h \cdot \zeta$, где ζ – переменная, принимающая значения от 1 до числа шагов) определить соответствующую величину $\widehat{x}_2^*$ (выражается из функции $f(x_{1j},x_2)$, такое выражение x_2 при заданных x_{1j} также обозначим $x_2\left(x_{1j}\right)$), верхнюю l_t и нижнюю l_t границы предиктивного интервала для x_2.

Шаг 4. Выбор среди значений $\widehat{x}_{1j}^*$ величин x_{1j}^* путем решения задачи оптимизации

$$\omega(x_1^*) = \sum_{j=1}^{v} k_1\gamma_{norm}(x_{1j}^*) + k_2\lambda_{norm}(x_{1j}^*) \to \max,$$

$$\gamma(x_{1j}^*) = \sum_{j=1}^{v}\left(\widehat{x}_{1j} - x_{1j}^*\right)^2 + \left(x_2\left(\widehat{x}_{1j}\right) - x_2\left(x_{1j}^*\right)\right)^2 = \sum_{j=1}^{v} \Delta\widehat{x}_{1j}^2 + \Delta\widehat{x}_2^2,$$

$$\gamma_{norm}(x_{1j}^*) = \frac{\max(\gamma(\widehat{x}_{1j}^*)) - \gamma(x_{1j}^*)}{\max(\gamma(\widehat{x}_{1j}^*)) - \min(\gamma(\widehat{x}_{1j}^*))},$$

$$\lambda(x_1^*) = -\frac{\left(l_t - x_2\left(x_1^*\right)\right)\left(x_2\left(x_1^*\right) - l_b\right)}{l_t - l_b},$$

$$\lambda_{norm}(x_1^*) = \frac{\max(\lambda(\widehat{x}_1^*)) - \lambda(x_1^*)}{\max(\lambda(\widehat{x}_1^*)) - \min(\lambda(\widehat{x}_1^*))},$$

где k_i – весовой коэффициент показателя i;

$\gamma\left(x_{1j}^*\right)$ – показатель расположения j-й переменной относительно начального решения;

$\lambda\left(x_1^*\right)$ – показатель расположения объясняемой переменной относительно величины предиктивного интервала;

$\gamma_{norm}\left(x_{1j}^*\right)$, $\lambda_{norm}\left(x_1^*\right)$ – нормированные значения показателей $\gamma\left(x_{1j}^*\right)$ и $\lambda\left(x_1^*\right)$ соответственно.

Целевая функция оптимизационной задачи включает две части. Величина $\gamma(x_{1j}^{*})$ характеризует степень удаленности от найденного на шаге 1 решения, выражение $\left(l_t - x_2\left(x_1^{*}\right)\right)\left(x_2\left(x_1^{*}\right) - l_b\right)$ при неизменных границах примет максимальное значение, когда величина $x_2(x_1^{*})$ будет находиться в середине предиктивного интервала (соответствовать точечной оценке). Расположенная в знаменателе разность $l_t - l_b$ представляет собой величину штрафа s за увеличение размера предиктивного интервала. Величина данного штрафа может быть увеличена путем использования квадратичной или экспоненциальной функции: $s = \left(l_t - l_b\right)^2$, $s = e^{l_t - l_b}$.

Рассмотрим задачу формирования прибыли при наличии зависимости объема продаж от цены (использованы данные ООО «ВокиФудТомск» о продукте «пшеничная лапша»).

Для построения уравнения зависимости количества от цены был использован метод наименьших квадратов. Полученное уравнение регрессии в случае линейной зависимости

$$K = 148{,}2 - 1{,}15 \cdot Ц\,. \qquad (6.1)$$

Исходное значение прибыли равно 4018 руб., цена 60 руб., количество – 98 шт., себестоимость равна 19 руб. Необходимо определить величины цены и объема таким образом, чтобы прибыль была равна 4900 руб. Зависимость объема выпуска от цены представлена линейным уравнением (6.1).

Определенные на первом шаге описанного выше алгоритма значения цены и количества равны 66 и 104 соответственно. Необходимо рассмотреть значения цены до 106 руб. с шагом 5 руб. и выбрать оптимальное значение (α=0,05, k_1=0,5, k_2=0,5).

Результаты вычислений представлены в таблице 6.1. Можно увидеть, что оптимальное решение задачи, при котором будет достигнуто заданное значение прибыли: $Ц^{*}$=76 руб., K^{*}=86 шт.

Таблица 6.1 – Решение задачи определения оптимальных значений аргументов (линейная зависимость)

$\widehat{Ц}^*$	$\widehat{K}^*$	$\Delta\widehat{Ц}$	$\Delta\widehat{K}$	l_b	l_t	$\gamma(\widehat{Ц}^*)$	$\lambda(\widehat{Ц}^*)$	$\omega(\widehat{Ц}^*)$
66,00	104,00	0	0,00	43,60	101,00	0	3,152	0,618
71,00	94,00	5	–10,00	39,75	93,35	125	0,658	0,891
76,00	86,00	10	–18,00	35,56	86,04	424	–0,042	$\boxed{0,934}$
81,00	79,00	15	–25,00	30,95	79,15	850	–0,145	0,891
86,00	73,00	20	–31,00	25,88	72,72	1361	0,278	0,777
91,00	68,00	25	–36,00	20,28	66,82	1921	1,215	0,596
96,00	64,00	30	–40,00	14,17	61,43	2500	2,707	0,349
101,00	60,00	35	–44,00	7,55	56,55	3161	3,693	0,151
106,00	56,00	40	–48,00	0,48	52,12	3904	4,172	0,000

На основе предложенного алгоритма реализована программа решения обратной задачи формирования маржинальной прибыли предприятия [224]. Описание программы приводится в приложении В.

6.3.3.2 Алгоритм решения задачи при прогнозных величинах аргументов

Рассмотрим случай, когда результирующий показатель связан с входными переменными детерминированной зависимостью, при этом значения входных переменных формируются путем прогноза на основе статистических данных [225].

Пусть даны значения аргументов x_j функции $f(x)$ за m промежутков времени: $x_{j,i}$ ($i=1,..,m$, $j=1..n$, n – число переменных). Необходимо определить величины аргументов $x_{j,m+1}$, обеспечивающие заданное значение функции $f(x)=y^*$.

Суть решения такой задачи заключается в прогнозировании значений аргументов x в момент $m+1$ на основе статистических данных. Для этого также используется уравнение регрессионной связи. На основе полученных величин аргументов происходит дальнейшая корректировка решения с помощью обратных вычислений.

Таким образом, метод решения данной задачи можно разбить на два этапа: определение аргументов с помощью регрессионной функции и решение обратной задачи.

Этап 1. Построение регрессионных функций *h* для прогнозирования аргументов .

Самой простой является линейная зависимость, в случае ее использования формулы расчета аргументов будут иметь вид

$$\widehat{x}_{j,m} = a_j + b_j \cdot m, \tag{7.1}$$

где a, b – неизвестные параметры регрессии.

Также могут быть рассмотрены другие модели, например, авторегрессионная

$$\widehat{x}_{j,m+1} = a_j + b_j \cdot \widehat{x}_{j,m}.$$

В результате решения задачи (7.1) будут найдены значения неизвестных параметров a, b. Путем подстановки m+1 в уравнения зависимости, получим величины аргументов $\widehat{x}_{j,m+1}$

$$\widehat{x}_{j,m+1} = a_j + b_j \cdot (m+1).$$

Значение функции для вычисленных значений аргументов будет равно: $f(\widehat{x}_{j,m+1})$.

Если условие $f(\widehat{x}_{j,m+1}) = y^*$ не выполняется, то необходимо скорректировать значения аргументов, и осуществляется переход ко второму этапу.

Этап 2. Решение обратной задачи определения аргументов. Вычисленные на первом этапе величины $\widehat{x}$ теперь рассматриваются как исходные значения аргументов. Необходимо вычислить их изменения, чтобы результирующий показатель был равен y^*. При минимизации суммы квадратов изменений аргументов задача будет иметь вид

$$\sum_{j=1}^{n} \left(\Delta\widehat{x}_{j,m+1}\right)^2 \to \min,$$

$$f(\widehat{x}_{j,m+1} + \Delta\widehat{x}_{j,m+1}) = y^*.$$

6.3.3.3 Алгоритм решения задачи при дискретных данных

Рассмотрим задачу оптимизации, в которой решение может принадлежать только набору возможных значений $\{\tilde{x}\}$. В этом случае алгоритм поиска решения будет включать следующие шаги.

Шаг 1. Решение обратной задачи с использованием исходных значений. Полученное решение – x^*.

Шаг 2. Поиск среди набора возможных значений аргументов величин, для которых эвклидово расстояние от полученного на шаге 1 решения минимально

$$\sqrt{\left(\tilde{x}_{1j} - x_1^*\right)^2 + \ldots + \left(\tilde{x}_{nj} - x_n^*\right)^2} \to \min .$$

Решением задачи будут значения $\tilde{x}_j$.

6.4 Выводы по главе 6

1) На основе объектно-ориентированного подхода разработана структура программной системы решения обратных задач. Ее основное отличие от предложенной в работах [163, 165] заключается в следующем: в основе декомпозиции лежит дерево целей, объект списка представляет собой переменную модели. Модификация базового подхода также заключается в вводе различных типов узлов, изменении процедуры обхода и включении новых классов.

2) Разработан алгоритм для решения общей задачи по формированию результирующего показателя дерева цели, позволяющий решать иерархические обратные задачи с ограничениями. Рассмотрено два варианта алгоритма в зависимости от способа решения подзадач: одновременное измененеи аргумента, выбор аргумента для изменения.

3) Использование выбранного подхода обуславливает возможность гибкой модификации и расширения системы: добавления новых показателей, механизмов расчёта и т.д. При его применении решение обратной задачи может быть выполнено пошагово для показателей, расположенных на различных уровнях. На каждом из шагов пользователь, оценив текущую ситуацию, может принять оптимальное решение, наилучшим образом использовать имеющиеся ресурсы для достижения той или иной цели.

4) Выполнена модификация и применение разработанных ранее алгоритмов для решения прикладных обратных задач: при использовании статистических данных, при наличии стохастичской зависимости между аргументами, при дискретных данных. Наличие статистических данных позволяет сформировать уравнение зависимости результирующего показателя от аргументов функции при стохастической зависимости, определять исходные значения аргументов для решения обратной задачи путем их прогноза, осуществлять поиск решения с учетом зависимости между аргументами функции, обеспечивая таким образом большее соответствие модели реальной системе и законам её поведения.

Предложенный алгоритм решения задачи при детерминированной зависимости результирующего показателя от формирующих его величин основан на прогнозировании значений аргументов и корректировке полученных величин методом обратных вычислений при минимизации суммы квадратов изменений аргументов. Предложенный алгоритм поиска решения при наличии стохастической зависимости между аргументами функции основан на вычислении показателя отдаленности от исходного решения, полученного путем решения обратной задачи, и характера расположения относительно предиктивного интервала.

Глава 7. Проблемно-ориентированное программное обеспечение решения задач на основе обратных вычислений

7.1 Разработка программы формирования прибыли

Программа решения иерархической многономенклатурной обратной задачи формирования прибыли предприятия [226, 227] предназначена для определения показателей образующих суммарную прибыль (прибыль по видам продукции; цена, себестоимость и объем продаж по видам продукции) с учётом ограничений, включая: загрузка и выгрузка данных в Excel; нахождение решения при использовании экспертной информации либо минимизации отклонений от исходных значений; решение задачи в случае двухуровневой и трехуровневой структуры дерева показателей. Программа может быть использована для определения значений цены, себестоимости, объема продаж, прибыли по наименованиям для достижения заданного значения суммарной прибыли. Использование программы позволяет автоматизировать расчёты и сократить временные затраты.

Предложенные алгоритмы и структура (п. 6.3.1, 6.3.2) были использованы для решения задачи формирования прибыли ресторана быстрого питания ООО «ВокиФудТомск». Испольуемая финансовая информация включала сведения о себестоимости каждого изделия, цене и объемах продаж за два года (2015–2016). На рисунках 7.1, 7.2 представлен фрагмент исходных данных.

Товары

с 11.01.2016 01:00:00 по 11.01.2016 23:59:59

Филиал: Основной

Точка продаж: Основная

Артикул	Наименование	Кол.	Стоимость	
Пельмешки ФРИ		4	76	руб.
	Кисло-сладкий	1	19	руб.
	Сладкий Васаби	1	19	руб.
	Соевый	2	38	руб.
устаревшие		1	150	руб.
	Суп : Том Кха с курицей	1	150	руб.
ЛапшаРИС ГОТОВЫЕ решения		18	3670	руб.
	.Рис неж. говядина Тонкацу 300г	1	190	руб.
	.Рис неж. говядина Тонкацу 450г	2	480	руб.
	300г	5	950	руб.
	.соба говяд. стр. фас. Тонкацу 300г	2	380	руб.
	.СОБА говяд. стр.фас.Тонкацу 450г	1	240	руб.
	.яичная ананас цып. терияки 300г	4	760	руб.
	.ЯИЧНАЯ ананас цып. терияки 450г	1	240	руб.
	.яичная свин. в устр соусе 300г	1	190	руб.
	.ЯИЧНАЯ свин. в устр соусе 450г	1	240	руб.
1. ОСНОВА		33	3520	руб.
	Гречневая лапша СОБА	4	400	руб.
	Гречневая лапша СОБА большая	1	120	руб.

Товары 11.01.2016 01-00-00-11.0

Рисунок 7.1 – Фрагмент данных о выручке предприятия ООО «ВокиФудТомск»

Себестоимость

на 14.11.2016

Наименование	Состав	Себестоимость	Цена
тигровые креветки.		40.5 руб. (32%)	125 руб.
Пшеничная лапша Удон .		18.35 руб. (15%)	119 руб.

Себестоимость 14.11.2016 (11)

Рисунок 7.2 – Фрагмент данных о себестоимости предприятия ООО «ВокиФудТомск»

Входные данные программы:

– исходные значения цены, себестоимости и объёма реализации для каждого вида продукции;

– коэффициенты важности, направления изменения, минимальные и максимальные значения прибыли от реализации продукции i-го наименования, цены, себестоимости и объёма продаж;

– целевое значение общей прибыли;

– способ решения задачи: с применением коэффициентов относительной важности и направлений изменения показателей; с применением коэффициентов относительной важности при наименьшем изменении показателей; при минимизации отклонений от исходных значений;

– вид задачи: двухуровневая или трехуровневая.

В случае двухуровневой задачи происходит определение цены, себестоимости, объема продаж по каждому наименованию для достижения заданного значения общей прибыли. При решении трехуровневой задачи сначала происходит определение прибыли по наименованию для достижения установленной суммарной прибыли, а затем решается задача нахождения себестоимости, объема продаж и цены для формирования прибыли по наименованию.

Выходные данные программы: значения цены, себестоимости, объема продаж и прибыли по наименованиям, обеспечивающие заданное значение целевой прибыли.

Дерево цели в случае трехуровневой модели представлено на рисунке 7.3. На данном рисунке представлен один из вариантов исходной экспертной информации (коэффициенты важности прибыли по наименованиям и направления изменения показателей), коэффициенты важности цены себестоимости и объема продаж для всех наименований равны 0,5, 0,2 и 0,3 соответственно.

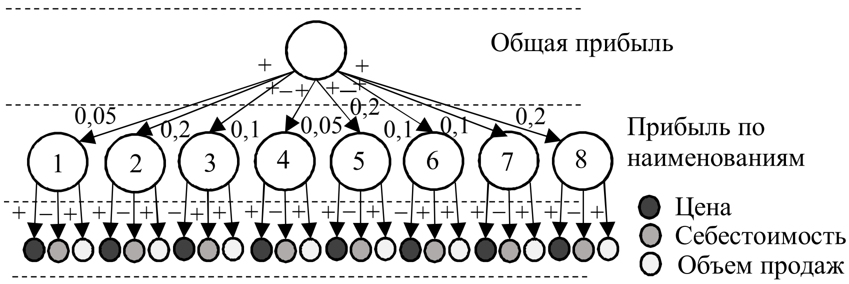


Рисунок 7.3 – Дерево цели задачи формирования прибыли

Тестирование программы осуществлялось путем вычисления метрик расстояния от заданных значений, решения отдельных подзадач с использованием математического пакета Excel, сравнения величин с установленными границами. Для проведения вычислительных экспериментов были использованы данные ресторана быстрого питания (таблица 7.1).

Таблица 7.1 – Исходные данные задачи

Продукт	Цена продажи, руб.	Себестоимость, руб.	Объем продаж	Прибыль, руб.
1. Гречневая лапша	100	19,1	153	12377,7
2. Пшеничная лапша	100	15,29	234	19822,14
3. Рис	100	12,02	280	24633
4. Рисовые спагетти	100	23,94	131	9963,86
5. Фунчоза	100	27,77	153	11051,19
6. Яичная лапша	100	17,25	387	32024,25
7. Вырезка из поросенка	70	21,08	147	7191,24
8. Говядина	90	24,52	178	11655,44

В таблице 7.2 представлены результаты решения задачи формирования прибыли, равной 160000 руб. (исходное значение общей прибыли 128718,82). Значения метрик расстояния от заданных значений представляют собой близкие к нулю величины.

Таблица 7.2 – Вычисленные значения показателей при использовании коэффициентов относительной важности и направлений изменения показателей (для первого уровня α=10^{-4}, для второго α=10^{-6})

Продукт	Цена продажи, руб.	Себестоимость, руб.	Объем продаж	Прибыль, руб.	$g(x)$	d
Гречневая лапша	109,62	15,25	158,77	14984,46	10^{-8}	$2{,}8\cdot10^{-5}$
Пшеничная лапша	126,05	4,87	249,63	30249,2	$9{,}96\cdot10^{-9}$	10^{-5}
Рис	85,17	17,95	288,89	19419,47	$9{,}99\cdot10^{-9}$	$1{,}6\cdot10^{-4}$
Рисовые спагетти	110,94	19,56	137,56	12570,62	$1{,}02\cdot10^{-8}$	$9{,}9\cdot10^{-5}$
Фунчоза	136,21	13,29	174,73	21478,25	$1{,}01\cdot10^{-8}$	$1{,}3\cdot10^{-5}$
Яичная лапша	89,59	21,41	393,24	26810,72	$1{,}02\cdot10^{-8}$	$8{,}4\cdot10^{-7}$
Вырезка из поросенка	90,65	12,82	159,39	12404,77	10^{-8}	$6{,}8\cdot10^{-5}$
Говядина	122,98	11,33	197,79	22082,5	$1{,}01\cdot10^{-8}$	$7{,}5\cdot10^{-5}$

Также рассмотрено решение этой задачи с учётом ограничений: для первых четырех изделий нижняя граница себестоимости равна 10, верхняя граница цены составляет 115, для последних четырех изделий верхняя граница цены равна 120, нижняя граница себестоимости – 15. Результаты решения представлены в таблице 7.3. В этом случае цены изделий с порядковыми номерами (таблица 7.3) 2, 5, 8 и себестоимости изделий с порядковыми номерами 2, 5, 7, 8 были скорректированы и равны граничному значению. Таким образом, были скорректированы коэффициенты относительной важности, полученные величины соотношения изменений приведены на рисунке 7.4.

Таблица 7.3 – Вычисленные значения показателей при наличии ограничений

Продукт	Цена продажи	Себестоимость	Объем продаж	Прибыль
Гречневая лапша	109,62	15,25	158,77	14984,46
Пшеничная лапша	115	10	288,09	30249,20
Рис	85,17	17,95	288,89	19419,47
Рисовые спагетти	110,94	19,56	137,56	12570,62
Фунчоза	120	15	204,55	21478,25
Яичная лапша	89,59	21,41	393,24	26810,72

Продукт	Цена продажи	Себестоимость	Объем продаж	Прибыль
Вырезка из поросенка	92,34	15	160,40	12404,77
Говядина	120	15	210,31	22082,50

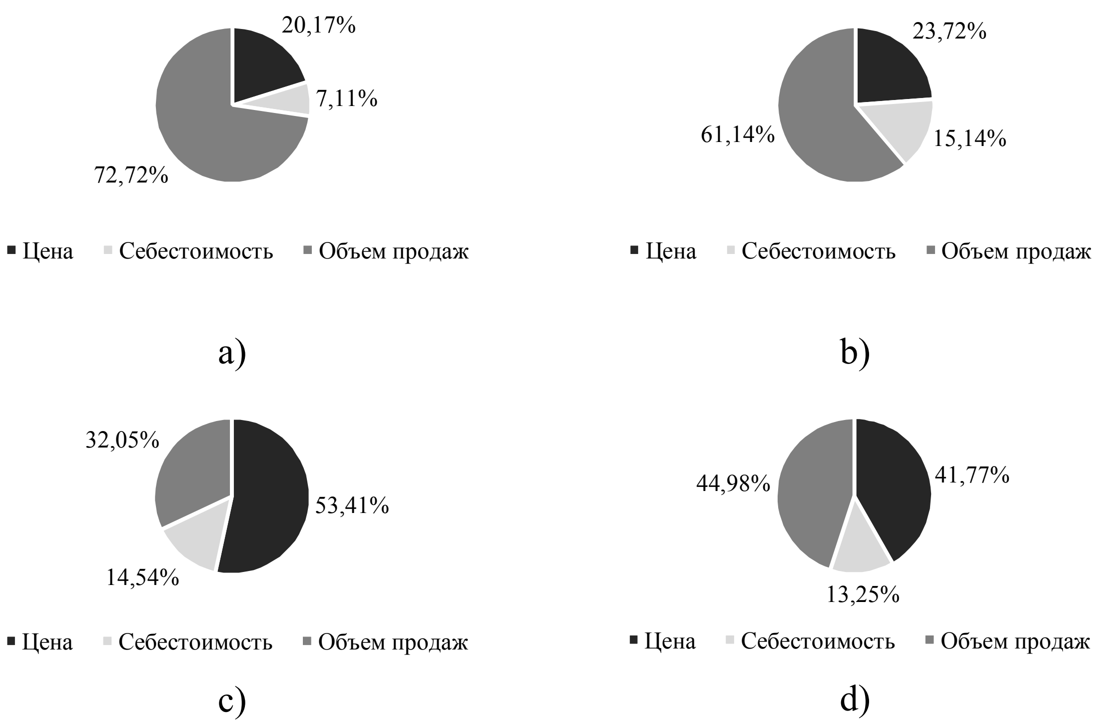


Рисунок 7.4 – Соотношение изменений показателей для продукции: a) пшеничная лапша; b) фунчоза; c) вырезка из поросенка; d) говядина

При минимизации суммы квадратов изменений показателей были определены две характеристики:

– $g(x)$ – сумма квадратов изменений показателей;

– d – абсолютная разница между полученным значением функции-ограничений $f(x)$ и заданным значением y^*.

В таблице 7.4 приведены результаты решения задачи формирования общей прибыли, а также прибыли по наименованиям с помощью реализованной программы и средств стандартного пакета. Рассматривая задачу как многокритериальную при минимизации двух параметров, можно сделать вывод, что для подзадачи первого уровня значения двух показателей оказались меньше для решения, полученного с помощью разработанного алгоритма. Для подзадач второго уровня использование алгоритма обеспечило меньшее значение показателя соответствия заданному ограничению, а использование математического пакета – функции $g(x)$.

Таблица 7.4 – Результаты решения задачи при минимизации суммы квадратов изменений аргументов

Подзадача	Решение с помощью программы		Решение с помощью пакета Excel	
	$g(x)$	d	$g(x)$	d
Формирование общей прибыли	122314059,014985	0,003995	122315034,395265	0,128719
Формирование прибыли, гречневая лапша	271,05273	$2{,}9\cdot10^{-5}$	270,8571151	$6\cdot10^{-3}$
Формирование прибыли, пшеничная лапша	138,413805	$4{,}8\cdot10^{-5}$	138,3721408	$3\cdot10^{-4}$
Формирование прибыли, рис	130,395605	$1{,}5\cdot10^{-4}$	130,3673111	$5\cdot10^{-4}$
Формирование прибыли, рисовые спагетти	346,375331	$1{,}6\cdot10^{-5}$	345,9028259	$4{,}3\cdot10^{-4}$
Формирование прибыли, фунчоза	275,231678	$8{,}8\cdot10^{-5}$	274,8970648	$2{,}3\cdot10^{-4}$
Формирование прибыли, яичная лапша	63,420252	$2{,}8\cdot10^{-4}$	63,41656061	$2\cdot10^{-2}$
Формирование прибыли, вырезка из поросенка	377,790356	10^{-4}	377,1783217	$3\cdot10^{-4}$
Формирование прибыли, говядина	216,896854	$1{,}4\cdot10^{-4}$	216,7093108	$4{,}6\cdot10^{-4}$

Также на основе предложенного в п. 6.3.2 подхода была реализована программа решения обратной задачи формирования прибыли с помощью стохастических алгоритмов [228], а также была реализована программа для проведения вычислительных экспериментов, результаты которых представлены в п. 2.2.2.

Описание интерфейса программы представлено в приложении Б.

7.2 Разработка программы формирования интегрального показателя

Дерево цели в случае формирования интегрального показателя (п.1.3.1.2) представлено на рисунке 7.5.

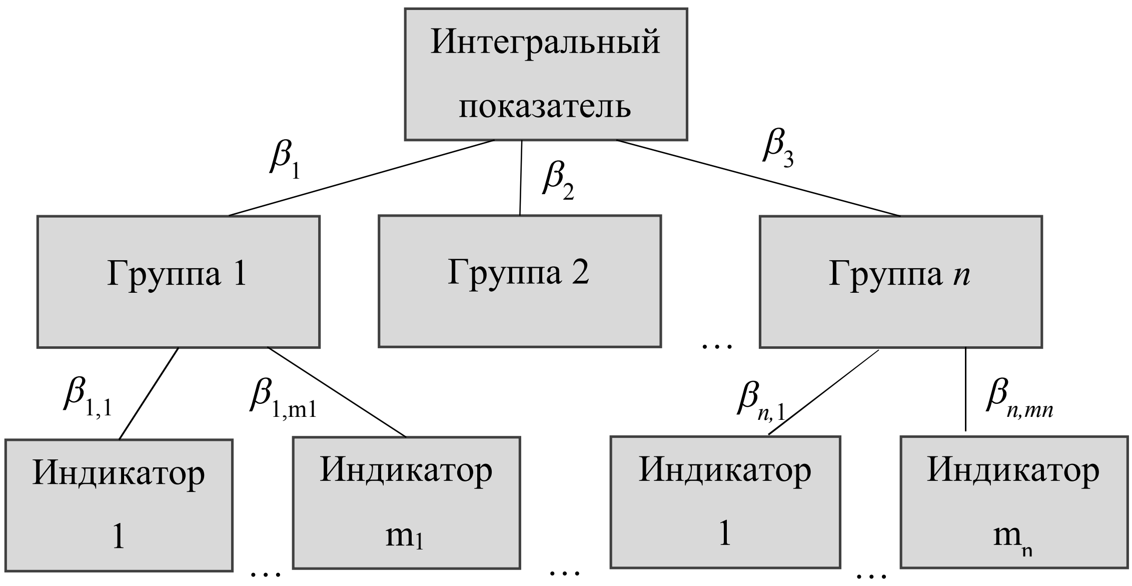


Рисунок 7.5 – Структура показателей

Последовательность решения задачи определения рейтинговой оценки представляется в виде графа, который содержит узлы трех типов: группа, индикатор и конечный узел (рисунок 7.6). Узлы типа «Индикатор» хранят исходные значения показателей и выполняют их нормировку, а также возврат к исходным значениям. Кроме того, в узлах такого типа осуществляется выбор значения для использования в расчётах (порядковый номер в списке либо прогнозное значение). Вершины «Группы» предназначены для преобразования значений своих потомков в единую интегральную характеристику. При этом также может быть выполнена нормировка с использованием значений узлов-потомков.

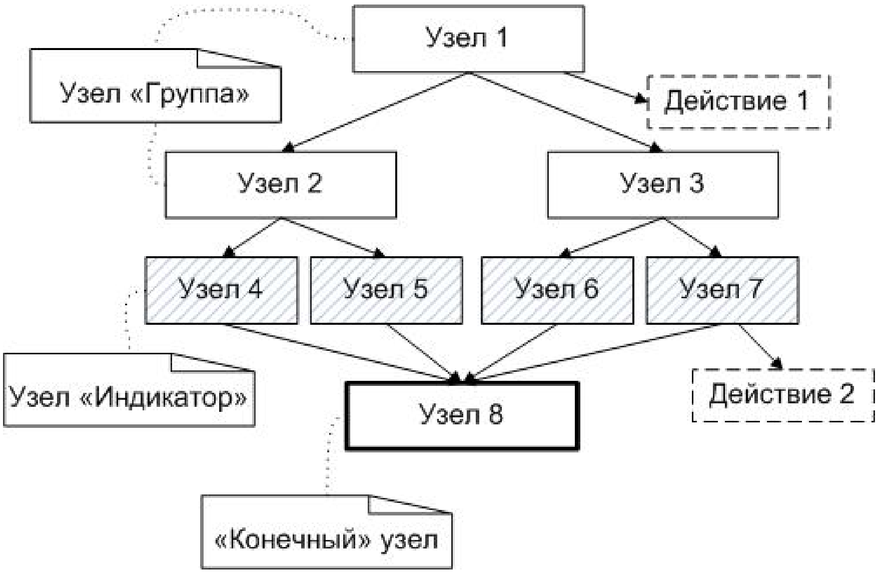


Рисунок 7.6 – Представление задачи в виде графа

Обход графа подразумевает обратный ход и прямой. В процессе прямого хода снизу/вверх осуществляется нормирование индикаторов и формирование интегральной характеристики каждого уровня. При прямом ходе вычисление узлов осуществляется последовательно, конечный узел является фиктивным и предназначен для запуска рекурсивной процедуры обхода графа, которая подразумевает вычисление текущей вершины только в том случае, если рассчитаны все её потомки. Алгоритм метода, вызываемого для фиктивного конечного узла представлен на рисунке 6.19. В результате его выполнения происходит рекурсивный вызов, метод `Расчёт` текущего узла вызывается в том случае, если значение атрибута `рассчитан` для всех потомков равно true. Так, для рисунка 7.6 в результате прямого хода последовательность вызова узлов представлена в таблице 7.6. Последовательность расчёта узлов следующая (указаны номера): 8, 4, 5, 2, 6, 7, 3, 1.

Обратный ход сверху/вниз запускается для узлов-групп, он предназначен для решения обратной задачи по определению значений узлов-потомков. При обратном ходе вычисление узлов, относящихся к одному предку, осуществляется параллельно. Таким образом, запускается рекурсивная процедура обхода потомков (на рисунке 7.7 представлена процедура `вызовПотомков`). Запустив обратный ход для узла с номером 1 графа на рисунке 7.5, получим, что сначала будут определены новые значения узлов 2 и 3 (решена обратная задача для узла 1), а затем 4, 5 (решена обратная задача для узла 2), 6, 7 (решена обратная задача для узла 3). Последовательность вызова узлов представлена в таблице 7.6.

Таблица 7.5 – Порядок обхода узлов и изменения их атрибута

Номер узла, для которого вызывается метод `вызовПредков`	Значение атрибута `рассчитан` (0–false, 1–true) после вызова метода `вызовПредков` узла							
	1	2	3	4	5	6	7	8
8	0	0	0	0	0	0	0	1
4	0	0	0	1	0	0	0	1
2	0	0	0	1	0	0	0	1
5	0	0	0	1	1	0	0	1
2	0	1	0	1	1	0	0	1
1	0	1	0	1	1	0	0	1
6	0	1	0	1	1	1	0	1
3	0	1	0	1	1	1	0	1
7	0	1	0	1	1	1	1	1
3	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1

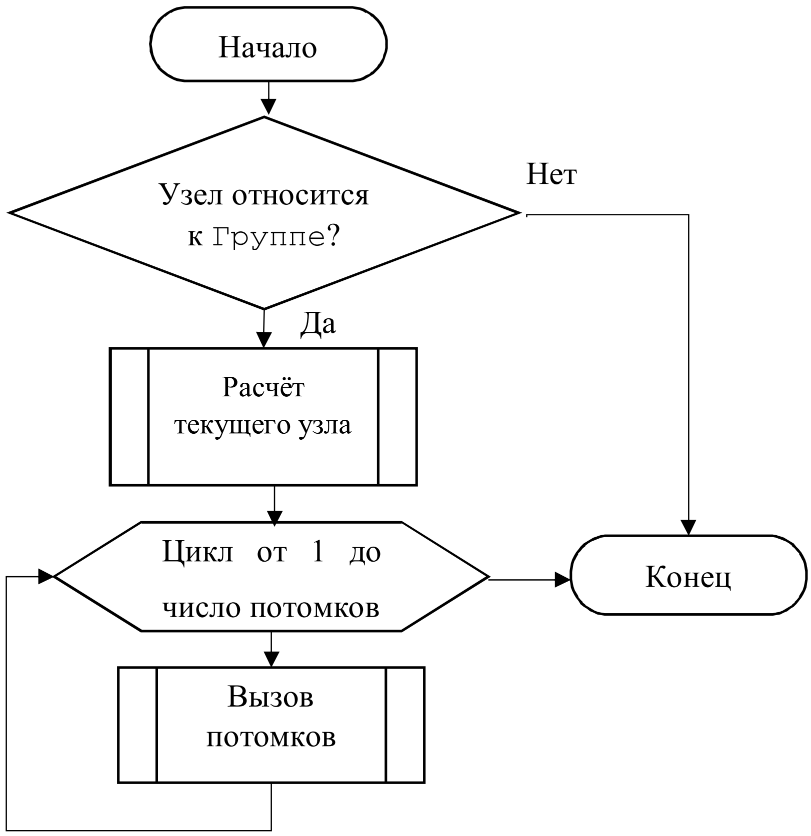


Рисунок 7.7 – Алгоритм метода `вызовПотомков`

Таблица 7.6 – Порядок обхода узлов при решении обратной задачи

Номер узла, для которого вызывается метод `вызовПотомков`	Признак решения обратной задачи узла							
	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	1
2	0	1	0	0	0	0	0	1
4	0	1	0	0	0	0	0	1
5	0	1	0	0	0	0	0	1
3	0	0	1	1	0	0	0	1
6	0	0	1	1	0	0	0	1
7	0	0	1	1	0	0	0	1

Диаграмма классов представлена на рисунке 7.8.

