Министерство высшего образования и науки Российской Федерации

МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ (ГУ)

ФАКУЛЬТЕТ АЭРОФИЗИКИ И КОСМИЧЕСКИХ ИССЛЕДОВАНИЙ

С.С. Гранин

СТАЦИОНАРНЫЕ СТРАТЕГИИ УПРАВЛЕНИЯ ЗАПАСАМИ ПРИ СЛУЧАЙНОМ СПРОСЕ В УСЛОВИЯХ ПАРАМЕТРИЧЕСКОЙ НЕОПРЕДЕЛЕННОСТИ

Диссертация на соискание ученой степени

кандидата технических наук

по специальности 05.13.10

«Управление в социальных и экономических системах»

Научный руководитель

д.т.н. А.С. Мандель

г. Москва

2021

Содержание

[Содержание 2](#_Toc68533659)

[Введение 4](#_Toc68533660)

[Глава 1. Модели управления запасами при случайном спросе: обзор и классификация 12](#_Toc68533661)

[1.1. Проблематика управления запасами при случайном спросе и ее связь с управлением цепями поставок: история, развитие и классификация моделей 12](#_Toc68533662)

[1.2. Модели управления запасами при случайном спросе 16](#_Toc68533663)

[1.3. Модели управления цепями поставок 22](#_Toc68533664)

[1.4. Волновой эффект в цепях поставок 23](#_Toc68533665)

[1.5. Методы синтеза алгоритмов управления в условиях нестационарности и неопределенности 32](#_Toc68533666)

[1.6. Выводы по главе 1 41](#_Toc68533667)

[Глава 2. Стационарные стратегии управления запасами при случайном спросе в условиях параметрической неопределенности 43](#_Toc68533668)

[2.1. Базовая модель оптимального управления запасами при случайном спросе 43](#_Toc68533669)

[2.2. Сравнение «близоруких», стационарных и «дальнозорких» стратегий 47](#_Toc68533670)

[2.3. Оптимальность *Rr*-стратегий управления запасами и связь между «дальнозоркими» и стационарными стратегиями управления 52](#_Toc68533671)

[2.4. Исследование зависимости оптимальных значений *R* и *r* стационарных стратегий управления запасами от эконометрических параметров базовой модели 54](#_Toc68533672)

[2.5. Обсуждение лестничного характера поведения параметра *R* стратегии управления запасами 58](#_Toc68533673)

[2.6. Скорость сходимости параметров *Rn* и *rn* к своим предельным значениям 64](#_Toc68533674)

[2.7. Выбор стратегии управления запасами при случайном спросе и изменении эконометрических параметров базовой модели 66](#_Toc68533675)

[2.8. Выводы по главе 2 70](#_Toc68533676)

[Глава 3. Модели управления запасами при случайном спросе в условиях срывов в процессе поставок 72](#_Toc68533677)

[3.1. Проблематика управления запасами при случайном спросе и ненадежных поставщиках 72](#_Toc68533678)

[3.2. Исходные положения для построения модели оптимального выбора стратегий управления запасами при случайном спросе и ненадежных поставщиках 73](#_Toc68533679)

[3.3. Модель оптимального выбора размеров поставок при случайном спросе в условиях ненадежности поставщиков 74](#_Toc68533680)

[3.4. Формирование правил рационального выбора поставщиков 75](#_Toc68533681)

[3.5. Компьютерное моделирование 77](#_Toc68533682)

[3.6. Выводы по главе 3 86](#_Toc68533683)

[Глава 4. Модели управления запасами при случайном спросе и отказах потребителей от поставленной им продукции 87](#_Toc68533684)

[4.1. Модель управления запасами с возвратами при случайном спросе 87](#_Toc68533685)

[4.2. Доказательство оптимальности (*Rn*, *rn*, *Sn*, *sn*)–стратегий 90](#_Toc68533686)

[4.3. Модель управляемой системы массового обслуживания и ее связь с моделью управления запасами с возвратами 92](#_Toc68533687)

[4.4. Алгоритмы построения оптимальных стратегий переключения каналов 93](#_Toc68533688)

[4.5. Аналогии между задачами теории управляемых систем массового обслуживания и задачами теории управления запасами и производством и пороговый характер оптимальных стратегий переключения каналов 94](#_Toc68533689)

[4.6. Компьютерное моделирование 96](#_Toc68533690)

[4.7. Выводы по главе 4 97](#_Toc68533691)

[Глава 5. Особенности компьютерного моделирования 99](#_Toc68533692)

[5.1. Общие положения 99](#_Toc68533693)

[5.2. Переход от последовательных к параллельные вычислениям 100](#_Toc68533694)

[5.3. Динамическая остановка цикла по *n* при стабилизации значений *Rn* 101](#_Toc68533695)

[5.4. Предварительный расчёт и «кэширование» значений избранных функций 101](#_Toc68533696)

[5.5. Дальнейшая возможная оптимизация скорости вычислений 102](#_Toc68533697)

[5.6. Методика работы с компьютерными моделями 105](#_Toc68533698)

[5.7. Выводы по главе 5 106](#_Toc68533699)

[Заключение 107](#_Toc68533700)

[Литература 108](#_Toc68533701)

[Приложение А. Исходный код модели задачи управления запасами для расчёта оптимальных параметров, суммарных средних затрат, построения параметрического «атласа» и моделирования случая с альтернативными поставщиками 121](#_Toc68533702)

[Приложение Б. Исходный код модели сравнения затрат «близоруких», стационарных и «дальнозорких» стратегий управления методом Монте-Карло 133](#_Toc68533703)

[Приложение В. Исходный код модели сравнения затрат методом Монте-Карло для задачи управления запасами при изменении параметров 142](#_Toc68533704)

[Приложение Г. Исходный код модели переключения каналов обслуживания 153](#_Toc68533705)

[Приложение Д. Акт о внедрении результатов кандидатской диссертационной работы 155](#_Toc68533706)

Введение

**Актуальность темы.** Широкое становление логистики запасов, как сугубо прикладной дисциплины, реализующей достижения теории управления запасами и опирающейся на анализ практического опыта специалистов в области управления снабжением, поставило новые вопросы к самой теории. И, прежде всего, что делать в условиях сбоев, когда, например, изменяется ситуация на рынке или законодательно налоговая база, возникают непредвиденные срывы в работе поставщиков или потребители, вопреки оговоренным условиям договоров, изъявляют желание отказаться от полученных ими поставок, желая, в конечном счете, возвратить купленное продавцу. Часть этих вопросов, благодаря полученным в диссертации новым результатам, удалось разрешить. И, прежде всего, в силу использования созданного в работе инструмента, позволяющего изучить параметрическое пространство базовой модели управления запасами методами компьютерного моделирования. На этой основе удалось по-новому осмыслить свойства оптимальных стратегий управления запасами, продвинув интуитивное понимание правильных приемов управления запасами и производством не только теоретиками, но и практиками. Кроме параметрической неопределенности в базовой модели были также учтены непредсказуемые срывы в работе поставщиков и неопределенность в поведении потребителей вплоть до их отказов от ранее приобретенных товаров. Это позволило использовать богатые теоретические и практические наработки из области управления снабжением для решения задач управления важным классом систем массового обслуживания.

В середине XX века в классической книге К. Эрроу, С. Карлина и Г. Скарфа [1] были заложены основы математической теории управления запасами и производством, в рамках которой к 70-80-м годам прошлого века усилиями Ф. Хэнсменна [2], Т. Уайтина и Дж. Хедли [3], Н. Прабху [4], а также отечественных ученых Ю.И. Рыжикова [5], А.А. Первозванского [6], Е.В. Булинской [7], [8] и многих других удалось разработать первые научно обоснованные алгоритмы управления запасами и производством. Параллельно велась разработка теории систем массового обслуживания. В нашей стране отметим монографии Б.В. Гнеденко и И.Н. Коваленко [9], а за рубежом Т. Саати [10]. Многие модели систем массового обслуживания, в конечном счете, оказались очень полезными в качестве альтернативных постановок задач математической теории управления запасами и производством, что первыми подметили А.А. Первозванский [6] и Н. Прабху [4].

К концу 80-х годов многочисленные попытки внедрения результатов теории в практику привели исследователей к выводу о том, что базовые модели теории нуждаются в серьезных обобщениях и развитии, поскольку большая часть перспективных (в плане применения научно обоснованных алгоритмов управления) прикладных ситуаций отличалась существенной априорной и апостериорной неопределенностью и не стационарностью протекающих в этих системах процессов. К этому времени Я.З Цыпкиным был разработан достаточно общий инструмент для решения задач управления в условиях неопределенности – теория адаптивного управления [11], [12], которая была подытожена Я.З. Цыпкиным в информационной теории идентификации [13]. Исследователи в области математической теории управления запасами и производством откликнулись на это предложение новыми продвинутыми результатами. Отметим, прежде всего, новую работу уже упомянутого Н. Прабху [14], книгу В.А. Лотоцкого и А.С. Манделя [15] и метод анализа иерархий Т. Саати [16].

В последнее десятилетие в связи с широким внедрением *ERP*-систем в области теории и приложений управления производственно-складскими системами, включая тот инженерно-экономический раздел этой теории, который получил название логистики [17], и, в частности, логистики запасов, все большее число работ посвящено решению проблем, собранных под общим наименованием «управление цепями поставок» (*supply chain management*) [18], [19]. Интерес к этому классу задач обусловлен прикладной актуальностью этой темы: все большее число предлагаемых моделей, алгоритмов и решений либо изначально разрабатывались для принятия решений в конкретных практических ситуациях, либо рано или поздно находят свои приложения. Однако при этом, важно подчеркнуть, далеко не все из существовавших теоретических наработок, особенно применительно к системам со случайным спросом и с теми или иными видами неопределенности были включены в состав современных *ERP*-систем. Это объяснялось тем, что при создании таких систем (в начале 90-х гг. прошлого века – первые *MRP1*-системы, за ними – *MRP2*-системы, а сегодня уже *ERP*) стали широко применяться только те модели и инструменты, которые были обкатаны и доведены, что называется, до ума. И это, прежде всего, различные модели и алгоритмы решения задач математического программирования – как правило, линейного или квадратичного. Эта работа шла и продолжается бурным ходом, а некоторые классические модели управления запасами при случайном спросе, например, та модель, которая будет рассмотрена во главе 2 диссертации, и которая будет названа базовой, оказались «приброшенными». Такие модели плохо вели себя в онлайновых системах управления, требовали сложных и времяемких расчетов, а в условиях неопределенностей разного толка это было неприемлемо. При этом в *ERP*-системах с неопределенностью справлялись за счет создания систем поддержки принятия решений (СППР), аккумулировавших практический опыт специалистов в области логистики запасов и производства. Это, на наш взгляд, ставит перед теорией новый вызов: адаптировать имеющиеся модели к возможностям современных *ERP*-систем. Именно это составляет основное содержание диссертационной работы.

Если проанализировать имеющуюся литературу по моделям управления цепями поставок, см., например, [20] – [21] [22] [23] [24] [25] [26], то нетрудно видеть, что во многих случаях параметры, вводимые в эти модели, порождаются изучением моделей управления запасами со случайным спросом, многие из которых, в том числе и самые классические, нуждаются в дальнейших уточнениях (об этом – в следующем абзаце). В работе [20], например, вводятся следующие параметры: – полные суммарные затраты хранения запасов, – полные суммарные затраты по подаче повторных заказов и – полные суммарные затраты по подаче альтернативных заказов на *n*-м шаге и все они проистекают из оптимальной стратегии управления запасами для продукта *j*. В работе [21] используются значения страховых запасов для отдельных элементов цепей поставок. В работе [22] вводится величайшее множество параметров, относящихся к локальным характеристикам процессов хранения и преодоления последствий проявления дефицита, а также используется предположение о том, что спрос детерминирован и известен для достаточно большого числа шагов для периода планирования. В работе [23] учитывается случайность времени запаздывания поставок, однако в качестве параметра рассматривается заранее выбранный размер пополнения запасов у первого в цепи поставщика (после производителя). В работе [24] рассматривается задача управления цепью поставок в условиях неопределенности, когда каждое звено цепи поставок оценивается в рамках трех вариантов прогностических решений – пессимистическом, нейтральном и оптимистическом. Каждый из этих вариантов нуждается в проработке с помощью развитой модели управления запасами. В работе [25] задача планирования поставок ассоциируется с заданием таких характеристик процесса, как сделанные ранее назначения размеров поставок и фактические размеры запасов, с которыми завершается очередной шаг планового периода. В работе [26] в задаче управления процессом работы сборочного конвейера задаются такие параметры, как количество единиц полуфабрикатов каждого вида, доступные каждому из потенциальных поставщиков. И все эти параметры могут быть оценены только в результате рассмотрения адекватных линейных моделей.

Усилия ученых в области расширения возможностей внедрения разрабатываемых прикладных систем управления запасами и производством были подытожены формированием нового представления о том, какие разделы этой теории нуждаются в дальнейшем развитии и доработке. Стало понятным, что использование адаптивных и робастных алгоритмов управления требует совершенствования алгоритмов расчета оптимальных параметров стратегий управления запасами, которые обычно являются двухуровневыми (что впервые было обнаружено Г. Скарфом [1]). Соответствующие усилия были предприняты А.С. Манделем и учениками [27] – [28] [29] [30] [31] [32] [33] [34] [35] [36] [37] [38] [39] [40]. Однако поначалу оказалось, что изначальные алгоритмы расчета оптимальных параметров стратегий управления запасами были слишком «долгоиграющими»: время расчета этих параметров составляло почти целые сутки. И это только для одного набора многочисленных эконометрических параметров модели, тогда как сама идея такого расчета предполагала, что нужно перебрать достаточно много таких наборов. В самом деле, новые идеи реализации оптимальных управлений в условиях неопределенности базировались на том, чтобы быть в готовности решать задачу управления при любых соотношениях между эконометрическими характеристиками рассматриваемой модели.

Кроме того, стало понятным, что работы в области управления запасами при случайном спросе при неопределенности в процессе поставок, которые активно велись до начала 90-х годов прошлого века и были приостановлены в связи с широким внедрением *MPR* и *ERP*-систем (см. выше)[[1]](#footnote-1), которые заняли большое число специалистов в области исследования операций, нуждаются в дальнейшем развитии для расширения возможностей их практического применения.

**Цель диссертационной работы** состоит в том, чтобы развить и адаптировать известные модели управления запасами со случайным спросом к потребностям современных систем управления производством и запасами в условиях параметрической неопределенности и меняющейся инфляции и очертить области их возможного применения.

В результате в работе были поставлены и решены следующие **задачи**:

1. Рассмотрена возможность использования оптимальных параметров стационарных стратегий управления запасами для синтеза правил управления запасами в системах снабжения со случайным спросом, действующих на реальном рынке. Выполнено сравнение эффективности «дальнозорких», стационарных и «близоруких» стратегий управления запасами. Исследованы различные подходы к синтезу оптимальных стационарных стратегий. Дана сравнительная оценка этих подходов и отмечены особенности их применения. Приведены результаты модельного оценивания оптимальных параметров стационарных стратегий управления запасами как функций от эконометрических параметров модели и уровня инфляции.

2. Рассмотрена задача управления запасами при случайном спросе, связанная с оптимизацией процессов, которые возникают в цепи поставок при наличии нескольких альтернативных поставщиков, обладающих разной степенью надежности и разными эконометрическими характеристиками. Исследовано два случая: (а) фиксированная часть затрат на поставку у всех поставщиков одна и та же при не совпадающих ценах на ед. товара и (б) все компоненты затрат у разных поставщиков различны.

3. Предложено существенное обобщение модели теории управления запасами со случайным спросом – модель с возвратами, применение которой на практике представляется весьма ограниченным, однако показано, что результаты, полученные на этой модели, могут быть перенесены на решение одного класса прикладных и широко распространённых задач управления системами массового обслуживания.

4. Исследованы аналогии между задачами теории управления запасами и теории управляемых систем массового обслуживания. На основе выявленных аналогий предложены новые постановки задач теории управления запасами и производством и новые пороговые алгоритмы решения задач управления системами массового обслуживания.

**Методы исследования.** Основным методом исследования являлось математическое моделирование. При этом использовался и был развит математический аппарат теории управления запасами и теории массового обслуживания. Для прикладной части разработаны специальные алгоритмы и создан оригинальный (без использования сторонних библиотек или приложений) код программ в среде *MATLAB*.

**Научная новизна.** Полученные в диссертации результаты существенно развивают понимание того, как строить оптимальные стратегии управления запасами и каковы их свойства в условиях параметрической неопределенности. Впервые выполнен качественный анализ сравнительных достоинств и недостатков «близоруких», стационарных и «дальнозорких» стратегий управления запасами. Существенно развита классическая модель управления запасами при ненадёжных поставщиках. Предложена новая модель управления запасами с возвратами. Еще одним продвижением стало обнаружение и изучение аналогий между задачами управления запасами с возвратами и некоторыми задачами теории управляемых систем массового обслуживания. Это позволило, с одной стороны, существенно продвинуть теорию управляемых систем массового обслуживания, а, с другой, наметить новые постановки задач теории управления запасами.

**Защищаемые положения.**

1. Для базовой модели продемонстрированы преимущества «дальнозорких» и стационарных стратегий управления запасами при случайном спросе по сравнению с «близорукими» стратегиями. Выявлены и проинтерпретированы свойства оптимальных стратегий управления запасами в условиях параметрической неопределенности.

2. Создан инструмент для построения «альбома» характеристик систем управления запасами при случайном спросе при изменении эконометрических характеристик базовой модели и даны рекомендации по процессу поддержки принятия решений в условиях изменения параметров модели.

3. Исследованы задачи управления запасами при случайном спросе и неопределенности в процессе пополнения запасов, обусловленной наличием нескольких альтернативных и разно надежных поставщиков. Для некоторого класса моделей построены новые конструктивные алгоритмы одновременного построения оптимальных стратегий управления запасами и выбора наилучших поставщиков. В общем случае, когда все компоненты затрат у разных поставщиков различны, предложена общая схема формирования оптимальных стратегий.

4. Описана принципиально новая модель управления запасами при случайном спросе – модель с возвратами. Продемонстрирована ее связь с моделями теории управляемых систем массового обслуживания. Построены новые алгоритмы оптимизации управляемых системам массового обслуживания – алгоритмы выбора пороговых стратегий переключения каналов.