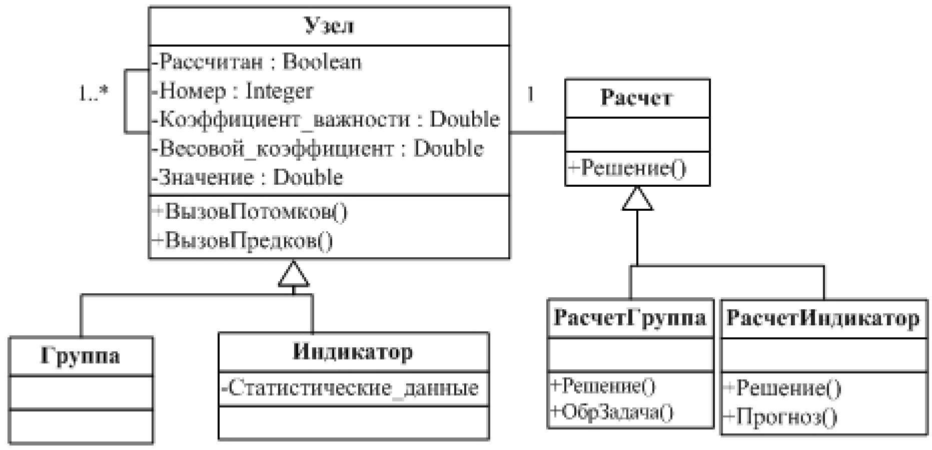


Рисунок 7.8 – Диаграмма классов

Данная структура была реализована на языке Java для создания программы формирования интегрального показателя социально-экономического объекта и системы рейтинговой оценки объектов экономики [229, 230].

7.2.1 Оценка и прогнозирование уровня социально-экономического развития регионов

Представленный алгоритм был использован для решения задачи моделирования рейтинга регионов Сибирского федерального округа (СФО) [230]. Интегральная характеристика социально-экономической деятельности региона (п.1.3.1.2) сформирована на основе восьми групп показателей (рисунок 7.9): уровень жизни, финансовая обеспеченность, эффективность сельскохозяйственного производства, эффективность строительства, обеспеченность трудовыми ресурсами, состояние системы здравоохранения, обеспеченность объектами образования, обеспеченность информационными и коммуникационными технологиями (ИиКТ). Группы также формируются из показателей более низкого уровня (всего было использовано 48 показателей, которые представлены в приложении Г). Для автоматизации расчета интегральной характеристики и хранения данных была реализована программа, описание которой представлено в приложении Д. Нормирование показателей выполнялось с помощью метода эталонного значения, который подразумевает деление значения показателя на максимальное либо деление минимального значения на величину показателя (в зависимости от того, какое значение является наилучшим: минимальное или максимальное). Таким образом, значения всех показателей изменяется в пределах от 0 до 1. На рисунках 7.10 – 7.12 представлены примеры исходных и нормированные значения показателей регионов. Расчет рейтинговой оценки был выполнен без использования коэффициентов относительной важности. Вычисление показателей групп (рисунок 7.9) происходит путем суммирования нормированных значений формирующих его показателей (расположенных уровнем ниже). Так, на рисунке 7.10 представлен пример вычисления интегрального показателя.

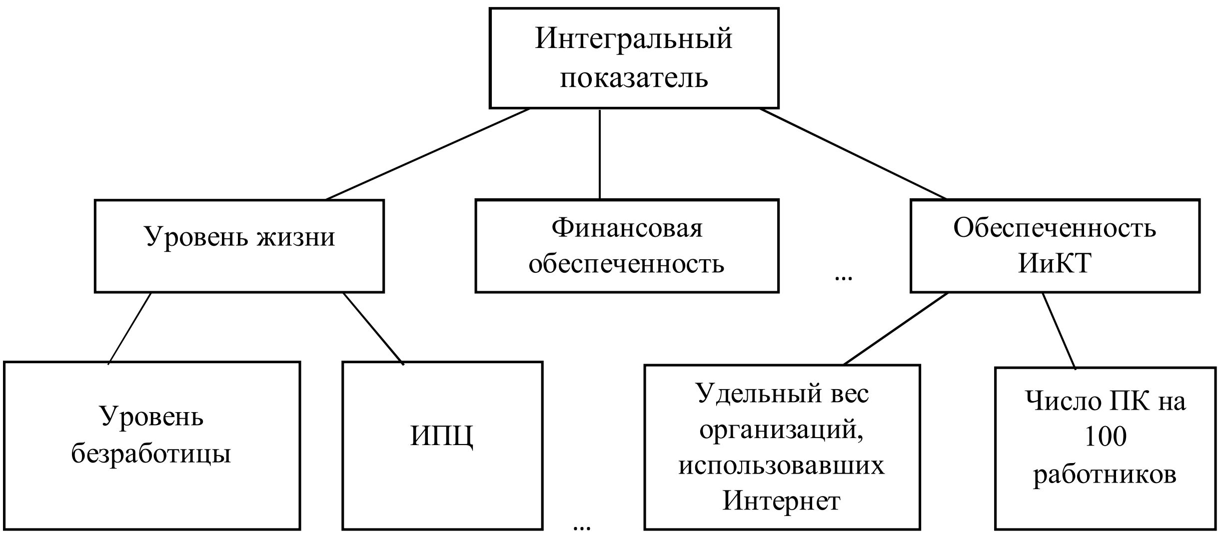


Рисунок 7.9 – Формирование интегрального показателя

	A	B	C	D	E	F	G	H	I
1			Коэф. естественного прироста (промилле)						
2			2010	2011	2012	2013	2014	2015	2016
3	1	Республика Алтай	8,3	10,5	10,9	9,8	9,6	7,9	8,1
4	2	Республика Бурятия	4,3	4,3	5,1	5,8	6	5,9	5,2
5	3	Республика Тыва	15,2	16,5	15,5	15,2	14,4	13,5	13,4
6	4	Республика Хакасия	1,2	1,7	2,6	2,6	2,1	1,3	1,3
7	5	Алтайский край	-2,3	-1,9	-0,9	-0,8	-1	1,5	-2
8	6	Забайкальский край	2,1	2,2	3,1	3,4	3,5	2,5	2,3
9	7	Красноярский край	0,1	0,5	1,5	1,7	1,7	1,7	1,4
10	8	Иркутская область	0,8	1,3	2	2,1	1,6	1,7	1,4
11	9	Кемеровская область	-3	-2,8	-1,4	-0,9	-1,4	-2	-2,2
12	10	Новосибирская область	-0,7	-0,5	0,3	0,7	0,7	1,1	0,8
13	11	Омская область	-0,7	0,1	1,1	1,3	1,9	1	0,1
14	12	Томская область	0,4	0,8	1,7	2,1	2	2,1	1,8
154			Индекс потребительских цен (в % к соотв.периоду прошлого года)						
155			2010	2011	2012	2013	2014	2015	2016
156		СФО	107,9	106,3	106,7	106,1	110,8	111,6	105
157	1	Республика Алтай	108,6	106,4	107	106,4	110,1	112,5	103,8
158	2	Республика Бурятия	109,4	107,5	106,8	107,5	111,8	110,7	104,2
159	3	Республика Тыва	108,3	107,3	107,5	105,4	109,5	111,4	104,3
160	4	Республика Хакасия	108,4	107,6	106	105,4	110,5	110,3	104,4
161	5	Алтайский край	108,2	104,8	107,2	107	111,2	112,4	105,8
162	6	Забайкальский край	109	107,8	105,6	108,3	111,2	114,3	105
163	7	Красноярский край	107,9	106,1	106,8	104,8	109,5	110,6	104,7

Ур-нь жизни / Фин.обеспеченности / С-х произ-во / Строительство / Обесп.труд.ресурсами / Сист

Рисунок 7.10 – Исходные значения показателей

		Коэф. естественного прироста (промилле)						
		2010	2011	2012	2013	2014	2015	2016
1	Республика Алтай	0,620879	0,689119	0,727811	0,664596	0,696203	0,63871	0,660256
2	Республика Бурятия	0,401099	0,367876	0,384615	0,416149	0,468354	0,509677	0,474359
3	Республика Тыва	1	1	1	1	1	1	1
4	Республика Хакасия	0,230769	0,233161	0,236686	0,217391	0,221519	0,212903	0,224359
5	Алтайский край	0,038462	0,046632	0,029586	0,006211	0,025316	0,225806	0,012821
6	Забайкальский край	0,28022	0,259067	0,266272	0,267081	0,310127	0,290323	0,288462
7	Красноярский край	0,17033	0,170984	0,171598	0,161491	0,196203	0,23871	0,230769
8	Иркутская область	0,208791	0,212435	0,201183	0,186335	0,189873	0,23871	0,230769
9	Кемеровская область	0	0	0	0	0	0	0
10	Новосибирская область	0,126374	0,119171	0,100592	0,099379	0,132911	0,2	0,192308
11	Омская область	0,126374	0,150259	0,147929	0,136646	0,208861	0,193548	0,147436
12	Томская область	0,186813	0,186528	0,183432	0,186335	0,21519	0,264516	0,25641
		Индекс потребительских цен (в % к соотв.периоду прошлого года)						
		2010	2011	2012	2013	2014	2015	2016
1	Республика Алтай	0,977901	0,984962	0,985047	0,984962	0,99455	0,980444	1
2	Республика Бурятия	0,97075	0,974884	0,986891	0,974884	0,979428	0,996387	0,996161
3	Республика Тыва	0,980609	0,976701	0,980465	0,994307	1	0,990126	0,995206
4	Республика Хакасия	0,979705	0,973978	0,99434	0,994307	0,99095	1	0,994253
5	Алтайский край	0,981516	1	0,983209	0,979439	0,984712	0,981317	0,981096
6	Забайкальский край	0,974312	0,972171	0,998106	0,967682	0,984712	0,965004	0,988571
7	[illegible]	[illegible]	[illegible]	[illegible]	1	1	[illegible]	[illegible]

Строительство / Обесп.труд.ресурсами / Система здравоохр. / Обесп.объект.обр. / Обесп.ИиКТ / Балл г

Рисунок 7.11 – Нормированные значения показателей

1 Республика Алтай								
№	Название группы	2010	2011	2012	2013	2014	2015	2016
1	Уровень жизни	7,66	7,76	7,64	7,52	7,75	7,83	7,75
2	Финансовая обеспеченность	4,41	3,62	3,31	3,35	3,52	3,58	3,70
3	Сельскохозяйственное производство	2,17	2,16	2,33	2,00	2,08	2,05	1,84
4	Строительство	2,36	2,41	2,08	1,90	2,00	2,09	2,63
5	Обеспеченность трудовыми ресурсами	3,94	3,92	3,89	3,90	3,84	3,87	3,81
6	Система здравоохранения	3,82	3,78	3,82	3,89	3,96	4,00	3,95
7	Обеспеченность объектами образования	1,76	1,76	1,84	1,80	1,71	1,80	1,80
8	Обеспеченность ИиКТ	4,68	4,61	5,10	5,11	5,36	5,21	5,24

Рисунок 7.12 – Вычисление оценок групп

На рисунке 7.13 представлен рейтинг регионов СФО за 2015 год. Можно сказать, что лидирующие позиции занимают такие регионы округа, как Красноярский край, Новосибирская, Иркутская, Кемеровская области.

Значения интегрального показателя было рассчитано как для сравнения региона с другими субъектами, так и для оценки динамики его развития. Нормированные значения исходных данных (групп показателей), на основе которых осуществляется расчет интегральной характеристики в этом случае, приведены в таблице 7.8. Рассмотрим обратную задачу: определение значений показателей, обеспечивающих в дальнейшем прирост интегральной характеристики на 2%. В этом случае ее значение должно получиться равным 7,104. При этом среднее значение показателя не может превышать 1 (обратная задача с ограничением).

Рисунок 7.13 – Рейтинг регионов Сибирского федерального округа в 2015 г.

В последнем столбце таблицы 7.7 представлено решение обратной задачи. Для того чтобы определить необходимые абсолютные значения показателей групп, выполняется решение обратной задачи с заданным значением интегральной характеристики группы. После этого осуществляется обратный переход от нормированных значений к величинам в исходных единицах измерения.

Таблица 7.7 – Решение задачи формирования интегрального показателя для региона Тыва

Название группы	Год						Прогноз	Решение
	2010	2011	2012	2013	2014	2015		
Уровень жизни	0,906	0,939	0,929	0,925	0,910	0,903	0,908	0,932
Финансовая обеспеченность	0,690	0,694	0,745	0,732	0,776	0,727	0,769	0,794
Сельскохозяйственное производство	0,631	0,694	0,699	0,736	0,758	0,768	0,805	0,830
Строительство	0,413	0,340	0,409	0,551	0,637	0,735	0,778	0,803
Обеспеченность трудовыми ресурсами	0,916	0,932	0,911	0,849	0,838	0,864	0,824	0,849
Система здравоохранения	0,970	0,958	0,965	0,955	0,937	0,941	0,933	0,957
Обеспеченность объектами образования	0,933	0,937	0,922	0,906	0,925	0,921	0,913	0,938
Обеспеченность ИиКТ	0,644	0,726	0,737	0,848	0,882	0,988	1,034	1,000
$f(x)$	6,102	6,219	6,316	6,502	6,662	6,847	6,965	7,104

7.2.2 Оценка групп социальной сети для реализации маркетинговых мероприятий

Также рассмотрим модель интегрального показателя, которая была использована для оценки групп социальной сети ВКонтакте с целью размещения рекламы [231–234] (приложение Е). Для формирования рейтинговой оценки были отобраны 50 групп и выбраны следующие характеристики групп:

– число подписчиков;

– активность – среднее количество записей на стене группы в день;

– популярность – сумма лайков, репостов и комментариев, деленная на число записей.

Дерево цели представлено на рисунке 7.14.

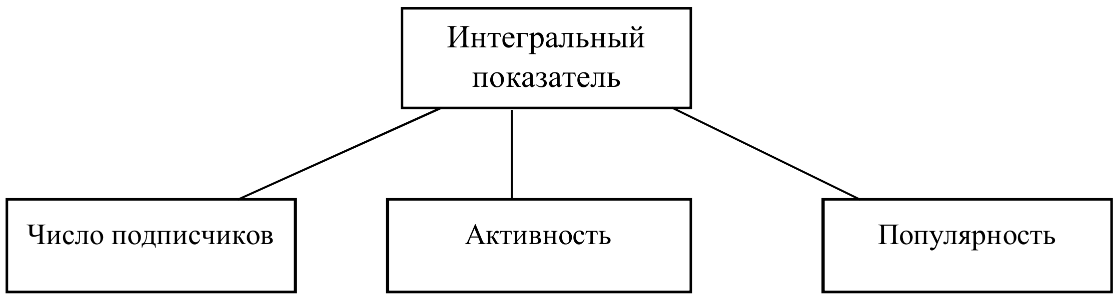


Рисунок 7.14 – Дерево цели для оценки групп социальной сети

Определение коэффициентов важности показателей было выполнено с помощью метода анализа иерархий. В качестве группы экспертов были выбраны менеджеры фирм, занимающиеся размещением рекламы в группах. Матрица попарных сравнений представлена в таблице 7.8 (полученные значения коэффициентов относительной важности: 0,701; 0,24; 0,059).

Таблица 7.8 – Матрица попарных сравнений

	Число подписчиков	Популярность	Активность
Число подписчиков	1	5	7
Популярность	1/5	1	7
Активность	1/7	1/7	1

Интегральная оценка вычисляется по формуле

$$R_j = 0,701x_{1j} + 0,24x_{2j} + 0,059x_{3j},$$

где x_{1j} – нормированное значение числа подписчиков j-той группы;

x_{2j} – нормированное значение показателя популярности j-той группы;

x_{3j} – нормированное значение показателя активности j-той группы.

Для выбранной группы нормированные значения числа подписчиков, популярности, активности равны соответственно 0,313, 0,004 и 0,029.

Рассмотрим задачу определения таких новых значений x, которые обеспечат значение рейтинговой оценки, равное 0,3 при минимизации сумм квадратов изменений аргументов

$$0,701(0,313 + \Delta x_1) + 0,24(0,004 + \Delta x_2) + 0,059(0,029 + \Delta x_3) = 0,3.$$

Согласно представленным в п.3.2.1 выражениям можно получить аналитическое решение данной задачи. При этом выражения для двух алгоритмов будут идентичны.

Решение задачи: Δx_1=0,099, Δx_2=0,034, Δx_3=0,008. Таким образом, искомые значения будут равны

$x_1^* = 0{,}313 + 0{,}099 = 0{,}412;$

$x_2^* = 0{,}004 + 0{,}034 = 0{,}038;$

$x_3^* = 0{,}029 + 0{,}008 = 0{,}037.$

Можно отметить, что в качестве модели интегральной оценки может быть рассмотрена регрессионная модель при стохастической зависимости результирующего показателя от аргументов. В этом случае определение коэффициентов осуществляется с помощью метода наименьших квадратов. Решение обратной задачи определяется с использованием полученного уравнения связи. В частности это может быть использвано для решения обратных задач с использованием имитационных моделей [235].

7.2.3 Оценка момента времени размещения сообщения в социальной сети

Данный алгоритм был использован для оценки момента времени размещения сообщения в группах онлайновой социальной сети. Онлайновые социальные сети являются популярными площадками по продвижению товаров и услуг, распространению информации социального и политического характера [236–240]. Социальные сети включают большое количество участников, поэтому грамотное определение способа размещения информации способно привести к её широкому распространению. Одним из важнейших факторов, определяющих число просмотров сообщения, является время его размещения. При выборе момента размещения сообщения необходимо выполнить анализ ряда показателей. Наилучшим временем размещения сообщения считается тот момент, когда большое число пользователей находятся в статусе «онлайн», так как в этом случае они с высокой вероятностью могут увидеть эту информацию. Кроме того, при определении момента размещения следует учитывать и ряд других показателей, таких, например, как число сообщений, публикуемых другими участниками социальной сети.

Для проведения исследования была реализована программа, выполняющая автоматический сбор информации из социальной сети ВКонтакте для выбранной группы («Golden time Anime») через каждые 5 минут [241–244]. Интерфейс программы представлен в приложении Е. На основе собранной информации были вычислены две характеристики:

1) Суммарная оценка просмотра сообщения, которая отражает тот факт, что участники находятся в режиме онлайн в данный момент. Наилучшим временем размещения сообщения считается тот момент, когда большее число пользователей находится в статусе «онлайн», так как в этом случае они с высокой вероятностью могут увидеть эту информацию.

2) Скорость обновления ленты, которая определяется числом публикуемых сообщений участниками. Если скорость новостной ленты очень высока, то опубликованное участником сообщение может быть смещенно другими постами и не прочитано.

В таблице 7.9 представлены сведения о средней доли участников, находящихся в статусе онлайн. Можно увидеть, что в вечернее время этот показатель достигает максимального значения.

Таблица 7.9 – Средняя доля участников, находящихся в статусе «онлайн»

Время	День недели						
	Пн.	Вт.	Ср.	Чт.	Пт.	Сб.	Вс.
0:00	0,2189	0,2220	0,2229	0,2218	0,2179	0,2222	0,2054
1:00	0,2143	0,2155	0,2220	0,2176	0,2117	0,2064	0,2173
2:00	0,2059	0,2131	0,2151	0,2188	0,2121	0,2025	0,2009
3:00	0,2001	0,1902	0,1954	0,1938	0,1888	0,1885	0,1885
4:00	0,1720	0,1525	0,1577	0,1579	0,1567	0,1548	0,1665
5:00	0,1311	0,1082	0,1126	0,1078	0,1090	0,1165	0,1401
6:00	0,0891	0,0734	0,0733	0,0722	0,0728	0,0841	0,0991
7:00	0,0603	0,0534	0,0532	0,0560	0,0547	0,0581	0,0649
8:00	0,0537	0,0505	0,0466	0,0469	0,0493	0,0487	0,0493
9:00	0,0474	0,0502	0,0520	0,0491	0,0497	0,0414	0,0442
10:00	0,0565	0,0622	0,0663	0,0610	0,0666	0,0521	0,0447
11:00	0,0713	0,0818	0,0897	0,0910	0,0889	0,0643	0,0558
12:00	0,0943	0,1202	0,1191	0,1183	0,1187	0,0897	0,0688
13:00	0,1163	0,1272	0,1221	0,1307	0,1351	0,1058	0,0966
14:00	0,1309	0,1375	0,1384	0,1453	0,1499	0,1290	0,1304
15:00	0,1603	0,1607	0,1587	0,1576	0,1640	0,1493	0,1595

Время	День недели						
	Пн.	Вт.	Ср.	Чт.	Пт.	Сб.	Вс.
16:00	0,1789	0,1718	0,1708	0,1736	0,1721	0,1710	0,1743
17:00	0,1792	0,1823	0,1779	0,1792	0,1805	0,1775	0,1921
18:00	0,1919	0,1915	0,1901	0,1797	0,1858	0,1867	0,1975
19:00	0,2007	0,2023	0,1961	0,1962	0,1965	0,1894	0,1965
20:00	0,2060	0,2065	0,1971	0,2045	0,2065	0,1963	0,2034
21:00	0,2118	0,2047	0,2096	0,2102	0,2016	0,1937	0,2078
22:00	0,2179	0,2142	0,2158	0,2208	0,1986	0,2038	0,2143
23:00	0,2193	0,2224	0,2221	0,2162	0,2159	0,2062	0,2203

На рисунке 7.15 представлены значения среднего числа публикуемых сообщений в новостной ленте, при этом рассмотрен период с 21:00 до 02:00 часов. Из рисунка 7.15 можно сделать вывод, что на начало каждого часа приходятся пиковые значения числа публикуемых сообщений, таким образом, это время является наиболее популярным для размещения сообществами постов.

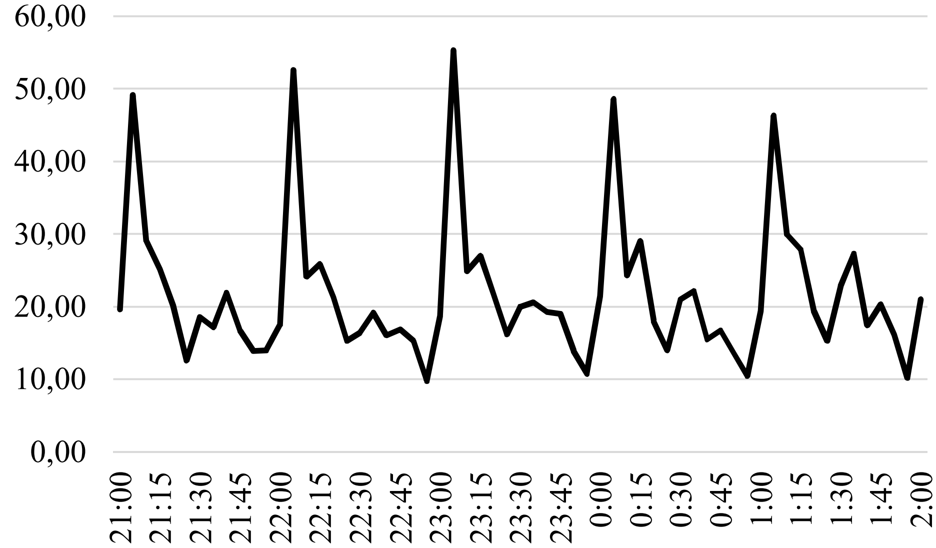


Рисунок 7.15 – Число публикуемых сообщений в новостной ленте

Величина суммарной оценки и скорость обновления ленты нормируются и выполняется линейная свёртка двух показателей. В результате будет получена интегральная оценка *I* момента времени (п.1.3.1.2)

$$I = k_1 \cdot x_1 + k_2 \cdot x_2,$$

где x_1 – нормированное значение суммарной оценки просмотра сообщения;

x_2 – нормированное значение скорости обновления ленты;

k_1, k_2 – устанавливаемые экспертом весовые коэффициенты показателей x_1, x_2 соответственно.

Дерево цели представлено на рисунке 7.16.

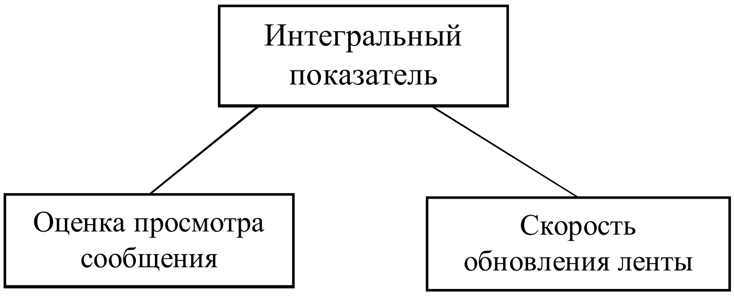


Рисунок 7.16 – Дерево цели для оценки момента времени размещения сообщения

В таблице 7.10 представлен фрагмент данных, в котором отражены для каждого момента времени нормированные значения оценки просмотра и скорости обновления ленты, а также величина интегрального показателя.

Таблица 7.10 – Данные, характеризующие моменты времени

Момент времени	Нормированное значение оценки просмотра	Нормированное значение скорости обновления ленты	Интегральный показатель
19:00	0,953	0,552	0,792
19:05	0,954	0,684	0,846
19:10	0,956	0,732	0,866
19:15	0,957	0,732	0,867
19:20	0,958	0,711	0,859
19:25	0,96	0,719	0,863
19:30	0,961	0,734	0,871
19:35	0,963	0,777	0,888
19:40	0,964	0,807	0,901
19:45	0,965	0,655	0,841
19:50	0,966	0,599	0,819
19:55	0,967	0,555	0,803
20:00	0,97	0,541	0,798
20:05	0,97	0,689	0,859

Момент времени	Нормированное значение оценки просмотра	Нормированное значение скорости обновления ленты	Интегральный показатель
20:10	0,975	0,722	0,874
20:15	0,978	0,727	0,877
20:20	0,980	0,735	0,882
20:25	0,983	0,738	0,885
20:30	0,985	0,761	0,896
20:35	0,988	0,802	0,913
20:40	0,99	0,789	0,909
20:45	0,992	0,654	0,857
20:50	0,994	0,59	0,832
20:55	0,996	0,534	0,811
21:00	0,997	0,532	0,811

Соответствующие наилучшему значению интегрального показателя нормированные значения характеристики просмотра и числа публикумых сообщений равны соответственно x_1 = 0,988, x_2= 0,802 (соответствующий момент времени равен 20:35, весовые коэффициенты k_1 и k_2 равны соответственно 0,6 и 0,4). Пусть необходимо изменить нормированные значения, таким образом, чтобы улучшить оценку скорости обновления ленты за счёт ухудшения оценки просмотра. Суммарное изменение нормированных значений должно составить –0,015, при этом коэффициенты относительной приоритетности будут равны 0,7 и 0,3 (для характеристики просмотра и скорости обновления ленты соответственно). Тогда система будет иметь вид

$$\begin{cases} \dfrac{\Delta x_1}{\Delta x_2} = -\dfrac{0,7}{0,3}, \\ 0,6\left(x_1 + \Delta x_1\right) + 0,4\left(x_1 + \Delta x_2\right) = 0,913 - 0,015. \end{cases}$$

Решением системы будут значения Δx_1 = –0,036, Δx_2 =0,015. Тогда новые значения нормированных величин составят

$$x_1^* = x_1 + \Delta x_1 = 0,952,$$

$$x_2^* = x_2 + \Delta x_2 = 0,817.$$

Далее определяется момент времени i, для которого отклонение полученных нормированных значений от исходных минимально

$$\sqrt{\left(\tilde{x}_{1i} - x_1^*\right)^2 + \left(\tilde{x}_{2i} - x_2^*\right)^2} \to \min.$$

Минимальное значение расстояния равное 0,016 соответствует моменту времени 19:40.

Таким образом, рассмотренный метод может быть применен для решения обратных задач при фиксированном наборе вариантов решения. В качестве ещё одного примера такой задачи может быть рассмотрена задача по выбору групп социальной сети для размещения рекламы [233].

7.3 Внедерение результатов диссертации на предприятиях и в учреждениях

Результаты диссертационного исследования в виде математических моделей, алгоритмов и программного обеспечения поддержки принятия решений были переданы следующим компаниям: ООО «Томская нефть», АО «Разрез «Степановский», ООО «Вокифудтомск», ООО «Титан», ООО «ФОРС», ООО «Интенс-строй», ООО «Дельта», МАОУ ДО Дворец творчества детей и молодежи, ООО «Гамарджоба», ООО «Сибмед», ООО «Система автоматизация бизнес», ФДО ТУСУР.

ООО «Томская нефть» ведет разработку десяти нефтяных месторождений (Столбовое, Федюшкинское, Грушевое, Дуклинское, Поселковое, Южно-Мыльджинское, Верхнесалатское, Соболиное, Гураринское, Ясное) и одного газоконденсатного (Речное) на территории Томской области. Предприятию была передана программа решения иерархической многономенклатурной обратной задачи формирования прибыли предприятия. Программа осуществляет расчёт прибыли по наименованиям, цены, себестоимости, объема продаж для достижения заданного значения суммарной прибыли. Использование программы обеспечивает применение новых методов для оперативного принятия решений и сокращение рабочего времени на проведение расчётов на 20%.

АО «Разрез «Степановский» – организация, расположенная в г. Новокузнецк, основным видом деятельности которой является добыча угля. Передана программа решеная иерархической многономенклатурной обратной задачи формирования прибыли предприятия для информационной поддержки принятия решений. Автоматизация расчётов позволяет сократить временные ресурсы (на 25%) и оперативно получить информацию для принятия решений управленческих решений.

ООО «Вокифудтомск» – это лапшичная паназиатской кухни «Woki». Директор организации планирует открытие новых торговых точек по продаже кофе, в связи с чем ему была передана программа формирования интегрального показателя для решения задачи выбора размещения новых торговых точек. Программа позволяет вычислять интегральный показатель размещения на основе совокупности характеристик: район, наличие поблизости парка, расстояние до центра и т.д. и статистических данных о действующих пунктах продажи (фрагмент статистических данных представлен на рисунке 7.17), и обеспечивает сокращение времени на анализ данных (повышение производительности труда более 300%).

2019	Число сотрудников	Выручка	Прибыль, млн.	УК	Октябрьский район	Ленинский район	Советский район	До остановки,м	До центра	Парки	Учебные заведения
ООО МАСТЕР КОФЕ	4	26,9	3,81	50000	0	1	0	114	1,84 км.	0	0
ООО КОФЕ-ШОП	5	6,83	0,358	23000	0	1	0	99	2,13	0	0
ООО КАКОЙ ВКУСНЫЙ КОФЕ	1	0,101	0,018	10000	0	0	0	78	3,95	0	0
ООО КОФЕ-ЛАЙТ	21	25,7	1,52	10000	0	0	0	41	2,48	1	1
ООО КОФЕ-ДАРК	1	1,72	0,45	10000	0	0	1	84	2	0	0
ООО СЕРВИС-КОФЕ	1	12,99	0,574	10000	1	0	0	73	7	0	0
ООО КОФЕ-АРТ	27	28,68	1,42	10000	0	1	0	26	0,558	1	1
ООО КОФЕ-ФРЕШ	26	31,2	1,87	10000	0	1	0	49	0,479	1	0

Рисунок 7.17 – Фрагмент статистических данных

Также с помощью данных, предоставленных данной организацией, была выполнена апробация алгоритмов и моделей формирование прибыли и выручки.

ООО «Титан» специализируется на деятельности по заготовке, хранению, переработке и реализации лома цветных и черных металлов. Переданные материалы включают модели формирования и прогнозирования выручки и программу «Анализ выручки», использование которых обеспечивает информацией для принятия решений относительно планирования финансового-хозяйственной деятельности (необходимый запас денежных средств в филиалах, периодичность организации вывоза металла и т.д.). Применение программы позволило получить экономический эффект в виде снижения времени, затрачиваемого на планирование, на 20%.

ООО «ФОРС» специализируется на торговле медицинскими изделиями, в том числе газовыми баллонами. С использованием данных организации о ежедневном спросе и остатках (рисунок 7.18) и модели управления запасами было выполнено исследование стратегий управления запасами для определения оптимальных значений управляемых переменных (объем доставки, минимальный уровень запаса). Предложенная стратегия позволяет сократить размер запаса на складе в среднем на 40% и таким образом сэкономить оборотные средства предприятия (рисунок 7.19).

Остаток полных больших баллонов на складе на 01.03.21

Газ	остаток на начало дня 26.02.2021	приход	расход	остаток на начало дня 01.03.2021
Ацетилен	768,00	75	3	840,00
Кислород	548,00	0	69	479,00
Пропан	69,00	44	27	86,00
Аргон	112,00	0	12	100,00
Углекислота	47,00	0	13	34,00
Азот	0,00	88	28	60,00

Рисунок 7.18 – Исходные данные о ежедневном спросе

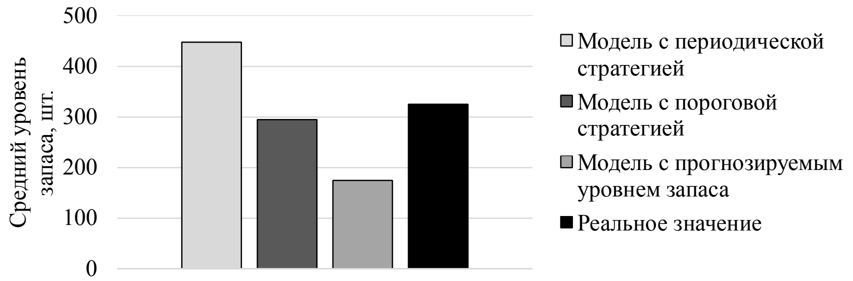


Рисунок 7.19 – Значения среднего уровня запаса

ООО «Интенс-строй» занимается предоставлением услуг по ремонту и обслуживанию зданий и помещений. Переданные материалы включают стохастический алгоритм и программу решения обратной задачи формирования прибыли с помощью стохастических алгоритмов. Программа позволяет формировать обоснованную стратегию финансово-хозяйственной деятельности с целью повышения финансовых результатов деятельности предприятия. Экономический эффект применения программы заключает в автоматизации расчётов и сокращении временных затрат на 30%.

ООО «Дельта» представляет собой фирму-франчайзи 1С и работает на рынке Юрги и Кемеровской области. Стратегия развития организации предполагает привлечение новых клиентов и партнеров, в связи с чем планируется проведение маркетинговых мероприятий в социальных сетях. Для организации продвижения организации в социальной сети ВКонтакте и выбора сообществ для эффективной рекламы руководству были переданы материалы в виде модели и программы оценки групп социальной сети для реализации маркетинговых мероприятий. Использование программы обеспечивает сокращение времени на обработку данных социальной сети более чем на 90%.

МАОУ ДО Дворец творчества детей и молодежи специализируется на предоставлении дополнительных образовательных услуг детям и молодежи г. Томска. Организация заинтересована в привлечении обучающихся и занимается продвижением своих услуг с помощью онлайновой социальной сети ВКонтакте, в том числе с помощью сообщества ("Дворец творчества детей и молодежи" г.Томск, https://vk.com/dtdm_tomsk), поскольку данная социальная сеть является наиболее популярной среди молодой аудитории. Заместителю директора по развитию и администратору был передана программа оценки времени размещения сообщения в группах онлайновой социальной сети для выбора наилучшего момента для размещения сообщений в группе "Дворец творчества детей и молодежи" г.Томск, а также постов рекламного характера в других сообществах с целью увеличения охвата аудитории. Сбор и расчёт показателей с использованием программы позволяет сократить время на обработку данных социальной сети более чем на 70%.

ООО «Гамарджоба» является кафе грузинской кухни «Гамарджоба». Руководство организации стремится к повышению её эффективности и оперативно принимает решения относительно ценовой политики, выбора поставщиков и закупаемых продуктов. Переданные материалы включают оптимизационные модели и программу формирования маржинальной прибыли и предназначены для поддержки принятия решений руководством с целью максимизации прибыли. Программное обеспечение позволило формировать варианты достижения желаемой прибыли и повышает обоснованность управленческих решений.

ООО «Сибмед» осуществляет деятельность по разработке и изготовлению новых моделей традиционного стоматологического инструмента. Для поддержки принятия решения руководством относительно значений управляемых показателей, формирующих прибыль предприятия, были переданы модели и программа формирования маржинальной прибыли предприятия, использование которой позволило сократить время решения задачи по определению необходимых изменений показателей и выбрать стратегии достижения заданного значения прибыли.

ООО «Система автоматизация бизнес» специализировалась на разработке компьютерного программного обеспечения, в том числе для администрации г. Томска в части проведения закупок товаров, работ, услуг для обеспечения государственных и муниципальных нужд в соответствии с законодательством Российской Федерации. Организации была передана программа «Аукцион», предназначенная для имитационного моделирования торгов, проходящих в форме аукциона. Программный продукт, реализующий два алгоритма проведения торгов, особенностью которых является возможность поиска предпоследнего участника и учёт мотивов претендентов, предназначен для специалистов по государственным закупкам для обучения проведению торгов, а также для определения оптимальной стратегии проведения аукциона, обеспечивающей минимальное количество числа шагов, цены продажи и эффективность поиска второго поставщика. Экономическая эффективность также заключается в снижении затрат на повторное проведение аукциона в случае отказа победителя от заключения договора за счёт нахождения предпоследнего поставщика.

ФДО ТУСУР представляет собой самый крупный факультет университета, предоставляющий образовательные услуги с применением дистанционных технологий, а также самая крупная организация, предоставляющая услуги дистанционного обучения в азиатской части России. Для проведения лабораторных работ по дисциплине «Имитационное моделирование экономических процессов», «Математическое и имитационное моделирование экономических процессов» была передана программа «Имитатор», которая использовалась в учебном процессе более 10 лет. Данная программа позволяет выполнять имитацию экономических объектов, с её помощью осуществлялось обучение студентов решению задач моделирования путем проведения вычислительных экспериментов (на рисунках 7.20–7.21 представлен пример задания и расчёта).

Также на данном факультете в виде учебных пособий и методических указаний по дисциплинам «Исследование операций и методы оптимизации», «Эконометрика» внедрены оптимизационные модели решения обратных задач, метод решения обратных задач на основе формирования уравнения зависимости между аргументами, метод решения обратных задач на основе статистических данных.

Задание 1

1. Менеджеру магазина необходимо принять решение об его оптимальной структуре. Количество клиентов, ежедневно нуждающихся в обслуживании точно неизвестно. Основные правила модели:

1) входной поток клиентов – простейший(время между соседними заявками имеет показательное распределение) с интенсивностью λ=20 заявок в час;

2) время обслуживания заявки является случайной величиной, которая имеет показательное распределение со средним значением tобс.ср. = 8 мин;

3) средний доход от обслуживания одного клиента B = 170 руб.;

4) максимальное время ожидания равно tож.max=15 мин.;

5) вновь поступившая заявка в том канале, который раньше всех освободится:

6) период работы равен 64 часа.

С помощью программы «Имитатор» определите оптимальную структуру магазина. Число каналов является переменной величиной. Составьте таблицу зависимости показателя эффективности модели от числа каналов.

Рисунок 7.20 – Пример задания на лабораторную работу

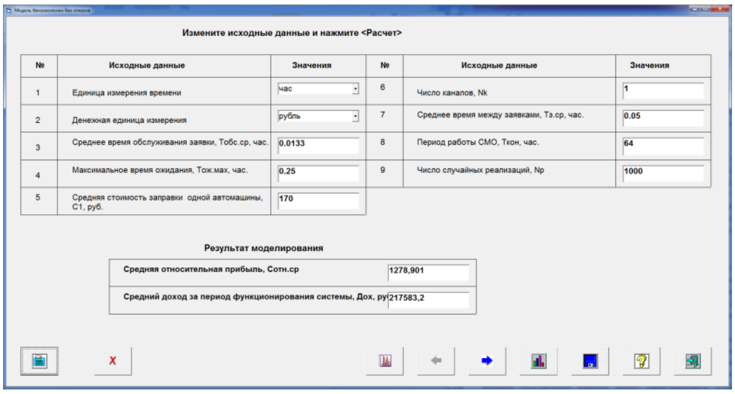


Рисунок 7.21 – Пример расчёта в ходе выполнения задания

Акты, подтверждающие внедрение и использование результатов диссертационной работы представлены в приложении Ж.

7.4Выводы по главе 7

1) Рассмотрено применение предложенного подхода для реализации двух групп программ: системы формирования прибыли и системы формирования интегрального показателя (рейтинга). Для решения задачи формирования рейтинга были определены два типа этапов (Группа, Индикатор), отличающихся процедурой расчета. Отличие от существующего подхода заключается в специфическом обходе графа решения задачи, для решения обратной задачи используется обратный обход, основанный на параллельном рекурсивном вычислении этапов-потомков. Для решения обратных задач также были определены дополнительные атрибуты класса `Узел` и методы класса `Расчёт`.

2) Реализованный проблемно-ориентированный комплекс программ позволяет осуществлять формирование ключевого показателя путем поэтапного решения иерархических задач с ограничениями на основе обратных вычислений и снизить временные затраты на обработку данных

3) Работоспособность созданных программ подтверждает реализуемость предложенного подхода.

Заключение

В диссертационной работе решена важная научная проблема по созданию и развитию методов решения обратных и оптимизационных задач прикладной математики, имеющая важное народно-хозяйственное значение. Диссертационная работа представляет собой совокупность новых научных результатов и положений, развивающих теоретические основы и методологические подходы в следующих направлениях научных исследований: методы и алгоритмы решения задач на основе обратных вычислений; методы и алгоритмы решения обратных задач; методы решения задач условной оптимизации; программное обеспечение решения обратных задач на основе объектно-ориентированного подхода.

Основные результаты работы:

– На основе анализа существующих подходов к решению обратных задач обоснована актуальность задачи разработки моделей, методов и алгоритмов решения задач на основе обратных вычислений, позволяющих минимизировать усилия эксперта и ошибки определения входных данных и выполнять решение широкого круга задач, а также не требующих применения трудоёмких вычислительных процедур, что позволяет сделать разработку программ быстрой и тиражируемой.

– С использованием литературных источников выбраны тестовые модели.

– Разработан метод решения задач с использованием обратных вычислений на основе построения уравнения зависимости между аргументами функции. При использовании данного метода отсутствует необходимость проверки согласованности дополнительной информации, поступающей от специалиста: соответствия поставленной цели коэффициентам относительной важности и направлениям их изменения.

– Для решения задач на основе обратных вычислений с ограничениями разработан стохастический метод, основанный на последовательном изменении результирующей величины и выбора аргумента для достижения результата с помощью моделирования полной группы несовместных событий.

– Разработаны оптимизационные модели для решения задач с использованием обратных вычислений, где в качестве целевой функции рассматривается мера отдаленности от исходных значений входных параметров. Разработаны методы и алгоритмы, в том числе итерационные, для решения обратных задач на основе минимизации суммы квадратов аргументов и минимизации суммы абсолютных изменений аргументов.

– Предложены модели и алгоритмы решения задач на основе обратных вычислений при максимизации соответствия экспертным целеполаганиям, а также при участии переменных в расчете нескольких показателей. Приведены результаты решения с помощью разработанного алгоритма классических задач оптимизации: закупок, цены, портфеля ценных бумаг, запасов предприятия.

– На основе объектно-ориентированного подхода разработана структура программного обеспечения решения задач с использованием обратных вычислений, основанная на формировании дерева цели. Предложен алгоритм для решения общей задачи формирования результирующего показателя дерева цели.

– Рассмотрена реализация предложенной структуры для создания программ: формирования прибыли и формирования интегрального показателя. Использование выбранного подхода обуславливает возможность гибкой модификации и расширения системы.

– Проведенные вычислительные эксперименты по формированию стандартных экономических показателей, разработанных показателей и решению классических оптимизационных задач показали соответствие результатов, полученных с помощью разработанных методов и алгоритмов, и решения с применением классических методов, а также математических пакетов.

– Результаты исследований внедрены на предприятиях и в учреждениях: ООО «Томская нефть», АО «Разрез «Степановский», ООО «Гамарджоба», г. Томск, ООО «ФОРС», г. Реутов, ООО «Сибмед», г. Томск, ООО «Вокифудтомск», г. Томск, ООО «Титан», г.Томск, ООО «Интенс-строй», г. Томск, ООО «Дельта», г.Юрга, МАОУ ДО Дворец творчества детей и молодежи, г. Томск.

– Результаты исследований внедрены в учебный процесс Томского государственного университета систем управления и радиоэлектроники в виде учебного пособия, методических указаний для выполнения практических, лабораторных и курсовых работ, компьютерных программ. Теоретические положения использовались для постановки задач научно-исследовательской работы студентов, 25-х выпускных квалификационных работ.

– Дальнейшее развитие данного исследования может быть связано с разработкой оптимизационных методов, например, методов решения задач линейного программирования, методов решения задачи квадратичного программирования при произвольной целевой функции. Также уже разработанные методы и алгоритмы могут быть использованы для решения задач прикладной математики. Так, например, модифицированный метод обратных вычислений может быть использован для решения задач факторного анализа при наличии зависимости между аргументами функции. Также развитие темы может быть связано с выявлением новых обратных задач, для решения которых использование разработанных методов будет затруднено или невозможно.

Список литературы

1) Groetsch, C. W. Inverse problems: activities for undergraduates. – Washington: Mathematical Association of America, 1999. – 222 p.

2) Воскобойников, Ю. Е. Нелинейный регуляризирующий алгоритм решения одного класса обратных задач теплопроводности / Ю. Е. Воскобойников // Инженерно-физический журнал. – 1989 – № 3 (56) – С. 29–35.

3) Воскобойников, Ю. Е. Регуляризирующий алгоритм восстановления сигналов и изображений с уточнением локальных отношений шум/сигнал / Ю. Е. Воскобойников, И. Н. Мухина // Автометрия. – 1999. – № 4. – С. 71–83.

4) Воскобойников, Ю. Е. Локальный регуляризирующий алгоритм восстановления контрастных сигналов и изображений / Ю. Е. Воскобойников, И. Н. Мухина // Автометрия. – 2000. – № 3. – С. 45–53.

5) Кабанихин, С. И. Обратные задачи естествознания и компьютерное моделирование / С. И. Кабанихин // Наука из первых рук. – 2013. – № 1(49). – С. 32–43.

6) Тихонов, А. Н. Об устойчивости обратных задач / А. Н. Тихонов // Докл. АН СССР. – 1943. – № 5(39). – С. 195–198.

7) Тихонов, А. Н. Методы решения некорректно поставленных задач / А. Н. Тихонов, В. Я. Арсенин. – М.: Наука, 1974. – 223 с.

8) Лаврентьев, М. М. О некоторых некорректных задачах математической физики / М.М. Лаврентьев. – Новосибирск: Изд-во СО АН СССР, 1962. – 92 с.

9) Лаврентьев, М. М. Теория операторов и некорректные задачи / М. М. Лаврентьев, Л. Я. Савельев. – Новосибирск: Изд-во Ин-та математики, 1999. – 702 с.

10) Иванов, В. К. Теория линейных некорректных задач и ее приложения / В. К. Иванов, В. В. Васин, В. П. Танана. – М.: Наука, 1978. – 208 с.

11) Латтес, Р. Метод квазиобращения и его приложения / Р. Латтес, Ж. Л. Лионс. – М.: Мир, 1970. – 224 с.

12) Лаврентьев, М. М. Некорректные задачи математической физики и анализа / М. М. Лаврентьев, В. Г. Романов, С. П. Шишатский. – М.: Наука, 1980. – 288 с.

13) Морозов, В. А. Регулярные методы решения некорректно поставленных задач / В. А. Морозов. – М.: Наука, 1987. – 239 с.

14) Лисковец, О. А. Вариационные методы решения неустойчивых задач / О. А. Лисковец. – Минск: Наука и техника, 1981. – 343 с.

15) Васин, В. В. Некорректные задачи с априорной информацией / В. В. Васин, А. Л. Агеев. – Екатеринбург: Наука, 1993. – 261 с.

16) Федотов, А. М. Некорректные задачи со случайными ошибками в данных / А. М. Федотов. – Новосибирск: Наука, 1990. – 280 с.

17) Вайникко, Г. М. Методы решения линейных некорректно поставленных задач в гильбертовых пространствах / Г. М. Вайникко. – Тарту: изд-во Тартус. ун-та, 1982. – 111 с.

18) Иванов, В. К. Дифференциально-операторные уравнения и некорректные задачи / В. К. Иванов, И. В. Мельникова, А. И. Филинков. – М.: Физматлит, 1995. – 176 с.

19) Танана, В. П. Оптимальные методы решения некорректно поставленных задач / В. П. Танана, А. И. Сидикова. – Челябинск: ЮУрГу, 2012. – 161 с.

20) Воскобойников, Ю. Е. Устойчивые алгоритмы решения обратных измерительных задач / Ю. Е. Воскобойников. – Новосибирск: НГАСУ, 2007. – 184 с.

21) Кабанихин, С. И. Обратные и некорректные задачи / С. И. Кабанихин. – Новосибирск: Сибирское научное издательство, 2009 – 458 с.

22) Tarantola, A. Inverse Problem Theory: Methods for Data Fitting and Model Parameter Estimation / A. Tarantola. – Amsterdam: Elsevier, 1987.

23) Parker, R.L. Geophysical Inverse Theory / R. L. Parker. – Princeton: Princeton University Press, 1994.

24) Groetsch, C. W. Inverse Problems in the Mathematical Sciences / C. W. Groetsch. – Braunschweig: Vieweg, 1993.

25) Самарский, А. А. Численные методы решения обратных задач математической физики / А. А. Самарский, П. Н. Вабищевич. – М.: УРСС, 2004. – 480 с.

26) Денисов, А. М. Введение в теорию обратных задач / А. М. Денисов. – М.: МГУ, 1994. – 207 с.

27) Vogel, C. R. Computational methods for inverse problems / C. R. Vogel. – Philadelphia: SIAM, 2002. – 183 p.

28) Воскобойников, Ю. Е. Синтез алгоритма наискорейшего спуска для решения плохо обусловленных систем линейных алгебраических уравнений / Ю.Е. Воскобойников // Труды Новосибирского государственного архитектурно-строительного университета (СИБСТРИН). – 2020. – № 1 (23). – С. 126–140.

29) Воскобойников, Ю. Е. Свойства регуляризированных решений обратной измерительной задачи при неточных характеристиках измерительной системы / Ю.Е. Воскобойников, В. А. Боева, С. В. Сыренов // ТРУДЫ Новосибирского государственного архитектурно-строительного университета (СИБСТРИН). – 2017. – № 3 (20). – С. 123–137.

30) Емелин, И. В. Правило останова в итерационных процедурах решения некорректных задач / И. В. Емелин, М. А. Красносельский // Автоматика и телемеханика. – 1978. – №12. – С. 59–63.

31) Gilyazov, S. F. Regularization of ill-posed problems by iteration methods / S. F. Gilyazov, N. L. Gol’dman. – Dordrecht ets.: Kluwer Acad. Publ., 2000. – 340 p.

32) Kabanikhin, S. I. Inverse and Ill-Posed Problems. Theory and Applications / S. I. Kabanikhin. – Deutschland: De Gruyter, 2011. – 459 p.

33) Матысик, О. В. Явные и неявные итерационные процедуры решения некорректно поставленных задач / О. В. Матысик. – Брест: БрГУ им. А.С. Пушкина, 2014. – 213 с.

34) Страхов, В. Н. К вопросу о скорости сходимости в методе простой итерации / В. Н. Страхов // Журн. вычисл. математики и мат. физики. – 1973. – № 6 (13). – С. 1602–1606.

35) Красносельский, М. А. Приближѐнное решение операторных уравнений / М. А. Красносельский, Г. М. Вайникко, П. П. Забрейко, Я. Б. Рутицкий, В. Я. Стеценко. – М.: Наука, 1969. – 456 с.

36) Shananin, A. A. Inverse problems in economic measurements / A. A. Shananin // Computational Mathematics and Mathematical Physics. – 2018. – № 58 (2). – P. 170–179.

37) Семенчин, Е. А. Об обратной задаче в математической модели самоорганизации рынка труда / Е. А. Семенчин, А. П. Невечеря // Фундаментальные исследования. – 2014. – № 6. – С. 1184 –1190.

38) Урусова, А. С. Обратная задача для экзогенных параметров модели Солоу / А. С. Урусова // Известия Российского государственного педагогического университета им. А.И. Герцена. – 2007. – № 53 (22). – С. 230–232.

39) Ekeland, I. An inverse problem in the economy theory of demand / I. Ekeland, N. Djitte // Annales de l'Institut Henri Poincare Non Linear Analysis. – 2006. – №. 2 (23). – P. 269–281.

40) Одинцов, Б. Е. Обратные вычисления в формировании экономических решений / Б. Е. Одинцов. – М.: Финансы и статистика, 2004. – 192 с.

41) Одинцов, Б. Е. Итерационный метод оптимизации управления предприятиями средствами обратных вычислений / Б. Е. Одинцов, А. Н. Романов // Вестник Финансового университета. – 2014. – № 2. – С. 60–73.

42) Одинцов, Б. Е. Проблемы создания информационных систем управления эффективностью бизнеса / Б. Е. Одинцов, А. Н. Романов // Вестник Финансового университета. – 2014. – № 6. – С. 22–36.

43) Цветков, М. А. "Возвратно-сетевой" метод совершенствования структуры кредитно-депозитной базы коммерческих банков / М. А. Цветков // Экономика и управление. – 2007. – №1. – С. 139–141.

44) Збарский, А. М. Математическое представление процесса приведения предприятия в равновесное состояние / А. М. Збарский // Российский экономический интернет-журнал. – 2009. – № 1.– С. 445–452.

45) Одинцов, Б. Е. Управление с учетом «золотых» пропорций плановых показателей / Б. Е. Одинцов // Управленческие науки в современном мире. – 2016. – №1. – С. 43–47.

46) Виштак, О. В. Использование технологии обратных вычислений при мониторинге качества дополнительного образования в ВУЗе / О. В. Виштак, И. А. Штырова // Вестник Астраханского государственного технического университета. – 2014. – № 2. – С.67–73.

47) Бармина, Е. А. Мониторинг качества коммерческой организации. Структурирование показателей. Применение когнитивных карт / Е. А. Бармина, И. Ю. Квятковская // Вестник Астраханского государственного технического университета. – 2010. – № 2. – С.15–20.

48) Блюмин, С. Л. Применение анализа конечных изменений и метода обратных вычислений в системах управления и поддержки принятия решений / С. Л. Блюмин, Г. С. Боровкова // Проблемы управления. – 2018. – №6. – С. 29–34.

49) Силкина, Г. Ю. Совмещение сбалансированной системы показателей и метода обратных вычислений как аналитический инструмент управления эффективностью компании / Г. Ю. Силкина, А. А. Переверзева // Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Экономические науки. – 2016. – №3.– С. 258–267.

50) Грибанова, Е. Б. Методы и алгоритмы решения обратных экономических задач с помощью модифицированного аппарата обратных вычислений / Е. Б. Грибанова. – Чебоксары: ИД «Среда», 2020. – 133 с.

51) Тимофеев, Ю. М. Математические аспекты решения обратных задач атмосферной оптики / Ю. М. Тимофеев, А. В. Поляков. – СПб: Изд-во С.-Петерб.ун-та, 2001 – 188 с.

52) Ватульян, А. О. Обратные задачи в механике деформируемого трердого тела / А. О. Ватульян. – М: Физматлит, 2007. – 223 с.

53) Aster, R. C., Parameter Estimation and Inverse Problems / R. C. Aster, B. Borchers, C. H. Thurber. – Waltham: Elsevier, 2013. – 355 p.

54) Ahuja, R. K. Inverse Optimization, Part1: Linear Programming and General Problem/ R. K.Ahuja, J. B. Orlin. – Cambridge: MIT, 1998. – 35 p.

55) Ghobadi, K. Robust inverse optimization / K. Ghobadi, T. Lee, H. Mahmoudzadeh, D. Terekhov // Operations Research Letters. - 2018. - № 3. - P. 339-344.

56) Zhenga, G.-H. Solving the backward problem for space-fractional diffusion equation by a fractional Tikhonov regularization method / G.-H. Zhenga, Q.-G. Zhang // Mathematics and Computers in Simulation. – 2018. – № 148. – P. 37–47.

57) Park, Y. Parameter determination for Tikhonov regularization problems in general form / Y. Park, L. Reichel, G. Rodriguez, X. Yud // Journal of Computational and Applied Mathematics. – 2018. – № 343. – P. 12–25.

58) Bai, Z.-Z. Modulus-based iterative methods for constrained Tikhonov regularization / Z.-Z. Bai, A. Buccini, K. Hayami, L. Reichel, J.-F. Yin, N. Zheng // Journal of Computational and Applied Mathematics. – 2017. – № 319. – P. 1–13.

59) Wang, H. Cauchy sparse NMF with manifold regularization: A robust method for hyperspectral unmixing / H. Wang, W. Yang, N. Guan // Knowledge-Based Systems. – 2019. – № 184. – P. 1–16.

60) Scardapane, S. Group sparse regularization for deep neural networks / S. Scardapane, D. Comminiello, A. Hussain, A. Uncini // Neurocomputing. – 2017. – № 241. – P. 81–89.

61) Денисов, А. М. Введение в теорию обратных задач. Учебное пособие / А. М. Денисов. – М.: Изд-во МГУ, 1994. – 207 с.

62) Boyd, S. Convex Optimization / S. Boyd, L. Vandenberghe. – Cambridge: Cambridge University Press, 2004 – 730 p.

63) Воскобойников, Ю. Е. Выбор параметра регуляризации при решении обратных измерительных задач / Ю. Е. Воскобойников, Н. Г. Преображенский // Автометрия. – 1984. – № 2. – С. 31–38.

64) Воскобойников, Ю. Е. Выбор параметра регуляризации и ошибки восстановления входного сигнала в методе статистической регуляризации / Ю. Е. Воскобойников // Автометрия. – 1975 – №4 – С. 10–18.

65) Xu, J. Assessment of Tikhonov-type regularization methods for solving atmospheric inverse problems / J. Xu, F. Schreier, A. Doicu, T. Trautmann // Journal of Quantitative Spectroscopy & Radiative Transfer. – 2016. – № 184. – P. 274–286.

66) Мицель, А. А. Методы оптимизации / А. А. Мицель, А. А. Шелестов. – Томск: Изд-во Томск. гос. ун-та систем управления и радиоэлектроники, 2004. – 256 с.

67) Lewis, R. M. Direct search methods: then and now / R. M. Lewis, V. Torczon, M. W.Trossetca // Journal of Computational and Applied Mathematics. – 2000. – № 124. – P. 191–207.

68) Luenbergr, D. G. Linear and Nonlinear Programming / D. G. Luenbergr, Y. Ye. – Springer Science, 2008 – 551 p.

69) Zhang, M. A simple sufficient Descent Method for Unconstrained Optimization / M. Zhang, Y. Xiao, D. Zhou // Mathematical Problems in Engineering. – 2010. – P. 1–9.

70) Грибанова, Е. Б. Стохастический алгоритм поиска глобального минимума функции / Е. Б. Грибанова // Прикладная информатика. – 2017. – № 2. – С. 130–139.

71) Растригин, Л. А. Адаптация сложных систем / Л. А. Растригин. – Рига: Зинатне, 1981. – 375 с.

72) Растригин, Л. А. Статистические методы поиска / Л. А. Растригин. – М.: Наука, 1968. – 376 с.

73) Мицель, А. А. Комбинаторная модель опционного портфеля / А. А. Мицель, М. Е. Семенов, М. Э. Фатьянова // Финансовая аналитика: проблемы и решения. – 2016. – № 25(307). – С. 2–13.

74) Thomas, J. Improved simple optimization algorithm for unconstrained non-linear optimization problems / J. Thomas, S. Mahapatra // Perspectives in Science. – 2016. – № 8. – P. 1–3.

75) Hamzacebi, C. Continuous function minimization by dynamic random search / C. Hamzacebi, F. Kutay // Applied Mathematical Modeling. – 2007. – № 10 (31). – P. 2189–2198.

76) Hamzacebi, C. A heuristic approach for finding the global minimum: adaptive random search technique / C. Hamzacebi, F. Kutay. // Applied Mathematics and Computation. – 2006. – № 173. – P. 1323–1333.

77) Toksari, M. D. Ant colony optimization for finding the global minimum / M. D. Toksari // Applied Mathematics and computation. – 2006. – № 176. – P. 308–316.

78) Toksari, M. D. A heuristic approach to find the global optimum of function / M. D. Toksari // Journal of computational and Applied mathematics. – 2007. – № 2 (209). – P. 160–166.

79) Lei, G. Adaptive random search in Quasi-Monte Carlo methods for global optimization / G. Lei // Computers and Mathematics with Applications. –2007. – № 6. – P. 747–754.

80) Жиглявский, А. А. Методы поиска глобального экстремума / А. А. Жиглявский, А. Г. Жилинскас. – М.: Наука, 1991. – 248 с.

81) Панов, Н. В. Объединение стохастических и интервальных подходов для решения задач глобальной оптимизации функций / Н. В. Панов // Вычислительные технологии. – 2009. – № 5(14). – С. 49–65.

82) Vanderbei, R. J. Linear programming. Foundation and extensions / R J. Vanderbei. – New York: Springer, 2014 – 466 p.

83) Ганичева, А. В. Метод решения некоторых классов оптимизационных задач / А. В. Ганичева // Моделирование, оптимизация и информационные технологии. – 2019. – № 2 (7). – С. 43–54.

84) Trunov, A. N. Modernization of means for analysis and solution of nonlinear programming problems / A. N. Trunov // Quantitative Methods in Economics. – 2015. – № 16 (2). – P. 133–141.

85) Qi, Y. An adaptive penalty-based boundary intersection method for many-objective optimization problem / Y. Qi, D. Liu, X. Li, J. Lei, X. Xu, Q. Miao // Information Sciences. – 2020. – № 509. – P. 356–375.

86) El-Sobky, B. A penalty method with trust-region mechanism for nonlinear bilevel optimization problem / B. El-Sobky, Y. Abo-Elnaga // Journal of Computational and Applied Mathematics. – 2018. – № 340. – P. 360–374.

87) Li, J. A QP-free algorithm without a penalty function or a filter for nonlinear general-constrained optimization / J. Li, Y. Zhenping // Applied Mathematics and Computation. – 2018. – № 316. – P. 52–72.

88) Мицель, А. А. Новый алгоритм решения задачи квадратичного программирования / А. А. Мицель, А. Н. Хващевский // Автометрия. – 1999. – № 3. – P. 93–98.

89) Morovati, V. Extension of Zoutendijk method for solving constrained multiobjective optimization problems / V. Morovati, L. Pourkarimi // European Journal of Operational Research. – 2019. – № 273 (1). – P. 44–57.

90) Hosseini, A. A non-penalty recurrent neural network for solving a class of constrained optimization problems / A. Hosseini // Neural Networks. – 2016. – № 73. – P. 10–25.

91) Darabia, A. Dual feasible direction-finding nonlinear programming combined with metaheuristic approaches for exact overcurrent relay coordination / A. Darabia, M. Bagheri, G.B. Gharehpetian // International Journal of Electrical Power & Energy Systems. – 2020. – №114. – C 1–8.

92) Антамошкин, А. Н. Поисковые алгоритмы условной псевдобулевой оптимизации / А. Н. Антамошкин, И. С. Масич // Системы управления, связи и безопасности. – 2016. – № 1. – С. 103–145.

93) Кремер, Н. Ш. Исследование операций в экономике. Учебное пособие для вузов / Н. Ш. Кремер, Б. А. Путко, И. М. Тришин, М. Н. Фридман, М. А. Войтенко, Т. С. Фофанова. – М.: ЮНИТИ, 2006. – 407 с.

94) Овчинников, В. А. Систематизация точных методов дискретной оптимизации / В. А. Овчинников // Наука и образование. – 2015. – № 6. – С. 288–304.

95) Гринченко, С. Н. Метод «проб и ошибок» и поисковая оптимизация: анализ, классификация, трактовка понятия «естественный отбор» / С. Н. Гринченко // Исследовано в России. – 2003. – № 104. – С. 1228–1271.

96) Glover, F. Tabu search. Part I / F. Glover // INFORMS Journal on Computing. – 1989. – № 1. – P. 190–206.

97) Pedroso, J. P. An evolutionary solver for linear integer programming / J. P. Pedroso // BSIS Technical Report. – 1998. – № 98. – C. 1–15.

98) Jansen, T. A comparison of simulated annealing with a simple evolutionary algorithm on pseudo-boolean functions of unitation / T. Jansen, I. Wegener // Theoretical Computer Science. – 2007. – № 386. – P. 73–93.

99) Галушин, П. В. Разработка и исследование эволюционных алгоритмов дискретной оптимизации / П. В. Галушин, О. Э. Семенкина // Вестник Сибирского государственного аэрокосмического университета. – 2011. – № 5. – С. 25–29.

100) Ferrario, E. Goal Tree Success Tree–Dynamic Master Logic Diagram and Monte Carlo simulation for the safety and resilience assessment of a multistate system of systems / E. Ferrario, E. Zioab // Engineering Structures. - 2014. - Vol.59. - P. 411-433.

101) Kuru, S. Goal-driven blackboard control architecture based on ex partially complete general trees / S. Kuru, F. Bek // Knowledge-Based Systems. - 1990. - № 2. - P. 236-241.

102) Stashchuk, O. Comprehensive System of Financial and Economic Security of the Enterprise / O. Stashchuk, A. Vitrenko, O. Kuzmenko, H. Koptieva, O. Tarasova, L. Dovgan // International Journal of Management. - 2020. - № 11 (5). - P. 330-340.

103) Bredikhina, N. V. Strategic aspects of the production and economic potential of urban investment and construction sector / N. V. Bredikhina // Journal of Applied Engineering Science. – 2021. – № 2. – P. 483-487.

104) Balan, O. Using the Pattern Method for the Comprehensive Organization of Recruitment and Selection of Personnel / O. Balan, H. Moskalyk, K. Peredalo, O. Hurman, I. Samarchenko, F. Revin // International Journal of Advanced Research in Engineering and Technology.- 2020. - № 11(4). - P. 290-300.

105) Shen, B. Profit optimization in service oriented data market: A Stackelberg game approach / B. Shen, Y. Shen, W. Ji // Future Generation Computer Systems. – 2019. – № 95. – P. 17–25.

106) Neill, B. Profit optimization for deterministic inventory systems with linear cost / B. Neill, S. Sanni // Computer & Industrial Engineering. – 2018. – № 122. – P. 303–317.

107) Lee, C. Meta-data envelopment analysis: Finding a direction towards marginal profit maximization / C. Lee // European Journal of Operation Research. – 2014. – № 237. – P. 207–216.

108) Парушина, Н. В. Анализ выпуска готовой продукции предприятий пищевой промышленности: влияние факторов и резервы роста / Н. В. Парушина, О. А. Булкина // Научные записки Орел ГИЭТ. – 2010. – №2. – С. 153–157.

109) Савицкая, Г. В.Анализ хозяйственной деятельности предприятия: Учебное пособие / Г. В. Савицкая. – Мн.: Новое значение, 2020. – 704 с.

110) Карминский, А. М. Энциклопедия рейтингов: экономика, общество, спорт / А. М. Карминский, А. А. Полозов. – М.: Инфра-М, 2016. – 36 с.

111) Солодов, А. А. Математические принципы построения рейтинговых систем // Экономика, статистика и информатика. – 2016. – № 1. – С. 75–81.

112) Сидоров, А. А. Подход к оценке территориальной дифференциации развития цифровой экономики / А. А. Сидоров, П. В. Сенченко, В. Ф. Тарасенко // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2020. – № 51. – С. 121–129.

113) Постюшков, А. В. Методика рейтинговой оценки предприятий / А. В. Постюшков // Имущественные отношения в Российской Федерации. – 2003. – № 1. – С. 46–54.

114) Гонова, О. В. Социально-экономическое развитие региона: модели рейтинговой оценки / О. В. Гонова // Современные наукоемкие технологии. Региональное приложение. – 2010. – № 3. – С. 40–46.

115) Гирина, А. Н. Методика оценки социально-экономического развития региона / А. Н. Гирина // Вестник ОГУ. – 2013. – № 8. – С. 82–87.

116) Гранберг, А. Г. Основы региональной экономики / А. Г. Гранберг. – М.: ГУ ВШЭ, 2000. – 495 с.

117) Цхай, А. А. Информационно-моделирующая система мониторинга деятельности сельхозпроизводителей региона / А. А. Цхай, Д. А. Рыков, А. В. Сибиряков, А. А. Шайдуров // Известия Алтайского государственного университета. – 2011. – № 1. – С. 126–130.

118) Мамаева, З. М. Оценка инновационного развития регионов: эконометрический подход / З. М. Мамаева // Вестник Нижегородского университета им. Н.И. Лобачевского. – 2012. – №2. – С.202–208.

119) Саати, Т. Принятие решений. Метод анализа иерархий / Т. Саати. – М.: Радио и связь, 1993. – 278 с.

120) Marzouk, M. Developing green bridge rating system using Simon's procedure / M. Marzouk, A. Nouh, M. El-Said // HBRC Journal. – 2014. – № 10. – С. 176–182.

121) Кендюхов, А. В. Использование метода главных компонент для оценки конкурентоспособности машиностроительных предприятий / А. В. Кендюхов, Д. О. Толкачев // Маркетинг и менеджмент инноваций. – 2013. – № 4. – С.219–227.

122) Грибанова, Е. Б. Методы решения обратных задач экономического анализа / Е. Б. Грибанова // Корпоративные финансы. – 2016. – №1. – С. 119–130.