**Практическая значимость.**

1. В рамках класса моделей, для которых оптимальны двухуровневые стратегии управления запасами, удалось оптимизировать процесс вычислений для задачи динамического программирования, что позволило совершить качественный скачок в производительности вычислений. Благодаря выявленным особенностям основного алгоритма динамического программирования, радикально сокращено общее число операций и тем самым уменьшено время обсчёта типовой серии с одних суток до нескольких минут. Это позволило строить «альбомы» характеристик систем управления запасами при изменении, а, стало быть, и неопределенности эконометрических характеристик и состояния внешней среды, являющиеся крайне ценным дополнением в системах поддержки принятия решений (СППР) в системах снабжения.

2. Для СППР разработаны рекомендации по выбору параметров стратегий управления запасами при случайном спросе при изменении эконометрических характеристик модели и уровня инфляции.

3. Предложенные алгоритмы выбора поставщиков и построенная общая схема формирования оптимальных стратегий управления позволяют снизить общие экономические затраты и повысить надёжность процесса пополнения запасов.

4. Подмеченная аналогия между задачами теории управления запасами при наличии возвратов и некоторых задач теории управляемых систем массового обслуживания. Предложенные алгоритмы позволили применить наработанный аппарат теории управления запасами для решения практических задач управления системами массового обслуживания.

**Личный вклад.** Все основные результаты получены автором самостоятельно при научном руководстве доктора технических наук, главного научного сотрудника Манделя А.С.

**Апробация результатов работы.** Основные результаты диссертационной работы докладывались на международных конференциях «*Управление развитием крупномасштабных систем*» в Москве (*MLSD’2015*; *MLSD’2017*; *MLSD’2018* и *MLSD’2019*), на международной конференции «*IEEE International Conference on Logistics, Informatics and Services Sciences*» в Пекине (*LISS 2018*) и на «*62-й Всероссийской научной конференции МФТИ*».

**Публикации.** По теме диссертационной работы было опубликовано 12 научных работ. Из них 2 публикации в журналах, рекомендованных ВАК РФ, и 5 публикаций в изданиях, индексируемых в международных научных базах данных.

**Структура и объём работы.** Диссертация состоит из введения, пяти глав, выводов, заключения, списка литературы и приложения в виде исходного кода программ, моделирующих задачи, рассматриваемые в данной работе. Работа, не считая приложений, содержит 120 страниц текста, список использованных источников включает в себя 163 наименования.

Первая глава настоящей работы, по сути, начинается с того, что исследуется, как технический прогресс вкупе с развитием технических и экономических наук создавали необходимые предпосылки для становления дисциплин, связанных с управлением запасами и производством. Современные методы производства и экономические отношения требовали передовых (а главное, научно обоснованных) методов управления хозяйственными системами.

Далее представлен обзор исследований в области передовых методов управления сложными хозяйственными системами. Подавляющее количество работ посвящено методам проектирования (и управления) складских систем с целью повышения их устойчивости к внешним негативным воздействиям случайной природы. Рассмотрены два самых популярных и перспективных направления (эффект хлыста и волновой эффект), вобравших в себя большой пласт научных исследований, связанных с теорией управления запасами. Волновой эффект может выступить в роли явления, которое в состоянии объединить исследование в области управления (и восстановления стабильности) нештатными ситуациями в цепях поставок, подобно эффекту хлыста (относительно колебаний времени выполнения заказа и спроса). Это может задать повестку для будущих исследований в области динамики, управления, непрерывности и управления сбоями в цепях поставок, делая их более робастными, гибкими и прибыльными. Рассмотрены достоинства и недостатки методов, представленных исследователями данной области.

Отдельно рассмотрены методы синтеза алгоритмов управления запасами в условиях неопределенности и нестационарности (процессов, протекающих внутри системы снабжения). Для задачи управления многономенклатурными запасами проанализирован общий подход к её решению. Он основан на том, что в условиях неопределенности и нестационарности и отсутствия достоверной информации о статистических характеристиках спроса практическое решение задачи должно опираться на представленную в работе многоэтапную процедуру. Суть процедуры заключается в том, что на первых этапах происходит классификация всех видов товаров по методу *ABC*-анализа. Далее для полученных классов осуществляется выделение трендов (включающих и сезонные компоненты спроса), на основе которых решается детерминированная многономенклатурная задача управления запасами. На последующих этапах решается проблема формирования дополнительных (страховых) запасов для компенсации флуктуаций спроса относительно выделенных трендов.

Во второй главе исследуется возможность использования оптимальных параметров стационарных стратегий управления запасами для синтеза правил управления запасами в системах снабжения со случайным спросом, действующих на реальном рынке. Рассмотрены известные в литературе основные подходы к формированию стационарных параметрических стратегий управления запасами и очерчены области их применения. Выполнено сравнение эффективности «дальнозорких», стационарных и «близоруких» стратегий управления запасами. Исследованы различные подходы к синтезу оптимальных стационарных стратегий. Дана сравнительная оценка этих подходов и отмечены особенности их применения.

Создан программный комплекс, позволивший выполнить моделирование предложенных алгоритмов управления запасами. С использованием этого комплекса выполнено исследование (на отдельных примерах) зависимости значений параметров стационарных стратегий управления запасами от уровня инфляции (задаваемое значением коэффициента дисконтирования α) и от других эконометрических параметров модели. Полученные результаты позволили выявить некоторые неожиданные свойства оптимальных стационарных стратегий и дать им теоретическое обоснование.

В третьей главе рассматривается задача управления запасами, связанная с оптимизацией процессов, которые возникают в цепи поставок при наличии нескольких альтернативных поставщиков, обладающих разной степенью надежности и разными эконометрическими характеристиками. Исследовано два случая: (а) фиксированная часть затрат на поставку у всех поставщиков одна и та же при не совпадающих ценах на ед. товара и (б) все компоненты затрат у разных поставщиков различны. Для случая (а) предложены конструктивные алгоритмы построения оптимальных стратегий управления запасами и выбора поставщиков. В случае (б) предложена общая схема формирования оптимальных стратегий. Приведены результаты численного моделирования.

Четвёртая глава посвящена исследованию аналогий между задачами теории управления запасами и теории управляемых систем массового обслуживания. На основе выявленных аналогий предложены новые постановки задач теории управления запасами и производством и новые пороговые алгоритмы решения задач управления системами массового обслуживания.

В пятой главе детально изучаются ключевые моменты компьютерного моделирования (основанного на решении уравнений дискретного динамического программирования) задач теории управления запасами. Описаны возникающие сложности (и даны практические рекомендации по их устранению), с которыми может столкнуться исследователь, занимающийся задачами данного класса. Также рассмотрены основные подходы к оптимизации расчётов модели и отмечены те из них, которые позволили автору данной работы снизить время обсчёта модели с одних суток до нескольких минут исключительно алгоритмическими методами.

# Модели управления запасами при случайном спросе: обзор и классификация

## Проблематика управления запасами при случайном спросе и ее связь с управлением цепями поставок: история, развитие и классификация моделей

Управления запасами является общей проблемой для предприятий, относящихся к любому сектору хозяйственной системы. Создание запасов необходимо в промышленности, в сельском хозяйстве, в розничной торговле, в оборонной отрасли и т.д. и т.п. Почему организации идут на создание запасов? На то есть обширный ряд причин. Принято отмечать тот факт, что зачастую либо экономически не целесообразно, либо просто физически невозможно организовать поступление товаров именно в момент возникновения спроса на них. В случае отсутствия товара потребитель не склонен его долго ждать. Ещё одна важная причина, особенно веская для предприятий, занимающихся розничной торговлей, заключается в том, что объёмы продаж, а, стало быть, и прибыль, можно увеличить путём хранения некоторого запаса товаров, который может быть предложен потребителю.

В управлении запасами возникает два ключевых вопроса – когда необходимо пополнять запас и в каких объёмах. То есть, вне зависимости от номенклатуры товаров и сложности системы снабжения, деятельность организации, управляющей запасами, в конечном итоге сводится к определению, когда и какого размера делать заказы на пополнение.

В математических моделях, описывающих системы снабжения, необходимо вводить некоторые приближения и допущения, так как не всегда возможно выявить, что представляет собой реальный мир. Не стоит забывать, что в математическом плане, модель может оказаться слишком сложной и её исследование на практике либо экономически не оправдано, либо, иногда, попросту невозможно. Зачастую, уточнения, полученные при помощи более сложных моделей, не компенсируют дополнительные расходы. На практике более простые модели могут дать отличный результат.

Несмотря на то, что хранение запасов и связанные с этим вопросы, столь же стары, как и сама история, только с начала прошлого века для их изучения начали предпринимать попытки использовать аналитические методы. По-видимому, одновременный рост промышленного производства и технических наук (наук об организации производства, в частности) дали толчок к применению математических методов для анализа систем управления запасами. В первую очередь в этом были заинтересованы отрасли, столкнувшиеся на практике с вопросами календарного планирования производства и хранения запасов в условиях серийного выпуска продукции (когда переналадка оборудования и перестройка технологической цепочки обходится очень дорого) и её хранения на заводском складе. В 1915 году Форд Харрис вывел формулу, которую часто называю формулой размера партии [41]. Данную формулу выводили многие исследователи и, в том числе, через 16 лет после Харриса Р. Уилсон получил её как один из результатов разработанной им схемы управления запасами. Так она и осталась в теории, как формула Уилсона. Ф. Е. Реймонд, сотрудник Массачусетского технологического института написал первую книгу, полностью посвящённую управлению запасами [42]. Книга посвящена практическому применению различных обобщений простой модели размера партии и не содержит ни теоретических обоснований, ни выкладок. Серьёзное внимание стохастической природе процессов управления запасами было уделено только по окончании второй мировой войны с развитием исследования операций и наук о методах управления и руководства. Так Т. Уайтин разработал стохастический вариант простой модели размера партии [43] и в 1953 г. опубликовал первую книгу, посвящённую подробному рассмотрению вероятностных моделей управления запасами.

Любопытно, что первоначально активный интерес к изучению и использованию аналитических методов для решения задач управления запасами проявили инженеры, занимающиеся решением практических задач, а не экономисты. Вероятно, это было вызвано тем, что на тот момент экономистов интересовали главным образом статические модели экономических систем. Однако с течением времени некоторые экономисты и математики стали проявлять интерес к моделям управления запасами. В первую очередь их интересовал ряд математических свойств и экономическая интерпретация, а не вопросы прямого практического применения. Одной из первых статей, дающих строгий математический анализ простой модели управления запасами, принадлежала авторству Р. Эрроу, Ф. Харриса и Ф. Маршака, вышла в 1951 г. [44]. В 1958 г. вышла книга Р. Эрроу, Т. Карлина и Г. Скарфа [1], полностью посвящённая математическим свойствам систем управления запасами.

Существующие складские системы можно различать по размерам, сложности, рабочим издержкам, характеру случайных процессов, затрагивающих систему, и характеру информации, поступающей ответственным за принятие решений лицам. Эти отличия могут быть трактованы, как структурные различия складских систем и могут иметь непосредственное влияние на стратегию функционирования, применяемую при управлении системы. Стратегия функционирования – это правило, отвечающее на вопрос: когда и сколько следует заказывать [3].

В зависимости от возможностей и потребностей организации, складская система может состоять из одного или нескольких складов. Если рассматривать системы снабжения армии, то мы увидим целые сети складов и ремонтных мастерских, необходимые для быстрого и бесперебойного обеспечения военных амуницией, припасами и деталями, в случае необходимости. В случае более скромной по размерам и ресурсам организации, будет вполне достаточно и одного склада для хранения всей, производимой предприятием, продукции.

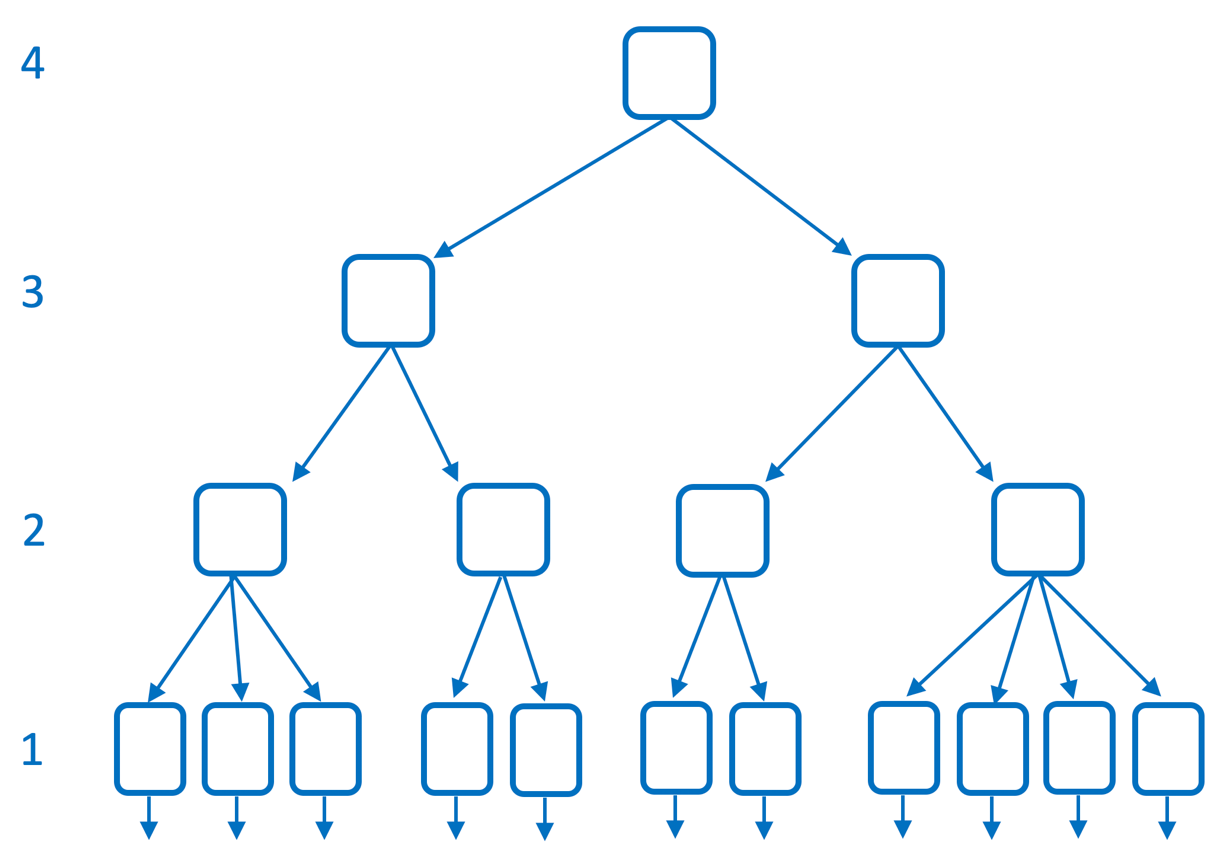


Рис. . Пример складской системы с разветвлённой структурой.

При наличии нескольких пунктов хранения запасов возможны различные формы взаимодействия между ними. Случай, когда один пункт служит складом для одного или нескольких других пунктов – является одной из простейших форм такого взаимодействия. Эта форма приводит к так называемой складской системе с разветвлённой структурой [3]. На рисунке изображён один из возможных типов такой системы. Стрелками обозначено нормальное направление потока запасов в системе, состоящей из четырёх уровней, называемых эшелонами. В данной системе, требования поступают только в пункты первого уровня. Пополнение запасов пунктов первого уровня производится со складов второго уровня, которые в свою очередь пополняются со складов третьего уровня и так далее. Данный пример представляет из себя только один из возможных вариантов структуры системы. Возможны варианты, когда требованию возникают на любом из уровней хранений. Пункты хранения могут пополняться не только с ближайшего верхнего уровня, но также от любого склада верхних уровней или, к примеру, из отдельных источников. Некоторые системы допускают перераспределение запасов между складами одного уровня.

Разветвлённая структура – одна из самых часто встречающихся структур систем управления запасами. Однако на практике не всегда нужно или корректно рассматривать всю систему, так как различные её участки находятся во владении и управляются разными организациями. К примеру, схема на рис. 1 может представлять систему распределения, в которой на четвёртом уровне находиться склад завода производителя, на третьем уровне работают региональные склады, на втором уровне – городские, а на первом – предприятия розничной торговли. При такой системе изготовитель имеет возможность управлять только заводским производством и его складом, при этом региональные склады могут управляться другими организациями, а городские склады и склады предприятий розничной торговли – третьими. Стоит помнить, что склады, находящиеся на одном уровне, тоже могут принадлежать различным организациям. Часто встречающаяся на практике ситуация, когда склады в разных городах принадлежат разным собственникам. В рамках такой системы стратегия управления запасами может выбираться организацией, для складов, находящихся в поле её подчинения. Обычно нет необходимости анализировать всю системы целиком, чтобы выбрать стратегию функционирования для каждого склада на каждом уровне. Но может возникнуть ситуация, при которой будет необходимо разработать оптимальную стратегию управления запасами, на одном из складов второго уровня. В таком случае в роли источника выступал бы склад третьего уровня, а в роли потребителей предприятия розничной торговли на первом уровне. В некоторых случаях система управление охватывает не только склады, но и источник снабжения, то есть завод изготовитель. В таком случае задача не ограничивается одним лишь управлением запасами, но и расширяется за счёт необходимости составления календарных планов производства.

На рис. 2 схематически изображена структура системы управления запасами в которой присутствует один источник снабжения и один пункт хранения. Пункт хранения агрегирует потребительские требования и в соответствующие моменты времени делает заказы у источника на пополнение своих запасов. Систему, представленную на рис. 1, крайне трудно изучить аналитически, так как даже анализ системы с относительно простой структурой может вызвать не малые трудности. Как было отмечено выше реальные системы обладают разветвлённой структурой, однако на практике достаточно ограничиться рассмотрением пунктов хранения по отдельности, если они управляются разными организациями. Даже в случае, когда одна организация управляет несколькими складами иногда уместно рассматривать пункты хранения независимо друг от друга из-за слабого взаимодействия между ними.