123) Грибанова, Е. Б. Решение обратных задач экономики с помощью модифицированного метода обратных вычислений / Е. Б. Грибанова // Проблемы управления. – 2016. – № 5.– С. 35–40.

124) Грибанова, Е. Б. Стохастические алгоритмы решения обратных задач экономического анализа с ограничениями / Е. Б. Грибанова // Доклады Томского государственного университета систем управления и радиоэлектроники. – 2016. – № 4. – С. 112–116.

125) Грибанова, Е. Б. Разработка стохастического алгоритма для решения обратных иерархических задач прикладной экономики / Е.Б. Грибанова // Информационные системы и технологии в образовании, науке и бизнесе: сб.

материалов всероссийской научно-практической конференции, Улан-Удэ. - Улан-Удэ: БГУ, 2022. – С. 23–28.

126) Demin, D. Synthesis of optimal control of technological processes based on a multialternative parametric description of the final state / D. Demin // Eastern-European Journal of Enterprise Technologies. – 2017. – № 3 (4 (87)). – P. 51–63.

127) Грибанова, Е. Б. Методы решения обратных задач экономического анализа с помощью минимизации приращений аргументов / Е. Б. Грибанова // Доклады Томского государственного университета систем управления и радиоэлектроники. – 2018. – № 2. – С. 95–99.

128) Сіницький, М. Є. До питання розв'язку обернених задач економічного спрямування / М. Є. Сіницький // Науковий вісник Національної академії статистики, обліку та аудиту. – 2018. – № 1–2. – С. 195–202.

129) Gribanova, E. B. Development of iterative algorithms for solving the inverse problem using inverse calculations / E. B. Gribanova // Eastern-European Journal of Enterprise Technologies. – 2020. – № 4 (3). – P. 27–34.

130) Gribanova, E. A method for solving the procurement optimization problem based on inverse calculations // Proceedings on Engineering Sciences. – 2020. – № 4. – P. 441–452.

131) Zhang, Q. Study on factors affecting corn yield based on the Cobb-Douglas production function / Q. Zhang, W. Dong, C. Wen, T. Li // Agricultural Water Management. – 2020. – № 228. – P. 1–11.

132) Sarmah, S. P. Coordination of a single-manufacturer/multi-buyer supply chain with credit option / S. P. Sarmah, D. Acharya, S.K. Goyal // International Journal of Production Economics. – 2008. – 111 (2). – P. 676–685.

133) Gribanova, E. B. Algorithm for solving the inverse problems of economic analysis in the presence of limitations / E. B. Gribanova // EUREKA: Physics and Engineering. – 2020. – № 1. – P. 70–78.

134) Gribanova, E. Stochastic Algorithm to Solve the Problem of Linear Programming with Backward Calculations / E. Gribanova // Proceedings of the international workshop "Applied methods of statistical analysis. Nonparametric methods

in cybernetics and system analysis", Krasnoyarsk. – Novosibirsk: NSTU publisher, 2017 – С. 196–203.

135) Грибанова, Е. Б. Алгоритм решения задачи линейного программирования с помощью обратных вычислений / Е. Б. Грибанова // Финансовая аналитика: проблемы и решения. – 2017. – №9. – С. 1062–1075.

136) Gribanova, E. Algorithm for regression equation parameters estimation using inverse calculations / E. Gribanova // Proceedings of the international workshop "Applied methods of statistical analysis. Statistical computation and simulation", Novosibirsk. – Novosibirsk: NSTU, 2019. – С. 357–364.

137) Gribanova, E. B. Development of a price optimization algorithm using inverse calculations / E. B. Gribanova // Eastern-European journal of Enterprise technologies. – 2019. – № 5 (4). – P. 18–25.

138) Chen, R. Capacitated assortment and price optimization for customers with disjoint consideration sets / R. Chen, H. Jiang // Operations Research Letters. – 2017. – № 45. – С. 170–174.

139) Gallego, G. Multiproduct price optimization and competition under the nested logit model with product-differentiated price sensitivities / G. Gallego, R. Wang // Operations Research. – 2014. – № 62 (2). – С. 450–461.

140) Ferreira, K. J. Analytics for an online retailer: demand forecasting and price optimization / K. J. Ferreira, B. H. A. Lee, D. Simchi-Levi // Manufacturing & Service Operations Management. – 2015. – № 18 (1). –С. 69–88.

141) Caro, F. Clearance pricing optimization for a fast-fashion retailer / F. Caro, J. Gallien // Operations Research. – 2012. – № 60 (6). – С. 1404–1422.

142) Harsha, P. A practical price optimization approach for omnichannel retailing / P. Harsha, S. Subramanian, M. Ettl // INFORMS Journal on Optimization. – 2019. – № 1 (3). – С. 241–264.

143) Salvietti, L. A profit-maximizing economic lot scheduling problem with price optimization / L. Salvietti, N. R. Smith // European Journal of Operational Research. – 2008. – № 184. – С. 900–914.

144) Katsifou, A. Joint product assortiment, inventory and price optimization to

attract loyal and non-loyal customers / A. Katsifou, A. R. W. Seifert, J.-S. Tancrez // Omega. – 2014. – № 46. – С. 36–50.

145) Chen, R. Capacitated assortment and price optimization under the multilevel nested logit model / R. Chen, H. Jiang // Operations Research Letters. – 2019. – № 47. – P. 30–35.

146) Qu, T. Demand prediction and price optimization for semi-luxury supermarket segment / T. Qu, J. H. Zhang, F. Chan, R. S. Srivastava, M. K. Tiwari, W. Park // Computers & Industrial Engineering. – 2017. – № 113. – P. 91–102.

147) Грибанова, Е. Б. Решение задачи оптимизации цены с помощью обратных вычислений / Е. Б. Грибанова // Моделирование, оптимизация и информационные технологии. – 2019. – Т. 7, № 3. – С. 1–10.

148) Katsifou, A. Joint product assortiment, inventory and price optimization to attract loyal and non-loyal customers / A. Katsifou, R. W. Seifert, J.-S. Tancrez // Omega. – 2014. – № 46. – P. 36–50.

149) Фарманов, Р. Ф. Оптимизация закупок материальных ресурсов в системе ресурсосбережения предприятий АПК / Р. Ф. Фарманов // Вопросы структуризации экономики. – 2008. – № 3. – С. 32–37.

150) Грибанова, Е. Б. Решение задачи оптимизации закупок с помощью обратных вычислений / Е. Б. Грибанова // Экономический анализ: теория и практика. – 2018. – № 3. – С. 586–596.

151) Kalayci, C. B. A comprehensive review of deterministic models and applications for mean-variance portfolio optimization / C. B. Kalayci, O. Ertenlice, M. A. Akbay // Expert Systems With Applications. – 2019. – № 125. – С. 345–368.

152) Gribanova, E. B. An Iterative Algorithm for Solving Inverse Problems of Economic Analysis Using Weighting Factors. Advances in Engineering Research / E. B. Gribanova. – New York: Nova Publishers. – 2021. – P. 49–79.

153) Tomazella, C. P. A comprehensive review of Branch-and-Bound algorithms: Guidelines and directions for further research on the flowshop scheduling problem / C. P. Tomazella, M. S. Nagano. // Expert Systems with Applications. – 2020. – № 158. – С. 1–19.

154) Gmys, J. A computationally efficient Branch-and-Bound algorithm for the permutation flow-shop scheduling problem / J. Gmys, M. Mezmaz, N. Melabb, D. Tuyttens. // European Journal of Operational Research. – 2020. – № 284. – P. 814–833.

155) Salman, R. Branch-and-bound for the Precedence Constrained Generalized Traveling Salesman Problem / R. Salman, F. Ekstedt, P. Damaschke // Operations Research Letters. – 2020. – № 48. – P. 163–166.

156) Гвоздев, Е. В. Методология анализа показателей влияния человеческого фактора на комплексную безопасность электроэнергетических предприятий / Е. В. Гвоздев, Е. Б. Грибанова, Ю. Г. Матвиенко // Безопасность труда в промышленности. – 2020. – № 12. – С. 38–43.

157) Gribanova, E. Construction of algorithms for solving the inverse problem when using indicators in several calculation functions / E. Gribanova // Eastern-European Journal of Enterprise Technologies. - 2022. - № 4 (115). - P. 44-50.

158) Крюков, А. С. Автоматизация решения обратных задач экономического анализа / А. С. Крюков // Информационные технологии в науке, управлении, социальной сфере и медицине : сборник научных трудов IV Международной научной конференции, Томск. – Томск : Издательство Томского политехнического университета, 2017. – Ч. 1. – С. 221–223.

159) Грибанова, Е. Б., Мустакимов, Р. Р. Разработка алгоритма решения обратной задачи формирования прибыли при ограничении целочисленности // Доклады Томского государственного университета систем управления и радиоэлектроники. 2022. – № 3. – С. 63–68.

160) Буч, Г. Объектно-ориентированный анализ и проектирование с примерами приложений на С++ / Г. Буч. – М.: «Издательство Бином»; СПб.: Невский диалект, 1999. – 560 с.

161) Иванова, Г. С. Объектно-ориентированное программирование: Учебник для вузов / Г. С. Иванова, Т. Н. Ничушкина, Е. К. Пугаев. – М.: МГТУ им. Н.Э. Баумана, 2003. – 368 с.

162) Фаулер, М. UML. Основы / М. Фаулер . – СПб.: Символ-Плюс, 2006. – 192 с.

163) Бойченко, И.В. Программное обеспечение моделирования, обработки и анализа данных лидарного зондирования газового состава атмосферы: дис. на соискание ученой степени канд. техн. наук: 05.13.18 / Бойченко Иван Валентинович. – Томск, 2002. – 113 с.

164) Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон Р. – СПб.: Питер, 2007. – 366 с.

165) Грибанова, Е. Б. Алгоритмы и комплекс программ для решения задач имитационного моделирования объектов прикладной экономики: дис. на соискание ученой степени канд. техн. наук: 05.13.18 / Грибанова Екатерина Борисовна. – Томск, 2009. – 154 с.

166) Мицель, А. А. Имитационное моделирование экономических объектов. Алгоритмы и комплекс программ / А. А. Мицель, Е. Б. Грибанова. – Saarbrucken: LAP LAMBERT Academic Publishing, 2011. – 148 с.

167) Грибанова, Е. Б. Разработка имитационных моделей экономических систем на основе объектно-ориентированного подхода / Е. Б. Грибанова // Современные техника и технологии: материалы XIV Международной научно-практической конференции, Томск. – Томск: Издательство Томского политехнического университета, 2008. – С. 276–277.

168) Грибанова, Е. Б. Использование метода имитационного моделирования для выбора методики проведения аукциона / Е. Б. Грибанова, О. В. Каштанова // Научная сессия ТУСУР: Материалы Всероссийской научно-технической конференции студентов, аспирантов и молодых специалистов, Томск. – Томск: В-Спектр, 2007. – С. 185–188.

169) Грибанова, Е. Б. Система имитационного моделирования торгов, проходящих в форме аукциона / Е. Б. Грибанова, О. В. Каштанова, А.А. Мицель А.А. // Доклады ТУСУР. – 2007. – №1 (15). – С. 63–70.

170) Грибанова, Е. Б. Разработка автоматизированной системы имитационного моделирования аукционов / Е. Б. Грибанова, И. В. Бойченко // Электронные средства и системы управления. Опыт инновационного развития:

доклады международной научно-практической конференции. – Томск. – Томск.: В: Спектр. – 2007. – С. 221–224.

171) Грибанова, Е. Б. Имитационная модель аукциона, проводимого с целью поставки товаров, услуг для государственных и муниципальных нужд / Е. Б. Грибанова // Доклады ТУСУР. – 2007. – №2 (16). – С. 204–210.

172) Мицель, А. А. Имитационная модель аукциона, проводимого с целью поставки товаров и услуг для государственных и муниципальных нужд / А. А. Мицель, Е. Б. Грибанова // Труды 6 Всероссийской научно-практической конференции с международным участием «Инновационные технологии и экономика в машиностроении», Юрга. – Томск: изд-во ТПУ, 2008. – С. 322–327.

173) Каштанова О.В., Грибанова Е.Б. Программа имитационного моделирования торгов, проходящих в форме аукциона «Аукцион», М., 2007. – зарег. в Отраслевом фонде алгоритмов и программ 17 апреля 2007, № 50200700854.

174) Мицель, А. А. Разработка системы имитационного моделирования управления запасами на основе объектно-ориентированной технологии / А. А. Мицель, И. В. Бойченко, Е. Б. Грибанова // Инфокоммуникационные технологии. – 2006. – т. 4. №3. – С. 59–64.

175) Бойченко И.В., Грибанова Е.Б. Программная система имитационного моделирования управления запасами «Запас», М., 2006. – зарег. в Отраслевом фонде алгоритмов и программ 26 октября 2006, № 50200601855.

176) Бойченко, И. В. Система имитационного моделирования управления запасами «Запас» / И. В. Бойченко, Е. Б. Грибанова // Компьютерные учебные программы и инновации. – 2008. – №5. – С. 145.

177) Мицель, А. А. Компьютерное имитационное моделирование экономических объектов / А. А. Мицель, Е. Б. Грибанова // Доклады ТУСУР. – 2005. – №3. – С. 49–56.

178) Грибанова, Е. Б. Имитационные модели экономических объектов / Е. Б. Грибанова // Актуальные проблемы экономики в творчестве студентов: Сб. статей. – СПб.: СПбГИЭУ, 2006. – С. 200–205.

179) Грибанова, Е. Б. Компьютерная система имитационного моделирования экономических объектов, разработанная на основе объектно-ориентированной технологии / Е. Б. Грибанова // Научная сессия ТУСУР: Материалы Всероссийской научно-технической конференции студентов, аспирантов и молодых специалистов, Томск. – Томск: В-Спектр, 2008. – С. 56–59.

180) Мицель, А. А. Разработка системы имитационного моделирования экономических объектов на основе объектно-ориентированного подхода / А. А. Мицель, Е. Б. Грибанова // Известия ТПУ. – 2007. – т. 311. №6. – С. 11–15.

181) Грибанова, Е. Б. Имитационные модели экономических объектов / Е. Б. Грибанова / Труды 6 Всероссийской научно-практической конференции с международным участием «Инновационные технологии и экономика в машиностроении», Юрга. – Томск: изд-во ТПУ, 2008. – С. 248–251.

182) Грибанова, Е. Б. Имитационное моделирование систем управления запасами / Е. Б. Грибанова // Современное образование: традиции и новации: материалы Всероссийской научно-методической конференции, Томск. – Томск: Томский государственный университет систем управления и радиоэлектроники, 2006. – С. 173–174.

183) Бойченко, И. В. Автоматизированная система имитационного моделирования управления запасами / И. В. Бойченко, Е. Б. Грибанова, А. А. Мицель // Информационные системы: тр. постоянно действующего науч.-техн. семинара. – 2006. – № 4 – С. 118–125.

184) Грибанова, Е. Б. Алгоритмические имитационные модели управления материальными запасами на складе / Е. Б. Грибанова, А. А. Мицель // Известия ТПУ. – 2006. – т. 309. №8. – С. 201–207.

185) Грибанова, Е. Б. Дискретно-событийное имитационное моделирование систем управления запасами / Е. Б. Грибанова, А. А. Мицель // Молодежь и современные информационные технологии: Сборник трудов V Всероссийской научно-практической конференции студентов, аспирантов и молодых специалистов, Томск. – Томск: Из-во ТПУ. – 2007 – С. 95–97.

186) Грибанова, Е. Б. Обучающие системы имитационного моделирования экономических процессов / Е. Б. Грибанова, А. А. Мицель // Доклады томского государственного университета систем управления и радиоэлектроники. – 2009. – № 1. – С.131–138.

187) Бойченко, И. В. Автоматизированная система имитационного моделирования экономических процессов / И. В. Бойченко, Е. Б. Грибанова // Научная сессия ТУСУР: материалы Всероссийской научно-технической конференции студентов, аспирантов и молодых специалистов, Томск. – Томск: В-Спектр. – 2006. – С. 164–166.

188) Грибанова, Е. Б. Система имитационного моделирования экономических объектов / Е. Б. Грибанова // Научная сессия ТУСУР: материалы Всероссийской научно-технической конференции студентов, аспирантов и молодых специалистов, Томск. – Томск: В-Спектр, 2005. – С. 196–198.

189) Грибанова, Е. Б. Система имитационного моделирования экономических объектов / Е. Б. Грибанова // Студент и научно-технический прогресс: материалы Международной научной студенческой конференции, Новосибирск. – Новосибирск: Редакционно-издательский отдел НГУ, 2005. – С. 50–54.

190) Мицель, А. А. Программа имитационного моделирования экономических объектов «Имитатор» / А. А. Мицель, Е. Б. Грибанова // Компьютерные учебные программы и инновации. – 2008. – № 5. – С. 144–145.

191) Мицель А.А., Грибанова Е.Б. Программа имитационного моделирования экономических объектов «Имитатор», М., 2006. – зарег. в Отраслевом фонде алгоритмов и программ 26 октября 2006, № 50200601854.

192) Мицель А. А. Практикум по имитационному моделированию экономических процессов / А. А. Мицель, Е. Б. Грибанова, Е. А. Ефремова. – Томск: ТУСУР, 2007. – 270 с.

193) Грибанова, Е. Б. Сборник задач по имитационному моделированию экономических процессов / Е. Б. Грибанова, А. А. Мицель. – Томск: ТУСУР, 2007 – 274 с.

194) Мицель, А. А. Имитационное моделирование экономических процессов. Учебное методическое пособие / А. А. Мицель, Е. Б. Грибанова. – Томск: ТМЦДО, 2007 – 50 с.

195) Мицель, А. А. Имитационное моделирование экономических процессов. Учебное пособие / А. А. Мицель, Е. Б. Грибанова. – Томск: ТМЦДО, 2007 – 143 с.

196) Грибанова, Е. Б. Сборник задач по математическому и имитационному моделированию экономических процессов / Е. Б. Грибанова, А. А. Мицель. – М.: «Горячая линия-Телеком», 2019. – 252 с.

197) Мицель, А. А. Имитационное моделирование экономических процессов: Учеб. пособие: в 2-х частях / А. А. Мицель, Е. Б. Грибанова – Томск.: Томский межвузовский центр дистанционного образования, 2005. – 236 с.

198) Мицель, А. А. Имитационное моделирование экономических объектов: Лабораторный практикум / А. А. Мицель, Е. Б. Грибанова. – Томск.: Изд-во НТЛ, 2005. – 160 с.

199) Мицель, А. А. Имитационное моделирование экономических процессов: Методические указания по выполнению лабораторных работ и курсового проекта / А. А. Мицель, Е. Б. Грибанова. – Томск: ТУСУР, 2006. – 108 с.

200) Грибанова, Е. Б. Игровые имитационные модели объектов экономики / Е. Б. Грибанова // Динамика систем, механизмов и машин. – 2016. – №1. – С. 320-323.

201) Gribanova, E. B. Gaming simulation models of economic entities / E. B. Gribanova // Advances in Computer Science Research, Proceedings of the IV International research conference "Information technologies in Science, Management, Social sphere and Medicine", Tomsk. – Atlantis Press, 2017. – P. 30–34.

202) Gribanova, E. Development of spreadsheet simulation models of gas cylinders inventory management / E. Gribanova, A. Mitsel, A. Shilnikov // Eureka: Physics and Engineering. – 2022. – № 2. – C. 116–127.

203) Грибанова, Е. Б. Процессно-ориентированное моделирование систем массового обслуживания в Excel / Е. Б. Грибанова // Прикладная информатика. – 2015. – № 6. – С. 83–90.

204) Грибанова, Е. Б. Табличная имитационная модель системы обслуживания ресторана быстрого питания / Е. Б. Грибанова, Е. А. Кармановская, И. Н. Логвин // Прикладная информатика. – 2019. – № 5. – С. 111–119.

205) Грибанова, Е. Б. Табличное моделирование как инструмент интерактивного обучения базовым понятиям эконометрики / Е. Б. Грибанова // Статистика и экономика. – 2016. – №1. – С. 40–45.

206) Грибанова, Е. Б. Имитационное моделирование экономических процессов. Практикум в Excel. Учебное пособие / Е. Б. Грибанова, И. Н. Логвин. – М.: Кнорус, 2020. – 228 с.

207) Логвин, И. Н. Сценарий многократной имитации системы табличного моделирования экономических процессов / И. Н. Логвин, Е. Б. Грибанова // Информационные технологии в науке, управлении, социальной сфере и медицине: сборник научных трудов V Международной конференции, Томск. – Томск: Издательство Томского политехнического университета, 2018. – Ч.2. – С. 82–84 .

208) Логвин, И. Н. Реализация сценария обработки формул системы табличного моделирования экономических процессов / И. Н. Логвин, Е. Б. Грибанова // Перспективы развития фундаментальных наук: сборник трудов XVI Международной конференции студентов, аспирантов и молодых учёных, Томск. – Томск: Изд-во Томского политехнического университета, 2019, Т.5. – С. 96-98.

209) Gribanova, E. Elaboration of an Algorithm for Solving Hierarchical Inverse Problems in Applied Economics / E. Gribanova // Mathematics. - 2022. - № 10 (15).

210) Айвазян, С. А. Прикладная статистика и основы эконометрики: Учебник для вузов / С. А. Айвазян, В. С. Мхитарян – М.: ЮНИТИ, 1998. – 1005 с.

211) Grigoreva, M. Research Activities of Students and Information Technology as a Method to Support Interdisciplinary Teaching in Training Process of Technical University / M. Grigoreva, E. Gribanova, T. Kust // Advances in Computer Science

Research: Proceedings of the 2016 Conference on Information Technologies in Science, Management, Social Sphere and Medicine, Tomsk. – Atlantis Press, 2016. – С. 322–327.

212) Грибанова, Е. Б. Модели прогнозирования выручки ресторана быстрого питания / Е. Б. Грибанова, Е. С. Соломенцева // Экономический анализ: теория и практика. – 2018. – № 4. – С. 754–767.

213) Gribanova, E. B. Models to forecast revenue of fast food restaurants / E. B. Gribanova, E. S. Solomentseva // Дайджест-финансы. – 2018. – № 2. – С. 212–220.

214) Грибанова, Е. Б. Эконометрика. Практикум. Учебное пособие / Е. Б. Грибанова. – М.: «Горячая линия-Телеком», 2019. – 148 с.

215) Терских, Д. В. Исследование регрессионных моделей прогнозирования онлайн продаж купонов на скидку / Д. В. Терских, Е. Б. Грибанова // Информационные технологии в науке, управлении, социальной сфере и медицине: сборник научных трудов V Международной конференции, Томск. – Томск: Издательство Томского политехнического университета, 2018. – Ч.2. – С.114-117 .

216) Терских, Д. В. Модель поддержки принятия решений о размещении купона на сайте cupon.tomsk.ru / Д. В. Терских, Е. Б. Грибанова // Инноватика–2019: сб. материалов XV Международной школыконференции студентов, аспирантов и молодых ученых, Томск. – Томск: STT, 2019. – С. 425–427.

217) Шакирзянова, А. М. Регрессионная модель оценки рыночной стоимости объектов недвижимости / А. М. Шакирзянова, Е. Б. Грибанова // Молодежь и современные информационные технологии: сборник трудов XII Международной научно-практической конференции студентов, аспирантов и молодых ученых, Томск. – Томск: Национальный исследовательский Томский политехнический университет, 2014. – С. 172–173.

218) Грибанова, Е. Б. Модель оценки удовлетворенности потребителей на основе онлайн-отзывов с помощью метода главных компонент / Е. Б. Грибанова, В. В. Саулин // Моделирование, оптимизация и информационные технологии. – 2021. – № 2.– С. 6–7.

219) Спицын, В. В. Моделирование влияния возраста на техническую эффективность в разрезе отраслей и временных рядов / В. В. Спицын, Л. Ю. Спицына, Е. Б. Грибанова // Вестник университета. – 2022. – № 10. – С. 59–68.

220) Спицына, Л. Ю. Цифровой капитал российских предприятий: тенденции развития в условиях цифровизации экономии и пандемии коронавируса / Л. Ю. Спицына, Е. Б. Грибанова, В. В. Спицын // Вестник университета. - 2022. - № 2. - С. 160-169.

221) Грибанова, Е. Б. Компьютерная обработка данных аналитических систем для исследования развития цифрового капитала организаций / Е. Б. Грибанова, Л. Ю. Спицына, К. А. Бозымбаева // Научная сессия ТУСУР: сборник избранных статей международной научно-технической конференции студентов, аспирантов и молодых ученых, Томск. – Томск: В-Спектр, 2022. – Ч.3. – С. 19–22.

222) Gribanova, E. B. Economic analysis inversion mechanism taking into account argument interrelation / E. B. Gribanova, I. N. Logvin. // Proceedings of the 1st International Scientific Conference "Modern Management Trends and the Digital Economy: from Regional Development to Global Economic Growth", Ekaterinburg. – Atlantis Press, 2019. – P. 86–92.

223) Грибанова, Е. Б. Система решения обратной задачи формирования маржинальной прибыли предприятия / Е. Б. Грибанова, И. Н. Логвин // Сборник трудов Всероссийской научно-практической конференции студентов, аспирантов и молодых учёных «Современные технологии принятия решений в цифровой экономике», Юрга. – Томск: Изд-во Томского политехнического университета, 2018. – С. 242–244.

224) Свидетельство о гос.регистрации программы для ЭВМ № 2021661430. Формирование маржинальной прибыли предприятия / Грибанова Е.Б., Логвин И.Н.; заявл. 07.07.2021; зарег. 12.07.21

225) Грибанова, Е. Б. Метод решения обратных задач экономического анализа на основе статистических данных / Е. Б. Грибанова, П. Э. Тугар-оол // Корпоративные финансы. – 2017. – №3. – С. 111–120.

226) Свидетельство о гос.регистрации программы для ЭВМ № 2022662734. Программа решения иерархической многономенклатурной обратной задачи формирования прибыли предприятия / Грибанова Е.Б.; заявл. 30.06.2022; зарег. 07.07.2022.

227) Грибанова, Е. Б. Разработка программы для решения иерархической обратной задачи формирования прибыли предприятия / Е. Б. Б. Грибанова // Математическое и информационное моделирование: сб. материалов Всероссийской конференции молодых учёных, Тюмень. – Тюмень: ТюмГУ-Press, 2022. – С. 63–66.

228) Свидетельство о гос.регистрации программы для ЭВМ № 2021615540. Программа решения обратной задачи формирования прибыли с помощью стохастических алгоритмов / Грибанова Е.Б.; заявл. 09.03.2021; зарег. 09.04.21

229) Грибанова, Е. Б. Информационная система рейтинговой оценки объектов экономики / Е. Б. Грибанова, А. Н. Алимханова, П. Э. Тугар-оол // Доклады Томского государственного университета систем управления и радиоэлектроники. – 2016. – №2. – С. 51–55.

230) Свидетельство о гос.регистрации программы для ЭВМ № 2021615840. Программа формирования интегрального показателя социально-экономического объекта / Грибанова Е.Б.; заявл. 09.03.2021; зарег. 13.04.21.

231) Грибанова, Е. Б. Модель оценки групп социальной сети для реализации маркетинговых мероприятий / Е. Б. Грибанова, А. В. Катасонова // Доклады Томского государственного университета систем управления и радиоэлектроники. – 2017. – №2. – С. 68–72.

232) Грибанова, Е. Б. Оптимизационные модели выбора групп социальной сети для размещения рекламы / Е. Б. Грибанова // Экономический анализ: теория и практика. – 2017. – №10. – С. 1989–2000.

233) Свидетельство о гос.регистрации программы для ЭВМ № 2021662008. Оценка групп социальной сети для реализации маркетинговых мероприятий / Грибанова Е.Б.; заявл. 07.07.2021; зарег. 20.07.21.

234) Gribanova, E. B. Econometric models for evaluation of marketing activities` indicators of social network / E. B. Gribanova, I. V. Shirenkov, A. V. Katasonova // Advances in Economics, Business and Management Research: Proceedings of the conference «Trends of Technologies and Innovations in Economic and Social Studies», Tomsk. – Atlantis Press, 2017. – P. 227–233.

235) Gribanova, E. B. Algorithms for Solving Inverse Problems of Simulation Modeling / E. B. Gribanova // International Journal of Computing. – 2021. – №3. – С. 433–439.

236) Катасонова, А. В. Распознавание оплаченных аккаунтов пользователей социальной сети "ВКонтакте" / А. В. Катасонова, Е. Б. Грибанова // Инноватика–2019: сб. материалов XV Международной школыконференции студентов, аспирантов и молодых ученых, Томск. – Томск: STT, 2019. – С. 388–392.

237) Грибанова, Е. Б. Алгоритмы моделирования распространения информации при маркетинговых мероприятиях в группах онлайновой социальной сети / Е. Б. Грибанова // Проблемы управления. – 2018. – № 1. – С. 66–73.

238) Грибанова, Е. Б. Алгоритм оценки маркетинговых мероприятий онлайновой социальной сети «ВКонтакте» на основе каскадной модели распространения информации / Е. Б. Грибанова, И. Н. Логвин, И. В. Ширенков // Доклады Томского государственного университета систем управления и радиоэлектроники. – 2018. – №3. – С. 69–74.

239) Законов, А. В. Сравнение алгоритмов обучения нейронной сети для идентификации потенциальных покупателей / А. В. Законов, Е. Б. Грибанова // Информационные технологии в науке, управлении, социальной сфере и медицине: сборник научных трудов V Международной конференции, Томск. – Томск: Издательство Томского политехнического университета, 2018. – Ч.2. – С. 42–45.

240) Законов, А. В. Система для оценки наличия потенциальных покупателей в группах онлайн социальной сети «ВКонтакте» / А. В. Законов, Е. Б. Грибанова // Научная сессия ТУСУР: Материалы Международной научно-технической конференции студентов, аспирантов и молодых ученых, Томск. – Томск: В-Спектр, 2019. – Ч.3. – С. 119–121.

241) Грибанова, Е. Б. Задача выбора наилучшего момента размещения сообщений в группах онлайновой социальной сети / Е. Б. Грибанова, К. А. Бозымбаева // Современное образование: интеграция образования, науки, бизнеса и власти: сб. материалов международной научно-методической конференции, Томск. – Томск: ТУСУР, 2022. – С. 361–364.

242) Gribanova, E. B. Algorithm for Estimating the Time of Posting Messages on Vkontakte Online Social Network / E. B. Gribanova, A.S. Savitsky // International Journal on Information Technologies and Security. – 2020. – №1. – P. 3–14.

243) Грибанова, Е. Б. Разработка программы оценки времени размещения сообщения в онлайновой социальной сети ВКонтакте / Е. Б. Грибанова, А. С. Савицкий // Моделирование, оптимизация и информационные технологии. – 2020. – №1. – С. 13–14.

244) Свидетельство о гос.регистрации программы для ЭВМ № 2021662008. Программа оценки времени размещения сообщений в группах онлайновой социальной сети ВКонтакте / Грибанова Е.Б., Савицкий А.С.; заявл. 07.07.2021; зарег. 09.07.21

Приложение А Программные системы имитационного моделирования

Программа «Имитатор»

Программа «Имитатор» реализована на языке Java. Выполнение программы возможно при использовании любой операционной системы, однако на компьютере должна быть установлена среда исполнения Java (JRE – Java Runtime Environment). Для обмена информацией между вычислительной частью и ГИП служит текстовый файл DataModel.txt.

Программа предназначена для исследования различных экономических объектов с помощью разработанных имитационных моделей.

Рассмотрим интерфейс системы. После запуска программы осуществляется загрузка моделей, номера которых указаны в файле. В качестве примера на рисунке А.1 приведен случай загрузки пяти моделей. Далее осуществляется выбор пользователем необходимой модели (на рисунке А.2 представлена форма модели управления запасами).

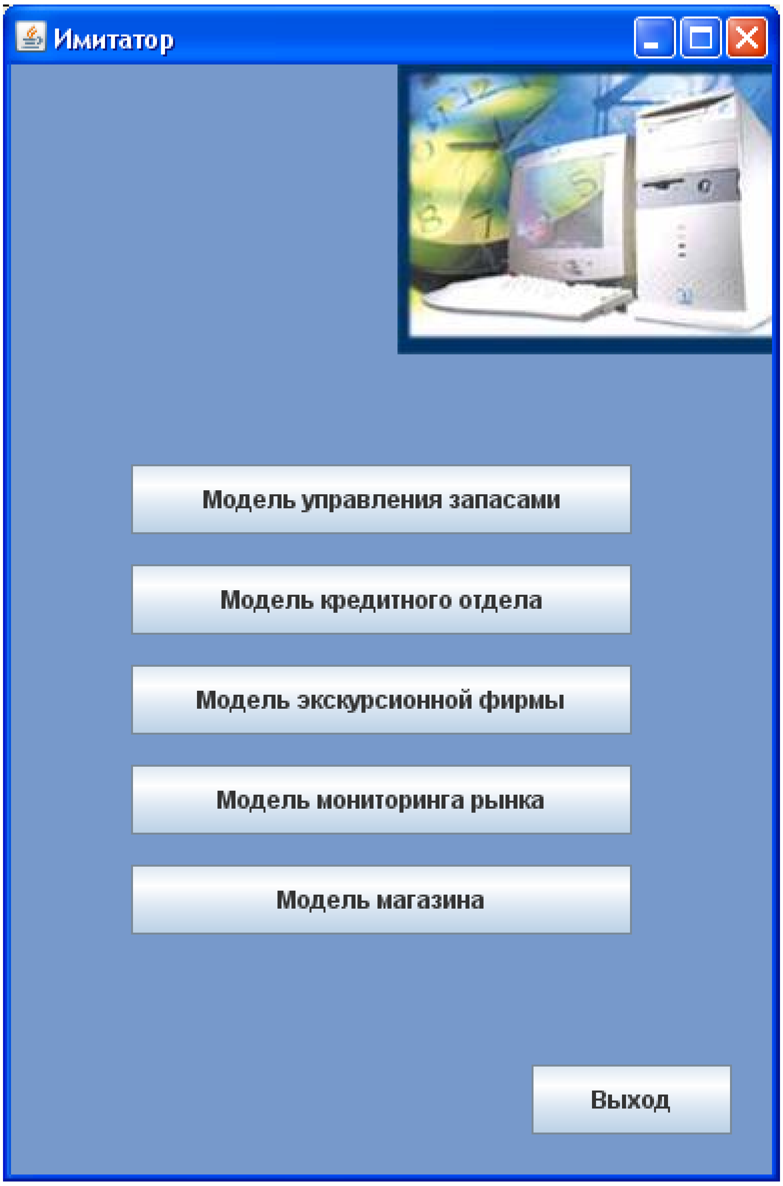
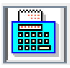


Рисунок А.1 – Стартовая форма

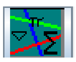
Окно модели включает кнопки для выполнения следующих действий:

 – имитация рассматриваемого экономического объекта;

 – получение справочной информации;

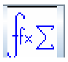
 – сохранение в файл исходных данных;

 – загрузка из файла входных параметров;

 – вызов диалогового окна «Оценка рисков»;

 – вызов окна «Свойства эксперимента»;

 – вызов окна «Свойства графика»;

 – вызов окна «Статистические характеристики»;

– выход в главное окно.