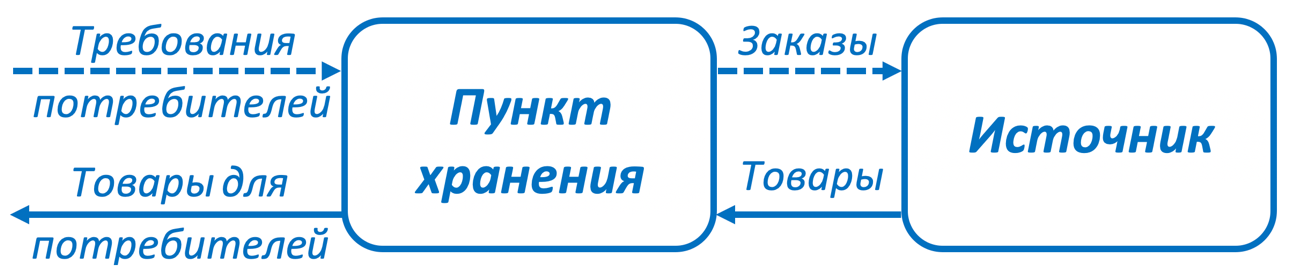


Рис. . Упрощённая модель системы управления запасами.

С развитием глобальных промышленных производств остро встал вопрос об их управлении в условиях нестационарности и не определённости. В 1961 году Джэй Форрестер опубликовал статью в журнале *Industrial Dynamics* в которой изложил концепцию эффекта хлыста (так же известную, как эффект Форрестера), выявленную на практике специалистами американской компании *Procter & Gamble*. Суть феномена заключается в том, что величина колебания потребительского спроса увеличивается в направлении от конечных потребителей до исходных поставщиков товаров или сырья. Иными словами, эффект хлыста – это ситуация, возникающая в цепи поставок, при которой небольшие изменения спроса на нижнем уровне цепочки способны спровоцировать, усиливающиеся (по направлению вверх по цепочке) колебания в планах по выпуску или хранению запасов.

Одна из основных причин возникновения эффекта хлыста – это ошибки в прогнозировании спроса. К примеру, спрос на определённый товар может резко вырасти из-за временной акции в магазине, что приведёт к увеличению размеров заказа у поставщика, который в свою очередь неверно интерпретирует ситуацию и нарастит производство (или закупки у вышестоящего участника цепи) с запасом. И так каждый вышестоящий член логистической цепи будет увеличивать выпуск или закупки всё больше и больше. Ключевой причиной в ошибках прогнозирования выступает отсутствие прозрачности всей цепи поставок. По сути, каждый участник выступает как самостоятельный агент и ориентируется только на полученные им заказы, что приводит к выбору стратегии оптимальной для него, но не для всей цепи поставок в целом. Для поставщика эффект хлыста означает потери, вызванные вложениями в основной капитал (для увеличения выпуска), которые не окупятся. Для предприятий розничной торговли, характерно простаивание товара на складе и сопутствующие этому издержки.

В последние десятилетия активно стало развиваться направление, которое получило собирательное название - волновой эффект. Суть заключается в том, что сбои в производстве или поставках (затрагивая конкретные предприятия) распространяют своё влияние на всю цепь поставок, словно волны на воде. Анализ того, как достигнуть запланированных экономических показателей в неопределенной и нестационарной среде производства, является жизненно необходимой и актуальной задачей во многих цепях поставок.

## Модели управления запасами при случайном спросе

Под сетевыми моделями цепей поставок подразумеваются системы с разветвлённой структурой, как показано на рис. 1. Ввиду сложности таких систем, зачастую очень затруднительно (а иногда и невозможно) получить аналитическое решение для задач управления запасами или задач календарного планирования производства. В современной литературе наблюдается тренд, который заключается в том, что для изучения и моделирования сетевых систем применяются результаты из теории управления запасами с весьма поверхностным рассмотрением того, что же на самом деле происходит на нижнем, локальном уровне. Зачастую исследователи ограничиваются использованием неких эвристических правил и/или интегральных характеристик (полученных из теории управления запасами) для моделирования и изучения глобальных процессов, происходящих на сетевом уровне. Рассмотрим подробнее наиболее актуальный работы по этой теме.

Работа [45] посвящена проблеме из области управления цепями поставок, а именно – задаче выбора размера заказа и составления расписания заказов. Для задачи постулируются две целевые функции – минимизация итоговой стоимости заказа (и хранения заказанных товаров), и минимизация требуемого объёма хранения. Зачастую, к сожалению, невозможно составить расписание пополнения запасов, которое минимизировало бы обе целевые функции одновременно.

Очевидно, что в случае склада, у которого отсутствует ограничение на вместимость, задача превращается в задачу минимизации стоимости заказа и хранения товаров. Однако, как показывает практика, после нескольких лет использования, вместимость склада может оказаться не достаточной и поэтому расчёт необходимого размера склада до его постройки (или модернизация уже построенного склада) является актуальной проблемой.

В настоящее время распространён подход, при котором первоначальная задача составления расписания пополнения запасов декомпозируется на две подзадачи, которые впоследствии решаются раздельно. Сначала определяется размер заказа (без составления расписания пополнений и без учёта ограничения по вместимости склада). Далее, для установленного размера заказа определяется расписание пополнений, которое минимизирует максимальную необходимую вместимость склада. Такой подход зачастую генерирует невозможные решения, которые приводят к превышению допустимой вместимости склада. Авторы работы не приводят уникального аналитического решения поставленные задачи, вместо этого они предлагают использовать два эвристических подхода, генерирующих серии Парето-оптимальных (или близких им) решений.

Рассматривается задача пополнения запасов с несколькими типами товара, поставляемыми множеством внешних поставщиков для удовлетворения постоянного спроса. Все товары хранятся на общем складе и цель – минимизировать максимальный объём склада. Два составных элемента стоимости – это цена заказа и расходы на хранение. Потери вследствие дефицита в данной модели не учитываются. Цена заказа – заданная величина, не зависимая для каждого вида товаров, расход на хранение единицы типа товара тоже заданная, постоянная величина. Необходимо, чтобы пополнение запасов было выстроено таким образом, чтобы промежутки времени между заказами для каждого типа товаров были постоянны. Так же, эти промежутки должны быть кратны выбранной целочисленной единице времени (одному дню, к примеру), которая называется фундаментальным (или базовым) циклом. Очередным любопытным допущением является отсутствие задолженного спроса, поэтому уровень запасов в начале каждого периода (перед принятием решения о размере заказа) может быть только не отрицательным. Авторы делают допущение (без потери общности), что единица товара каждого типа занимает 1 единицу объёма хранения.

Рассмотрим первый метод поиска решений задачи, названный авторами «исследовательский метод» (*The exploratory method*), который заключается в поиске решений либо при помощи эвристики, либо путём расчёта модели и дальнейшего отбора Парето-оптимальных решений. Результаты моделирования показали, что значение целевой функции объёма склада *S* в среднем лучше (меньше) на 2,58% для математической модели, нежели для эвристики, но при этом время расчётов для эвристики оказалось в 640 раз меньше, чем время расчёта модели (42 секунды против 7,5 часов).

Второй метод назван в работе эволюционный алгоритм двух популяций (*The two-population evolutionary algorithm*). Многочисленные исследования свидетельствуют о том, что эволюционные алгоритмы хорошо подходят для решения задач оптимизации с несколькими целевыми функциями, так как они генерируют множество решений, из которых можно выделить близкие к Парето-оптимальным.

В конце концов, в работе [45] предлагается объединить лучшие решения, полученные эвристикой и эволюционным алгоритмом для формирования общего Парето-фронта решений, который в дальнейшем исследователь сможет использовать посредством введения весовой функции и задавая вес каждой из первоначальных целей.

Главным достоинством модели, предложенной в работе [45], следует считать рассмотрение многономенклатурной задачи и учет дополнительного ограничения на объём склада. Представленные эвристики сильно сократили время расчётов (19 минут против 7,5 часов), при этом потеря качества целевой функции (объём склада) была минимальной. К недостаткам следует отнести, прежде всего, то, что модель, по сути, «близорука», так как учитываются затраты только на одном шаге. Спрос считается детерминированным, а не стохастическим. Не учитываются потери вследствие дефицита.

В работе [46] поставлена задача поиска алгоритма, который мог бы стать инструментом для специалистов, управляющих промышленным производством с обширным парком различных станков. Забота управляющего – подбор стратегии пополнения запасных деталей для того, чтобы избежать простоев, вызванных поломкой станка, и при этом минимизировать стоимость хранения этих деталей. Представлена байесовская модель потребления запасных деталей, вкупе с модернизированной стратегией управления запасами типа (*mT, r\*, R*). Оригинальность данного исследования заключается в том, что процесс потребления запасных частей описывается набором типовых сценариев («сценарии потребления»), которые задаются системой индикаторов производительности, использующих состояния переменных в байесовской модели. После того, как определены выше упомянутые индикаторы, задаётся байесовская сеть, позволяющая путём байесовского моделирования, рассчитать для заданного периода планирования искомые комбинации таких параметров, как: длина периода восстановления, размер заказа, вид запасных частей (новые или бывшие в употреблении), риск простоя (и связанные с этим убытки), стоимость заказов и стоимость хранения.

Проблема поддержания производственных станков в рабочем состоянии является типичной для любой из ветвей производства. Задача управления запасами запчастей на предприятии осложняется их высокой стоимостью и дороговизной хранения, а также нестабильностью рынка (которая выражается во флуктуации цен на запчасти, доступности и сроков поставки). Ко всему вышесказанному добавляется перманентное желание собственников сэкономить, что, в свою очередь, понижает управляемость системы в случае поломок.

Классическая стратегия пополнения запасов (*T, r, R*) хороша тем, что предотвращает небольшие по размеру заказы в случае низкого спроса в период наблюдения, но плохо подходит для составления расписания закупки запасных частей (различного типа) для парка станков с длительным сроком службы. Также, стратегия пополнения должна подразумевать возможность принудительных пополнений, даже если уровень запасов достиг безопасного. Для случая, когда рабочая нагрузка и частота поломок являются переменными величинами, предлагается использовать модифицированную стратегию управления запасами (*mT, r\*, R*).

Авторы выбрали две формулировки каузальной модели:

1. Определить оптимальный период принудительных пополнений (с постоянной частотой) в условиях неопределённости, вызванной флуктуацией времён поставки и флуктуацией потребности в запасных частях.

2. Определить периоды пополнения, минимизирующих затраты для постоянных уровней запасов: полностью укомплектованного и безопасного.

В результате моделирования было получено 97 сценариев, 32 из которых были отринуты ввиду того, что на практике они не встречаются. Оставшиеся 65 были проанализированы и переработаны в 12 классов графиков, соответствующих различным конфигурациям. Обсчёт каждой конфигурации даёт искомые параметры: уровень запасов, общие расходы на хранение, расходы на закупку запасных частей, общее время простоя. В итоге авторы получили 32 комбинации из 97 сценариев.

Для конечного пользователя (то есть специалиста, уполномоченного принимать решения о закупках) эти сценарии являются базовыми блоками (для каждого типа запасных частей), помогающими принимать решения о закупках. Эти блоки возможно рассчитать при наличии информации об уровне запасов в начале каждого периода, стоимости запасных частей (каждого вида) и указании типа поставки (отложенная или принудительная). Итоговая стратегия управления запасов будет складываться из таких блоков (что является предметом отдельного исследования).

Модель (*T, r\*, R*) позволяет использовать принудительные пополнения запасов, в то время как модифицированная модель (*mT, r\*, R*) с варьируемым периодом контроля, позволяет минимизировать стоимость закупок. Этот подход был верифицирован моделью, построенной в BayesiaLab.

Сильной стороной данной работы является учёт не только стоимости хранения запасных частей, но и стоимости простоя в случае поломки и отсутствия деталей (что можно сравнить с потерями в следствии дефицита). Кроме того, в модели также учитывается время поставки и вероятность её запаздывания (то есть случай, когда поставка не успела прийти до конца текущего периода контроля) и имеется возможность задать режим принудительных поставок.

К недостаткам следует отнести то, что данная модель одношаговая и не учитывает задолженного спроса. Кроме того, она не даёт специалисту, осуществляющему управление складом, простого ответа (в виде параметра, к примеру) на вопрос – когда и сколько следует заказывать?

В работе [47] рассматривается задача выбора поставщиков и составление расписаний производства и дистрибуции при наличии не нулевой вероятности срывов внутри вытроенной цепи поставок. Для данной задачи автор предложил формулировку стохастического смешанного целочисленного программирования. Поставщики расположены в различных географических регионах и их поставки могут быть подвержены многоуровневым локальным сбоям (для каждого отдельного поставщика) и двухуровневым региональным сбоям (для всех поставщиков, находящихся в одном регионе). Рассматриваются одновременно две конфликтующие цели – минимизация ожидаемых затрат и максимизация ожидаемого уровня обслуживания. Оптимальное решение достигается при помощи метода агрегирования взвешенной суммы.

В глобальных цепях поставок, где материальные потоки подвержены различного рода сбоям, крайнюю важность обретает задача интеграции и координации планирования операций снабжения, производства и дистрибуции. Литература по данной тематике ограничена, в основном, детерминированными моделями с отдельно рассматриваемой частью снабжения. В своих предыдущих работах автор предложил новый стохастический метод смешанного целочисленного программирования для выбора поставщиков, размера заказов и планирования заказов конечных потребителей в многоуровневой цепи поставок, при наличии риска сбоев. В цитируемой работе развит метод для одновременной оптимизации поставок исходного сырья, производства и дистрибуции готовой продукции. В отличие от большинства статей, которые рассматривают шаблон сбоев по типу «всё или ничего», в данной работе все уровни локальных сбоев разделены на три класса: незначительный сбой, значительный сбой и полное закрытие. В то время как региональные сбои относятся к типу «всё или ничего».

Рассматривается трёх-эшелонная цепь поставок, ориентированная на потребителей. Эта цепь состоит из нескольких поставщиков деталей одного типа (критически необходимого), из одного производителя единственного типа конечного продукта и из нескольких центров дистрибуции, которые обслуживают набор конечных (для этой модели) потребителей. То есть, используя критически необходимые детали единого типа, которые производятся несколькими поставщиками, производитель собирает готовые изделия и отправляет их в несколько центров дистрибуции для удовлетворения спроса потребителей. Модель обладает рядом интересных особенностей. К примеру, производителя можно оштрафовать за невыполнение сроков или полный срыв заказов потребителей, вызванных недостатком деталей. Любой заказ потребителя может быть выполнен только в течении одного периода. Предполагается, что заказ на детали размещается на старте горизонта планирования, когда все заказы потребителей на конечный продукт известны. Каждый поставщик имеет достаточные мощности для удовлетворения полного спроса на детали и для подготовки поставки в течении одного периода планирования. Далее все детали от поставщика доставляются единой партией (в рамках одного периода). Уровень локального сбоя определяется долей (от полного) заказа (деталей), которая может быть отгружена производителю. Помимо независимых локальных сбоев (для каждого поставщика), существуют угроза региональных стихийных бедствий, способных полностью прервать работу всех поставщиков, относящихся к одному региону. Примеры таких стихийных бедствий включают в себя наводнения, землетрясения и так далее. Региональные сбои для каждой местности и локальные сбои для каждого поставщика считаются независимыми событиями.

Цель управления интегрированной цепью поставок при наличии риска сбоев состоит в том, чтобы распределить общий спрос на детали между подмножеством избранных поставщиков и для каждого сценария сбоя (из множества всех возможных сценариев) составить планы потребительских заказов и поставок готовых продуктов в центры дистрибуции, при этом оптимизировать отношение между ожидаемыми затратами и ожидаемым уровнем обслуживания. Модель WCS (*Weighted sum of normalized expected Cost and expected Service level*) – это выбор поставщиков, планирование потребительских заказов и дистрибуции с целью минимизировать взвешенную сумму *f1* нормированной ожидаемой стоимости продукта и *f2* нормированного ожидаемого уровня обслуживания. Необходимо минимизировать функцию *α f1 +* (1-*α*) *f2*, где 0 ≤ *α* ≤ *1.*

Пример, рассмотренный в разделе «компьютерное моделирование» работы [39], показал, что самые дешёвые (и не надёжные поставщики) из одного из регионов никогда не выбирались. Для случая, в котором выбиралась минимальная итоговая стоимость (*α* = 1), предпочтение было отдано самому дешёвому поставщику из числа самых надёжных. Для случая максимизации уровня обслуживания (*α* = 0) модель выбрала трёх самых надёжных и дорогих поставщика. Представленная модель продемонстрировала, что набор решений, направленных на повышение уровня обслуживания (*α →*0) более разнообразен, чем набор решений, направленных на минимизацию стоимости (*α* *→* 1). Так же моделирование продемонстрировало, что высокий ожидаемый уровень обслуживания зачастую связан с низкой ожидаемой долей удовлетворённого спроса. Это может указывать на то, что для достижения максимума ожидаемого уровня обслуживания, отменяются заказы потребителей, которые не могут быть удовлетворены в срок (заданный самим потребителем). С ростом *α*, то есть в случае смещения приоритета от максимума уровня обслуживания к минимуму стоимости, происходит выбор более дешёвых и менее надёжных поставщиков. Расписание поставок деталей и соответствующие расписания поставок готовой продукции становятся смещены ближе к концу горизонта планирования.

Солвер *Gurobi 6* на ноутбуке автора статьи [39] находил оптимальные решения за время из диапазона от нескольких минут до нескольких часов.

Данная работа хороша тем, что в ней рассматривается сетевая модель с учётом потерь в следствии срыва сроков поставок (аналогия дефицита в задачах теории управления запасами). Учёт различных сбоев, а также две целевые функции даёт больше простора специалисту, уполномоченному составлять расписания.

К недостаткам можно отнести то, что в общем случае распределение вероятности срыва поставки конкретным поставщиком неизвестно. В рамках допущения, касающегося единой совместной поставки различных заказов, модель не учитывает стоимости транспортировки и ограничения, связанные с производственными мощностями поставщика. Данная задача относиться к классу задач, рассматриваемых на сети производственных и распределительных центров, которые сводятся к различным вариациям задач математического программирования и лишь опосредованно (через некие упрощенные интегральные характеристики) учитывают стохастический характер процессов снабжения и производства.

## Модели управления цепями поставок

Начиная с конца 1960-х гг., всё большее практическое применение стал находить один из фундаментальных результатов теории управления запасами, который принадлежит Герберту Скарфу [48] и заключается в том, что при достаточно общих предположениях о характере процессов, происходящих в системе снабжения, и выборе критерия управления запасами в форме минимизации суммарных средних затрат в периоде планирования, оптимальные стратегии управления запасами оказываются параметрическими [3], [15].

Они относятся к классу так называемых (*R*, *r*)-стратегий управления запасами. Это означает, что за *n* шагов до конца периода для любого *n* найдутся два такие числа *Rn* и *rn*, что, если уровень запасов в системе больше, чем *rn*, то размер заказа на пополнение запасов в этот момент времени подавать не нужно, а если уровень запасов меньше или равен *rn*, то размер заказа *un* выбирается равным *Rn* – *rn*. Было также установлено [3], что существуют пределы и , представляющие собой параметры оптимальных стационарных стратегий управления запасами. При этом номер шага *n* представляет собой так называемое «обратное» время, отсчитываемое от конца периода планирования. Иначе говоря, если период планирования *T* достаточно велик (*T* = *N*τ, где τ – интервал между моментами принятия решений о размере заказа), то с первых шагов (в обратном времени – *N*-м, (*N* – 1)-м, …) могут использоваться оптимальные параметры оптимальных стационарных стратегий управления запасами. Однако оказалось, что процедуры расчета как динамических параметров (*Rn*, *rn*), так и стационарных параметров (*R*, *r*) оптимальных стратегий управления запасами чрезвычайно вычислительно сложны и требуют прокрутки времяемких алгоритмов динамического программирования. Именно поэтому в качестве альтернативы были предложены существенно более простые «близорукие» (myopic) стратегии управления запасами, суть которых состоит в том, что в момент принятия решения о размере заказа учитываются не суммарные средние затраты, а средние затраты на ближайшем, примыкающем к текущему моменту временн*о*м шаге. В отличие от них стратегии, основанные на минимизации суммарных средних затрат в периоде планирования, будем называть «дальнозоркими».

В последнее десятилетие в связи с бурным развитием абсолютно прикладной науки, которая в нашей стране известна под названием «логистика» [49] – [50] [51] [52] и вобрала в себя значительную часть теоретического багажа математической теории управления запасами и производством, стало очевидным, что именно эти «близорукие» стратегии управления запасами оказались основным инструментом практического управления запасами. Однако сегодня с появлением мощных вычислительных средств представляется возможным вернуться к применению «дальнозорких» стратегий управления запасами, исследовав при этом как параметры (в том числе, и стационарные) этих стратегий зависят от стабильности рынка.

## Волновой эффект в цепях поставок

За два прошлых десятилетия проектирование цепей поставок было важной темой в области производства, исследования операций и управления цепями поставок. Большое количество подходов было предложено для проектирования цепей поставок в [53] – [54] [55] [56] [57] [58]. Как правило, оптимизация стоимости или уровня сервиса были включены в целевые функции. Во многих случаях модели учитывали уровень запасов, задержку поставок и колебания спроса [59] – [60] [61] [62].

Риски, связанные с неопределённостью спроса и предложения, провоцируются неопределённостями случайной природы и типичными для бизнеса ситуациями. Эти риски также известны как текущие или эксплуатационные риски [63] – [64] [65]. Управляющие цепями поставок достигли существенных результатов в управлении цепями поставок и смягчении текущих рисков посредством улучшенного планирования и исполнения [66]. Риски сбоев представляют собой новую проблему для управляющих снабжением, которые сталкиваются с волновым эффектом, возникающим в результате уязвимостей, нестабильности и сбоев в цепях поставок [67].

Волновой эффект описывает влияние сбоя на производительность цепи поставок и последующие структурные и параметрические изменения в цепи [67]. Вслед за возникшим сбоем, волны его воздействия распространяются по цепи поставок. Направление волны (и её воздействия) на экономические показатели зависит как от запасов надежности (к примеру, избыточность в форме буферных запасов), так и скорости принятия и масштаба мер, направленных на восстановление [68] – [69] [70] [71] [72]. Так как научное сообщество различает эксплуатационные риски и риски сбоев, волновой эффект может быть отнесён к рискам сбоев, в то время как эффект хлыста к эксплуатационным рискам. Различия представлены в табл. 1.

На стадии проектирования (прогностическое управление), разрабатываются резервные планы (например, альтернативные поставщики или маршруты) [73] – [74] [75] [76]. На стадии исполнения (управление по обратной связи) восстановление должно происходить быстро, ускоряя стабилизацию и адаптацию, чтобы гарантировать непрерывность цепи поставок и избежать долгосрочных последствий [66], [68] и [77].

|  |  |  |
| --- | --- | --- |
| **Параметр** | **Волновой эффект** | **Эффект хлыста** |
| Риски | Нештатные ситуации (к примеру, взрыв завода) | Операционные (к примеру, флуктуации спроса) |
| Подверженные области | Структуры и критические параметры (такие как, время доставки и уровень запасов) | Критические параметры, такие как упущенные продажи |
| Восстановление | Средне и долгосрочное; работы по координации и инвестированию | Краткосрочная координация, направленная на балансировку спроса и предложения |
| Подверженные показатели производительности | Итоговый показатель производительности, к примеру, годовой доход | Текущий показатель производительности, такой как потери в следствии дефицита за день |

Табл. . Сравнение волнового эффекта и эффекта хлыста.

В таких условиях жизненно необходимо расширить существующие модели проектирования цепей поставок, путём внедрения таких целевых функций, как гибкость, робастность, стабильность, упругость в мультикритериальные процедуры отбора проекта цепи поставок [64], [78] – [79] [80] [81]. Подобные исследования могут предоставить управленцам полезные инструменты для анализа цепи поставок на предмет производительности и устойчивости одновременно.

**Прогностический подход.**

Исследования [64], [67], [71], [72], [74], [76], [78] – [79] [80], [82] – [83] [84] [85], указали на то, что понимание и поиск проекта цепи поставок с эффективной комбинацией экономических показателей, сложности, робастности, гибкости, адаптивности и устойчивости, является многообещающей областью исследований с высокой практической применимостью.

**Смешанное целочисленное программирование** с применение надёжного проекта цепи поставок является широким полем для исследований на протяжении последних 10 лет. Точнее, в основном исследовались задачи о размещении объектов (*incapacitated fixed charge location model – UFL*) и задача *P*-медианы. Надёжная модель размещения была впервые представлена [78]. Модель *UFL* ставит задачу найти оптимальный проект цепи поставок с назначением потребителей к локациям с целью минимизировать сумму фиксированных и транспортных расходов в цепи поставок. В исследование [78] предполагалось, что оценки вероятности сбоев равны для всех узлов цепи поставки и был рассмотрен случай с 49 городами США.

Данная модель была расширена [74], [86] путём релаксации предположения об однородной вероятности сбоя. Вдобавок, в работе [74] обращено внимание на тот факт, что полная стоимость транспортировки в цепи поставок не должна увеличиваться после сбоя. Эта модель предоставляет решение без повышения транспортных расходов для обоих режимов работы: нормального и со сбоями. Как указано в работе [76], для задач средней размерности Лагранжева релаксация также позволяет найти оптимальное решение за разумное время.

Дальнейшее развитие моделей смешанного целочисленного программирования, в контексте волнового эффекта, может быть рассмотрено относительно укрепления объектов. В работе [84] рассматривается абсолютно надёжный запасной поставщик, к которому обращаются в случае, если основной поставщик уничтожен. Связанные затраты включены в целевую функцию, в то время как бюджет укрепления остался за её пределами. В работе [76] эта модель расширена посредством накладывания ограничений на бюджет укрепления в случае однономенклатурной задачи с восемью дистрибьюторами и при наличии многих покупателей (до 150).

Кроме того, соображения насчёт уровня запасов тоже были включены. В работе [87] представлена совмещённая модель «запасы-локация» при наличии вероятностного риска сбоев на объектах. В работе [75] рассматривается проект цепи поставок с ненадёжными поставщиками. Цель – минимизировать фиксированные суммарные затраты, состоящие из затрат, связанных с локацией, затрат на хранение запасов и затрат на страховой запас в центрах дистрибуции и стоимости заказа и транспортировки на всей цепи поставок. Модель нелинейного смешанного целочисленного программирования решается при помощи лагранжевой релаксации. Управление запасами при наличии сбоев в цепи поставок включает нелинейные компоненты стоимости. Зачастую используются эвристические методы решения. Кроме того, могут быть добавлены ограничения на неопределённость в сроках поставки [88], [89].

В работе [90] разработана исчерпывающая модель для постановки задачи с многими типами товаров и множеством периодов. Учитываются уровни запасов, задолженный спрос, доступные производственные мощности и трудовые резервы для каждого источника, транспортные мощности на каждом промежуточном узле доставки и доступные складские площади в каждом пункте назначения. Также принимается во внимание укрепление объектов путём введения запасного поставщика (с зарезервированными мощностями) и запасного промежуточного перегрузочного узла, что поможет удовлетворить покупательский спрос при более высокой цене, но без сбоев в цепи. Решение модели основывается на генетическом алгоритме с расстановкой приоритетов.

**Стохастическое программирование**. Необходимо различать модели классического стохастического программирования [91] – [92] [93], в которых спрос считается неопределённым параметром и модели робастного стохастического программирования [94], где вдобавок сбои на объектах и стоимость расширения объёмов рассматриваются как неопределённые. Недавно в работах [81], [95] и [96] была сделана попытка расширения существующих моделей путём одновременного рассмотрения неопределённостей, относящихся как к спросу, так и к предложению (запасам).

Так, в работе [75] рассматривается двухпериодная модель цепи поставок, в которой выбранные поставщики надёжны в первый период и могут сорвать поставку во второй. Соответствующая задача размещения объектов/надёжности поставщиков формулируется как задача нелинейного стохастического программирования. Авторы используют оптимизационный подход Монте-Карло вкупе с лагранжевой релаксацией. В работе [97] разработана модель стохастического программирования для: выбора интегрированных поставщиков, определения размеров заказов и составления расписания исполнения заказов покупателей при наличии риска сбоев в цепи поставок.

**Оптимизация нечёткая, робастная и целевая.** Работа [98] стала одной из первых по применению нечёткой оптимизации к проектированию цепей поставок с неопределёнными спросом и внешними поставками. Целью было определение уровня запасов и размеров заказов для установления приемлемых сроков доставки в цепи поставок при разумной итоговой цене. В работе [99] применено нечёткое математическое программирование с нечёткой целевой функцией, решаемой генетическим алгоритмом. По аналогии с [98] в работе [99] потребительский спрос и объём производства считаются неопределёнными. Цель заключалась в том, чтобы обеспечить адекватный компромисс между максимизацией прибыли и уровнем сервиса. В [100] представлен подход с использованием нечёткого программирования для стратегического проектирования распределительных сетей.

В работе [101] сформулирована стохастическая модель для нескольких объектов, которые обслуживают покупателей единственным продуктом, и при учете вероятности дефицита как случайного ограничения. С опорой на робастную оптимизацию выполнены численные эксперименты, чтобы проиллюстрировать производительность различных робастных формулировок. В [102] представлен подход с использованием вероятностного программирования для социально-ответственного проектирования цепи поставок.

**Моделирование, системный анализ и теория управления.** Моделирование зарекомендовало себя как подходящий инструмент для анализа проектов цепей поставок на предмет волнового эффекта. В работе [103] представлен подход, основанный на сети Петри, для моделирования распространения сбоев в цепи поставок и оценки влияния этих сбоев на производительность цепи поставок. Другое применение моделирования цепей поставок, основанного на сети Петри, представлено в работе [104] для оценки воздействия нескольких сценариев сбоев (сбои в спросе, транспортировке и качестве) и для поиска возможных путей восстановления производительности цепи поставок.

Метод Монте-Карло, основанный на обобщённом полумарковском процессе, используется для оценки сбоев, вызванных конкретным типом опасности в цепи поставок [105]. Эта модель оценивает распределение вероятностей потери производительности, вызванной возникновением опасностей в цепи поставок. В работе [106] представлена модель анализа сбоев в цепи поставок, основанной на раскрашенных сетях Петри (для лучшей визуальной репрезентации).

В работе [107] анализируются риски сбоев в портах ввоза с использованием вероятности и длительности закрытия, которые моделируются полностью моделируемой однородной марковской цепью. Разработана модель с периодической проверкой уровня запасов, которая показала (для опробованных сценариев), что маржа операционной прибыли может уменьшиться на 10 % в случае закрытия портов ввоза на разумное время или может быть полностью потеряна в случае полного отсутствия планирования непредвиденных обстоятельств. Средние издержки и штрафы могут увеличиться на 20 % из-за ожидаемой возросшей нагрузки на порты ввоза.

В работах [108] и [109] для динамического анализа цепи поставок применены многоагентные системы и принцип адаптивности. В [110] анализируются решения по увеличению производительности для продуктов с коротким жизненным циклом и непредсказуемо высоким спросом при помощи подхода системной динамики. Авторы работы [111] используют программу *AnyLogic* при моделировании цепи поставок в виде многоагентной системы для изучения сбоев со стороны поставщиков и стратегии восстановления в цепи поставок на уровне обслуживания.

В работе [112] выполнен анализ стабильности, основанный на мультиклассовой сети массового обслуживания. Изучены различные дестабилизационные факторы и сформулирована математическая программа, которая минимизирует необходимую производительность сети, при этом обеспечивая желаемый уровень робастности.

Для изучения влияния сбоев транспортировки на производительность цепи поставок авторы работы [113] разработали подход к управлению структурной динамикой для проектированию цепи поставок с одновременным рассмотрением нескольких составляющих частей цепи поставок (таких, как материалы, информация, товары, технологии и финансы) и их динамики. Представлены методы решения, основанные на комбинации оптимального управления и математического программирования.

В работах [114] и [115] исследуются проблемы устойчивости цепи поставок (путём повышения гибкости), а для анализа воздействия отсрочки на производительность цепи поставок используются модели массового обслуживания. В работе [116] для анализа производительности стратегии изготовления «под заказ» в условиях неопределённости применяются моделирование и оптимизация.

В работах [117] и [118] представлена модель для многопериодного и многономенклатурного проекта цепи поставок с рассмотрением структурной динамики. Оригинальность идеи этих исследований заключается в описании цепи поставок в виде нестационарной динамической системы управления вкупе с моделью линейного программирования. В отличие от задачи смешанного целочисленного линейного программирования, в данном случае статические и динамические параметры распределены между моделями линейного программирования и моделями управления.

**Подход, основанный на обратной связи.**

Инвестиции в защиту цепи поставок могут помочь избежать множества проблем, вызванных неблагоприятными событиями, однако полностью избежать сбоев невозможно. Таким образом, адаптация необходима для оперативного изменения планов, расписаний и стратегии управления запасами для достижения желаемого уровня производства [71].

Моделирование цепей поставок и операций с использованием системной динамики и методов классической теории управления (с применением дифференциальных уравнений) представляет большой интерес [119] – [120] [121]. Это вызвано тем, что многие из определяющих характеристик задачи можно без труда представить в динамической форме. Далее, используя широкий набор инструментов и методик, можно осмыслить в динамику системы. Динамические методы, также полезны тем, что могут послужить проводниками в область адаптации/восстановления.

В работе [64] представлена модель системной динамики для многоступенчатой цепи поставок. Были промоделированы различные сбои в транспортировке и оценено их влияние на темп выполнения потребительских заказов и флуктуаций запасов. Наиболее сильные воздействия связаны с нарушениями в процессе транспортировки от поставщиков первого уровня на склад.

В работе [83] при помощи моделирования получена количественная оценка риска сбоя в мультиэшелонной цепи поставок. Риск сбоя измеряется в неделях восстановления как усиление сбоя. В работе [122] проанализировано влияние сбоев на время доставки и на итоговые затраты цепи поставок при помощи симуляции, основанной на *ARENA*-модели.

В работе [123] рассмотрены сбои в информационной структуре, как возможные задержки информации и неполнота политики подачи заказов (для принятия решений, связанных с уровнем запасов) при помощи линейной теории управления. В [124] проанализированы динамические процессы в цепи поставок с точки зрения теории хаоса. В работе [125] промоделированы стратегии, включающие удовлетворение спроса из альтернативных локаций цепи; обеспечение материалов и транспорта из альтернативного источника или маршрута; и хранение стратегического резерва запасов на протяжении всей цепи.

Обширная область исследования адаптации цепи поставок – это управление с прогнозирующими моделями [126]. В управлении с прогнозирующими моделями, используется модель системы с измерением текущих процессов (и их истории) для предсказания поведения системы в избранные промежутки времени в будущем. Целевая функция, связанная с управлением, оптимизируется для расчёта последовательности управления, которая должна удовлетворить ограничениям системы.

Применение управления с прогнозирующими моделями к задачам мультиэшелонного производства/управления запасами уже рассматривалось в литературе. В работе [127] выполнено моделирование системы с несколькими заводами и несколькими товарами на базе прогнозирующих моделей с использованием различных уравнений и оптимизации расписания. В [128] описано децентрализованное управления с прогнозирующими моделями для задачи с сетью спроса из шести узлов, двух товаров и трёх эшелонов. В исследовании [129] была применена многоступенчатая стохастическая модель. В [130] анализируется производительность цепи поставок при наличии неопределённости параметров модели и спроса с учётом уровня сервиса цепи поставок.

Кроме теоретико-управленческого подхода в данной области применялись и другие инструменты. В работе [131] разработана модель, основанная на сценариях, вкупе со стратегией адаптивной маршрутизации. В [132] используется нечёткая программа оценки и предложен метод обзора для расчёта времени завершения операций цепи поставок в случае серьёзного сбоя. В [133] рассматриваются сбои в снабжении и применяются реактивные стратегии в цепи поставок с исполнением «под заказ» с чувствительным ко времени спросом. В [117] исследуется возможность реконфигурации транспортировки в случае сбоя внутри цепи поставок в проект цепи поставок в многопериодную модель, основанную на комбинации моделей линейного программирования и оптимального управления. В [111] разработан подход к предсказанию устойчивости цепи поставок путём введения мер восстановления, который использует аналогию с биологическими клетками, имеющими способность к самоадаптации и самовосстановлению. В [134] рассмотрены серии сбоев в течении долгого времени и представлена модель управления запасами для того, чтобы выработать оптимальную политику восстановления управления в внештатной ситуации в реальном времени для двухступенчатой системы серийного производства с учётом надёжности. Рассмотрены несколько сбоев и случаев, в которых последующие сбои могут или не могут повлиять на план восстановления после более ранних сбоев.

**Типы рисков.**

Проанализированная литература предлагает четыре базовых типа рисков сбоя, которые должны учитываться управляющими цепью поставок:

* Сбои при производстве и транспортировке, особенно в глобальных цепях поставок.
* Риски сбоев, связанные с товаром и узкой специализацией поставщика.
* Сбои потока информации.
* Сбои в финансовых потоках.