Некоторые из названных окон представлены на рисунке А.3.

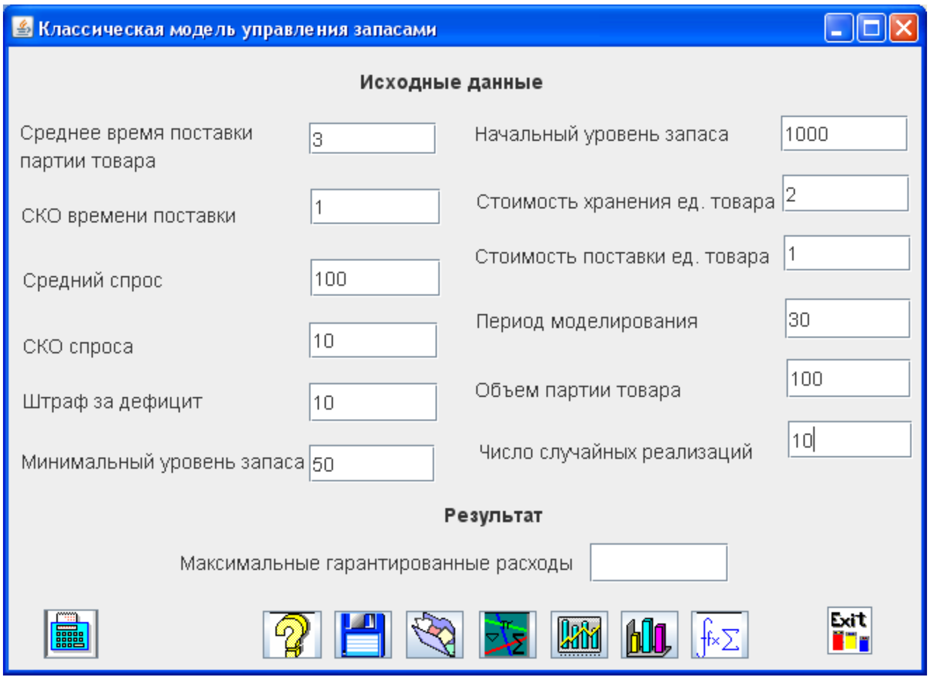
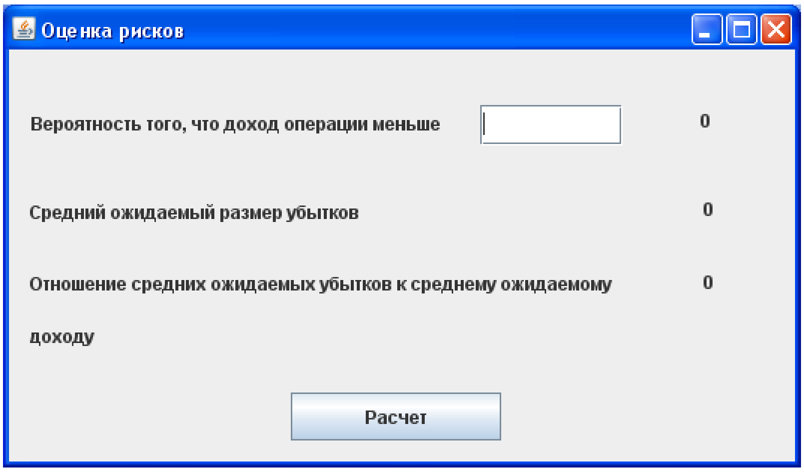


Рисунок А.2 – Окно модели

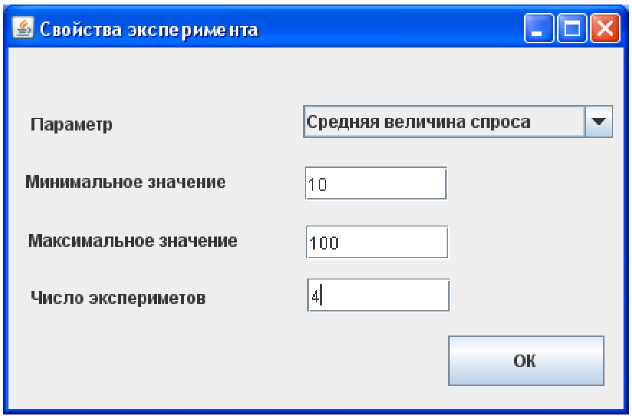
Программа обладает графическим пользовательским «однодокументным» интерфейсом. Это означает, что в каждый момент времени доступным является только одно окно приложения. На рисунке А.4 приведен пример схемы вызова окон программы.

Работа с имитационной моделью заключается в следующем. Пользователем вводятся исходные данные, после чего программа осуществляет имитацию функционирования исследуемой системы во времени. Затем могут вызваны этапы, осуществляющие выполнение дополнительных подзадач: оценку рисков, статистическую обработку результатов моделирования и т.д.

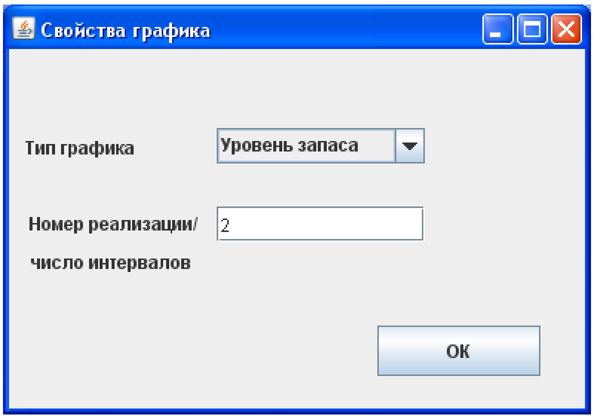
Простота интерфейса приводит к минимальным затратам времени на изучение программы пользователем.



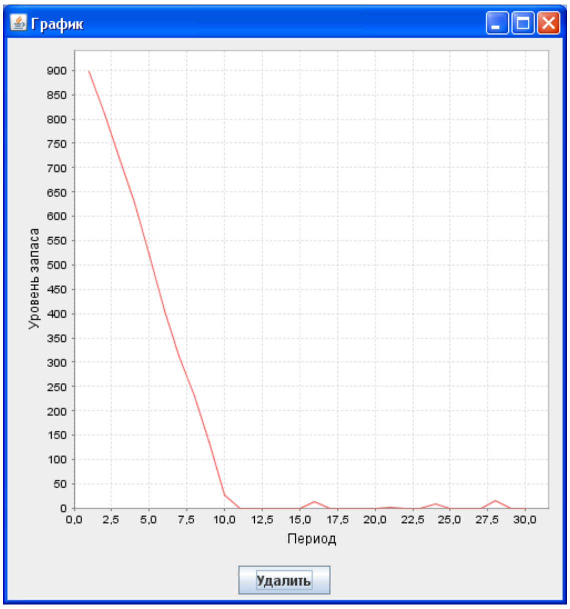
а)



б)



в)



г)

Рисунок А.3 – Основные формы программы «Имитатор»: а) окно оценки рисков; б) окно установки свойств эксперимента; в) окно установки свойств графика; г) окно графика

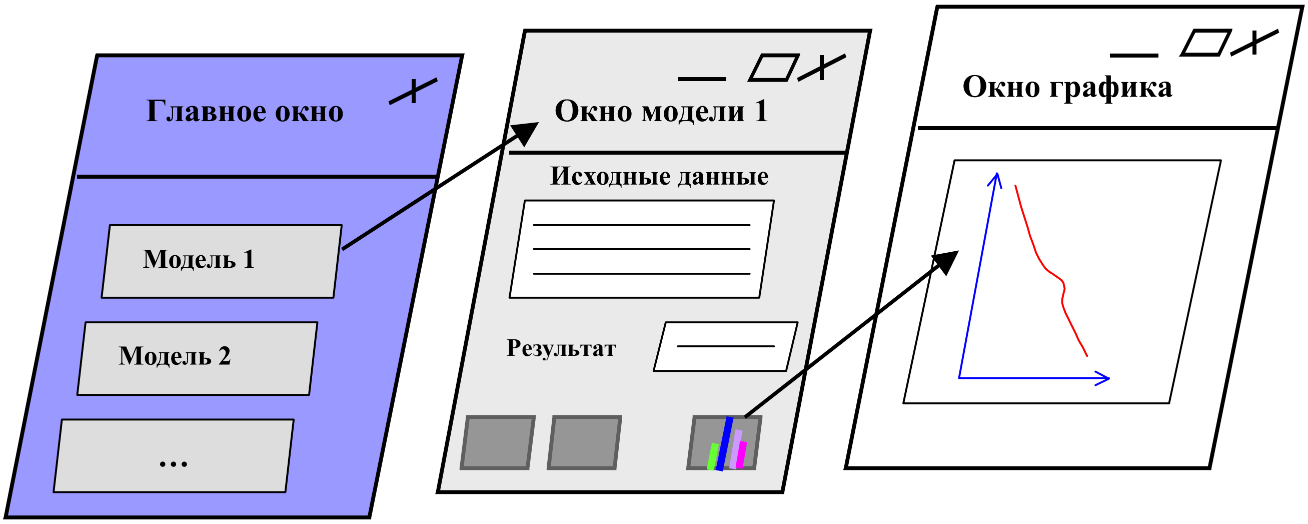


Рисунок А.4 – Схема вызова окон программы

Дополнительно был создан электронный учебник (в виде связанных html – страниц), описывающий теоретические основы имитационного моделирования и этапы разработки включенных моделей, а также средство самоконтроля (рисунок А.5–А.6).

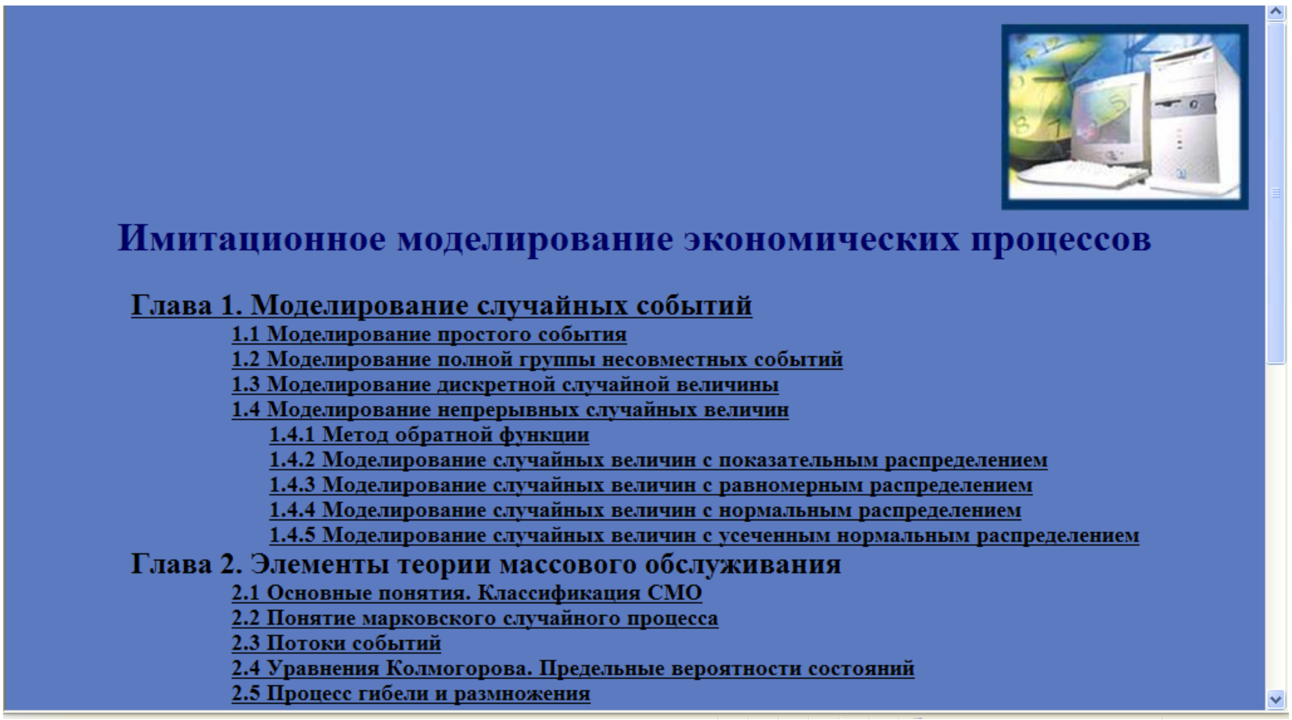


Рисунок А.5 – Электронный учебник

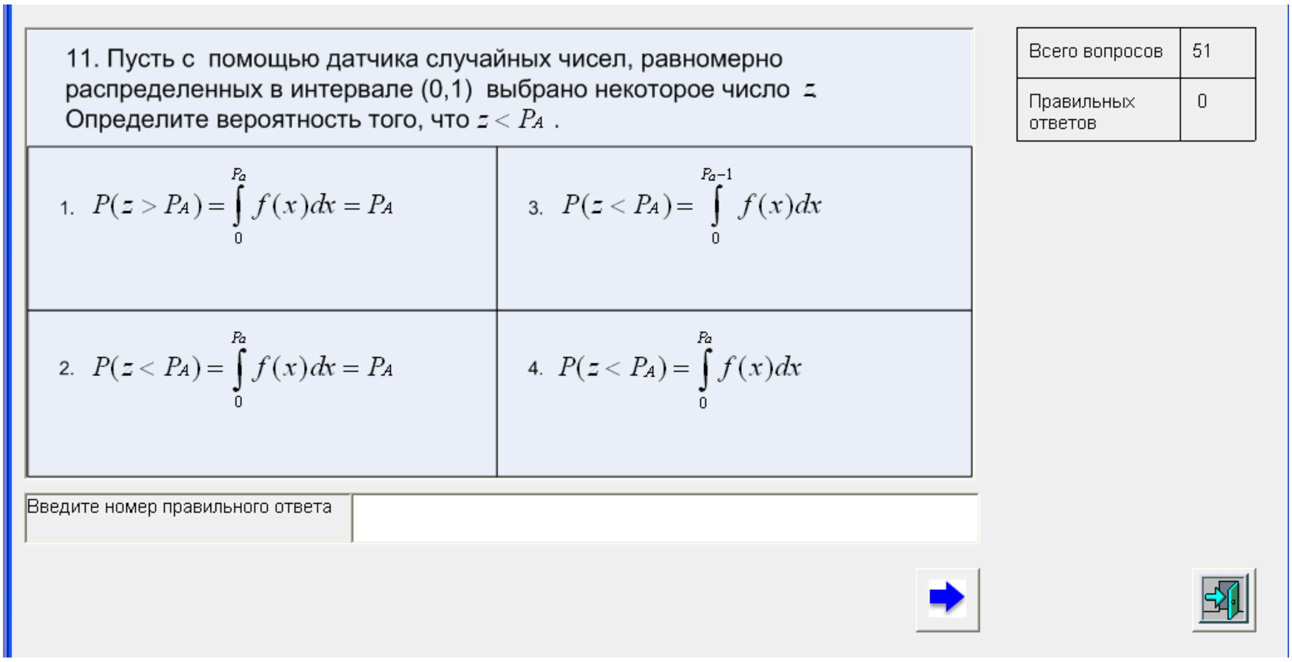


Рисунок А.6 – Форма самоконтроля

Программа «Запас»

Программа «Запас» реализован в среде Visual Basic 6.0 и предназначена для имитационного моделирования систем управления запасами с различными характеристиками: с периодической и пороговой стратегиями подачи заявок, случайным, детерминированным, замкнутым, отложенным спросом и т.д. Она позволяет оценивать системы с заданными правилами функционирования, анализировать возможные изменения, происходящие в результате действия различных факторов.

На рисунке А.7 представлены основные элементы интерфейса системы.

В качестве стиля интерфейса системы был выбран MDI (интерфейс со многими документами). Его техника заключается в использовании одного первичного окна, называемого родительским, которое может содержать набор связанных с ним дочерних окон.

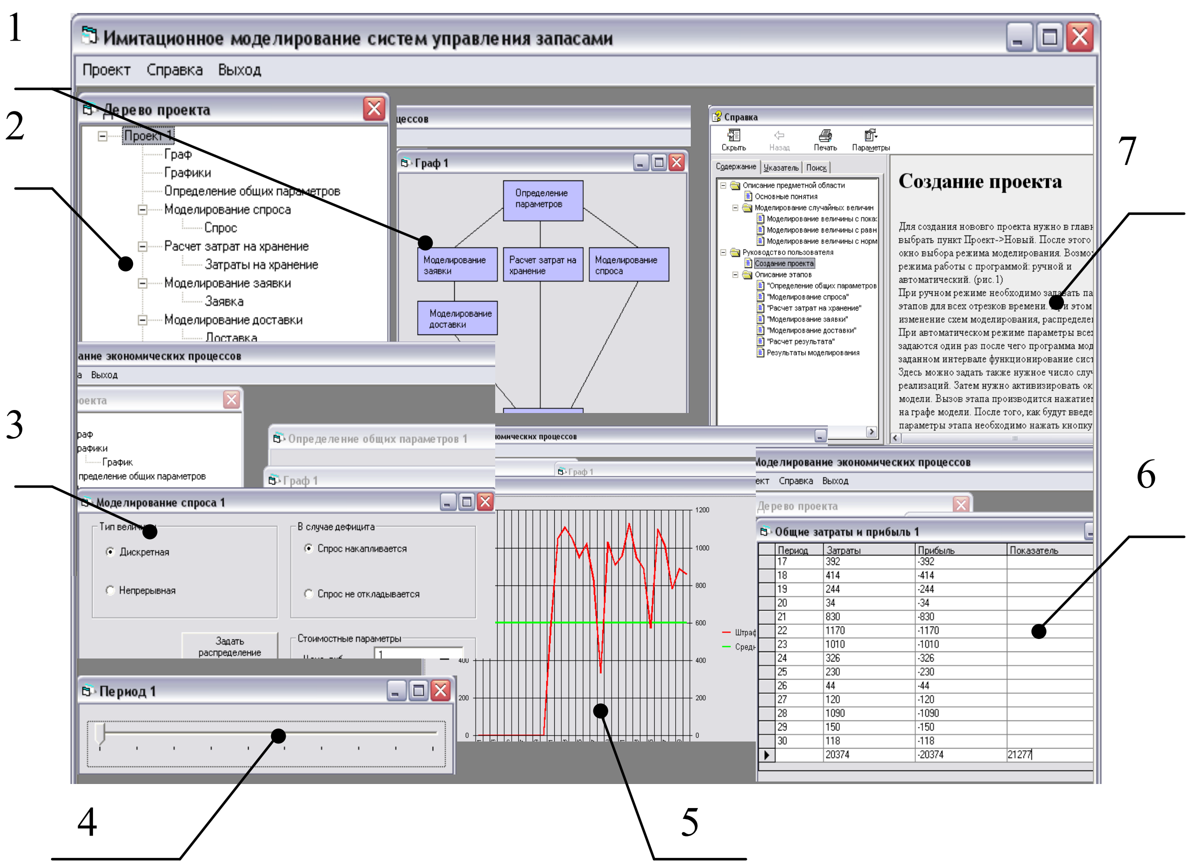


Рисунок А.7 – Интерфейс системы: 1 – Граф алгоритма; 2 – Дерево проекта; 3 – Окно ввода параметров этапа; 4 – Элемент управления модельным временем; 5 – Окно графика; 6 – Окно таблицы с результатами моделирования; 7 – Окно справки.

Родительское окно включает дерево проекта, которое обеспечивает доступ к графу алгоритма и рассчитанным этапам. Программа может включать несколько проектов, объекты которых доступны из дерева проекта.

Процесс моделирования включает имитацию отдельных событий функционирования склада, каждому из которых соответствует этап схемы на рисунке Е.7. Вызов диалогового окна ввода параметров этапа осуществляется с помощью графа алгоритма. После задания всех исходных значений параметров происходит решение текущей задачи узла и пересчет уже рассчитанных данных.

Выходные данные системы представляются в виде статистических таблиц и графиков.

Возможны два режима работы с программой: ручной и автоматический. Ручной режим основан на последовательном запуске этапов в течение всего периода моделирования и позволяет понять, какие действия происходят на каждом этапе и как это влияет на состояние всей системы. При работе в этом режиме пользователь с помощью элемента управления получает доступ к данным каждого момента времени и может их корректировать в зависимости от ситуации (а также рассматривать системы с изменяющимися во времени исходными данными). Например, если на какой-то день были обнаружены значительные потери из-за дефицита товара, то пользователь может вернуться в предыдущий период, подать заявку на доставку или заказать партию большего объема и посмотреть, как изменятся значения параметров модели, и увеличит ли такое решение эффективность системы. Таким образом, можно осуществить поиск оптимальной стратегии управления запасами. Кроме того, ручной режим работы увеличивает степень доверия к результатам моделирования.

При автоматическом режиме пользователь сначала вводит все параметры, и затем программа без его участия моделирует поведение системы в заданном промежутке времени в течение определенного количества случайных реализаций. После окончания моделирования пользователь имеет возможность просмотра всех статистических данных за каждую реализацию.

Программа «Аукцион»

Программа «Аукцион» реализована на языке Java с помощью среды Eclipse. Для построения графиков была использована библиотека JfreeChart. Для обмена данными между интерфейсом и вычислительной частью предназначен файл DataModel.txt.

Система предназначена для имитационного моделирования аукционов, проводимых согласно ФЗ №94, в ходе которых определяется победитель и предпоследний участник. Реализовано два механизма проведения аукциона и три стратегии участия в них (истинного, случайного и максимального предложения). Программа позволяет сравнивать характеристики рассматриваемых торгов и осуществлять выбор оптимального механизма проведения аукциона в выбранных условиях. Поэтапное выполнение имитации обеспечивает возможность работы с системой в двух режимах: ручном поэтапном и автоматическом.

Рабочее пространство системы включает форму, которая содержит главное меню и панель с тремя вкладками: «Ввод данных», «Моделирование», «Результаты». Работа с программой начинается с выбора с помощью главного меню режима работы и механизмов аукционов, используемых при имитации, а также ввода исходных данных (рисунок А.8). Для генерирования случайных чисел предназначена форма, представленная на рисунок А.9. Кроме того, исходные данные можно загрузить из файла. В таблице значений характеристик участников доступ к данным каждой реализации осуществляется с помощью стрелок «←» и «→».

Следующий этап работы включает проведение имитации аукционов с выбранными механизмами. В зависимости от установленного режима работы с программой действия пользователя могут заключаться либо в нажатии кнопки <ОК> (при автоматическом режиме) (рисунок А.10 а) для выполнения имитации торгов вплоть до их завершения, либо в проведении поэтапного моделировании, требующего определения поведения участников на каждом шаге аукциона (при ручном режиме) (рисунок А.10 б). Таблица, включающая информацию о результатах имитации, состоит из следующих граф:

– «Шаг» – номер шага аукциона;

– «Цена_Схема1», «Цена_Схема2» – цены, объявленные на текущем шаге аукционов, использующих первую и вторую схему (механизм) поиска предпоследнего участника соответственно;

– «Согласные_участники» – участники, выразившие согласие с предложенной на данном шаге ценой (порядок их отображения в таблице соответствует последовательности выражения ими согласия);

– «Величина_шага» – величина шага аукциона в долях.

Наконец, на последнем этапе осуществляется определение наилучшего механизма проведения аукциона. Для этого вводятся коэффициенты важности таких характеристик, как эффективность предложения второго участника, число шагов, величина предложения предпоследнего участника, а затем рассчитывается целевая функция и определяется позиция каждого механизма (механизм с большим значением целевой функции занимает более высокую позицию и считается предпочтительным).

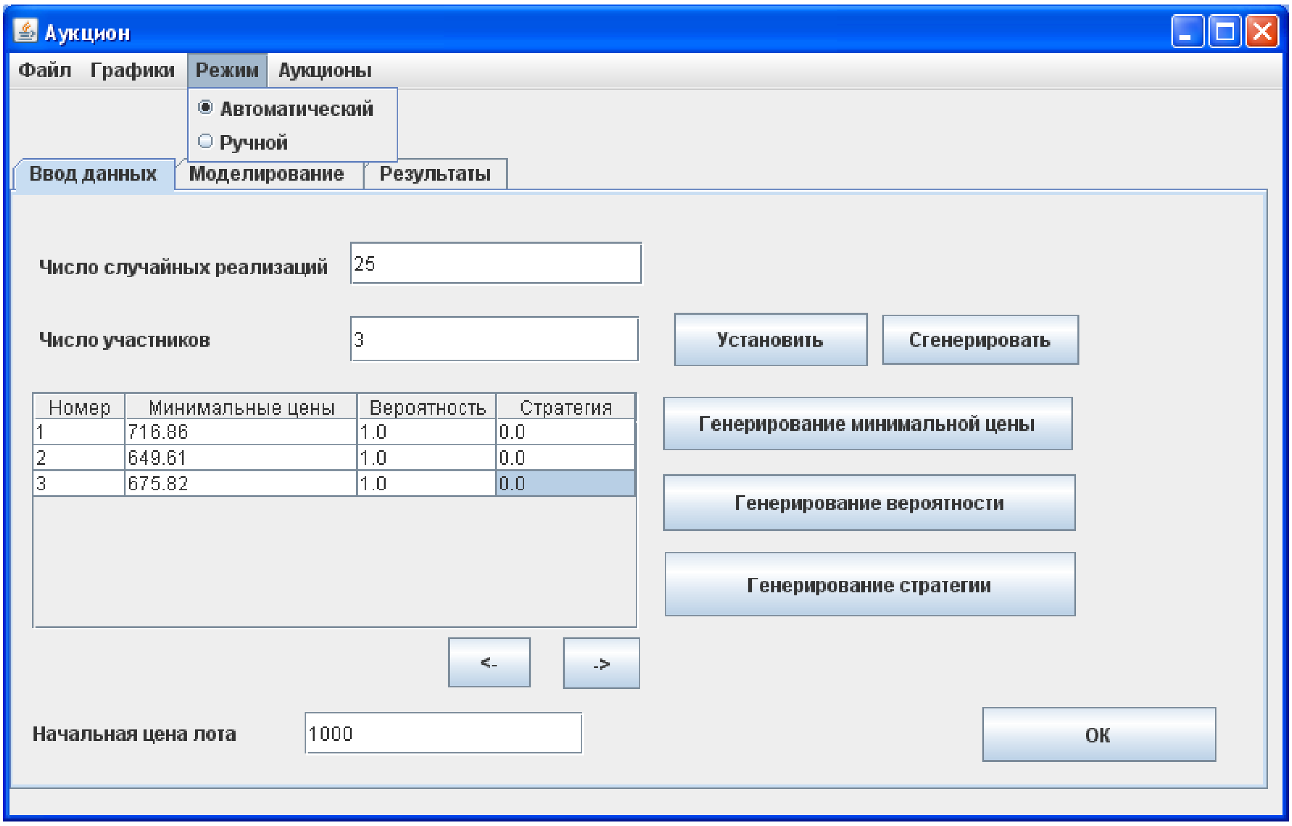


Рисунок А.8 – Интерфейс системы. Форма ввода исходных данных

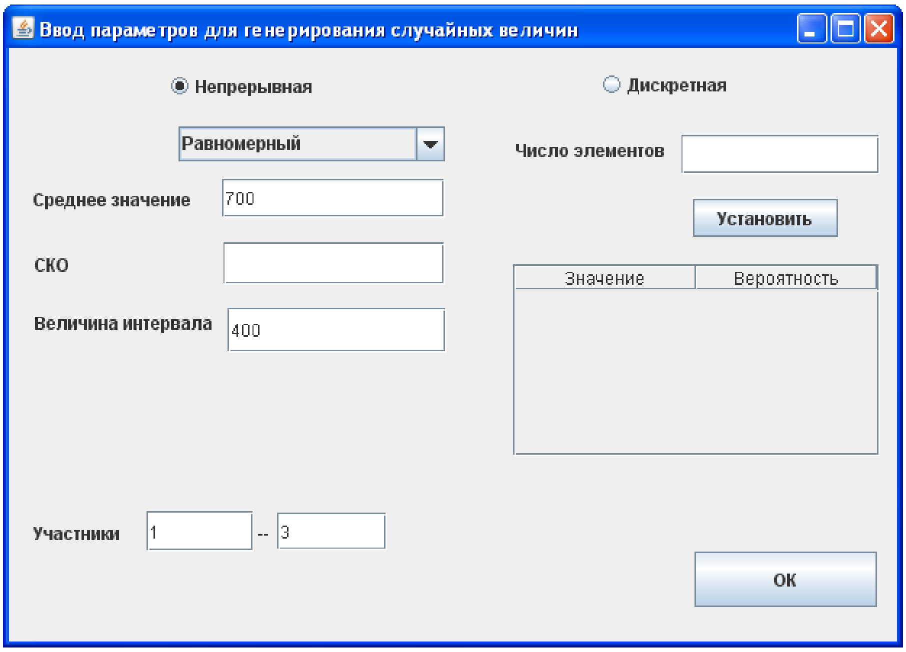
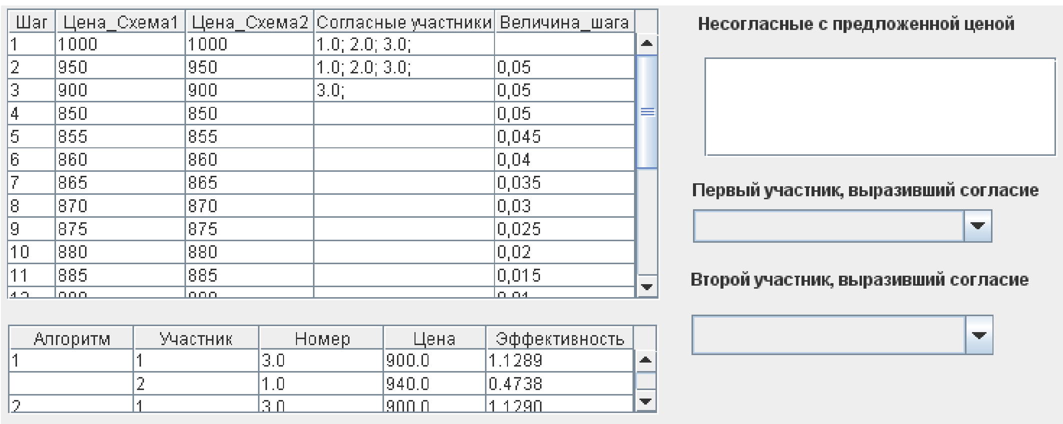
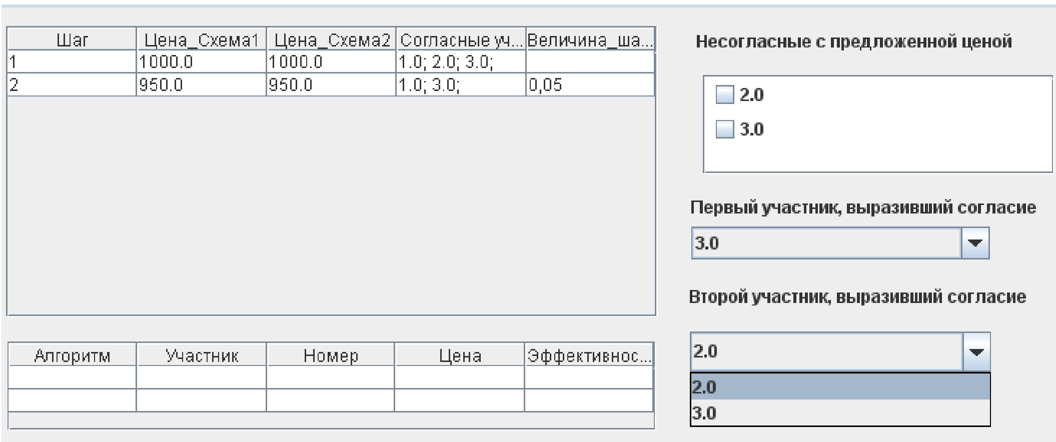


Рисунок А.9 – Окна ввода параметров для генерирования случайных величин



а)



б)

Рисунок А.10 – Форма с результатами имитации: а) автоматический режим; б) ручной режим

Приложение Б Программа решения иерархической многономенклатурной обратной задачи формирования прибыли

Реализация программы выполнена на языке C# в среде Visual Studio. Для реализации алгоритмов использован объектно-ориентированный подход с представлением задачи в виде дерева. Далее представлено описание интерфейса.

После запуска программы будет открыта следующая форма (рисунок Б.1).