Во-первых, глобализация и склонность к аутсорсингу усложняют цепи поставок и делают их менее прозрачными и управляемыми. В соответствии с теорией сложности, такие системы становятся более чувствительны к сбоям. Особое внимание в этой области уделяется сбоям в транспортных каналах. Во-вторых, парадигмы эффективности бережливого производства, единственного источника и т.д. показали себя не пригодными в ситуациях сбоев. Как следствие, цепи поставок стали более уязвимы даже для слабых возмущений. Любое возмущение в глобальной цепи поставок, особенно в части её снабжения, немедленно влияет на всю цепь поставок. В-третьих, с возросшими специализацией и географической концентрированностью производств, возмущения в одном или нескольких узлах влияют на практически все узлы и связи в цепи поставок. В-четвёртых, информационные технологии (ИТ) стали важным элементом глобальных цепей поставок, так как сбои в ИТ могут оказать серьёзное влияние на нарушение материальных потоков [135] и [136]. В-пятых, проблемы координации контрактов, расходов и прибыли важны для анализа рисков сбоев в цепях поставок.

**Прогностическая защита цепей поставок.**

В недавних публикациях обсуждались различные стратегии смягчения рисков. Шесть элементов защиты цепей поставок (прогностическое управление) можно классифицировать как:

* Запасные поставщики, депо и каналы транспортировки.
* Буферы запасов и объёмов производства.
* Локализация цепей поставок и сегментация.
* Гибкость товара и процесса.
* Координация и заключение контрактов.
* Резервные ИТ-системы.

**Стратегии восстановления цепей поставок, основанные на обратной связи.**

Управление по время внештатной ситуации может быть осуществлено 4 основными способами (в зависимости от тяжести ситуации):

* Параметрическая адаптация.
* Адаптация процесса и товара.
* Структурная адаптация.
* Системная адаптация.

Параметрическая адаптация представляет собой простейший случай, когда добиться стабилизации и восстановления можно путём настройки некоторых критических параметров (например, времени доставки или уровни запасов). Адаптация процесса и товара относиться к запасам гибкости. Структурная адаптация рассматривает резервного поставщика в рамках планирования транспортировки в случае непредвиденных обстоятельств. Системная адаптация является высшим уровнем адаптации, при которой должны быть восстановлены стратегия и организация.

Подходы, основанные на обратной связи, могут строиться, либо исключительно на восстановительных стратегиях (без какой либо прогностической защиты цепи), либо интегрировано использоваться вместе с прогностическими подходами. Рассмотрим второй случай. Многие прогностические методы содержат, на самом деле, элементы подхода, основанного на обратной связи. Формулировки задач смешанного целочисленного программирования с учётом укрепления объектов инфраструктуры рассматривают переориентацию на резервных поставщиков в случае сбоя у основного поставщика. При управлении с прогнозирующими моделями применяется стратегия поэтапного планирования, явно включающая элементы перепланирования. Модели управления запасами тоже предлагают стратегии восстановления. При моделировании рассматриваются «что-если»-сценарии, которые могу быть использованы управляющими цепью поставок в случае возникновения внештатной ситуации для быстрой оценки стратегий восстановления и влияния на эксплуатационные и финансовые показатели.

Модели смешанного целочисленного программирования дают управленцам интересные подсказки и могут быть использованы в случаях, когда вероятность сбоя может быть довольно точно определена. Большинство решений моделей смешанного целочисленного программирования предлагают открытие новых объектов. Это увеличивает общие расходы, даже если стоимость транспортировки не возросла.

Тем не менее, как было отмечено в статьях [66] и [77], – почти невозможно определить вероятность возникновения пожаров на фабриках, природных катаклизмов или пиратства в определённом регионе. Поэтому следует концентрироваться, в основном, на стратегиях смягчения и определения влияния возмущения на финансовые и эксплуатационные параметры вне зависимости от природы возмущений.

Вдобавок, общий недостаток существующих исследований, как было отмечено в [74] и [76], заключается в том, что не учитывается динамика процессов в цепи поставок. Сбои, в основном, считаются статическими событиями, без учёта их продолжительности и стратегий стабилизации/восстановления.

Аналогично моделям смешанного целочисленного программирования, предположения касательно известной надёжности поставщика и параметрических вероятностей, делают модели стохастического программирования сложнее в управлении и применении. Кроме того, подход, основанный на сценариях, экспоненциально увеличивает количество переменных и ограничений в формулировке стохастической модели. Для ознакомления с некоторыми практическими проблемами и решениями в этом направлении, см. [137].

Зачастую применение нечёткой или робастной оптимизации относится к эксплуатационным рискам (к примеру, флуктуации спроса) и уровню тактического планирования с эпизодическими отсылками к проектированию цепей поставок. То же самое можно сказать и про управление с прогнозирующими моделями. Кроме того, необходимо отметить общий недостаток робастной оптимизации – склонность к достаточно пессимистическому решению. На практике, тяжело представить, что управляющие утвердят проект цепи поставок с низкой эффективностью и высокой фиксированной стоимостью только из-за ожидания худшего развития событий.

## Методы синтеза алгоритмов управления в условиях нестационарности и неопределенности

Основной результат теории управления запасами заключается в том, что в рамках достаточно общих предположений о распределении спроса (лишь бы он оставался стационарным случайным процессом) структура функции суммарных средних затрат обеспечивает оптимальность двухуровневых (*R*, *r*)-стратегий управления запасами [3]. Это означает, что если до конца периода планирования осталось *n* моментов принятия решений о размере заказа (их называют шагами), то для любого *n* можно указать такие два числа *Rn* и *rn* (*Rn* > 0 и *rn < Rn*), что решение об оптимальном размере заказа *un*(*xn*), где *xn* – текущий уровень запасов в момент принятия решений, выбирается по следующему правилу: если *xn*> *rn*, то заказ не подается вообще, а если *xn*≤ *rn*, то подается заказ размера *Rn* – *xn*, то есть размер заказа определяется следующей формулой:

|  |  |
| --- | --- |
|  | (1) |

Этот результат оказался очень удобным для приложений, поскольку задавал предельно простое правило определения момента подачи заказа на пополнение запасов. Именно так действовали и прежде менеджеры торговых и производственных компаний. Только два этих уровня *Rn* и *rn* они считали по своим собственным, основанном на их личном опыте, правилам. Иначе говоря, практические специалисты по логистике запасов обычно применяют параметрические стратегии управления запасами. На практике используются различные параметрические стратегии, каждая из которых является разновидностью (*R*, *r*)-стратегии. Среди них: (а) сами (*R*, *r*)-стратегии, (б) (*R*, *r*,*τ*)-стратегии, когда решения о подаче заказов принимаются периодически с периодом *τ*, (в) (*Q*, *r*)-стратегии, когда размер заказа фиксирован и равен *Q*, (г) (*Q*, *r*,*τ*)-стратегии, (д) (*nQ*, *r*,*τ*)-стратегии, когда размер заказа должен быть кратен величине *Q* и другие.

Формирование зависящих от номера шага *n* параметров стратегий *Rn* и *rn* требует использования достаточно времяемких процедур решения уравнения динамического программирования [15]:

|  |  |
| --- | --- |
|  | (2) |

где – значение суммарных средних затрат (с учетом переоценки) для *n*-шагового процесса, который начинается с фиктивного уровня запаса *x* (т.е. уровня запаса с учетом задолженного спроса и «запасов в пути», см. [3]) и ведется с использованием оптимальной стратегии управления запасами, *A* – фиксированные затраты на одну поставку, **1**(**.**) – функция единичного скачка (функция Хэвисайда), *c* – закупочная цена единицы товара, *g*(*x*,*u*) – среднее значение так называемых одношаговых затрат, α – коэффициент дисконтирования (0 ≤ α < 1), *F*(*z*) – распределение спроса. Функция *g*(*x*, *u*) задаётся следующей формулой:

|  |  |
| --- | --- |
|  | (3) |

где *h* – это удельные затраты на хранение продукции (стоимость хранения единицы продукции в единицу времени), а *d* – это удельные издержки вследствие дефицита (финансовые потери от существования единичного дефицита в течение одной единицы времени).

Из-за сложности процедур решения уравнений динамического программирования нередко от них отказываются, подменяя критерий минимума суммарных средних затрат на критерий минимума средних затрат только на очередном шаге – шаге принятия решений. Тогда вместо приведенного ранее уравнения динамического программирования появляется одно единственное статическое уравнение:

|  |  |
| --- | --- |
|  | () |

Соответствующие стратегии называются «близорукими» в отличие от «дальнозорких» стратегий, строящихся с помощью уравнения (2). В подавляющем большинстве современных пакетов прикладных программ для решения задач управления производством и запасами считаются так или иначе проинтерпретированные параметры «близоруких» стратегий. И дело тут не только в том, что такой расчет значительно проще, но и еще в нескольких причинах. Во-первых, это оправдано тем, что позволяет в условиях неопределенности (когда, статистические характеристики случайных процессов заранее неизвестны, о чем более подробно речь пойдет чуть позже) воспользоваться достаточно простыми адаптивными алгоритмами оценки параметров оптимальных стратегий управления запасами. Эквивалентного универсального инструмента для «дальнозорких» стратегий пока еще нет. Во-вторых, в реальной складской системе стохастические процессы (например, спрос), как правило, нестационарны. Поэтому возникают немалые сомнения в целесообразности применения динамических моделей типа уравнения (2) для стационарных случайных процессов. Тут действует известный принцип: «лучше синица в руках, чем журавль в небе». О том, во что может обойтись такая «синица», говорится в разделе 2.2, посвящённом сравнению затрат в случае использования «близоруких» и «дальнозорких» стратегий.

Кстати, еще одним фундаментальным результатом теории управления запасами является существование пределов и . Это означает, что при достаточно большой длине периода планирования *T = Nτ* (*N*  → ∞) вместо зависящих от времени параметров стратегий *Rn* и *rn* можно воспользоваться их предельными значениям *R* и *r*. Иначе говоря, параметрами стационарных «дальнозорких» стратегий управления запасами. Не является ли это серьезным аргументом в пользу того, чтобы работать с критерием суммарных средних затрат (2), а не с «близоруким» одношаговым критерием (3)?

На этом пути отметим вышедшую в 70-е годы работу Н. Прабху [4], который первым подметил возможность использования методов теории восстановления для решения задач теории управления запасами. В самом деле, теория восстановления – это прикладной раздел теории вероятностей, посвященный исследованию свойств сумм случайных величин. В теории восстановления – объектом изучения являются суммы времен работы последовательно заменяемых элементов. Аналогичные суммы случайных величин неявно фигурируют и в задачах теории управления запасами и представляют собой значения суммарного спроса в интервалах между очередными поставками в систему снабжения. На базе этой аналогии Н. Прабху предложил использовать еще один альтернативный критерий в задачах управления запасами, описывающийся следующим функционалом:

|  |  |
| --- | --- |
| , | (5) |

где *G*(*x*) – стационарное распределение уровня запасов в системе снабжения.

На базе этой модели были построены адаптивные алгоритмы расчета параметров стационарных «дальнозорких» стратегий [30]. Однако нельзя не отметить одной особенности модели (5), которая вынуждает с осторожностью относиться к полученному Прабху результату, а, стало быть, и синтезированным алгоритмам.

Дело в том, что, как следует из ранее сказанного, стационарные в смысле модели (2) параметры оптимальной предельной стратегии управления запасами можно использовать при управлении запасами с первых шагов достаточно продолжительного по числу шагов периода планирования, поскольку в модели (2) используется так называемое «обратное» время, то есть время, оставшееся до конца периода планирования. А в модели (5), предложенной Прабху, стационарность распределения уровня запасов понимается в обычном для теории случайных процессов прямом времени. Итак, времена для моделей (2) и (5) «разнополярны»!

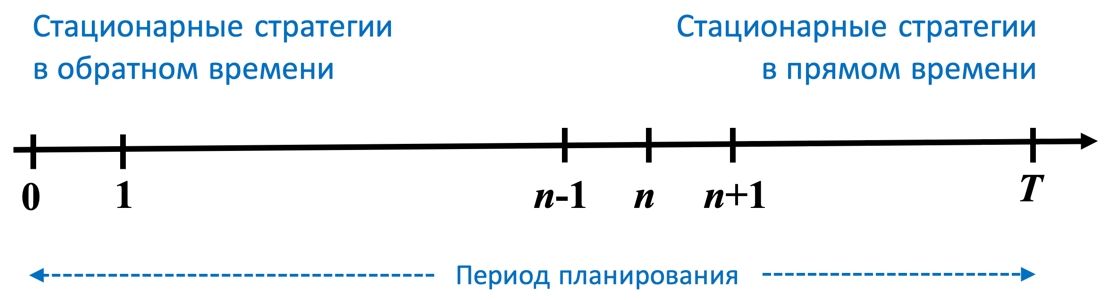


Рис. . Смена стратегий в периоде планирования.

Чтобы соотнести эти времена, А.С. Манделем было проделано специальное исследование [138], результаты которого суммированы в следующей теореме:

**Теорема 1** [15]. Решение задачи (5) задает двухуровневую стационарную стратегию управления запасами с параметрами *R* и *r*, которые определяются в результате решения задачи (2) при N → ∞ и α → 1.

Из теоремы ясно, что использование результатов, полученных по модели Прабху, для построения адаптивных алгоритмов вычисления оптимальных параметров стационарной (в обратном времени) «дальнозоркой» стратегии управления запасами корректно только в том случае, когда коэффициент дисконтирования будущих затрат *α* равен 1. Что из этого следует? А то, что модель (5) гораздо более чувствительна к отклонению встроенных в модель случайных процессов от стационарности. Мало того, что в этом случае распределение спроса не должно меняться во времени, так еще и внешняя среда должна быть абсолютно стационарна, да еще с инфляцией, равной 0.

И, сделав этот вывод, перейдём, наконец, к рассмотрению задачи управления запасами в условиях неопределенности.

Итак, далее будет рассматриваться ситуация, когда исследуемая с целью ее оптимизации система снабжения функционирует в условиях неопределенности и группа исследователей операций, которой доверено синтезировать эффективную систему управления запасами также действует в этих условиях. То есть в условиях, когда статистические свойства случайных процессов, развивающихся в рассматриваемой многономенклатурной системе снабжения, априори неизвестны, причем характер протекания этих процессов принципиально нестационарен и вбирает в себя не только такие источники нестационарности, как внутринедельные «качания» и сезонность спроса, но и различные источники дополнительных случайных вариаций, включая макроэкономические изменения на рынке.

За последние 15 лет удалось сформировать [139], [140] достаточно общую схему действий, которую можно использовать в качестве общей схемы решения задач управления запасами в условиях неопределенности. Рассматриваемая схема состоит из шести основных блоков, которые перечислены ниже:

1. Классификации типов товаров по группам при помощи *ABC*-анализа.
2. Прогнозирование трендов изменения спроса (экспертно-статистический подход и разработанный в его рамках метод аналогов).
3. Решение детерминированной задачи управления многономенклатурными запасами по выявленным трендам спроса (вариационные и комбинаторные задачи оптимизации).
4. Исследование компонентов, формирующих стохастический спрос.
5. Предварительная минимаксная оценка страховых запасов (робастные и адаптивные алгоритмы управления).
6. Алгоритмы управления страховыми запасами (метод фильтрации).

Эволюцию современной теории управления можно представить в виде дерева. Дальнейший рост этого дерева требует, во-первых, совершенствования специфических инструментов научного анализа для каждой из ветвей, а, во-вторых, освоения методик и техник, наработанных на более разросшихся верхних сучьях дерева теории. Для теории управления запасами и производством это означает умение заимствовать наиболее совершенные методы и алгоритмы управления техническими системами.

Отметим, что еще 20 лет назад в связи с широким распространением гибких производственных систем дебатировался вопрос о том, нужны ли запасы вообще. Не правильнее во всем жить по принципу *Just in Time*. Споры эти постепенно стихли, поскольку жить без запасов может и правильнее, но жизнь такая абсолютно невозможна. Во-первых, никакого другого способа, кроме создания запасов, для согласования разнотемповых производств не существует. Во-вторых, мы живем в стохастическом мире, в котором жесткому планированию поддается далеко не все. Что, например, делать с модой и изменяющимися со временем вкусами людей, которые неожиданно рождают спрос на все новые и новые виды товаров. Такая среда не только стохастична, но и плохо предсказуема и эволюционирует в условиях неопределенности.

И даже одна организация, например, завод или большой склад живут как иерархическая мультиагентная система, поскольку разные департаменты, управления и отделы одного и того же завода имеют разные целевые функции. И, скажем, если управление сбыта заинтересовано в больших запасах готовой продукции, то финансово-экономическому управлению это категорически не нравится.

Итак, вернёмся к решению задачи управления запасами и производством в условиях неопределенности. Преобразовав традиционные критерии в задаче управления запасами к той форме, которая позволила применить для их решения методы управления техническими системами, мы пришли к следующей уже продемонстрированной 6-этапной схеме. Далее последовательно рассмотрим содержание и методы решения задач, возникающих на каждом из этих этапов.

*ABC*-анализ является важным инструментом решения задач управления запасами (многономенклатурными) и задач логистики. Суть *ABC*-анализа заключается в том, чтобы разбить все типы товаров на группы (3, как правило) так, чтобы в группу *A* вошли наиболее значимые в стоимостном выражении товары (небольшого числа типов), в группу *B* – среднее количество типов «средних» по цене товаров, а в группу *C* – оставшиеся. В итоге, группа *A* оказывается заполнена 5–10% типами товаров, стоимостной вклад которых в суммарном обороте составляет 40–70% от общей суммы средств, затраченных на складские запасы. К группе *B* относят 15–30% товаров, общая стоимость которых может варьироваться в пределах 15–25% затраченных средств. В последнюю группу, *C* – включают примерно 65–80% типов товаров, причем так, что в суммарном выражении доля их стоимости равняется 5–15% [139]. Хорошее решение задачи *ABC*-анализа напрямую влияет на эффективность работы автоматизированных систем управления снабжением, так как даёт возможность дифференцированно применять различные алгоритмы/схемы для сбора данных, их обработки и управления товарами, которые относятся к разным группам (от наиболее дорогостоящих и времяёмких для группы *A* до самых дешёвых и непритязательных для группы *C*). Самые разные методы применяются для решения задачи *ABC*-классификации товаров - от чисто эмпирических до хорошо формализованных (кластеризация, автоматическая классификация, метод *k*-средних и другие) [141].

Переходя ко второй проблеме – проблеме прогнозирования трендов, – заметим, что ее решение, особенно для новых видов товаров, связано с чрезвычайно малыми объемами исходной статистической информации вплоть до полного ее отсутствия. Предлагается использовать для ее решения разрабатываемые в институте методы экспертно-статистической обработки информации. В данном случае – это метод аналогов. Суть процесса экспертно-статистической обработки информации заключается в том, чтобы, образно говоря, «на равных» употребить всю доступную об объекте управления информацию и объективного, и субъективного происхождения [142]. Для каждой прикладной проблемы, и, в частности, в логистике, это требует разработки специальных математических моделей, часть которых (для описания предметной области) мы называем базисными, а часть – интеграционными, рассчитанными на объединение результатов приборных измерений и информации экспертного происхождения в рамках единой системы обработки данных.

Напомню, что в данном случае речь идет о прогнозировании временных рядов по очень коротким выборкам. Ниже приведены условия, при выполнении которых для предсказания можно использовать экспертно-статистический метод аналогов [143]:

* Очень короткие временные выборки;
* Прогнозирование должно быть быстрым;
* Имеются достаточно полные данные о реализациях для объектов-аналогов;
* Команда специалистов с опытом в исследуемой области.

Идея метода аналогов состоит в том, чтобы, опираясь на анализ базы данных о спросе на разные товары и об атрибутах этих товаров, осуществить кластеризацию многомерного пространства характеристических признаков товаров. Это создает основу для того, чтобы при прогнозировании спроса на новые виды изделий можно было воспользоваться историями продаж других товаров, которые являются аналогами впервые выпускаемого на рынок продукта.

Переходя к решению третьей проблемы – детерминированной задачи управления запасами по выявленным трендам спроса, – отметим, что её удалось свести к вариационным и комбинаторным задачам оптимизации. В результате были существенно обобщены и развиты классические инструменты теории управления запасами и, в частности, классическая формула экономичного размера партии поставки (*economic order quantity*).

Однако для того, чтобы проблема оптимального управления запасами в условиях неопределенности была решена полностью, необходимо синтезировать средства компенсации случайных отклонений от трендов. В данном случае это сводится к созданию компенсационных или, как называют их экономисты, страховых запасов (*safety stocks*). И разработка методов их расчета составляет содержание трех последних этапов общей схемы управления запасами в условиях неопределенности. Научная основа одного из них заложена приведенной ранее теоремой.

Впрочем, чтобы рассчитать эти параметры, надо знать распределения вероятностей, а мы работаем в условиях неопределенности. И благодаря упомянутой ранее трансформации классических критериев, удалось построить адаптивные алгоритмы расчета оптимальных параметров двухуровневой стратегии управления запасами (правда, только для условий безинфляционного развития) [144]. Выигрыш по уровню суммарных средних затрат от использования стационарных «дальнозорких» стратегий управления запасами по сравнению с более простыми «близорукими» (одношаговыми) стратегиями может достигать 25%.

Возможность применения таких адаптивных алгоритмов отнесём к так называемому квазистационарному случаю, когда степень нестационарности, протекающих в системе случайных процессов, не слишком велика: если справедливо считать, что динамические изменения статистических свойств, скажем, потребительского спроса становятся заметными только после 15–20 шагов процесса управления, который в данном случае реализуется как процесс принятия управленческих решений.

В общем случае для предварительной оценки компенсационных запасов воспользуемся результатами, полученными в [15]. Вопрос обеспечения малой вероятности того, что система окажется перенасыщенной товаром или «провалиться» в дефицит, решается при создании страховых запасов. При этом суммарные затраты уже были минимизированы на этапе решения задачи (детерминированной) управления запасами по трендам. Рассматриваемые случайные отклонения от трендов, колеблют систему управления запасами в пределах 5-10% от минимизированной на прошлом шаге суммы затрат. Это говорит о том, что на этапе создания страховых запасов можно использовать критерии, обеспечивающие заданные уровень обслуживания потребителей на каждом шаге периода планирования, вместо затратных критериев [140].

Пусть **z***t* – вектор случайных отклонений от тренда (спроса) на шаге *t* (прямого времени): , где  – спрос на продукт *i*, , на шаге *t.* T обозначает знак транспонирования. Тогда страховой (дополнительный) запас , где  – страховой запас продукта *i*,  на шаге *t*, должен быть выбран таким образом, чтобы следующая вероятность была максимальной

|  |  |
| --- | --- |
| , | (6) |

где  – вектор запаса на шаге *t*, удовлетворяющий следующему бюджетному ограничению:

|  |  |
| --- | --- |
| , | (7) |

где  – вектор цен товаров (хранящихся на складе), а Φ – бюджетное ограничение. Трудности решения задачи (6)-(7) обсуждаются в [15].

Другой вариант критерия:

|  |  |
| --- | --- |
| , | (8) |

вместе с ограничением (7).

Решение задачи (7)-(8) обладает важным свойством:

|  |  |
| --- | --- |
| . | (9) |

Другим свойством решения этой задачи является то, что оно зависит только от дисперсий и средних значений одномерных распределений спроса. Данные о функционировании склада (поступающие в процессе его эксплуатации) могут быть использованы для оценки (хотя бы грубой) этих величин.

Тогда, если записать задачу стабилизации (по критерию минимума дисперсии флуктуации процесса относительно заданного нормативного уровня) компенсационных запасов в предлагаемой выше форме, то выяснится, что выполнены все условия применимости для ее решения алгоритмов типа калмановской фильтрации [145].

Пусть **K** – матрица ковариации, введённая для описания векторного спроса (как авторегрессионного процесса первого порядка с произвольными межноменклатурными корреляциями). Рассмотрим задачу стабилизации страховых запасов (см. также [6]), то есть задачу поддержания дополнительных запасов на уровне максимально близком к *st*при неизвестном распределении спроса.

|  |  |
| --- | --- |
| , | (10) |

где **u***t* – это размер заказа, а E – оператор вычисления математического ожидания.

Введём матрицу **H** для описания авторегрессионного процесса:

|  |  |
| --- | --- |
| **H** =. | (11) |

Пусть **A** = **HK**-1, **B** = [**K** – **HK**-1**H**T] -1/2 и E**z***t* = **m***t*. Запишем модель формирования спроса:

|  |  |
| --- | --- |
| , | (12) |

и уравнение наблюдений:

|  |  |
| --- | --- |
| , | (13) |

где  и  – стационарные белые шумы.

Запишем оптимальный размер заказа (страхового), используя гипотезу разделимости:

|  |  |
| --- | --- |
| , | (14) |

где  – калмановская оптимальная оценка вектора спроса:

|  |  |
| --- | --- |
| , | (15) |

а

|  |  |
| --- | --- |
| , | () |
|  | () |

и

|  |  |
| --- | --- |
| . | () |

Каким образом оценить эффективность предложенной схемы? И, вообще, насколько применимы те критерии оптимальности, которые в большинстве случаев сводятся к вычислению математических ожиданий затрат на достаточно больших интервалах времени, в условиях нестационарности развивающихся в системах снабжения случайных процессов?

Чтобы ответить на этот вопрос, сделаем одно важное замечание. В условиях нестационарности нам крайне редко будет удаваться вычислять эти математические ожидания теоретически. Только для каких-то предельно простых и, стало быть, малоинтересных практически случаев. Поэтому, как нам представляется, единственным конструктивным инструментом оценки средних значений суммарных затрат становится выполнение достаточно объемного имитационно-экспериментального моделирования системы снабжения при использовании действующих и вновь предлагаемых правил управления.

Всё выше сказанное позволяет сделать предположение о возможности развития для соответствующих систем стохастических дифференциальных уравнений декомпозиционного метода анализа типа метода малого параметра. Это можно назвать первой из еще не решенных проблем. Разработка универсального имитационно-экспериментального стенда для оценочного моделирования стратегий управления запасами можно обозначить как вторую проблему. Третья нерешённая проблема – это синтез универсальных адаптивных алгоритмов для расчета параметров оптимальных «дальнозорких» стратегий управления запасами. Четвертой проблемой можно назвать разработку и существенное расширение множества статистических критериев для проверки гипотез о степени нестационарности наблюдаемых в системах снабжения случайных процессов.

## Выводы по главе 1

1. Описанный в главе волновой эффект способен выступить в роли явления, которое может объединить исследования в области управления нештатными ситуациями при управлении запасами в цепях поставок и в области восстановления от нештатных ситуаций. Это задает повестку для будущих исследований в области динамики, управления, непрерывности и преодоления сбоев при управлении запасами в цепях поставок, делая их более робастными, гибкими и прибыльными.
2. Наука исследования операций вкупе с анализом системной динамики и использованием методов общей теории управления позволяет сформулировать ряд полезных методов, которые могут быть использованы для изучения волнового эффекта и уменьшения негативных последствий от его проявления. Для различных задач могут быть разработаны конкретные методы решения, но ни один из них не станет панацеей.
3. Математические методы решения детерминированных и стохастических задач оптимизации запасов заняли свою нишу на стратегическом и тактическом уровнях планирования и проектирования цепей поставок и систем управления запасами. Однако они не способны в полной мере раскрыть динамику процессов, протекающих в цепях поставок и в системах управления запасами. Результаты, достигнутые при помощи стратегического проектирования цепей поставок и тактического планирования производительности на этапе восстановления, могут быть усилены использованием моделей, основанных на динамике происходящих в системе процессов.
4. Для задачи управления многономенклатурными запасами в условиях неопределенности и нестационарности проанализирован общий подход к её решению. Рассмотрена многоступенчатая процедура решения задачи, суть которой заключается в том, чтобы: (а) упорядочить множество всех видов товаров посредством их классификации при помощи *ABC*-анализа; (б) затем для каждого класса товаров осуществить выделение трендов (включающее, в том числе, и сезонные компоненты спроса), на основе которых решается детерминированная многономенклатурная задача управления запасами; (в) далее исследуется проблема формирования дополнительных (страховых) запасов с целью компенсации случайных флуктуаций спроса при отклонении от ранее выделенных трендов (для решения последней задачи разобран подход, основанный на применении фильтра Калмана).

# Стационарные стратегии управления запасами при случайном спросе в условиях параметрической неопределенности

## Базовая модель оптимального управления запасами при случайном спросе

Рассмотрим модель управления однономенклатурным запасом с запоминанием задолженного спроса[[2]](#footnote-2) в течение периода планирования *T* = (0, *N*τ), где *N* – достаточно большое натуральное число. Спрос описывается моделью независимых в совокупности, одинаково распределенных случайных величин { *z*(*n*), *n* = 1, 2, …, *N* } с функцией распределения *F*(*z*).

Такой моделью, которую в дальнейшем мы будем называть базовой, хорошо описываются системы управления запасами продуктов массового потребления, спрос на которые не подвержен (или мало подвержен) сезонным изменениям. Среди таких продуктов можно назвать основные продукты питания (хлебобулочные изделия, мясо, молочные продукты и т.д.), лекарственные средства от хронических заболеваний и многие-многие другие.

Пусть время запаздывания поставки θ = 0[[3]](#footnote-3), то есть время от момента подачи заказа до его поступления на склад. Тогда, если в качестве критерия планирования принят минимум суммарных средних затрат в периоде планирования для оптимальной стратегии пополнения запасов, можно записать уравнения динамического программирования [3], [15]:

|  |  |
| --- | --- |
|  | (19) |
|  | (20) |

Здесь – минимально возможное значение суммарных средних затрат (с учетом переоценки) для *n*-шагового процесса, который начинается с фиктивного уровня запасов[[4]](#footnote-4) *x* и ведется с использованием оптимальной стратегии управления запасами.

**Замечание**. Отметим, что в уравнениях (19) и (20) в отличие от процитированных источников и вообще от классической теории управления запасами в пределах интегрирования появилась конструкция max{*x* + *u*, 0} вместо привычного *x* + *u*. Это объясняется тем, что расчеты по прежним формулам (с пределом интегрирования *x* + *u*) при использовании фиктивного уровня запасов приводили к ошибкам, поскольку при отрицательном значении *x* + *u* (чего нельзя исключать) интеграл принимает отрицательное значение, что невозможно, так как *h* × *Ih* – это размер издержек хранения, и тогда получилось бы, что в отсутствие запаса плата за хранение возвращается плательщику[[5]](#footnote-5). Подобная корректировка никак не отражается на основных выводах теории управления запасами (сохраняются свойства *A*-выпуклости и оптимальность двухуровневых стратегий управления запасами), поскольку, как нетрудно убедиться, при такой модификации записи интегралов производные от подынтегральных выражений по *u* остаются непрерывными.

Теперь перепишем уравнения (19) и (20), введя функции:

|  |  |
| --- | --- |
|  | (21) |

Известно [48], что решение уравнения динамического программирования (19)-(20) принадлежит к классу так называемых (*R*,*r*)-стратегий управления запасами. Техника доказательства опирается на понятие *A*-выпуклости [48] и введение вспомогательной функции , которая задается следующей формулой:

|  |  |
| --- | --- |
|  | (22) |

где *n* = 1, 2, …, а начальное условие 0.

Учитывая функции (21) и (22), уравнения (19) и (20) можно переписать в виде:

|  |  |
| --- | --- |
|  | (23) |

Ясно, что если *A* задано, то оптимальная стратегия функционирования будет зависеть только от формы кривой . Предположим, что кривая имеет вид, показанный на рис. 4. Именно такие кривые называются *A*-выпуклыми (в данном случае при *A =* *A*1).

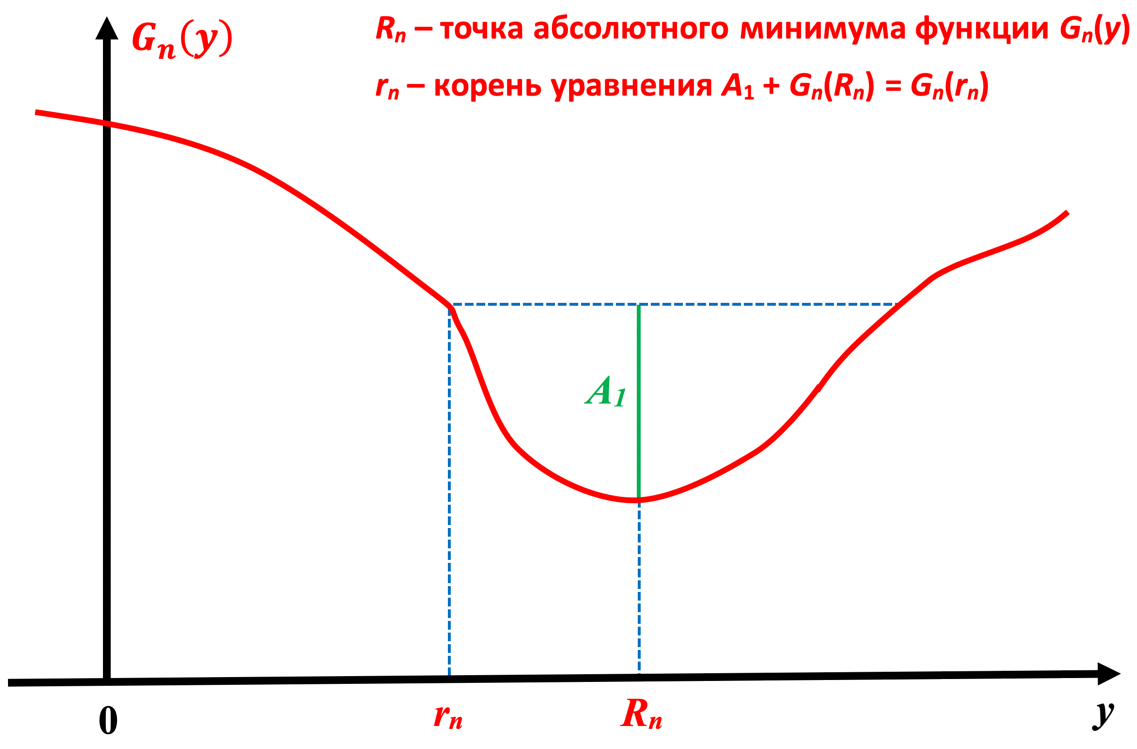


Рис. . График функции *Gn*(*y*).

Известно [3], [15], что в системе, оптимизация которой задается уравнением (23), оптимальны двухуровневые (*R*, *r*)-стратегии управления запасами. Это означает, что для каждого момента времени *n* (при отсчете времени от конца периода планирования) существует пара чисел *Rn*, *Rn* > 0, и *rn*, *rn < Rn*, таких, что правило подачи заказов *u*(*x*) за *n* шагов до конца периода планирования может быть задано формулой (1).

Повторим здесь эту формулу и дадим ей эконометрическую интерпретацию:

|  |  |
| --- | --- |
|  |  |

где *x* – уровень запасов в момент принятия решений (начало шага) о размере заказа , *r* – точка подачи заказа (сокращенно, точка заказа), то есть тот критический уровень запасов, пересечение которого сверху вниз требует подачи заказа размера , а *R* – это уровень пополнения запаса, то есть максимальный уровень запаса продукта в системе, который достигается после прихода поставки. Это иллюстрируется следующим рисунком.

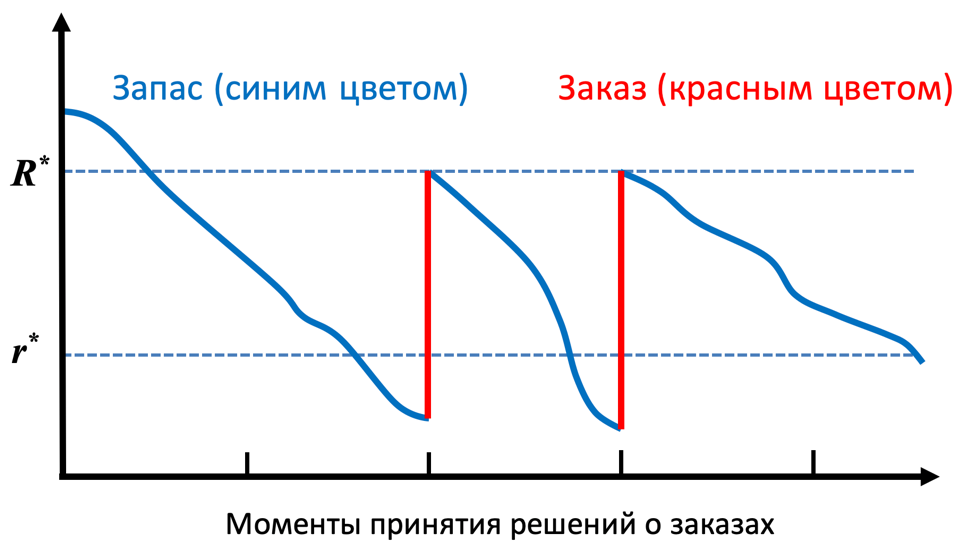


Рис. . Принятие решений о размере заказа.