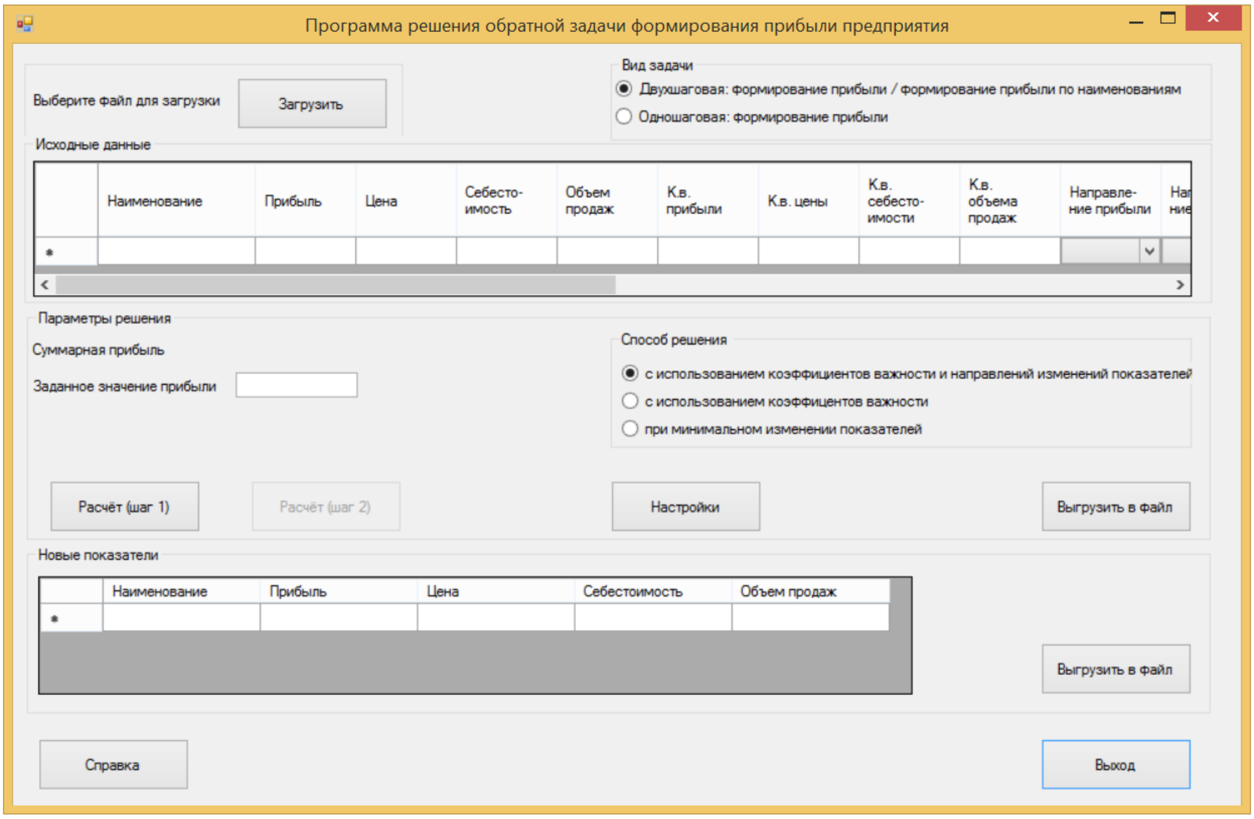


Рисунок Б.1 – Главная форма

Для работы программы необходимо загрузить данные из Excel-файла. Структура входного файла приведена в файле «Исходные данные.xls» (рисунок Б.2).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	
1	Наименование	Цена	Себестоим	Объем пр	К.в. Прибі	Направле	Мин.приб	Макс.прив	К.в. Цены	К.в. Себес	к.в. Объем	Направле	Направле	Направле	Мин
2	Гречневая лапша С	100	19,1	153	0,05	+	0	100000	0,5	0,2	0,3	+	-	+	
3	Пшеничная Удон	100	15,29	234	0,2	+	0	100000	0,5	0,2	0,3	+	-	+	
4	РИС	100	12,025	280	0,1	-	0	100000	0,5	0,2	0,3	-	+	+	
5	Рисовые спагетти	100	23,94	131	0,05	+	0	100000	0,5	0,2	0,3	+	-	+	
6	ФУНЧОЗА	100	27,77	153	0,2	+	0	100000	0,5	0,2	0,3	+	-	+	
7	ЯИЧНАЯ	100	17,25	387	0,1	-	0	100000	0,5	0,2	0,3	-	+	+	
8	Вырезка из поросе	70	21,08	147	0,1	+	0	100000	0,5	0,2	0,3	+	-	+	
9	ГОВЯДИНА мал.	90	24,52	178	0,2	+	0	100000	0,5	0,2	0,3	+	-	+	

Рисунок Б.2 – Главная форма

Для загрузки данных необходимо нажать кнопку «Загрузить» рядом с надписью «Выберите файл для загрузки».

Откроется диалоговое окно открытия файла, где необходимо выбрать файл с исходными данными и нажать кнопку «Открыть».

После загрузки файла значения появятся в таблице. При этом экспертную информацию и ограничения можно редактировать (рисунок Б.3).

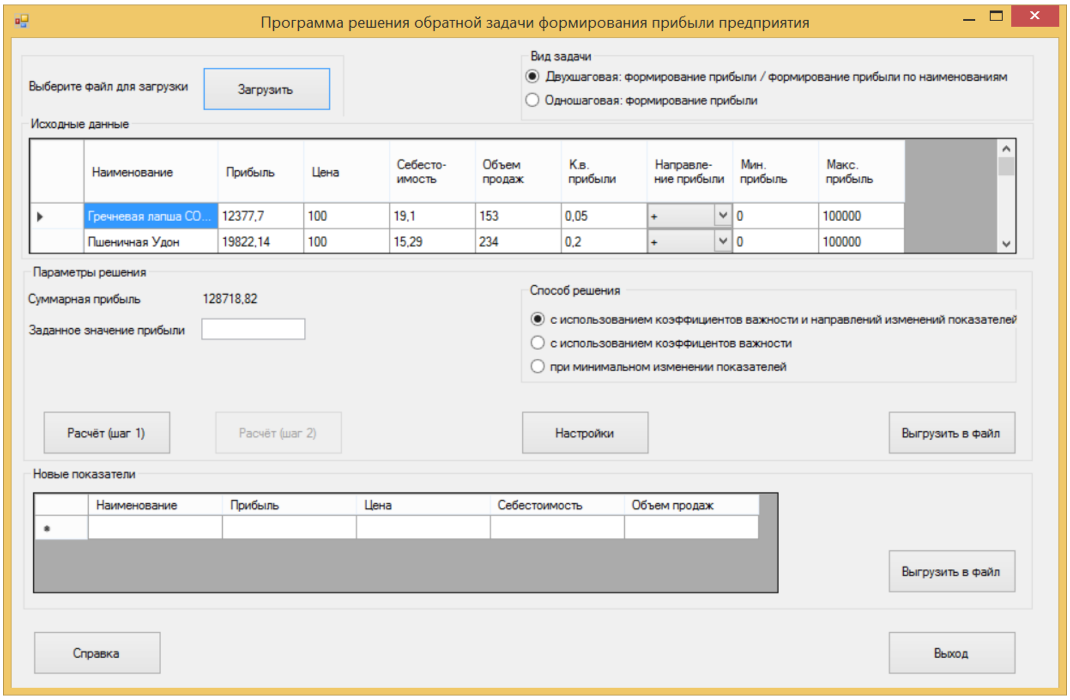


Рисунок Б.3 – Заполненная таблица исходных данных

В открывшейся форме необходимо указать следующие условия.

Вид задачи:

– Двухшаговая: решение задачи формирования суммарной прибыли происходит в два этапа. На первом этапе происходит определение прибыли по наименованию для достижения заданного значения суммарной прибыли. На втором этапе осуществляется расчёт цены, себестоимости, объема продаж для достижения заданного значения прибыли по наименованию.

– Одношаговая: определение цены, себестоимость, объема продаж для достижения заданного значения суммарной прибыли.

Способ решения задачи:

– с использованием коэффициентов относительной важности и направлений изменений показателей: необходимо для искомых значений указать направление изменения («+» – положительное, «-» – отрицательное) и коэффициенты относительной важности, которые соответствуют пропорциям изменения показателей;

– с использованием коэффициентов относительной важности: необходимо для искомых значений указать коэффициенты относительной важности, которые соответствуют пропорциям изменения показателей (направление изменения будет определено исходя из наиболее «быстрого» способа достижения цели, требующего меньшее изменение показателя);

– при минимальном изменении показателей: значения показателей определяются таким образом, чтобы их изменение было как можно ближе к нулю.

Также необходимо указать заданное значение суммарной прибыли.

После этого необходимо нажать кнопку «Расчёт (шаг1)». После завершения вычислений в таблицу «Новые показатели» будут записаны новые значения.

Двухшаговая задача

В случае двухшаговой задачи в таблице «Новые показатели» будут получены значения прибыли по наименованиям, которые позволяют получить заданную суммарную прибыль.

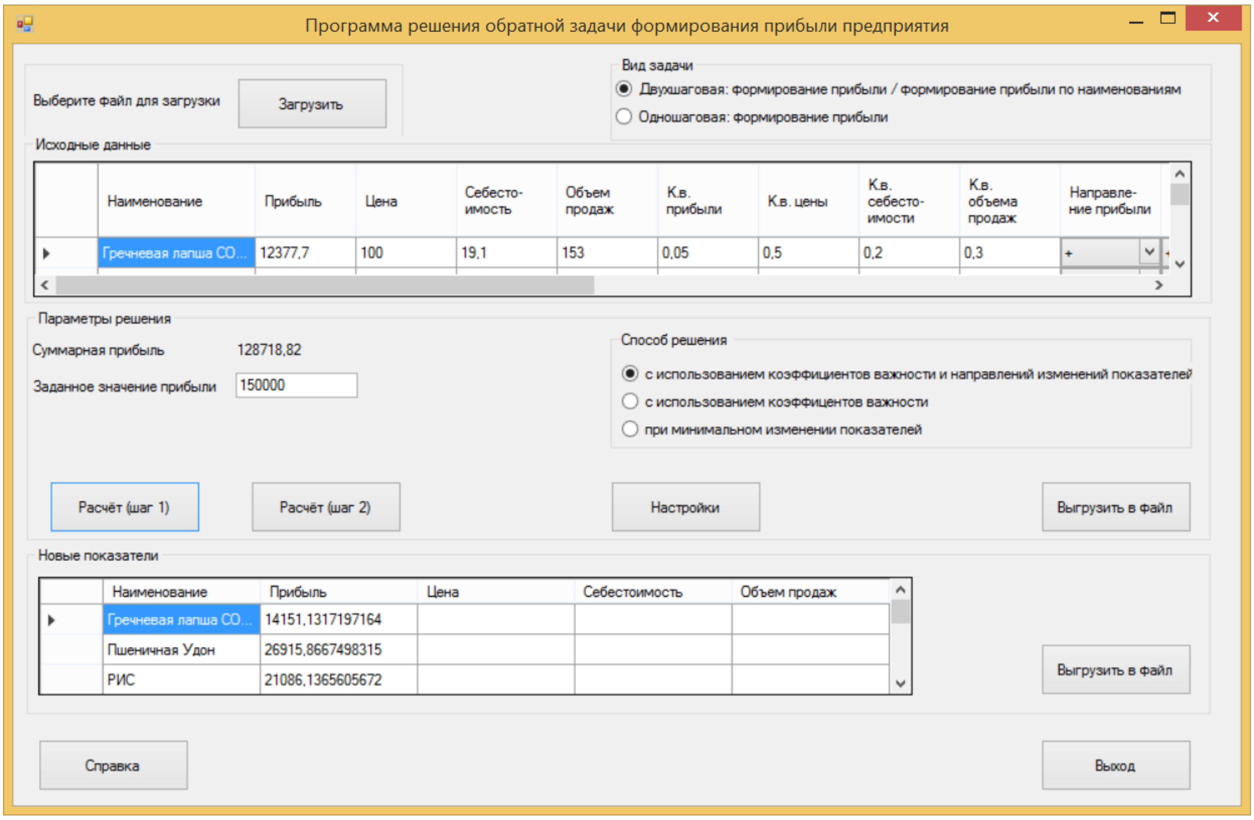


Рисунок Б.4 – Расчёт прибыли по наименованию

Далее при необходимости может быть выполнена корректировка коэффициентов относительной важности, ограничений показателей, вычисляемых на следующем уровне (цена, себестоимость, объем продаж). После этого необходимо нажать кнопку «Расчёт (шаг 2)». В таблице появятся значения цены, себестоимости, объема продаж, обеспечивающее полученное значение прибыли по наименованию (рисунок Б.5).

Одношаговая задача

В случае одношаговой задачи необходимо в таблице исходных данных определить граничные значения цены, себестоимости и объема продаж (при необходимости коэффициенты важности/направления изменения цены, себестоимости, объема продаж) (рисунок Б.6).

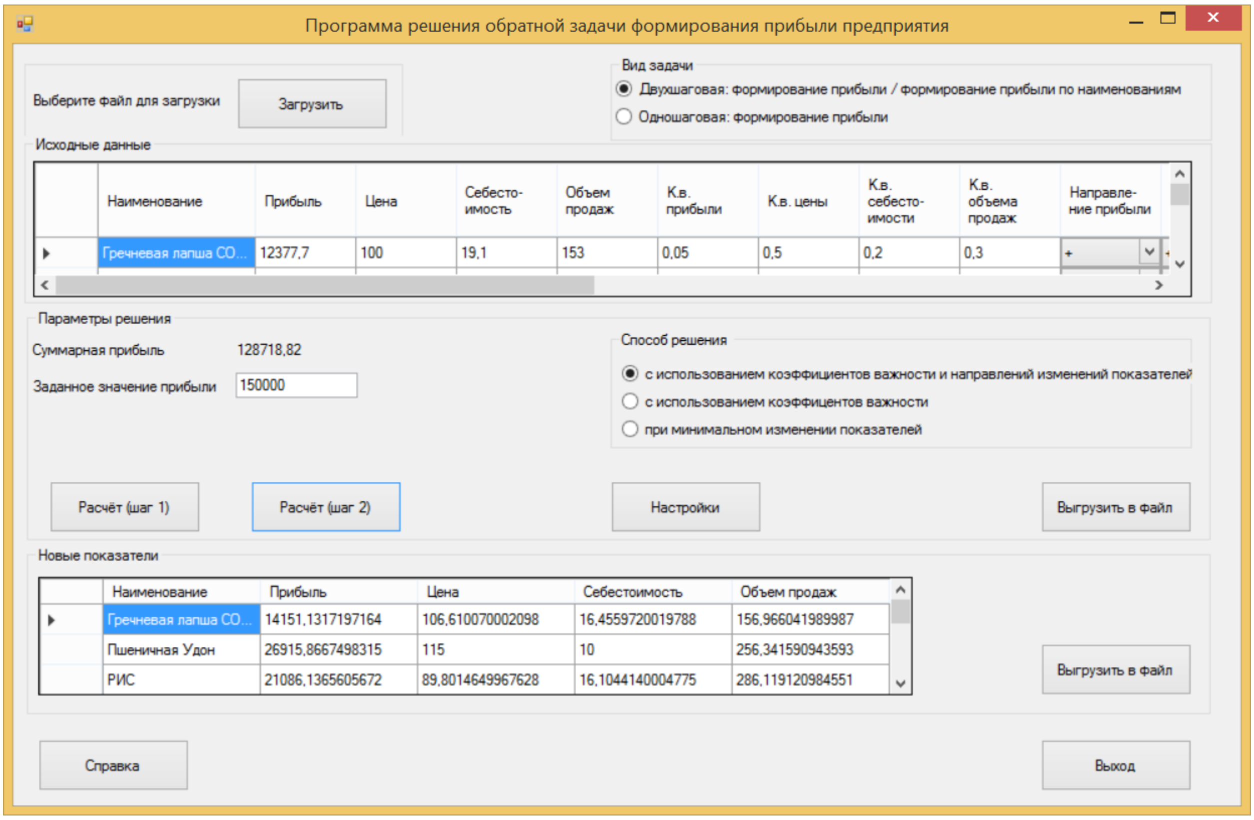


Рисунок Б.5 – Расчёт цены, себестоимости и объема продаж

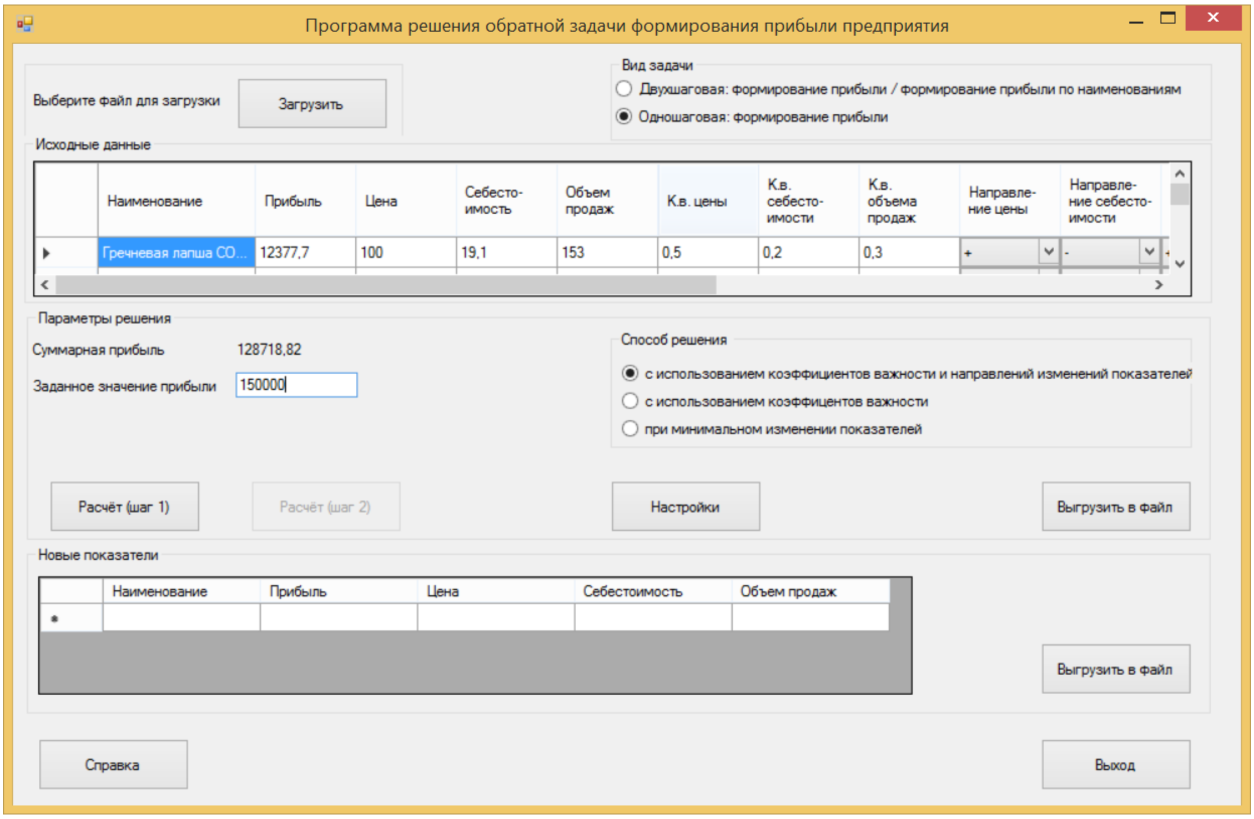


Рисунок Б.6 – Определение исходных значений

После нажатия кнопки «Расчёт (шаг 1)» будут вычислены новые значения цены, себестоимости, объема продаж для каждого наименования (рисунок Б.7).

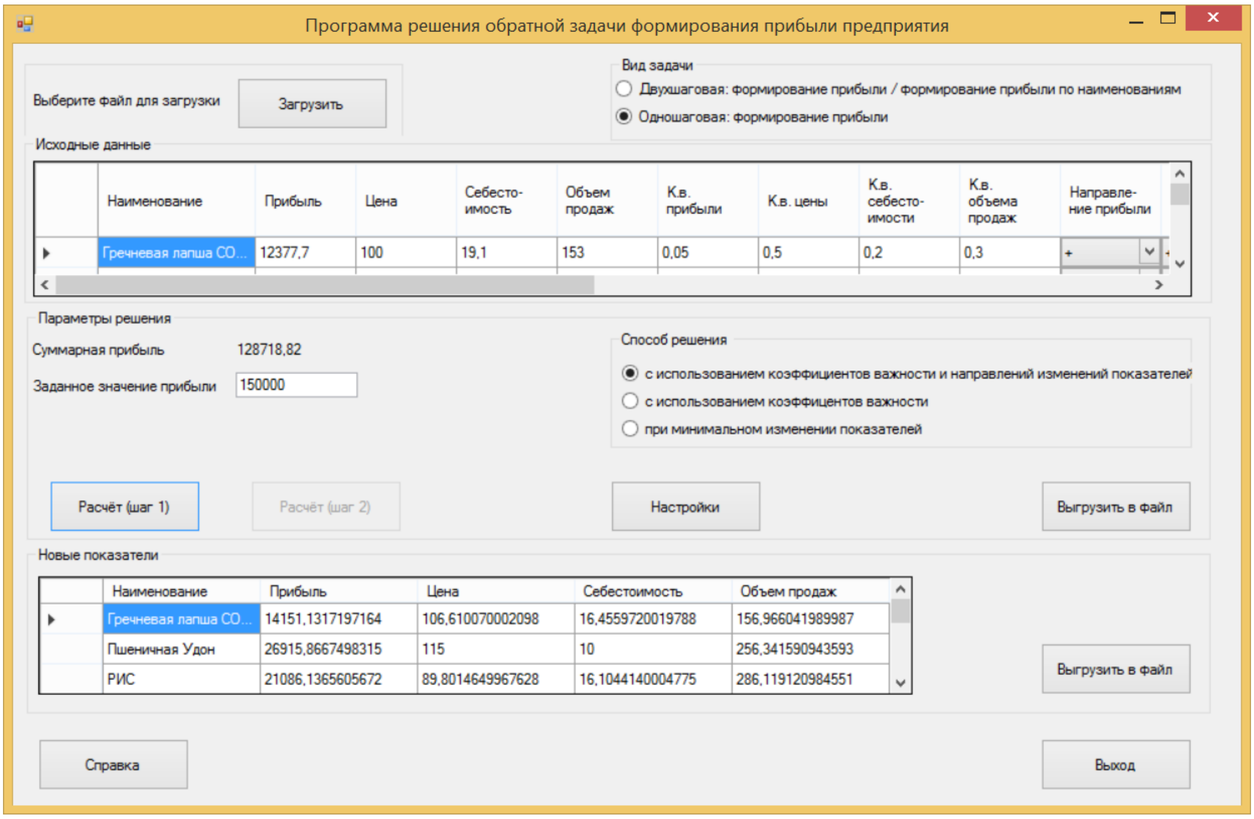


Рисунок Б.7 – Расчёт цены, себестоимости и объема продаж при одношаговой задаче

Настройка точности расчётов

Путем нажатия кнопки «Настройки» можно изменить точность решения. Чем меньше значение параметра, тем точнее будет найдено решение, но вычисления будут продолжаться дольше. Значение параметра для первого уровня устанавливает шаг изменения прибыли по наименованию для определения общей прибыли, значение параметра второго уровня устанавливает шаг изменения цены, себестоимости, объема продаж для достижения прибыли по наименованию (рисунок Б.8).

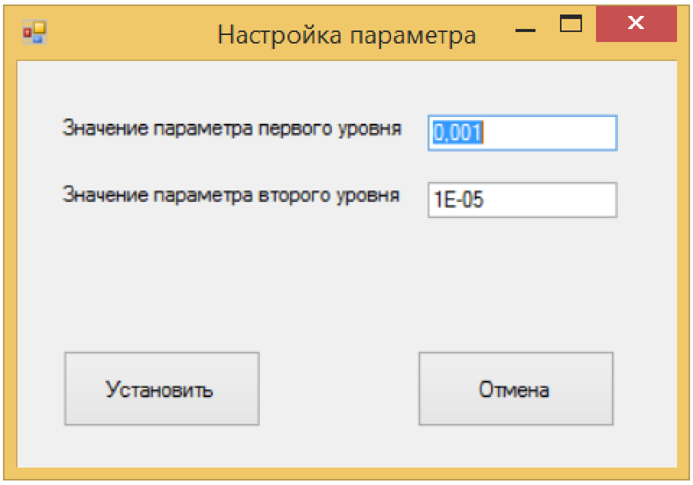


Рисунок Б.8 – Настройка точности расчётов

Выгрузка данных

Путем нажатия кнопки «Выгрузить в файл» в области «Исходные данные» можно выгрузить данные таблицы «Исходные данные» в файл Excel, которые могут быть впоследствии снова загружены для расчётов. Новый файл Excel с названием «Исходные данные» появится в папке «Результат».

Путем нажатия кнопки «Выгрузить в файл» в области «Новые показатели» можно выгрузить в файл Excel новые значения вычисленных показателей (прибыль, цена, себестоимость, объем продаж). Новый файл Excel с названием «Результат» появится в папке «Результат».

Приложение В Программа формирования маржинальной прибыли предприятия

Программа реализована на языке программирования Java в среде разработки IntelliJ IDEA. Для работы с Excel документом используется библиотека Apache-POI.

Программа имеет графический интерфейс, с помощью которого пользователь вводит входные данные, после чего запускается процесс вычислений. При этом проверяются различные варианты ввода некорректных значений в поля и таблицу, а также вычисляется дискриминант перед выполнением основных расчетов, т.к. при отрицательном дискриминанте решения не будет. В этом случае пользователю выдается текст с рекомендацией по входным параметрам α и β. В случае успешного выполнения программы формируется Excel-файл «output.xlsx» с формулами расчета, вычисленными в программе значениями, а также входными данными и описанием ячеек.

На рисунках В.1 и В.2 представлен интерфейс программы. На вкладке «Исходные данные» вводятся исходные значения: цена изделия, количество, себестоимость единицы, направления изменения показателей, коэффициенты важности цены и объема выпуска, весовые коэффициенты показателей соответствия зависимости и исходным значениям, желаемая прибыль, максимальное значение цены и шаг её изменения. На этой вкладке также может быть выбран язык интерфейса: русский, английский. Для формирования зависимости объема выпуска от цены (п.7.1.2) на вкладке «Статистические данные» вводятся значения цены и количества.

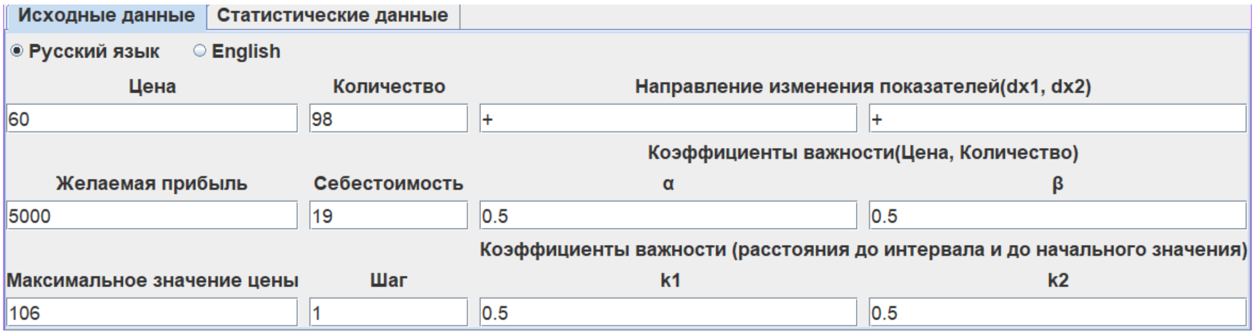


Рисунок В.1 – Интерфейс программы (вкладка «Исходные данные»)

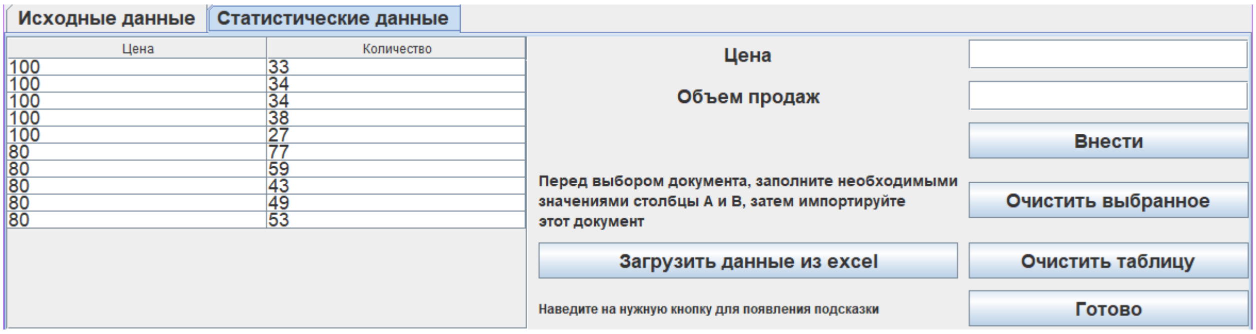


Рисунок В.2 – Интерфейс программы (вкладка «Статические данные»)

Выходной документ содержит 3 листа: «Обратная задача», «Решение» и «Предиктивный интервал». На лист «Обратная задача» (рисунок В.3) отображается результат решения обратной задачи (шаг 1 алгоритма, п.7.1.2), при этом величины цены и количества округляются таким образом, чтобы полученное значение прибыли имело минимальное отклонение от установленного значения. В ячейки A12, B12 программа заносит новые значения цены и количества. Эти величины необходимо округлить, поэтому в ячейках A16:D19 определены варианты округления величин и выбран тот, для которого абсолютное отклонение от заданного значения будет минимальным (для значений на рисунке И.3 минимальное отклонение расположено в ячейке D18). Автоматически формируемые формулы:

[A10]=A6/B6*B10

[B10] =ЕСЛИ(И(A8="+";B8="+");МИН(ABS(E9);ABS(F9));ЕСЛИ(И(A8="+";B8="-");МИН(ABS(E11);ABS(F11));ЕСЛИ(И(A8="-";B8="+");МИН(ABS(E13);ABS(F13));МИН(ABS(E15);ABS(F15)))))

[A12]=A3+A10

[B12]=B3+B10

[A16]=ОКРУГЛВВЕРХ(A12;0)

[B16]=ОКРУГЛВВЕРХ(B12;0)

[C16]=(A16-$E$3)*B16

[D16]=ABS(C16-$C$12)

	A	B	C	D	E	F	G	H	L
1									
2	Цена	Количество		Желаемая	Себестоимость		Прибыль=количество*(цена-себестоимость)		
3	60	98		5000	19				
4	Коэффициенты важности						4018		
5	α	β							
6	0,5	0,5							
7	Направление изменения показателей								
8	+	+			Решение квадратного уравнения для увеличения всех показателей				Если дискриминант отрицательный то решения не будет
9	dx1	dx2			6,738114	-145,738		Из двух решений нужно выбрать положительное и минимальное по модулю (таким же образом и далее)	
10	6,738114	6,738114	Значения изменени		Решение квадратного уравнения при увеличении цены и уменьшении количества				
11	Новая цен	Новое кол	Новая прибыль		#ЧИСЛО!	#ЧИСЛО!			
12	66,73811	104,7381	5000		Решение квадратного уравнения при уменьшении цены и увеличении количества				
13	Искомые значения				#ЧИСЛО!	#ЧИСЛО!			
14	Округление - выбирается результат, наиб				Решение квадратного уравнения при уменьшениивсех показателей				
15	Цена	Количеств	Прибыль	Отклонен	145,7381	-6,73811			
16	67	105	5040	40					
17	66	104	4888	112					
18	67	104	4992	8	Минимальное отклонение				
19	66	105	4935	65					
20									
21									
22									
23									

Рисунок В.3 – Лист «Обратная задача»

Лист «Решение» (рисунок В.4) включает таблицу со значениями цены, изменяемой с заданным шагом, количества, величин изменения показателей, границ интервалов, показателей расположения относительно границ предиктивного интервала и начального решения, нормированные показателей расположения относительно границ предиктивного интервала и начального решения, интегрального показателя. Цена, соответствующая набольшему значению целевой функции, отмечается в качестве решения задачи (на рисунке В.5 ячейка В17).

Формируемые программой на листе формулы расчёта:

[D5]=B5-$B$5

[E5]=C5-$C$5

[H5]=D5^2+E5^2

[I5]=-(G5-C5)*(C5-F5)/(G5-F5)

[J5]=(МАКС($H$5:$H$44)-H5)/(МАКС($H$5:$H$44)-МИН($H$5:$H$44))

[K5]=(МАКС($I$5:$I$44)-I5)/(МАКС($I$5:$I$44)-МИН($I$5:$I$44))

[L5]=$L$3*J5+$M$3*K5

	A	B	C	D	E	F	G	H	I	J	K	L	M
1		Прибыль, руб.	5000			Максимальная цена, руб.	106					Коэффициенты важности	
2												k1	k2
3		Шаг, руб.	1			Себестоимость, руб.	19					0,5	0,5
4		Цена (x1), руб.	Количество (x2)	Изменение цены (dx1)	Изменение количества (dx2)	Нижняя граница	Верхняя граница	Удаленность от начального решения	Расположение относительно предиктивного интервала	Удаленность от начального решения, норм.	Расположение относительно предиктивного интервала, норм.	Интегральный показатель	
5		67,00	104,00	0	0,00	42,851	99,449	0,000	4,917	1,000	0,196	0,598	
6		68,00	102,00	1	-2,00	42,094	97,906	5,000	4,395	0,999	0,308	0,653	
7		69,00	100,00	2	-4,00	41,326	96,374	20,000	3,865	0,995	0,422	0,708	
8		70,00	98,00	3	-6,00	40,544	94,856	45,000	3,326	0,988	0,538	0,763	
9		71,00	96,00	4	-8,00	39,750	93,350	80,000	2,781	0,979	0,655	0,817	
10		72,00	94,00	5	-10,00	38,941	91,859	125,000	2,228	0,966	0,774	0,870	
11		73,00	93,00	6	-11,00	38,118	90,382	157,000	2,750	0,958	0,662	0,810	
12		74,00	91,00	7	-13,00	37,281	88,919	218,000	2,165	0,942	0,788	0,865	
13		75,00	89,00	8	-15,00	36,428	87,472	289,000	1,573	0,923	0,915	0,919	
14		76,00	88,00	9	-16,00	35,558	86,042	337,000	2,034	0,910	0,816	0,863	
15		77,00	86,00	10	-18,00	34,673	84,627	424,000	1,410	0,886	0,950	0,918	
16		78,00	85,00	11	-19,00	33,770	83,230	482,000	1,833	0,871	0,859	0,865	
17		79	83,00	12	-21,00	32,850	81,850	585,000	1,177	0,843	1,000	0,922	Максимум
18		80,00	82,00	13	-22,00	31,911	80,489	653,000	1,558	0,825	0,918	0,871	
19		81,00	81,00	14	-23,00	30,954	79,146	725,000	1,926	0,806	0,839	0,822	
20		82,00	79,00	15	-25,00	29,978	77,822	850,000	1,207	0,772	0,993	0,883	
21		83,00	78,00	16	-26,00	28,983	76,517	932,000	1,529	0,750	0,924	0,837	
22		84,00	77,00	17	-27,00	27,968	75,232	1018,000	1,834	0,727	0,859	0,793	
23		85,00	76,00	18	-28,00	26,933	73,967	1108,000	2,120	0,703	0,797	0,750	
24		86,00	75,00	19	-29,00	25,877	72,723	1202,000	2,387	0,678	0,740	0,709	
25		87,00	74,00	20	-30,00	24,800	71,500	1300,000	2,634	0,651	0,687	0,669	
26		88,00	72,00	21	-32,00	23,703	70,297	1465,000	1,765	0,607	0,873	0,740	

Рисунок В.4 – Лист «Решение»

Последний лист «Предиктивный интервал» (рисунок В.5) является вспомогательным и предназначен для вычисления границ предиктивного интервала при заданном значении объясняющей переменной (цены).