Если в формуле (1) выбраны оптимальные по какому-то критерию значений параметров стратегии *R*\* и *r*\*, то график оптимального размера заказа *u*\*(*x*), как функции от текущего запаса *x* в момент принятия решений, будет выглядеть так, как показано на рис. 5.

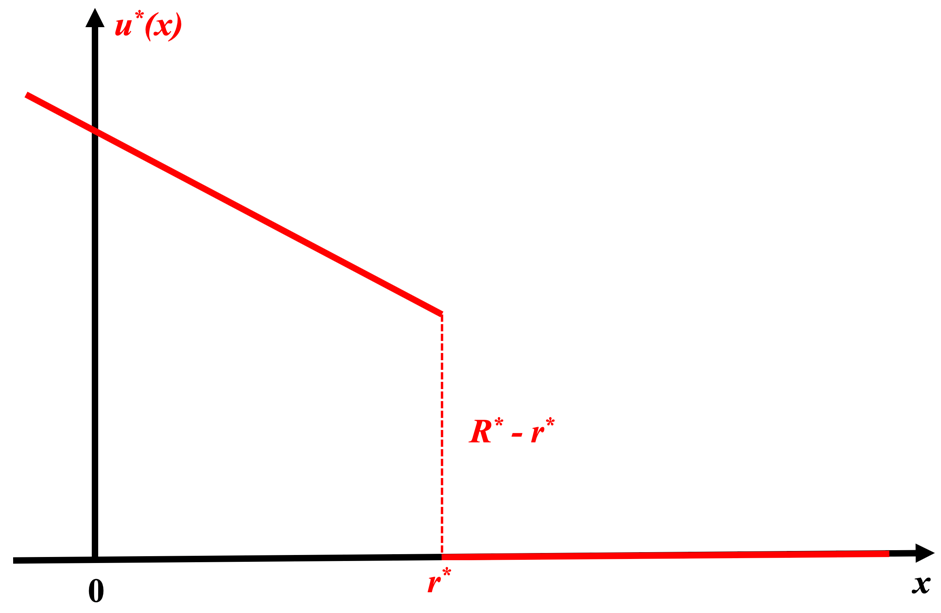


Рис. . Вид функции *u*\*(*x*) размера заказа от фиктивного уровня запасов *x*.

Вернемся к многошаговой задаче. Величина *Rn* является точкой абсолютного минимума функции G*n*(*x*), а величина *rn* – решением уравнения:

|  |  |
| --- | --- |
| *Gn*(*rn*) = *A* + *Gn*(*Rn*). | () |

При этом стратегии управления запасами с переменными на каждом шаге параметрами *Rn* и *rn*, которые рассчитываются с применением алгоритма (19), (20) и формулы (24), называются «дальнозоркими».

Известно также, что в стационарном режиме[[6]](#footnote-6) (когда *N* и *n* стремятся к бесконечности) в силу существования пределов

|  |  |
| --- | --- |
| и | (25) |

существуют такие два числа *R* и *r*, которые полностью задают оптимальную стратегию управления запасами для достаточно больших периодов планирования почти на всем их протяжении (продолжение этого анализа см. в п. 2.2).

При коэффициенте дисконтирования α < 1 эти числа (через соотношение (1)) определяют решение функционального уравнения (20).

Как нетрудно видеть, «близорукие» стратегии (с параметрами *R*1 и *r*1, которые мы выделим и назовем *R*б и *r*б) задаются следующим образом:

|  |  |
| --- | --- |
|  | (26) |
|  | (27) |

В отличие от них параметрами «дальнозорких» стационарных стратегий, которые в дальнейшем мы будем называть просто стационарными стратегиями, служат значения *R* и *r*, которые для наглядности мы иногда будем называть *R*д и *r*д.

## Сравнение «близоруких», стационарных и «дальнозорких» стратегий

Прежде чем перейти к сравнению «близоруких», стационарных и «дальнозорких» стратегий, напомним о результате, полученном Н. Прабху [4], [14] и развитом в работе [144]. В работе [4] было предложено вместо задачи (19), (20) рассматривать задачу вида: найти такие значения параметров двухуровневой стратегии управления запасами (связанные с размером поставки *u*), чтобы минимизировать функционал (они и будут параметрами стационарной стратегии *R*с и *r*с):

|  |  |
| --- | --- |
|  | (28) |

где *G*(*x*) – стационарное распределение уровня запасов в начале временн*о*го шага, а функция ϕ(*y*) определяется формулой (21). В работах [4], [14] на базе методов теории восстановления выведены соотношения, позволяющие рассчитать стационарное распределение *G*(*x*).

В работах [33], [138] были отмечены те проблемы, которые возникают в связи с тем, что стационарность распределения уровня запасов понимается как обычная стационарность (в прямом времени), а стационарность стратегий управления запасами связана с обратным временем, названные собирательно «проблемой разнополярности». Кроме того, было указано [30], что в связи с проблемой разнополярности полная эквивалентность задач (19), (20) и (28) имеет место только в достаточно экзотическом случае отсутствия инфляции, т. е. когда коэффициент дисконтирования α = 1.

Тем не менее, анализ модели (28) на предмет сравнения средних затрат при использовании параметров «близорукой» (*R*б, *r*б)-стратегии и стационарной (*R*с, *r*с)-стратегии гораздо проще, чем анализ модели (19), (20), т е. изучению «дальнозоркой» стратегии [144]. Развивая тематику сравнительного анализа «близоруких» и стационарных стратегий управления запасами, было принято решение о создании дополнительной модели (приложение *Б*), которая рассчитывает значения суммарных средних затрат *C*(*x*) методом Монте-Карло для «близорукого», стационарного и «дальнозоркого» (на каждом шаге *n* свои значения *Rn* и *rn*) случаев.

В фундаментальных исследованиях [2], [3] и [6] отмечено, что богатый эмпирический материал, полученный при анализе систем управления запсами, подсказывает типичные соотношения эконометрических характеристи модели. Принято за опорную величину брать стоимость единицы товара *c*. Стоимость хранение единицы товара в единицу времени *h* – принимает значение от 0,1*c* и ниже. Потери вследствие дефицита единицы товара в единицу времени *d* стоит выбирать не меньше двух значений *c*. Стоимость единичной поставки *A* может составлять порядка 5*c* – 10*c*. Вот для таких соотношений и была обсчитана представленная ниже модель. При этом необходимо помнить, что для товаров массового потребления длительность горизонта планирования зачастую составляет величину порядка 1 – 6 месяцев и выше.

Был рассмотрен случай, когда цена единичной поставки *A* = 100; цена единицы товара *c* = 10; стоимость хранения единицы товара в единицу времени *h* = 1; потери вследствие дефицита единицы товара за единицу времени *d* = 20; коэффициент дисконтирования 𝛂 = 0,99. Количество экспериментов по методу Монте-Карло для одного значения *x* равнялось 105 для трёх функций распределения спроса: экспоненциальной (λ = 0,2), Гауссовой (𝜇 = 5, 𝛿 = 1) и равномерной (*a* = 0, *b* = 10) со средним значением спроса равным 5. Если один шаг принять за один день, то на рис. 7 – рис. 9 и в табл. 2 представлены результаты моделирования для периода равного месяцу, кварталу и полугодию.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| *Экспоненциальное* | *Гауссово* | *Равномерное* |
| Рис. 7. Графики функций суммарных средних затрат *Cблиз*(*x*) для «близорукой» (красные), *Cстац*(*x*) для стационарной (синие) и *Cдаль*(*x*) для «дальнозоркой» (зеленые) стратегий управления запасами. Случай *N* = 30. | | |
|  |  |  |
| *Экспоненциальное* | *Гауссово* | *Равномерное* |
| **Рис. 8. Графики функций суммарных средних затрат *Cблиз*(*x*) для «близорукой» (красные), *Cстац*(*x*) для стационарной (синие) и *Cдаль*(*x*) для «дальнозоркой» (зеленые) стратегий управления запасами. Случай *N* = 90.** | | |
|  |  |  |
| *Экспоненциальное* | *Гауссово* | *Равномерное* |

Рис. . Графики функций суммарных средних затрат *Cблиз*(*x*) для «близорукой» (красные), *Cстац*(*x*) для стационарным (синие) и *Cдаль*(*x*) для «дальнозоркой» (зеленые) стратегий управления запасами. Случай *N* = 180.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Переход от**  **«близорукой» к**  **«дальнозоркой»**  **стратегии** | **Переход от**  **«близорукой» к**  **стационарной**  **стратегии** | **Переход от**  **стационарной к**  **«дальнозоркой»**  **стратегии** |
| ***N* = 30** | **Гауссово** | 30,2% | 17,3% | 15,6% |
| **Экспонент.** | 30,0% | 20,2% | 12,3% |
| **Равномерное** | 31,1% | 20,2% | 13,7% |
| ***N* = 90** | **Гауссово** | 28,5% | 22,5% | 7,7% |
| **Экспонент.** | 34,4% | 26,7% | 10,4% |
| **Равномерное** | 30,2% | 25,5% | 6,3% |
| ***N* = 180** | **Гауссово** | 28,1% | 23,7% | 5,9% |
| **Экспонент.** | 38,2% | 28,0% | 14,2% |
| **Равномерное** | 30,0% | 26,6% | 4,6% |

Табл. . Значения уменьшения суммарных средних затрат в процентах при переходе от использования одной стратегии управления запасами к другой.

Из результатов моделирования можно сделать вывод, что переход от «близорукой» к стационарной стратегии может сэкономить порядка трети расходов, а переход от «близорукой» к стационарной – порядка четверти. При этом, стоит отметить, что переход от стационарной стратегии к «дальнозоркой» даст экономию не выше порядка 15% для большинства случаев и это значение будет падать с увеличением периода планирования. При этом мы понимаем, что стационарная (тоже «дальнозоркая») стратегия гораздо проще реализуема на практике, чем «дальнозоркая» (динамическая).

Рассмотрим более «экзотические» соотношения эконометрических параметров задачи, когда, к примеру, удельные издержки содержания запасов становятся сравнимы с издержками вследствие дефицита (*h* = 5, *d* = 10) или значение *d* стремится к *c* (*d = с* = 10). При моделировании использовалось Гауссово распределение спроса (𝜇 = 5, 𝛿 = 1) с периодом планирования *N* = 90; коэффициент дисконтирования 𝛂 = 0,99; количество экспериментов по методу Монте-Карло равнялось 105. Результаты моделирования представлены в табл. 3.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***A*** | ***c*** | ***h*** | ***d*** | ***RБ*** | ***rБ*** | ***Rс*** | ***rс*** | **Близ.→ Дальн.** | **Близ.→Стац.** | **Стац.→ Дальн.** |
| **100** | **10** | **1** | **80** | 6,1 | 3,35 | 32,55 | 4,95 | 46,4% | 41,7% | 8% |
| **20** | 4,9 | -5,85 | 31,45 | 3,35 | 28,5% | 22,5% | 7,7% |
| **10** | 2,9 | -497,45 | 30,66 | 1,815 | 95,8% | 95,4% | 7,2% |
| **5** | **80** | 5,9 | 3,25 | 16,4 | 4,25 | 33,3% | 20% | 16,7% |
| **20** | 4,7 | -5,95 | 14,9 | 1,75 | 8,9% | -7% | 14,9% |
| **10** | 2,8 | -497,55 | 14,1 | -0,85 | 94,5% | 93,8% | 11,1% |
| **5** | **1** | **20** | 5,6 | -2,15 | 31,93 | 3,402 | 28,3% | 21% | 9,3% |
| **10** | 4,9 | -15,85 | 31,22 | 1,834 | 44,3% | 39,1% | 8,5% |
| **5** | **20** | 5,3 | -2,35 | 14,9 | 1,75 | 4,4% | -16,1% | 17,6% |
| **10** | 4,6 | -16,05 | 14,2 | -0,85 | 22,6% | 10,7% | 13,3% |
| **50** | **10** | **1** | **20** | 4,9 | -0,85 | 21,9 | 3,85 | 12,3% | 4,7% | 8% |
| **10** | 2,6 | -497,75 | 21,39 | 2,75 | 96,3% | 96% | 7,8% |
| **5** | **20** | 4,7 | -0,95 | 10,4 | 2,75 | 1,9% | -20,8% | 18,7% |
| **10** | 2,5 | -497,85 | 9,9 | 0,85 | 95,4% | 94,5% | 16,4% |
| **5** | **1** | **20** | 5,6 | 1,15 | 22,1 | 3,95 | 31% | 23% | 10,4% |
| **10** | 4,9 | -5,85 | 21,7 | 2,75 | 20,1% | 11,5% | 9,8% |
| **5** | **20** | 5,3 | 1,05 | 10,5 | 2,75 | 23,1% | -0,5% | 23,5% |
| **10** | 4,6 | -6,05 | 10 | 0,85 | 2,5% | -22,6% | 20,5% |
| **10** | **10** | **1** | **20** | 4,9 | 3,15 | 11,6 | 4,85 | 8,8% | -0,2% | 9% |
| **10** | 2,6 | -97,75 | 11,2 | 4,15 | 87,9% | 86,6% | 9% |
| **5** | **20** | 4,7 | 3,05 | 5,8 | 4,35 | 21% | -6,8% | 26% |
| **10** | 2,5 | -97,85 | 5,4 | 3,45 | 86,8% | 82% | 26,1% |
| **5** | **1** | **20** | 5,6 | 3,95 | 11,7 | 4,85 | 15 % | 2,2% | 13,1% |
| **10** | 4,9 | 2,15 | 11,2 | 4,15 | 14,3% | 1,6% | 12,9% |
| **5** | **20** | 5,3 | 3,75 | 5,8 | 4,35 | 32,6% | -4,1% | 35,3% |
| **10** | 4,6 | 1,95 | 5,4 | 3,45 | 31% | -7,3% | 35,7% |

Табл. . Оптимальные параметры «близорукой» и стационарной стратегий управления запасами и «выигрыши» по величине функции суммарных средних затрат в процентном соотношении при переходе от одной стратегии управления к другой.

Из приведённой выше таблицы, можно сделать вывод, что при стремлении значения коэффициента издержек вследствие дефицита *d* к величине цены товара *c* мы получим ещё больший выигрыш при переходе от «близорукой» к стационарной стратегии (порядка 82% – 95%), при этом выигрыш от перехода со стационарной к «дальнозоркой» стратегии будет тем меньше, чем больше значение *A* относительно *c*. В некоторых специфических случаях, когда значение *h* становилось сравнимо со значением *d* переход с «близорукой» стратегии на стационарную мог оказаться невыгоден ввиду того, что «близорукое» значение *r* позволяло поддерживать в системе небольшой дефицит, который оказывался дешевле, чем хранение товаров по высокой ставке *h*.

Результаты сравнения показывают, что в отдельных случаях элементарной заменой значений *R* и *r* можно сократить расходы на треть без каких-либо дополнительных вложений. Отсюда следует, что именно стационарные стратегии желательно, да и попросту необходимо внедрять в практическую логистику. Возможности современных компьютеров, несомненно, позволяют осуществлять требуемые расчеты «дальнозорких» (динамических) стратегий с помощью алгоритма (19), (20). При этом, если мы готовы пожертвовать простотой стационарной (тоже «дальнозоркой») стратегии в угоду сокращения расходов – мы можем применять «дальнозоркую» (динамическую) стратегию управления запасами.

При этом необходимо учесть, что реальное значение коэффициента дисконтирования α может изменяться во времени. В этому случае вместо константного значения α будут использоваться значения α*n* соответствующего шага. Поведение графиков оптимальный параметров *R* и *r* можно посмотреть в разделе 2.4 на рис. 16. В рассмотренном случае графики зависимостей *r*(α) и *R*(α) представляют собой монотонно возрастающие функции от α. В самом деле, при большем значении α мы все дальше «заглядываем» в будущее и выясняется, что заказывать нужно больше, поскольку все меньше смысла ограничиваться «интересами» только ближайшего шага принятия решений. Значениями *r*(0) и *R*(0) описываются параметры «близорукой» стратегии управления запасами.

В результате выполненного анализа мы видим, что использование стационарных стратегий вместо «близоруких» может дать весьма ощутимый экономический эффект. В чем же причины того, что «близорукие» стратегии нашли в практической логистике гораздо более широкое применение? Этих причин две. Во-первых, эти стратегии очень просты: достаточно задать всего два числа *R*б и *r*б, чтобы задать стратегию управления запасами на всем периоде планирования. Такие стратегии абсолютно понятны и удобны практическим менеджерам в области логистики запасов. Во-вторых, эти стратегии, которые требуют решения всего двух простых уравнений (26) и (27), а не вычислительно трудоемких уравнений динамического программирования (19) и (20). Однако, как следует из существования пределов *R* и *r* (25) для динамически изменяющихся параметров *Rn* и *rn* «дальнозорких» стратегий, в стационарном режиме управления также можно ограничиться только двумя параметрами *R* и *r*. Впрочем тут же возникает два вопроса, как считать эти параметры и можно ли ими подменять динамически изменяющиеся параметры *Rn* и *rn* «дальнозорких» стратегий на протяжении всего периода планирования.

Ответим сначала на второй вопрос. Как показано в п. 2.6, в большинстве случаев уже примерно к 10-му шагу реальные значения *Rn* и *rn* практически неотличимы от *R* и *r*. При этом моделирование показало, что подмена реальных *Rn* и *rn* предельными значениями *R* и *r* приводит к дополнительным затратам в пределах до 15% от значения минимально возможных средних затрат. Это делает актуальным вопрос о замене «близоруких» параметров *R*б и *r*б на стационарные параметры *R* и *r*.

При этом поскольку задача решается на нестационарном рынке, когда значения эконометрических характеристик системы и коэффициент дисконтирования постоянно меняются (заметим, что, как правило, гораздо медленнее, чем мы переходим от одного периода планирования к другому), желательно не просто найти конкретные значения параметров *R* и *r*, а выстроить настоящий альбом значений этой пары для всех возможных значений эконометрических характеристик *A*, *c*, *h* и *d* и коэффициента дисконтирования *α*.