Формулы расчёта ячеек:

[H7]=СРЗНАЧ(B4:B13)

[H9]=ПРЕДСКАЗ(H8;C4:C13;B4:B13)

[H11]=КВАДРОТКЛ(B4:B13)

[H12]=H10*КОРЕНЬ(1+1/H5+(H8-H7)^2/H11)

[H14]=H9-H13*H12

[H15]=H9+H13*H12

	A	B	C	D	E	F	G	H	I	J	K
1											
2		Исходные данные									
3		Цена (x)	Объем продаж (y)								
4		100	33								
5		100	34				n	10			
6		100	34				df	8	Число эл-ов - 2		= n - 2
7		100	38				mean(x)	90,00	Среднее значение x		= AVERAGE(x)
8		100	27				x0	106	Задаваемая цена		
9		80	77				ŷ0	26,3	Определение прогн		= FORECAST(y,x,x0)
10		80	59				sRes	9,61509	Вычисление стандар		= STEYX(y,x)
11		80	43				SSx	1000	Сумма квадратов от		= DEVSQ(x)
12		80	49				se	11,19653			= sRes*SQRT(1/n+(x0-x̄)^2/SSx)
13		80	53				t-crit	2,30600	Хранение таблицы :		= TINV(0.05,df)
14							lower	0,48076	Нижняя граница инт		= ŷ0 - t-crit * se
15							upper	52,11924	Верхняя граница ин		= ŷ0 + t-crit * se
16											

Рисунок В.5 – Лист «Предиктивный интервал»

Приложение Г Показатели оценки деятельности региона

Название показателя	Единица измерения
1. Уровень жизни	
1) коэффициент естественного прироста населения	промилле
2) ожидаемая продолжительность жизни при рождении	лет
3) уровень занятости	процент
4) уровень безработицы	процент
5) соотношение среднедушевых денежных доходов с величиной прожиточного минимума	рубль
6) среднедушевые денежные доходы населения	рубль
7) численность населения с денежными доходами ниже величины прожиточного минимума	процент
8) общая площадь жилых помещений, приходящаяся в среднем на одного жителя	квадратный метр
9) удельный вес расходов домашних хозяйств на оплату жилищно-коммунальных услуг	процент
10) индекс потребительских цен	процент
2. Финансовая обеспеченность	
1) соотношение поступлений и расходований в бюджете пенсионного фонда	коэффициент
2) соотношение поступлений и расходований в бюджете фонда социального страхования	коэффициент
3) соотношение поступлений и расходований в бюджете территориального фонда обязательного медицинского страхования	коэффициент
4) удельный вес прибыльных организаций	процент
5) скорректированный показатель кредиторской задолженности организаций	процент
6) скорректированный показатель дебиторской задолженности организаций	процент
3. Эффективность сельскохозяйственного производства	
1) производительность труда	процент
2) фондоотдача	млн. рублей
3) стоимость основных фондов полная учетная	млн. рублей
4) удельный вес прибыльных организаций	процент
5) индексы производства продукции сельского хозяйства	процент
4. Эффективность строительства	
1) производительность труда	процент
2) фондоотдача	млн. рублей
3) удельный вес прибыльных организаций	процент
4) ввод в действие жилых домов на 1000 человек населения	квадратный метр/человек

Название показателя	Единица измерения
5) ввод в действие зданий жилого и нежилого назначения на 1000 человек населения	квадратный метр/человек
5. Обеспеченность трудовыми ресурсами	
1) население в трудоспособном возрасте	тыс. человек
2) скорректированный коэффициент демографической нагрузки	коэффициент
3) уровень экономической активности населения	процент
4) выпуск на 10 000 человек населения квалифицированных рабочих	тыс. человек
5) выпуск на 10 000 человек населения специалистов со средним профессиональным образованием	тыс. человек
6) выпуск на 10 000 человек населения специалистов с высшим профессиональным образованием	тыс. человек
6. Состояние системы здравоохранения	
1) число больничных коек на 10 000 человек населения	единица
2) мощность амбулаторно-поликлинических учреждений на 10 000 человек населения	посещений в смену
3) численность врачей на 10 000 человек населения	человек
4) численность среднего медицинского персонала на 10 000 человек населения	человек
5) заболеваемость на 1000 человек населения	человек
7. Обеспеченность объектами образования	
1) обеспеченность детей дошкольного возраста местами в дошкольных образовательных учреждениях	единица
2) численность детей, стоящих на учете для определения в дошкольные образовательные учреждения	человек
3) численность обучающихся государственных и муниципальных общеобразовательных учреждений (без вечерних (сменных)) общеобразовательных учреждений	тыс. человек
4) удельный вес обучающихся в государственных и муниципальных общеобразовательных учреждений (без вечерних (сменных)), занимающихся во вторую и третью смены	процент
5) численность студентов, обучающихся по программам бакалавриата, специалитета, магистратуры на 10000 человек населения	тыс. человек
8. Обеспеченность ИиКТ	
1) удельный вес организаций, использовавших персональные компьютеры, % от общего числа обследованных организаций соответствующего субъекта РФ (без субъектов малого предпринимательства)	процент

Название показателя	Единица измерения
2) удельный вес организаций использовавших Интернет, % от общего числа обследованных организаций соответствующего субъекта РФ (без субъектов малого предпринимательства)	процент
3) удельный вес организаций имевших веб-сайт, % от общего числа обследованных организаций соответствующего субъекта РФ (без субъектов малого предпринимательства)	процент
4) число ПК на 100 работников (без субъектов малого предпринимательства)	штук
5) число персональных компьютеров с доступом к сети Интернет на 100 работников (без субъектов малого предпринимательства)	штук
6) наличие персональных компьютеров в домашних хозяйствах, на 100 домохозяйств	штук

Приложение Д Программа оценки и прогнозирования уровня социально-экономического развития регионов Сибирского федерального округа

Для разработки программы была выбрана система программирования Delphi версии 7.

На рисунке Д.1 представлено главное меню информационной системы, которое имеет вкладки: «Ввод данных», «Рейтинг», «Прогноз», «Графики» и «Выход».

Рисунок Д.1 – Главное меню

По центру расположены названия всех субъектов СФО. При нажатии на них можно увидеть краткое описание каждого региона (рисунок Д.2).

Рисунок Д.2 – Краткое описание Томской области

Вкладка «Ввод данных» предназначена для редактирования исходных данных (рисунок Д.3, Д.4).

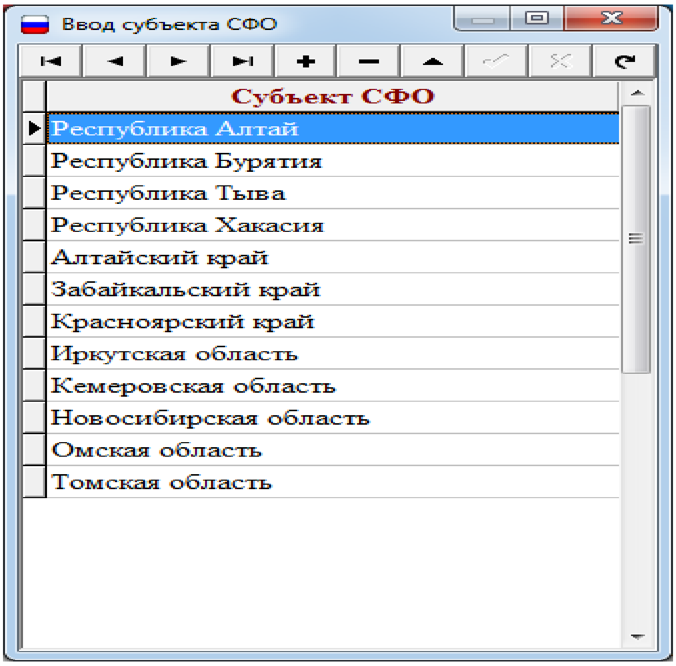
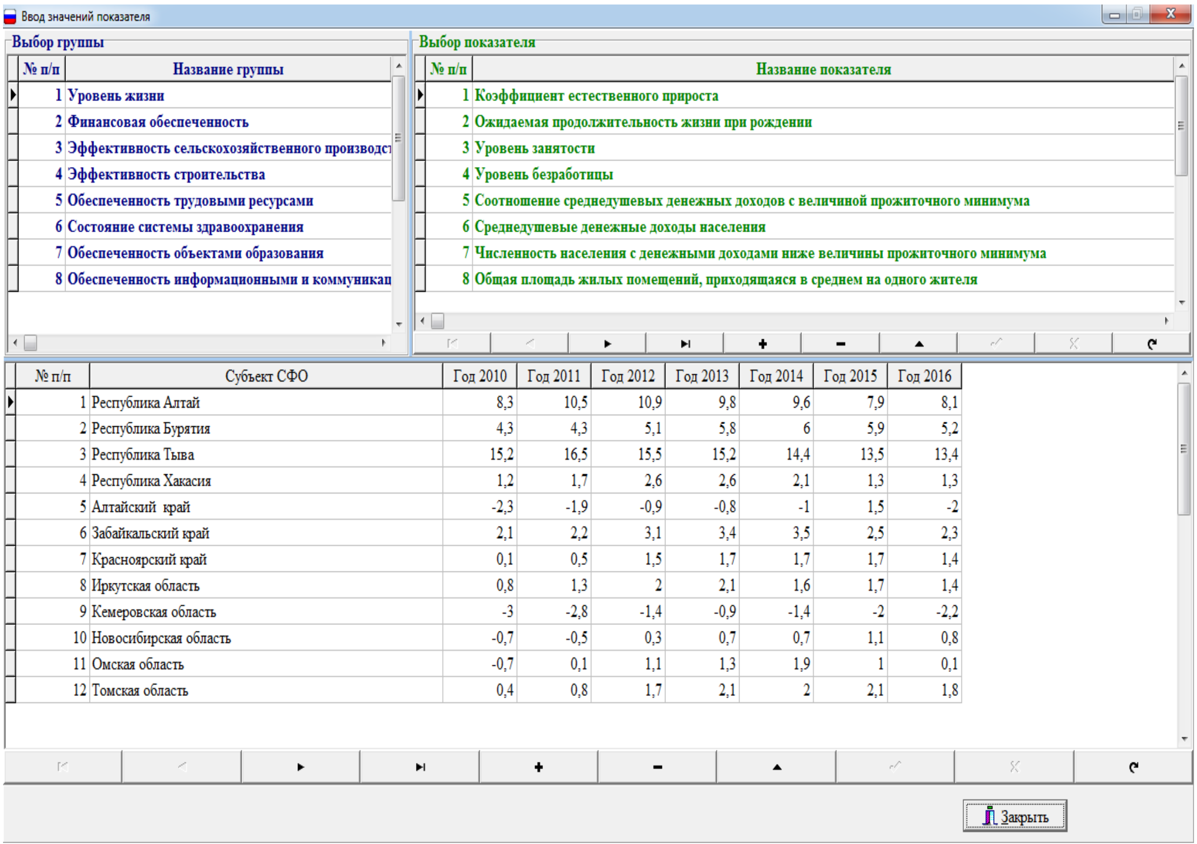


Рисунок Д.3 – Форма «Субъекты»



Ввод значений показателя

Выбор группы

№ п/п	Название группы
1	Уровень жизни
2	Финансовая обеспеченность
3	Эффективность сельскохозяйственного производст
4	Эффективность строительства
5	Обеспеченность трудовыми ресурсами
6	Состояние системы здравоохранения
7	Обеспеченность объектами образования
8	Обеспеченность информационными и коммуникац

Выбор показателя

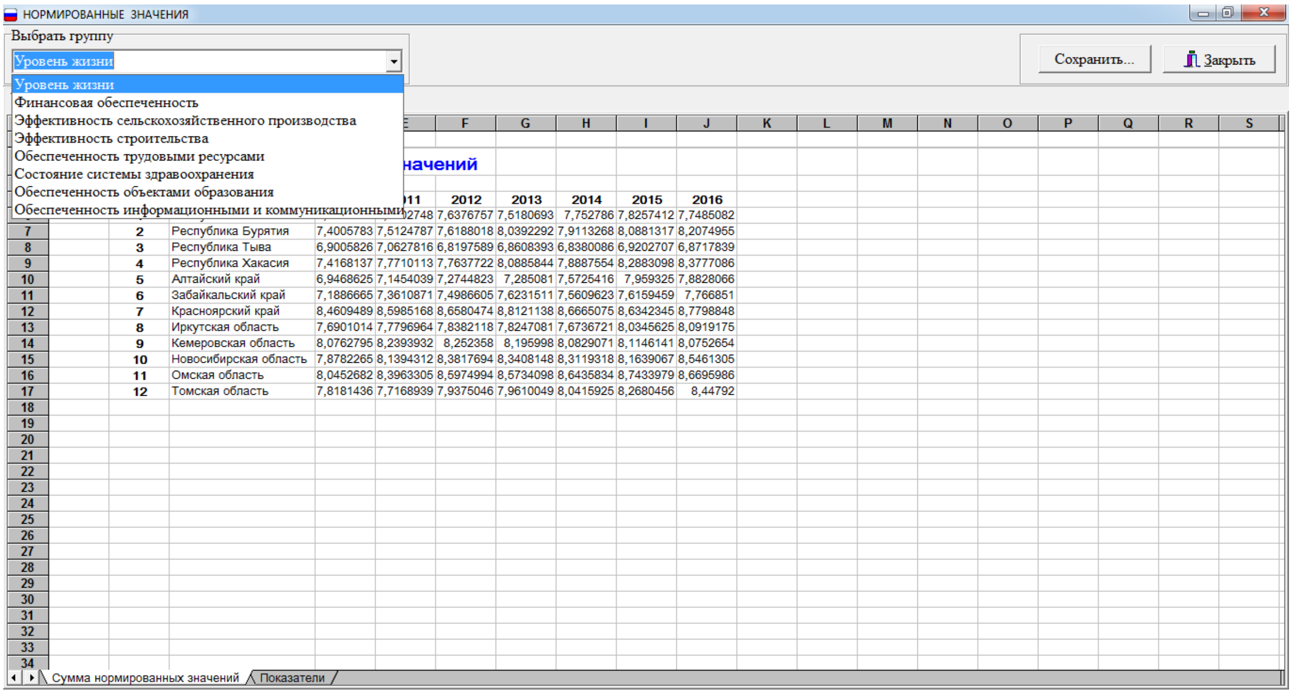
№ п/п	Название показателя
1	Коэффициент естественного прироста
2	Ожидаемая продолжительность жизни при рождении
3	Уровень занятости
4	Уровень безработицы
5	Соотношение среднедушевых денежных доходов с величиной прожиточного минимума
6	Среднедушевые денежные доходы населения
7	Численность населения с денежными доходами ниже величины прожиточного минимума
8	Общая площадь жилых помещений, приходящаяся в среднем на одного жителя

№ п/п	Субъект СФО	Год 2010	Год 2011	Год 2012	Год 2013	Год 2014	Год 2015	Год 2016
1	Республика Алтай	8,3	10,5	10,9	9,8	9,6	7,9	8,1
2	Республика Бурятия	4,3	4,3	5,1	5,8	6	5,9	5,2
3	Республика Тыва	15,2	16,5	15,5	15,2	14,4	13,5	13,4
4	Республика Хакасия	1,2	1,7	2,6	2,6	2,1	1,3	1,3
5	Алтайский край	-2,3	-1,9	-0,9	-0,8	-1	1,5	-2
6	Забайкальский край	2,1	2,2	3,1	3,4	3,5	2,5	2,3
7	Красноярский край	0,1	0,5	1,5	1,7	1,7	1,7	1,4
8	Иркутская область	0,8	1,3	2	2,1	1,6	1,7	1,4
9	Кемеровская область	-3	-2,8	-1,4	-0,9	-1,4	-2	-2,2
10	Новосибирская область	-0,7	-0,5	0,3	0,7	0,7	1,1	0,8
11	Омская область	-0,7	0,1	1,1	1,3	1,9	1	0,1
12	Томская область	0,4	0,8	1,7	2,1	2	2,1	1,8

Закрыть

Рисунок Д.4 – Форма редактирования значений показателей

С помощью вкладки «Рейтинг» осуществляется расчет нормированных значений показателей, а также формирование рейтинга (рисунок Д.5).



		...11	2012	2013	2014	2015	2016
		...02748	7,6376757	7,5180693	7,752786	7,8257412	7,7485082
2	Республика Бурятия	7,4005783 / 7,5124787	7,6188018	8,0392292	7,9113268	8,0881317	8,2074955
3	Республика Тыва	6,9005826 / 7,0627816	6,8197589	6,8608393	6,8380086	6,9202707	6,8717839
4	Республика Хакасия	7,4168137 / 7,7710113	7,7637722	8,0885844	7,8887554	8,2883098	8,3777086
5	Алтайский край	6,9468625 / 7,1454039	7,2744823	7,285081	7,5725416	7,959325	7,8828066
6	Забайкальский край	7,1886665 / 7,3610871	7,4986605	7,6231511	7,5609623	7,6159459	7,766851
7	Красноярский край	8,4609489 / 8,5985168	8,6580474	8,8121138	8,6665075	8,6342345	8,7798848
8	Иркутская область	7,6901014 / 7,7796964	7,8382118	7,8247081	7,6736721	8,0345625	8,0919175
9	Кемеровская область	8,0762795 / 8,2393932	8,252358	8,195998	8,0829071	8,1146141	8,0752654
10	Новосибирская область	7,8782265 / 8,1394312	8,3817694	8,3408148	8,3119318	8,1639067	8,5461305
11	Омская область	8,0452682 / 8,3963305	8,5974994	8,5734098	8,6435834	8,7433979	8,6695986
12	Томская область	7,8181436 / 7,7168939	7,9375046	7,9610049	8,0415925	8,2680456	8,44792

Рисунок Д.5 – Значения показателей групп

Во вкладке «Прогноз» рассчитываются прогнозные значения интегральных показателей и групп показателей.

Приложение Е Программы анализа данных онлайновой социальной сети ВКонтакте

Программа оценки групп социальной сети для реализации маркетинговых мероприятий

Программа позволяет автоматически собрать информацию со стены сообщества социальной сети ВКонтакте (количество последних постов задается пользователем): число просмотров каждого сообщения, даты постов, число лайков, комментариев и репостов сообщения. На основе этой информации вычисляются средние значения: среднее число просмотров сообщения, среднее число сообщений в день, среднее число суммы лайков репостов и комментариев. После этого формируется рейтинг сообществ по данным критериям. Программа может быть использована для оценки динамики развития сообщества во времени, и соответственно эффективности работы администратора группы. Также программа может быть использована для сравнения нескольких сообществ между собой, например, для выбора группы для размещения рекламы или для планирования развития сообщества на основе показателей конкурентов. Использование программы позволяет автоматизировать сбор данных и сократить временные затраты.

Реализация программы выполнена на языке Java (среда разработки системы – Eclipse)

Для работы программы необходимо активное интернет-соединение. После запуска программы будет открыта форма (рисунок Е.1).

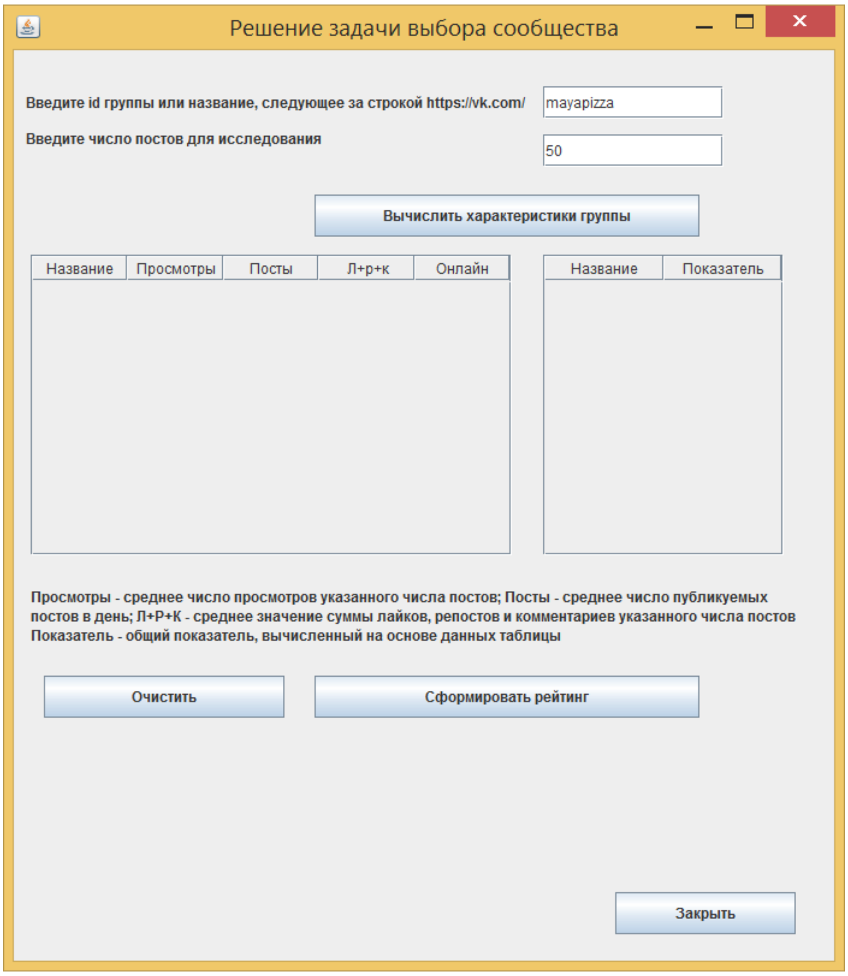


Рисунок Е.1 – Основная форма

Решение задачи выбора сообщества осуществляется путем заполнения двух полей:

– Id сообщества или его название, указанное в адресной строке после vk.com/.

– Число постов для исследования: количество последних постов на стене (начиная со 2-го), на основе которых будет вычислена статистика.

После заполнения этих двух полей необходимо нажать кнопку «Вычислить характеристики группы». В результате в таблице появится запись с вычисленными характеристиками (рисунок Е.2):

– Просмотры – среднее количество просмотров одного поста (сумма просмотров указанного числа постов, деленная на число постов).

– Посты – среднее число публикуемых постов в день.

– Л+р+к – среднее количество лайков, репостов и комментариев, приходящихся на один пост (для указанного числа постов определяется сумма лайков, репостов и комментариев и делится на число постов).

– Онлайн – число участников сообщества, находящихся в текущий момент статусе «онлайн».

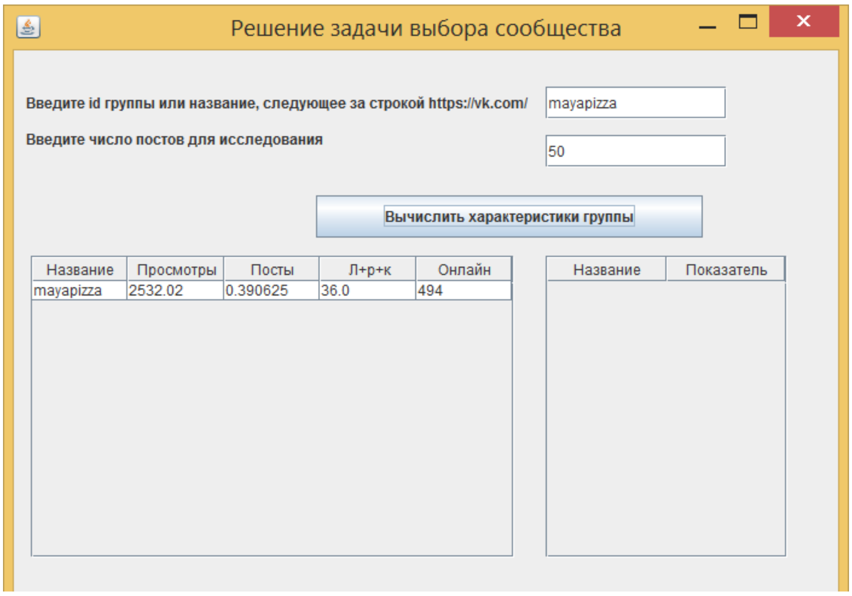


Рисунок Е.2 – Вычисленные характеристики

Аналогичным образом заполняется информация о других группах, которые рассматриваются в качестве потенциальной площадки для размещения рекламы: вводится id или название и нажимается кнопка «Вычислить характеристики группы».

Для удаления всех строк таблицы используется кнопка «Очистить».

После расчёта характеристик всех рассматриваемых групп необходимо нажать на кнопку «Сформировать рейтинг» (рисунок Е.3).

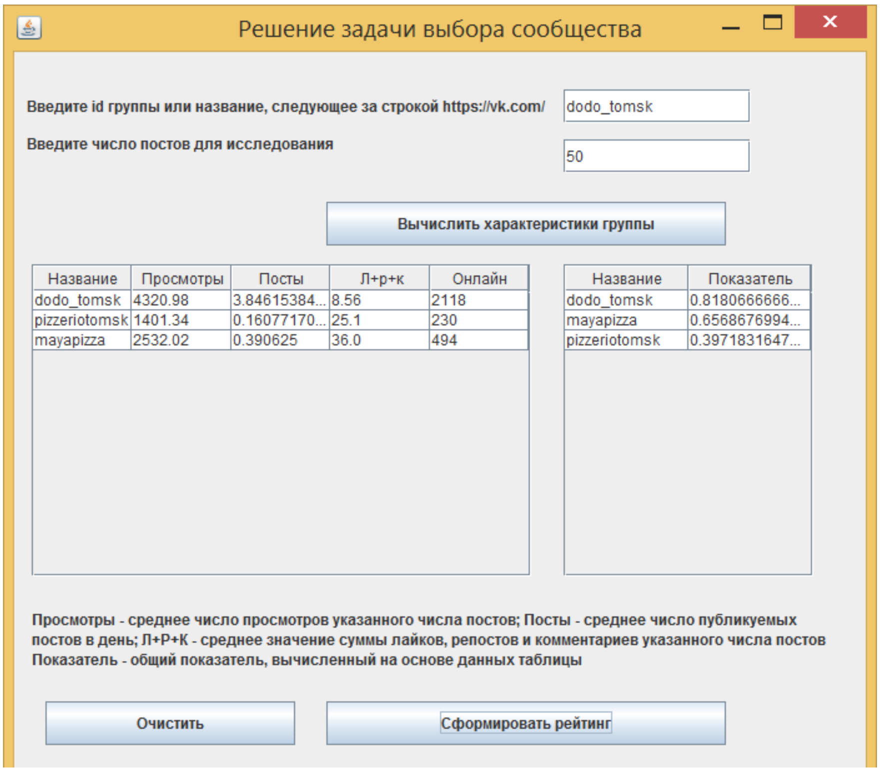


Рисунок Е.3 – Сформированый рейтинг

В результате в таблице справа будет сформирован рейтинг: вверху будет расположена наиболее привлекательная для размещения рекламы группа, в самом низу – наименее привлекательная. Рейтинг формируется на основе показателя, который вычисляется с помощью рассчитанных характеристик каждой группы (просмотры, посты, л+р+к).

Программа оценки времени размещения сообщений в группах онлайновой социальной сети ВКонтакте

Программа предназначена для определения наилучшего момента времени для размещения сообщения в сообществе социальной сети ВКонтакте. Наилучшим моментом времени считается момент, когда максимальное число участников группы находится в статусе «онлайн» и когда скорость обновления новостной ленты минимальна, т.к. в этом случае пользователь может увидеть сообщение в новостной ленте в более высокой позиции и сообщение не будет смещено другими постами. Программа позволяет: автоматически осуществлять сбор данных о статусах пользователей (каждые 5 минут, 10 минут…); обрабатывать данные и вычислять оценки моментов на основе статусов участников; формировать рейтинг моментов времени на основе оценок статусов участников и скорости обновления ленты.

Реализация программы выполнена на языке C# (среда разработки системы – Microsoft Visual Studio 2015), при этом были использованы стандартные методы API ВКонтакте: groups.getMembers, users.get, wall.get, newsfeed.get и execute. Результаты сбора и обработки данных хранятся в Excel документе.

Интерфейс системы оценки времени размещения сообщений в группах социальной сети ВКонтакте представлен на рисунке Е.4. В начале работы с программой пользователю требуется предоставить доступ к своим данным в ВКонтакте (рисунок Е.5), т.к. это необходимо для получения токена, который обеспечивает доступ к запросам VK API.

Для получения токена нужно вставить следующий адрес в браузере: https://oauth.vk.com/authorize?client_id=7121736&display=page&redirect_uri=https://oauth.vk.com/blank.html&scope=friends,status,wall,groups,offline&response_type=token&v=5.103.

После подтверждения предоставления доступа к данным (рисунок Д.5), открывается новая страница где написано «Пожалуйста, не копируйте данные из адресной строки для сторонних сайтов. Таким образом Вы можете потерять доступ к Вашему аккаунту.» Это ссылка выглядит следующем образом: https://oauth.vk.com/blank.html#access_token=XXXXXX&expires_in=0&user_id=146659440.

В этой ссылке, в части access_token, токеном будет XXXXXX. Этот токен нужно скопировать и вставить в систему в «Токен 1», а для «Токен 2» нужно вставить другой токен, проделав те же операции.

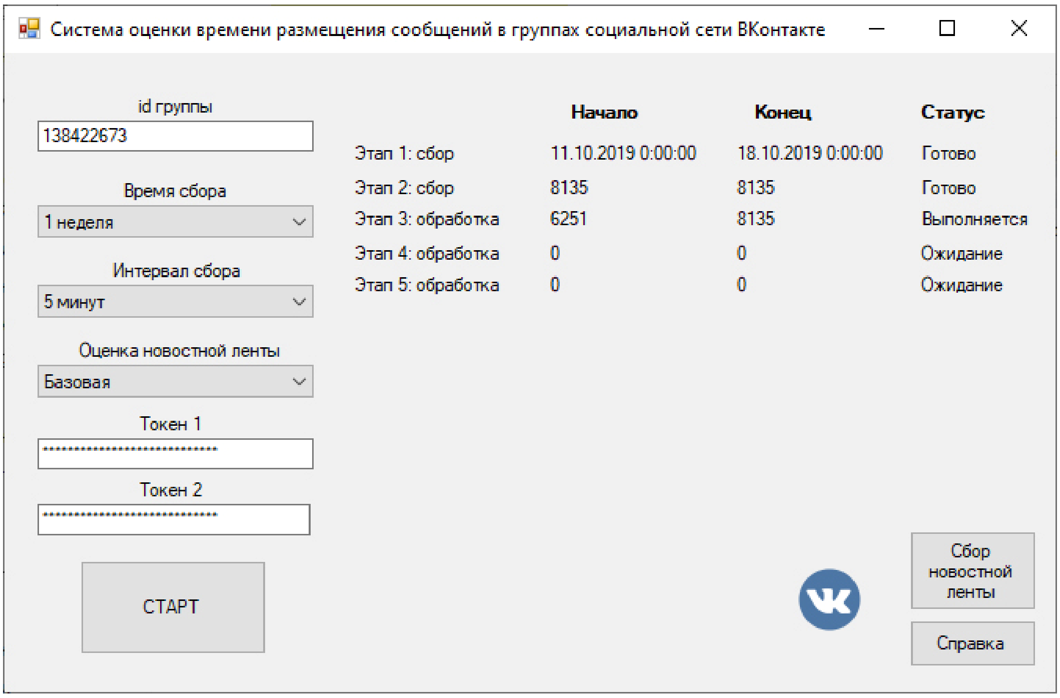


Рисунок Е.4 – Интерфейс системы

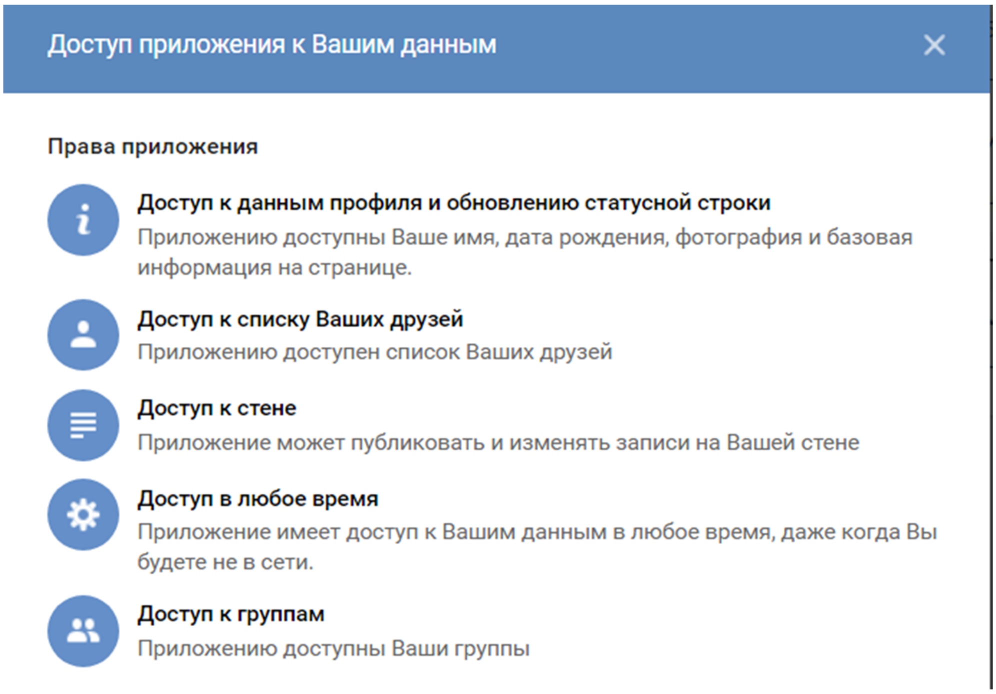


Рисунок Е.5 – Права доступа

После предоставлений прав доступа, заполняются входные данные, такие как: id группы, время сбора (от 1 дня до 2 недель), интервал сбора (от 5 минут до 1 часа), оценка новостной ленты (базовая или пользовательская) и два токена.

У пользователя есть два варианта выбора оценки новостной ленты: это «базовая» или «пользовательская» (рисунок Е.6). При выборе «пользовательской» оценки пользователю необходимо загрузить данные новостной ленты в систему, указав путь к файлу Excel, либо осуществить их сбор путем нажатия кнопки «Сбор новостной ленты» (рисунок Е.7). При выборе «базовой» оценки используются данные, заложенные в системе, которые были получены на основе сбора данных 300 групп в течение недели.

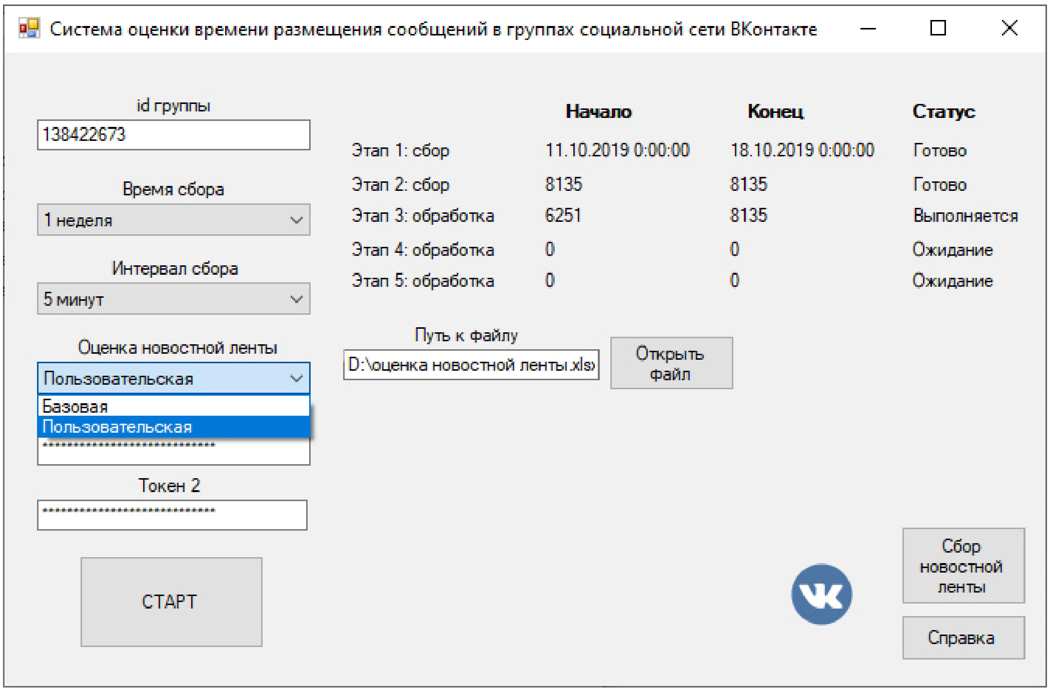


Рисунок Е.6 – Выбор оценки новостной ленты

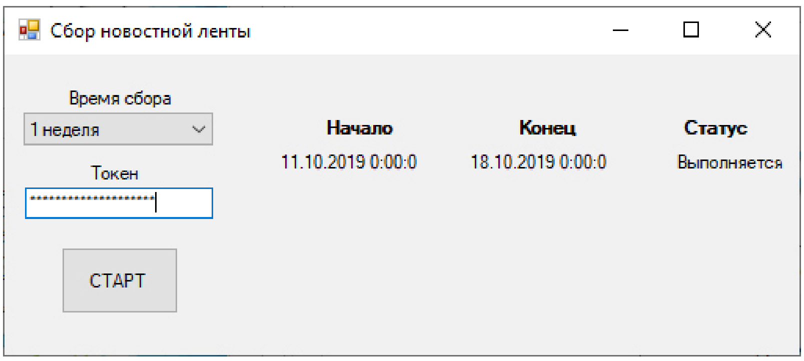


Рисунок Е.7 – Интерфейс формы «Сбор новостной ленты»

Процесс работы с программой для формирования интегральной оценки включает следующие шаги:

Этап 1. Сбор данных о статусе пользователей с заданным интервалом (программа автоматически собирает сведения: id участника и статус).

Этап 2. Сбор данных о характеристиках участников.

Данные этапа 1 и 2 выгружаются в Excel, и в последующих трёх этапах выполняется их обработка.

Этап 3. Расчет характеристик индивидуальных периодов активности пользователей (на основе данных этапа 1).

Этап 4. Вычисление суммарной оценки на основе характеристик пользователей (с использованием данных этапа 2 и 3).

Этап 5. Вычисление интегральной оценки момента размещения сообщения (на основе данных этапа 4).

На рисунках Е.8–Е.12 представлены фрагменты выгруженных в файлы Excel данных, полученных в результате выполнения каждого этапа.

11.10.2019								
0:00		0:05		0:10			23:55	
Id	Онлайн	Id	Онлайн	Id	Онлайн	...	Id	Онлайн
62845	1	62845	1	62845	1	...	62845	0
67654	0	67654	1	67654	0	...	67654	0
87980	1	87980	1	87980	0	....	87980	1
101472	0	101472	0	101472	0	...	101472	0
103068	0	103068	0	103068	1	...	103068	1

Рисунок Е.8 – Результат обработки данных на первом этапе

Id	Друзья	Группы	Сумма лайков, репостов, комментариев
62845	416	112	0
67654	943	377	100
87980	235	737	1
101472	29	728	17
103068	141	103	96
123988	58	118	15
124398	516	2983	0

Рисунок Е.9 – Результат обработки данных на втором этапе

id	0:00	0:05	0:10	0:15	...	23:55
62845	0,051852	0,054688	0,057851	0,06034	...	0,056452
67654	0,083333	0,090909	0,0875	0,09459	...	0,084337
87980	0,006157	0,006195	0,006233	0,00627	...	0,006119
101472	0,094595	0,1	0,106061	0,10938	...	0,088608
103068	0,011925	0,012069	0,012216	0,01217	...	0,011785

Рисунок Е.10 – Результат обработки данных на третьем этапе

	суммарная оценка
0:00	762,5811272
0:05	762,6060303
0:10	757,6423215
...	...
23:50	771,0930691
23:55	785,7505141

Рисунок Е.11 – Результат обработки данных на четвертом этапе

№	Максимальная оценка интегрального показателя	Время
1	0,794286	20:35
2	0,793982	20:40
3	0,791246	20:45
4	0,789898	21:40
5	0,789467	21:20
6	0,788878	20:30
7	0,788682	21:45
8	0,788286	20:20
9	0,786766	22:20
10	0,786570	20:25
...	...	...
286	0,392620	4:55
287	0,389850	3:55
288	0,389312	3:50

Рисунок Е.12 – Результат обработки данных на пятом этапе

Приложение Ж Акты внедрения, свидетельства о регистрации программ

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
"ТОМСКАЯ НЕФТЬ"

УТВЕРЖДАЮ
Заместитель генерального директора
по экономике и финансам
Казакова С.В.
«15» июня 2022 г.

АКТ О ВНЕДРЕНИИ

Комиссия в составе:
председателя – Перетягина Н.В., начальник финансового отдела

членов комиссии:
Борзунов В.А., заместитель начальника планово-бюджетного отдела
Козлов С.В., начальник отдела сопровождения бухгалтерских и финансовых систем

составила настоящий Акт о нижеследующем.

В 2022 году было выполнено внедрение программы решения иерархической многономенклатурной обратной задачи формирования прибыли предприятия, разработанной Грибановой Екатериной Борисовной, для информационной поддержки принятия решений.

В программе реализованы три алгоритма формирования показателя, особенностью которых является возможность решения иерархических задач с ограничениями. Отличие алгоритмов заключается в способе достижения результата, который может основываться на использовании экспертной информации либо минимизации отклонения от исходных значений. Программа позволяет осуществлять работу в двух режимах: поэтапном и одношаговом. Результатом работы программы являются значения цены, себестоимости, объем продаж, прибыли по наименованию для достижения заданного значения суммарной прибыли. Программа представляет возможность загрузки и сохранения результатов моделирования в файл Excel.

Автоматизация расчётов позволяет сократить временные ресурсы (на 25%) и оперативно получить информацию для принятия решений управленческих решений.

Председатель комиссии: _______________ / Перетягина Н.В./

Члены комиссии: _______________ / Борзунов В.А./

_______________ / Козлов С.В. /

Общество с ограниченной ответственностью ОГРН 1117017010945 РуссНефть "ТОМСКАЯ НЕФТЬ" ИНН 7017287178 Томск Россия

УТВЕРЖДАЮ
директор АО «Разрез «Степановский»
___________ Камышанов М.Д.
«   » ________ 2022 г.

АКТ О ВНЕДРЕНИИ

Комиссия в составе: председатель Главный экономист Баранова Л.В.; члены комиссии: Начальник планово-экономического отдела Пронина И.А., Ведущий экономист Вытенкова А.А. составили настоящий акт, подтверждающий факт экспериментального внедрения в АО «Разрез «Степановский» программы решения иерархической многономенклатурной обратной задачи формирования прибыли предприятия, разработанной Грибановой Екатериной Борисовной, доцентом кафедры Автоматизированных систем управления Томского государственного университета систем управления и радиоэлектроники.

Программа осуществляет расчёт прибыли по наименованиям, цены, себестоимости, объема продаж для достижения заданного значения суммарной прибыли. Возможны два способа решения задачи: одношаговый и двухшаговый расчёт. При двухшаговом расчёте на первом этапе вычисляются значения прибыли по наименованиям, на втором – цена, себестоимость и объем продаж по каждому наименованию. В одношаговом варианте вычисление прибыли по наименованиям не происходит. Нахождение решения может быть выполнено при использовании экспертной информации (коэффициенты относительной важности и направления изменения показателей) либо при минимизации отклонений от исходных значений.

Использование программы обеспечивает применение новых методов для оперативного принятия решений и сокращение рабочего времени на проведение расчётов на 25%.

Председатель комиссии:
_____________________/Баранова Л.В./

Члены комиссии:
_____________________/Пронина И.А./
_____________________/Вытенкова А.А./

Общество с ограниченной
ответственностью «Вокифудтомск»
ИНН 7017361777 КПП 701701001
ОГРН 1147017018796
634041, г. Томск, ул. Белинского, 50, 14

УТВЕРЖДАЮ
Генеральный директор
ООО «Вокифудтомск»
Малащук В.В.
2021 г.

АКТ О ВНЕДРЕНИИ

Настоящий акт подтверждает факт внедрения в ООО «Вокифудтомск» программы формирования интегрального показателя. Данная программа была адаптирована под задачу организации для выбора размещения новых торговых точек по продаже кофе. Программа позволяет вычислять интегральный показатель объектов (торговых точек) на основе совокупности характеристик расположения (район, наличие поблизости парка, расстояние до центра и т.д.) и статистических данных о действующих пунктах продажи. Также программа выполняет решение обратной задачи по определению набора характеристик для формирования интегрального показателя.

Использование программы позволяет получить полезную информацию для принятия решения относительно размещения торговых точек (о влиянии характеристик расположения на выручку, об интегральном показателе каждой точки, о необходимом изменении характеристик) и обеспечивает значительное сокращение времени на обработку и анализ данных (повышение производительности труда более 300%).

Генеральный директор
ООО «Вокифудтомск»
Малащук В.В.

Исполнитель:
доцент каф. АСУ
Грибанова Е.Б.

Общество с ограниченной ответственностью «Титан»
ИНН 7017381237 КПП 701701001
ОГРН 1157017012680
634027, г. Томск, ул. Мостовая, д. 20, офис 301

УТВЕРЖДАЮ
Директор ООО «Титан»
Козловский Сергей Андреевич
20.03.2021 г.

Акт внедрения

результатов диссертационного исследования Грибановой Екатерины Борисовны

Настоящий акт подтверждает факт использования результатов диссертационной работы Грибановой Екатерины Борисовны в ООО «Титан», а именно моделей формирования и прогнозирования выручки, программного обеспечения «Анализ выручки». Модель прогнозирования выручки основана на регрессионной модели и позволяет выполнять краткосрочный прогноз в зависимости от будних и праздничных дней на основе статистических данных, а также путем решения обратной задачи определять моменты возникновения возможного дефицита для планирования движения денежных средств. Программа автоматизирует расчёт параметров модели, показателей её качества (индекс детерминации, ошибка аппроксимации) и прогнозных значений, скрывает от пользователя сложные расчёты и не требует от пользователя знаний в области регрессионного анализа. Применение программы позволило выявить модель с максимальной точностью решения, обеспечивающую снижение ошибки предсказания на 55% по сравнению с использованием среднего значения. Использование данных результатов обеспечивает информацией для принятия решений относительно планирования финансово-хозяйственной деятельности. Применение программы позволило также получить экономический эффект в виде снижения времени, затрачиваемого на планирование, на 20%.

Директор ООО «Титан» Козловский С.А.

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ ОГРН 1157017012680 "ТИТАН" ИНН 7017381237 РОССИЯ ТОМСК

ООО «ФОРС»
ИНН: 5041205221
КПП: 504101001
ОГРН: 1175053001100
Юридический адрес: 143960, Московская обл, Реутов г, Фабричная ул, дом № 7, литера Е2, помещение 135
Телефон: +7(968)0515522

УТВЕРЖДАЮ
Генеральный директор
ООО «ФОРС»
Файзутдинов А.Ф.
«30» апреля 2021 г.

АКТ

Об использовании результатов диссертационной работы
Грибановой Екатерины Борисовны

Настоящим актом подтверждается, что результаты диссертационной работы Грибановой Е.Б. были использованы в ООО «ФОРС» как часть стратегии управления запасами газовых баллонов. Построение модели было обусловлено необходимостью снижения ситуаций дефицита и затоваривания склада с целью экономии оборотных средств. Модель управления запасами, представленная в работе Грибановой Е.Б., позволила решить обозначенные проблемы путем выполнения вычислительных экспериментов с целью определения оптимальных значений управляемых переменных (объем доставки, минимальный уровень запаса, дефицит). Это позволило построить улучшенную стратегию управления запасами и снизить затоваривание склада. Применение полученных результатов позволяет сократить размер запаса на складе в среднем на 40% по сравнению с используемой ранее стратегией управления и таким образом сэкономить оборотные средства предприятия.

Генеральный директор
ООО «ФОРС»
_______________ Файзутдинов А.Ф.

РОССИЙСКАЯ ФЕДЕРАЦИЯ • МОСКОВСКАЯ ОБЛАСТЬ • г. Реутов
ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ ОГРН 1175053001100
ФОРС
ИНН 5041205221

ООО «ИНТЕНС-СТРОЙ»
ИНН 7017388306 КПП 701701001
ОГРН 1157017019576
634059, Томская область, г. Томск, ул. Интернационалистов, д. 31, кв. 189

г. Томск 18.03.2021 г.

Справка

о внедрении результатов диссертационной работы

Грибановой Екатерины Борисовны

Результаты диссертационной работы Грибановой Екатерины Борисовны применяются в практической деятельности ООО «Интенс-строй», в частности используется разработанный стохастический алгоритм решения обратных задач с ограничениями, а также его реализация в виде компьютерного приложения «Программа решения обратной задачи формирования прибыли с помощью стохастических алгоритмов». Программа предназначена для вычисления себестоимости, объема выпуска и цены продукции для достижения заданного значения прибыли с учетом экспертной информации, в качестве которой выступают коэффициенты относительной важности показателей и ограничения на значения величин, включая: определение искомых показателей путем генерации случайных чисел из допустимого диапазона; расчёт искомых показателей с использованием стохастического алгоритма на основе итерационного изменения прибыли с заданным шагом.

Программа помогает формировать обоснованную стратегию финансово-хозяйственной деятельности с целью повышения финансовых результатов деятельности предприятия. Экономический эффект применения программы заключается в автоматизации расчётов и сокращении временных затрат на 30%.

Директор ООО «Интенс-строй» Жидков А.А.

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ "Интенс-Строй" ОГРН 1157017019576 ИНН 7017388306 ТОМСК РОССИЯ

УТВЕРЖДАЮ
Директор
ООО «Дельта»
_____________Важдаев А.Н.
«18» августа 2021 г.

Акт о внедрении

результатов диссертационной работы Грибановой Екатерины Борисовны

Комиссия в составе:
председателя Важдаева Андрея Николаевича, директора,
членов комиссии Думчева Дениса Игоревича, инженера по сопровождению баз данных,
Важдаевой Марии Владимировны, заместителя директора

рассмотрев результаты диссертационной работы Грибановой Е.Б. пришла к заключению:

В ООО «Дельта» внедрены следующие результаты диссертационной работы:

1. Модель оценки групп социальной сети для реализации маркетинговых мероприятий. Данная модель позволяет формировать интегральную оценку сообществ на основе ряда показателей, определяющих популярность группы среди пользователей социальной сети и частоту обновления информации в сообществе: среднее число просмотров записей; среднее число публикуемых сообщений в день; среднее число «лайков», «репостов» и комментариев одной записи. С помощью метода анализа иерархий были определены весовые коэффициенты данных характеристик. Применение модели позволяет формировать рейтинг сообществ, на основе которого может быть выполнен выбор наиболее подходящей группы для размещения рекламы с точки зрения большего охвата аудитории. В связи с этим данная модель является эффективным инструментом для поддержки принятия решений в области разработки плана маркетинговых мероприятий в социальной сети.
2. Программа для ЭВМ «Оценка групп социальной сети для реализации маркетинговых мероприятий» позволяет автоматизировать сбор данных сообществ социальной сети, а также расчёт характеристик и интегрального показателя, выполняемый на основе входной информации, в качестве которой выступают наименование сообщества и число исследуемых постов на стене сообщества. На основе интегрального показателя формируется рейтинг групп социальной сети. Программа имеет простой интерфейс и её использование не требует от пользователя выполнения трудоемких вычислений. Применение данной программы обеспечивает сокращение времени на обработку данных социальной сети более чем на 90%.

_____________ Важдаев А.Н.

_____________ Думчев Д.И.

_____________ Важдаева М.В.

УТВЕРЖДАЮ
заместитель директора по развитию
МАОУ ДО ДТДиМ
_______________ Еремина Е.Г.
«16» сентября 2021 г.

АКТ О ВНЕДРЕНИИ

результатов диссертационной работы Грибановой Екатерины Борисовны

Настоящий акт составлен о том, что в муниципальном автономном образовательном учреждении дополнительного образования Дворец творчества детей и молодежи г.Томска внедрены результаты диссертационной работы Грибановой Е.Б., а именно модель и программа оценки времени размещения сообщения в группах онлайновой социальной сети ВКонтакте. Публикация сообщений в сообществах социальной сети представляет собой эффективный инструмент распространения информации, в том числе рекламного характера. При этом одним из важнейших факторов, определяющих число просмотров сообщения, является время его размещения. Выбор момента размещения сообщения осуществляется путем анализа ряда показателей. Наилучшее время размещения сообщения определяется числом пользователей, находящихся в статусе «онлайн». Кроме того, при определении момента публикации следует учитывать и количество сообщений, размещаемых другими участниками социальной сети, которые могут смещать сообщение, делая его менее заметным. Разработанная Грибановой Е.Б. программа автоматизирует сбор и обработку данных, вычисляет характеристики, основанные на статусе участников и скорости обновления новостной ленты, рассчитывает обобщенный показатель и формирует рейтинг, на основе которого можно определить момент, когда размещение сообщения будет наиболее удачным с точки зрения возможности просмотра его участниками сообщества. Выгруженные данные и вычисленные показатели хранятся в файле Excel. Сбор и расчёт показателей с использованием программы позволяет сократить время на обработку данных социальной сети более чем на 70%.

Заместитель директора по развитию _______________ Еремина Е.Г.

<u>Общество с ограниченной ответственностью «Гамарджоба»</u>

ИНН 7017478430 КПП 701701001 ОГРН 1207000010975

Юридический адрес: г. Томск, ул. Гоголя д.63-10, эл. почта: d.utoyan@yandex.ru

Адрес осуществления предпринимательской деятельности: г.Томск, пр. Фрунзе, д.35

УТВЕРЖДАЮ

Директор ООО «Гамарджоба»

____________ Утоев Д.Р.

« 10 » марта 2021 г.

Акт о внедрении

результатов диссертационной работы Грибановой Екатерины Борисовны

Комиссия в составе:

Утоев Давид Рустамович, директор

Иванова Зина Морофовна, заведующая кафе «Гамарджоба»

составила настоящий акт, подтверждающий факт внедрения в ООО «Гамарджоба» результатов диссертационной работы Грибановой Е.Б.:

1. Оптимизационные модели формирования маржинальной прибыли, позволяющие определять изменения показателей для достижения заданного значения прибыли.
2. Программа для ЭВМ «Формирование маржинальной прибыли предприятия», позволяющая автоматизировать расчёты на основе оптимизационных моделей. В зависимости от выбранной модели решение задачи формирования прибыли может быть выполнено при минимальном суммарном изменении аргументов, при наличии стохастической зависимости объема продаж от цены, при использовании коэффициентов относительной важности показателей.

Разработанное математическое и программное обеспечение позволяет выполнять решение обратной задачи формирования прибыли и повышает обоснованность управленческих решений при организации деятельности в сфере общественного питания.

Члены комиссии:

____________________/Утоев Д.Р./

____________________/Иванова З.М./

Общество с ограниченной ответственностью “Сибмед”
(ООО “Сибмед”)

634040, г. Томск, Иркутский тракт,
дом 194, кв. 43
ИНН 7017408182, КПП 701701001

17.03.2021 № 5/н

Справка о внедрении
результатов диссертации на соискание учёной степени доктора технических наук Грибановой Екатерина Борисовны

Настоящая справка подтверждает факт внедрения в ООО «Сибмед» результатов диссертационной работы Грибановой Е.Б., а именно комплекса моделей, предназначенных для исследования вариантов формирования прибыли, и программы для ЭВМ «Формирование маржинальной прибыли предприятия».

В программном продукте реализованы три алгоритма решения обратной задачи формирования маржинальной прибыли. В результате работы программы происходит вычисление изменений аргументов (цена, объем продаж, себестоимость) для достижения заданного значения прибыли с учётом определяемых условий решения задачи.

Использование программы позволило сократить временные затраты на решение обратной задачи, а также исследовать различные стратегии достижения результата.

Директор Д.А. Азбукин

«17» марта 2021г.

Общество с ограниченной ответственностью ИНН 7017408182 «Сибмед» ОГРН 1167031068225 РОССИЯ • ТОМСК

УТВЕРЖДАЮ

Проректор по учебной работе ТУСУР

______________ П.В. Сенченко

«3» 09 2021

АКТ

о внедрении результатов диссертационного исследования

Грибановой Екатерины Борисовны в учебный процесс

Комиссия в составе:

Черкашиной И.П. (декан факультета дистанционного обучения), Романенко В.В. (зав. кафедрой АСУ, к.т.н.), Исаковой О.Ю. (начальник учебно-методического отдела факультета дистанционного обучения института инноватики) установила следующее.

Результаты диссертационной работы Грибановой Е.Б. используются студентами факультета дистанционного обучения направления подготовки 09.03.03 Прикладная информатика при изучении курсов «Эконометрика», «Исследование операций и методы оптимизации».

В вышеизложенных курсах использованы следующие результаты диссертационного исследования: метод решения обратных задач на основе статистических данных при стохастической зависимости между показателями, метод решения обратных задач с помощью формирования уравнения зависимости между аргументами функции, метод решения обратных задач при минимизации суммы квадратов изменений аргументов (задача нелинейного программирования).

Вид внедрения: учебно-методические комплексы дисциплин «Эконометрика», «Исследование операций и методы оптимизации», а именно учебные пособия и методические указания: Грибанова Е.Б. Эконометрика: учебное пособие. – Томск: факультет дистанционного обучения ТУСУРа, 2014. – 156 с.; Грибанова Е.Б. Эконометрика: методические указания по практическим и самостоятельным работам. – Томск: факультет дистанционного обучения ТУСУРа, 2014. – 57 с.; Грибанова Е.Б., Мицель А.А. Исследование операций и методы оптимизации: учебное пособие. – Томск: ФДО, ТУСУР, 2017. – 185 с.; Грибанова Е.Б. Исследование операций и методы оптимизации: методические указания по выполнению лабораторных работ. – Томск: ФДО, ТУСУР, 2017. – 110 с.

Члены комиссии: ______________ / И.П. Черкашина

______________ / В.В. Романенко

______________ / О.Ю. Исакова

УТВЕРЖДАЮ

Проректор по учебной работе ТУСУР

П.В. Сенченко

« 03 » 09 2021

Акт

внедрения в учебный процесс

результатов диссертационной работы Грибановой Екатерины Борисовны

Комиссия в составе:

председателя Романенко В.В. (зав. кафедрой АСУ, к.т.н.), членов комиссии Исаковой А.И. (методист каф. АСУ, к.т.н., доцент), Григорьевой М.В. (доцент каф. АСУ, руководитель ОПОП бакалавриата направления 09.03.03 Прикладная информатика, профиль Прикладная информатика в экономике, к.т.н.), Мицеля А.А. (профессор каф. АСУ, руководитель ОПОП магистратуры направления 09.04.01 Информатика и вычислительная техника, профиль Автоматизированные системы обработки информации и управления в экономике, д.т.н.)

составила настоящий акт о нижеследующем.

Результаты диссертационной работы Грибановой Е.Б. используются в учебном процессе кафедры Автоматизированных систем управления при чтении курсов «Технико-экономический анализ деятельности предприятий», «Эконометрика», «Исследование операций и методы оптимизации», «Математическое и имитационное моделирование экономических процессов» для подготовки бакалавров, обучающихся по направлению подготовки 09.03.03 Прикладная информатика. В частности, результаты исследований отражены в следующих учебных пособиях, используемых студентами при изучении теоретического материала и выполнении практических работ: Грибанова Е.Б. Технико-экономический анализ деятельности предприятия. – Томск: ТУСУР, 2016. – 105 с.; Грибанова Е.Б. Эконометрика. Практикум. – М.: Горячая линия – Телеком, 2019. – 148 с.; Грибанова Е.Б., Мицель А.А. Сборник задач по математическому и имитационному моделированию экономических процессов. – М.: Горячая линия – Телеком, 2019. – 252 с.; Грибанова Е.Б., Логвин И.Н. Имитационное моделирование экономических процессов. Практикум в Excel. Учебное пособие. – М.: Кнорус, 2020. – 228 с.; Грибанова Е.Б., Мицель А.А. Исследование операций и методы оптимизации: учебное пособие. – Томск: ТУСУР, 2017. – 185 с.

На основе предложенных в диссертационной работе моделей, методов, алгоритмов осуществлялась постановка задач для научно-исследовательской работы студентов, выпускных квалификационных работ бакалавров и магистрантов. Всего было защищено 14 бакалаврских работ и 10 магистерских диссертаций. Результаты научно-исследовательских работ студентов опубликованы в рецензируемых журналах, были представлены на научных конференциях и конкурсах различного уровня, четверо студентов становились победителями конкурса на получение стипендии Правительства по приоритетным направлениям.

Председатель комиссии: _______________ / В.В. Романенко

Члены комиссии: _______________ / А.И. Исакова

_______________ / М.В. Григорьева

_______________ / А.А. Мицель

УТВЕРЖДАЮ	УТВЕРЖДАЮ
директор ООО «Система Автоматизация Бизнес»	проректор ТУСУР по научной работе, д.т.н., профессор
______ Сим А.Б.	______ Ремпе Н.Г.

АКТ ВНЕДРЕНИЯ

программной системы "Аукцион", предназначенной для имитационного моделирования торгов

Комиссия в составе 3 человек составила настоящий акт в том, что в ООО «Система Автоматизация Бизнес» внедрен пакет программ "Аукцион" для целей имитационного моделирования торгов, проходящих в форме аукциона.

В программном продукте реализованы два алгоритма проведения торгов, особенностью которых является возможность поиска предпоследнего участника, а также учет мотивов претендентов. Программа позволяет осуществлять работу в двух режимах: ручном и автоматическом, а также учитывать возникающие в системе стохастические факторы, связанные с поведением и характеристиками участников. Для сравнения результатов работы каждого алгоритма, по таким критериям как эффективность, число шагов и цена продажи второго участника используются следующие методы: определение множества Парето, абсолютная и относительная свертка критериев. Пакет предоставляет возможности графического представления результатов моделирования и их статистического анализа.

Технические преимущества состоят в автоматическом воспроизведении процесса торгов по заданному алгоритму с расчетом его характеристик.

Экономическая результативность заключается в следующем: 1) снижение затрат на повторное проведение аукциона в случае отказа победителя от заключения договора за счет нахождения предпоследнего поставщика; 2) определение оптимальной стратегии проведения аукциона в заданных условиях, обеспечивающей минимум числа шагов аукциона, цены продажи и эффективности второго поставщика.

Социальная значимость определяется использованием электронно-вычислительных устройств с применением программного обеспечения, в основе которого лежат алгоритмы проведения аукциона, непосредственно на экономических объектах.

ЗАКАЗЧИКА	ИСПОЛНИТЕЛЬ
Директор,	Научный руководитель, д.т.н., профессор
Сим А. Б.______	Мицель А.А. ______
Начальник отдела,	Исполнители:
Симонова О. В.______	Каштанова О.В. ______
Ведущий программист,	Грибанова Е.Б. ______
Чабанец Н.А.______	
«17» апреля 2007 г.	«10» апреля 2007 г.

УТВЕРЖДАЮ

Проректор ТУСУР по УР, к.т.н.

_______ М.Т. Решетников

«27» 04 2007 г.

АКТ

о внедрении результатов кандидатской диссертации
Грибановой Екатерины Борисовны в учебный процесс

Результаты диссертационной работы Е.Б. Грибановой существенно используются при изучении студентами ТУСУР курса: «Имитационное моделирование экономических процессов» по специальности 080801 – «Прикладная информатика (в экономике)» на кафедре АСУ и в ТМЦ ДО.

В вышеизложенном курсе использованы результаты 3-й и 4-й глав диссертации, в которых рассматриваются имитационные модели экономических объектов и система «Имитатор».

Вид внедрения: Учебные пособия «Имитационное моделирование экономических процессов», «Лабораторный практикум по имитационному моделированию экономических процессов», руководство к лабораторным работам и курсовому проекту по дисциплине «Имитационное моделирование экономических процессов», пакет программ «Имитатор».

Директор ТМЦ ДО ТУСУР, к.э.н.

_______ А.Ф. Уваров

Зав. отделом УМО ТМЦ ДО, к.т.н.

А.И. Воронин

Зав. кафедрой АСУ д.т.н., профессор

А.М. Кориков

Научный руководитель,

д.т.н., профессор

_______ А.А. Мицель

Исполнитель,

аспирантка каф. АСУ

_______ Е.Б. Грибанова

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

ФГНУ «ГОСУДАРСТВЕННЫЙ КООРДИНАЦИОННЫЙ ЦЕНТР ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»

ОТРАСЛЕВОЙ ФОНД АЛГОРИТМОВ И ПРОГРАММ

СВИДЕТЕЛЬСТВО ОБ ОТРАСЛЕВОЙ РЕГИСТРАЦИИ РАЗРАБОТКИ

№ 8167

Настоящее свидетельство выдано на разработку:

Программа имитационного моделирования торгов, проходящих в форме аукциона «Аукцион»

зарегистрированную в Отраслевом фонде алгоритмов и программ.

Дата регистрации: **17 апреля 2007 года**

Авторы: **Каштанова О.В., Грибанова Е.Б.**

Организация-разработчик: **Томский государственный университет систем и радиоэлектроники**

Директор _______________ **Е.Г. Калинкевич**

Руководитель ОФАП _______________ **А.И. Галкина**

Дата выдачи 25.04.2007

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

ГОСУДАРСТВЕННЫЙ КООРДИНАЦИОННЫЙ ЦЕНТР ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ОТРАСЛЕВОЙ ФОНД АЛГОРИТМОВ И ПРОГРАММ

СВИДЕТЕЛЬСТВО ОБ ОТРАСЛЕВОЙ РЕГИСТРАЦИИ РАЗРАБОТКИ

№ 7080

Настоящее свидетельство выдано на разработку:

Программа имитационного моделирования экономических объектов «Имитатор»

зарегистрированную в Отраслевом фонде алгоритмов и программ.

Дата регистрации: **24 октября 2006 года**

Авторы: **Мицель А.А., Грибанова Е.Б.**

Организация-разработчик: **Томский государственный университет систем управления и радиоэлектроники**

Директор ______________ **Е.Г. Калинкевич**

Руководитель ОФАП ______________ **А.И. Галкина**

Дата выдачи 31.10.2006

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

ГОСУДАРСТВЕННЫЙ КООРДИНАЦИОННЫЙ ЦЕНТР ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ОТРАСЛЕВОЙ ФОНД АЛГОРИТМОВ И ПРОГРАММ

СВИДЕТЕЛЬСТВО ОБ ОТРАСЛЕВОЙ РЕГИСТРАЦИИ РАЗРАБОТКИ

№ 7081

Настоящее свидетельство выдано на разработку:

Программная система имитационного моделирования управления запасами «Запас»

зарегистрированную в Отраслевом фонде алгоритмов и программ.

Дата регистрации: **24 октября 2006 года**

Авторы: **Бойченко И.В., Грибанова Е.Б.**

Организация-разработчик: **Томский государственный университет систем управления и радиоэлектроники**

Директор ____________ **Е.Г. Калинкевич**

Руководитель ОФАП ____________ **А.И. Галкина**

Дата выдачи 31.10.2006

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2021615840

«Программа формирования интегрального показателя социально-экономического объекта»

Правообладатель: ***Грибанова Екатерина Борисовна (RU)***

Автор(ы): ***Грибанова Екатерина Борисовна (RU)***

Заявка № **2021612929**

Дата поступления **09 марта 2021 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ ***13 апреля 2021 г.***

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2021615540

«Программа решения обратной задачи формирования прибыли с помощью стохастических алгоритмов»

Правообладатель: ***Грибанова Екатерина Борисовна (RU)***

Автор(ы): ***Грибанова Екатерина Борисовна (RU)***

Заявка № **2021612974**

Дата поступления **09 марта 2021 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ ***09 апреля 2021 г.***

Руководитель Федеральной службы по интеллектуальной собственности

Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2021661344

Программа оценки времени размещения сообщений в группах онлайновой социальной сети ВКонтакте

Правообладатель: ***Федеральное государственное бюджетное образовательное учреждение высшего образования «Томский государственный университет систем управления и радиоэлектроники» (RU)***

Авторы: ***Грибанова Екатерина Борисовна (RU), Савицкий Александр Сергеевич (RU)***

Заявка № **2021660554**

Дата поступления **07 июля 2021 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ ***09 июля 2021 г.***

Руководитель Федеральной службы по интеллектуальной собственности

Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2021661430

Формирование маржинальной прибыли предприятия

Правообладатель: ***Федеральное государственное бюджетное образовательное учреждение высшего образования «Томский государственный университет систем управления и радиоэлектроники» (RU)***

Авторы: ***Грибанова Екатерина Борисовна (RU), Логвин Игорь Николаевич (RU)***

Заявка № **2021660555**

Дата поступления **07 июля 2021 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ ***12 июля 2021 г.***

Руководитель Федеральной службы

по интеллектуальной собственности

Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2021662008

Оценка групп социальной сети для реализации маркетинговых мероприятий

Правообладатель: ***Федеральное государственное бюджетное образовательное учреждение высшего образования «Томский государственный университет систем управления и радиоэлектроники» (RU)***

Автор(ы): ***Грибанова Екатерина Борисовна (RU)***

Заявка № **2021660609**

Дата поступления **07 июля 2021 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ ***20 июля 2021 г.***

Руководитель Федеральной службы по интеллектуальной собственности

Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2022662734

«Программа решения иерархической многономенклатурной обратной задачи формирования прибыли предприятия»

Правообладатель: ***Грибанова Екатерина Борисовна (RU)***

Автор(ы): ***Грибанова Екатерина Борисовна (RU)***

Заявка № **2022662345**

Дата поступления **30 июня 2022 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ ***07 июля 2022 г.***

Руководитель Федеральной службы

по интеллектуальной собственности

Ю.С. Зубов