Теперь приходится вспомнить об отмеченном выше первом вопросе – как считать параметры *R* и *r*. Это требует быстрого (очень быстрого) решения уравнений динамического программирования (19) и (20). Известные до сих пор попытки их компьютерного решения приводили к тому, что один сеанс расчета (для конкретного набора всех исходных характеристик) занимал примерно сутки. В связи с этим была предпринята попытка оптимизировать процесс расчета по критерию минимума времени.

## Оптимальность *Rr*-стратегий управления запасами и связь между «дальнозоркими» и стационарными стратегиями управления

Следуя [3], продемонстрируем качественно, как определить оптимальную стратегию функционирования в системах, находящихся в стационарном режиме, используя подход, связанный с функциональными уравнениями.

Когда *А* является заданной величиной, оптимальная стратегия управления запасами зависит только от формы кривой функции *G*(*y*). Пусть кривая *G*(*y*) имеет форму как показано на рис. 4. Положим, что *G*(*R*\*) – это абсолютный минимум функции *G*(*y*), а *R*\* – это единственная точка, в которой этот минимум достигается (*R*\* будет функцией *y*). При этом, пусть *r*\* будет таким значением y (меньше *R*\*), для которого *G*(*r*\*) = *A* + *G*(*R*\*). Если значение фиктивного уровня запасов x будет находиться между *r*\* и *R*\*, то оптимально не делать заказ вообще, так как делая любой заказ мы не можем снизить значение функции *G*(*y*) ниже, чем *G*(*R*\*). В этом случае *G*(*x*) < *A* + *G*(*R*\*), что означает рост ожидаемых издержек из-за поданного заказа. Если x < *r*\*, то оптимально сделать заказ, при этом его размер должен быть таким, чтобы функция *G*(*y*) приняла минимально возможное значение, то есть размер заказа должен быть равен *R*\* - *x*, тогда будет выполняться неравенство *G*(*x*) > *A* + *G*(*R*\*). Более того, ожидаемые переоценённые издержки будут больше *A* + *G*(*R*\*) при всех значениях размера заказа *u* отличных от *R*\* – *x.* В случае *x > R*\* не следует подавать заказ по тем же соображениям – это приведёт к увеличению издержек, но важно отметить, что *x* никогда не будет больше *R*\* в случае нормального функционирования системы.

Покажем, что *Rr*-стратегия управления запасами будет оптимальной в данном случае. Если провести некоторые рассуждения, можно показать, что в случае нулевых затрат на поставку (*A* = 0), стратегия, пополняющая фиктивный уровень запасов до значения *R\**, будет оптимальной. Этот результат интуитивно понятен, так как цена товара не меняется, а подача заказа не требует дополнительных расходов, то оптимально будет подавать заказы в начале каждого периода. При этом размер заказа должен быть таковым, чтобы фиктивный уровень запасов доводился до значения *R*\*. *Rr*-стратегия будет оптимальной в случае, если *G*(*y*) будет *A*-выпуклой функцией от *y*.

На рис. 10 показан пример вида функции *G*(*y*) при котором *Rr*-стратегия уже не будет оптимальной. В этом случае оптимальным будет следующее правило: если фиктивный уровень запасов *x* < *r*\*1, то размер заказа должен быть равен *R*\* - *x*; если *r*\*1 < *x* < *r*\*2, то заказ не подаётся; если *r*\*2 < *x* < *r*\*3, то размер заказа будет равен *R*\* - *x*; если *x* > *r*\*3, то заказ не подаётся. Мы полагаем, что вид кривой *G*(*y*), продемонстрированный нарис. 10 не будет встречаться на практике.

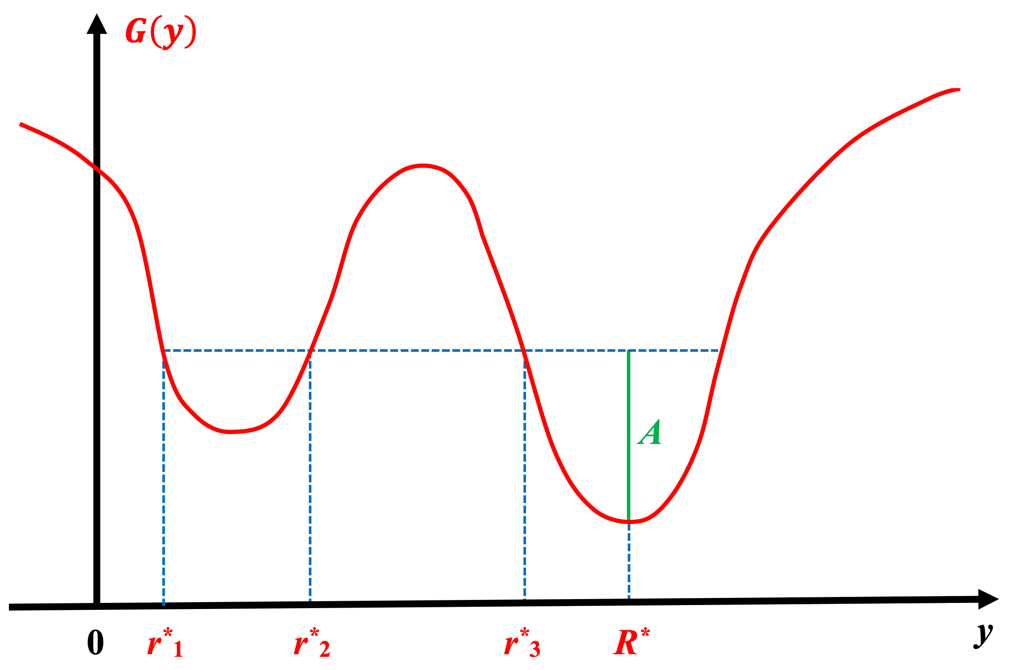


Рис. . Альтернативный вид функции *G*(*y*).

Заметим, что функция *G*(*y*) (за исключением члена *cy*) представляет собой ожидаемые переоценённые издержки, если рассматриваемый период функционирования системы начинается с фиктивного уровня запасов x (после подачи любого заказа) и управление осуществляется при помощи оптимальной стратегии. В случае относительно малых *y*, ожидаемые издержки будут велики (относительно) из-за издержек по учёту за текущий период. С ростом *y* эти издержки будут в среднем убывать до тех пор, пока y достаточно велико. Уменьшение издержек по учёту будет преобладать над ростом издержек содержания. После достижения баланса этих составляющих и при дальнейшем увеличении y, ожидаемые издержки тоже будут расти. Форма кривой не изменится при добавлении члена *cy*. Поэтому кривая *G*(*y*) будет иметь вид, показанный на рис.  4, а не рис. 10.

Таким образом, *Rr*-стратегия будет оптимальной для стационарного режима работы системы, если *g*(*y*) является выпуклой функцией (функция *G*(*y*) при этом должна быть *А*-выпуклой) и цена товара фиксирована. Факт *А*-выпуклости исключает возможность случаев, представленных на рис. 10, где *Rr*-стратегия не будет оптимальной, но не исключает возможности случаев, как показано на рис. 11 (*Rr*-стратегия управления запасами не будет оптимальной).

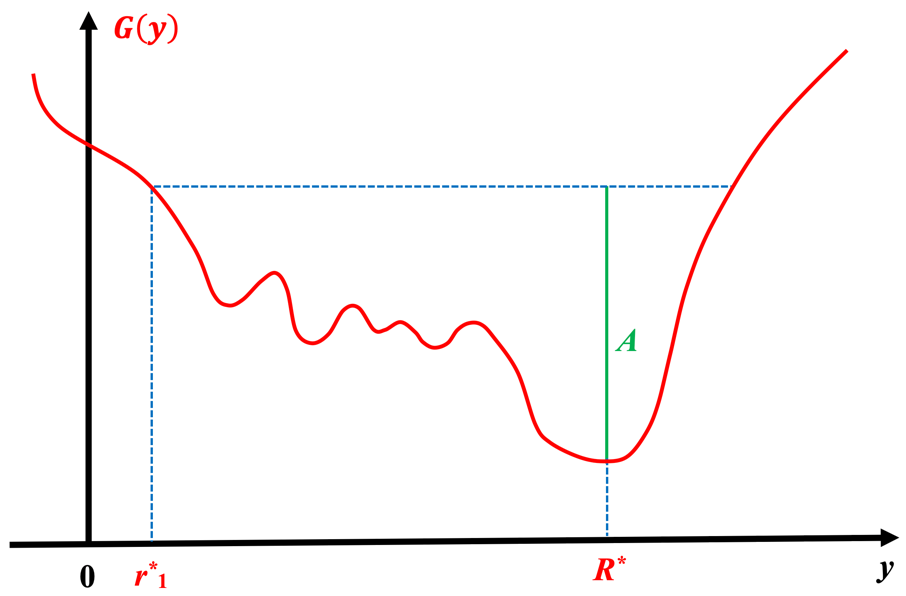


Рис. . Вид функции *G*(*y*) для которой *Rr*-стратегия не будет оптимальной.

## Исследование зависимости оптимальных значений *R* и *r* стационарных стратегий управления запасами от эконометрических параметров базовой модели

Представляет интерес изучение свойств стационарных значений параметров *R* и *r* в зависимости от значений стоимостных параметров *h*, *d*, *c* и *A*. Проблема в том, что для этого нужно решать упомянутые выше уравнения дискретного динамического программирования. Процедура эта – сложная и времяемкая. Удалось оптимизировать программный подход решения этой задачи, который позволил радикально ускорить процесс вычислений. Новый способ базируется на некоторых свойствах алгоритма динамического программирования, позволяющих упростить вычисления за счет использования ранее полученной расчетной информации. Этот способ находится в ряду методов, который А.А. Лазарев и Е.М. Гафаров назвали графическим методом [146], но опирается на другие свойства задач дискретной оптимизации. Рассматриваемые в настоящей работе вероятностные задачи дискретного динамического программирования не укладываются в схему графического метода, предложенного для детерминистских постановок задач.

Приведем несколько графиков, полученных в результате моделирования. При этом рассматривалось три варианта распределений спроса на одном шаге: (а) Гауссово распределение, (б) экспоненциальное распределение и (в) равномерное распределение. Все эти распределения были выбраны так, чтобы, как правило, их математические ожидания *μ* совпадали (было выбрано *μ* = 5). Параметр λ экспоненциального закона распределения был равен 0,2. Среднеквадратическое отклонение 𝛿 гауссова распределения равнялось 1. Равномерное распределение рассматривалось на отрезке от 0 до 10.

Для удобства рассмотрения поведения функций *R*, *r* от того или иного параметра удобно ввести новую функцию *umin* = *R* – *r*. Смысл новой функции заключается в следующем: если уровень запасов *x* в начале шага перед принятием решения (о подаче заказа на пополнение запасов и о его размере) ниже или равен *r*, то величина размера заказа будет ограничена снизу значением *umin.* То есть *un*(*x*) ≥ *umin.*

Рассмотрим поведение параметров *R* и *r* от стоимости единичной поставки товара *A*. Значение параметров модели: *A* = 0 ÷ 100; *c* = 10; *h* = 1; *d* = 20; 𝛂 = 0,9. Результаты моделирования представлены на рис. 12. С ростом значений *А*, графики значений функции *R*(*A*) показывают монотонный рост, при монотонном убывании значений функции *r*(*A*). С практической точки зрения это означает ровно одно – становиться экономически целесообразно увеличивать размер каждого заказа, чтобы вклад стоимости единичной поставки в итоговые расходы был как можно меньше. При дальнейшем росте значения *А* значения функции *r*(*А*) будут монотонно убывать и уходить в отрицательную область, что будет означать приемлемость небольшого дефицита товара в системе.

Рис. . Графики зависимостей оптимальных параметров *R*(*A*) и *r*(*A*) для экспоненциального, гауссова и равномерного распределений спроса. Значение параметров: *A* = 0 ÷ 100; *c* = 10; *h* = 1; *d* = 20; 𝛂 = 0,9.

Перейдём к рассмотрению поведения параметров *R* и *r* от *c* стоимости единицы товара. Значение параметров модели: *A* = 100; *c* = 0 ÷ 100; *h* = 1; *d* = 20; 𝛂 = 0,9.

Рис. . Графики зависимостей оптимальных параметров *R*(*c*) и *r*(*c*) для экспоненциального, гауссова и равномерного распределений спроса. Значение параметров: *A* = 100; *c* = 0 ÷ 100; *h* = 1; *d* = 20; 𝛂 = 0,9.

Значения функций *R*(*c*) и *r*(*c*) монотонно убывают с ростом значения *c* стоимости единицы товара. Темп снижения значений функции *R*(*c*) и *umin*(*c*) убывает с ростом значения *c*, при этом значения *r*(*c*) постепенно (и монотонно) заходят в отрицательную область. На практике это означает, что с ростом цены за единицу товара уменьшается размер заказа и постепенно растёт уровень допустимого дефицита в системе.

Далее изучим зависимость параметров *R* и *r* от стоимости хранения единицы товара *h*. Значение параметров модели: *A* = 100; *c* = 10; *h* = 0 ÷ 20; *d* = 20; 𝛂 = 0,9.

Рис. . Графики зависимостей оптимальных параметров *R*(*h*) и *r*(*h*) для экспоненциального, гауссова и равномерного распределений спроса. Значение параметров: *A* = 100; *c* = 10; *h* = 0 ÷ 20; *d* = 20; 𝛂 = 0,9.

На графике можно наблюдать монотонно убывающие функции *R*(*h*) и *r*(*h*). При этом темп их убывания падает с ростом *h*. Минимальное значение размера заказа *umin*(*h*) монотонно убывает с ростом значения *h*. На практике это значит следующее: с ростом стоимости хранения единицы товара становиться не выгодно делать большие заказы из-за высоких расходов на содержание склада. При дальнейшем росте значения *h*, значения функции *r*(*h*) могут принять отрицательные значения. Это вызвано тем фактом, что в какой-то момент стоимость хранения товаров перевешивает потери от их дефицита, то есть в системе допускается небольшой дефицит.

Изучим поведение параметров *R* и *r* от коэффициента удельных потерь вследствие дефицита *d*. Значение параметров модели: *A* = 100; *c* = 10; *h* = 1; *d* = 0 ÷ 100; 𝛂 = 0,9.

Рис. . Графики зависимостей оптимальных параметров *R*(*d*) и *r*(*d*) для экспоненциального, гауссова и равномерного распределений спроса. Значение параметров: *A* = 100; *c* = 10; *h* = 1; *d* = 0 ÷ 100; 𝛂 = 0,9.

Поведение графиков функций *R*(*d*) и *r*(*d*) показывает, что при очень малых значениях *d* потерь вследствие дефицита товара (в данном примере при *d* < 1,5) модель не в состоянии найти адекватные значения *R* и *r*. Это значит, что нет смысла применять двухуровневую стратегии управления запасами потому, что потерями от дефицита товара можно пренебречь по сравнению с другими компонентами расходов. При достижении определённого значения *d* модель позволяет рассчитать значения функции *r*(*d*), что говорит о целесообразности применения двухуровневой стратегии управления запасами. При малых значениях *d* значения функции *r*(*d*) принимают сильно отрицательные значения, что говорит о том, что дефицит товара, даже в большом количестве экономически более выгоден, чем хранение товара на складе. При дальнейшем росте значения *d*, значение минимального размера заказа *umin* изменяется не сильно, так как функции *R*(*d*) и *r*(*d*) растут с примерно одинаковым темпом, а значения функции *r*(*d*) переходят в положительную область и монотонно растут, что свидетельствует о невыгодности допущения дефицита товара в системе.

Рассмотрим поведение параметров *R* и *r* от значений коэффициента дисконтирования 𝛂. Значение параметров модели: *A* = 100; *c* = 10; *h* = 1; *d* = 20; 𝛂 = 0 ÷ 1.

Рис. . Графики зависимостей оптимальных параметров *R*(𝛂) и *r*(𝛂) для экспоненциального, гауссова и равномерного распределений спроса. Значение параметров: *A* = 100; *c* = 10; *h* = 1; *d* = 20; 𝛂 = 0 ÷ 1.

На графике видно, что функции *R*(𝛂) и *r*(𝛂) монотонно возрастают с увеличением значений 𝛂. Ввиду того, что темп роста значений для функции *R*(𝛂) выше, чем для функции *r*(𝛂), значение функции минимального размера заказа *umin*(𝛂) тоже будет монотонно возрастать. На практике это явление объясняется тем, что при больших значениях 𝛂 мы всё дальше «заглядываем» в будущее и обнаруживаем, что экономически целесообразно делать большие заказы, а не ограничиваться «интересами» только ближайшего шага принятия решений. Так же с ростом значения 𝛂 уменьшается уровень допустимого дефицита товара, что демонстрирует график функции *r*(𝛂) для случая экспоненциального распределения спроса. В разделе 2.2 было показано превосходство использования «дальнозорких» двухуровневых оптимальных стратегий управления над «близорукими».

На выше представленных графиках для гауссова распределения спроса с параметрами 𝜇 = 5, 𝛿 = 1 и равномерного распределения с параметрами *a* = 0 и *b* = 10, параметр *R* может иметь вид «лестницы». Рассмотрим подробнее данное явление в следующем разделе.

## Обсуждение лестничного характера поведения параметра *R* стратегии управления запасами

Необходимо отметить, что выявление «лестничных» зависимостей параметра *R* от коэффициента дисконтирования *α* оказалось неожиданностью. Чтобы разобраться в этом, важно вспомнить, что нормальный или равномерный законы распределения описывают случайные величины, которые более «детерминированы», нежели случайные величины, описываемые экспоненциальным законом распределения. В самом деле, при стремлении среднеквадратического отклонения *σ* гауссова закона к 0 это распределение вырождается в чисто детерминистский скачок. То же самое можно сказать о равномерном законе распределении при стремлении его правой *b* и левой *a* границ к среднему значению *μ*. В этом контексте только экспоненциальный закон распределения максимально далек от описания детерминированных величин.

Поэтому, чтобы понять, откуда берется полученная слегка наклонная «лесенка», рассмотрим случай детерминированного спроса, который на каждом дискретном шаге будет равен 5 (заданному выше среднему значению случайного спроса). Тогда, сохраняя все остальные эконометрические параметры задачи прежними, кроме *α*, которое выберем равным 1, построим следующую дискретную и детерминированную модель управления запасами, обобщающую модель, рассмотренную в монографии [3].

Требуется определить оптимальный размер заказа *u*, который в силу однородности и симметрии задачи будет фиксированным и равным *Q*. Это значение однозначно определит значение интервала между очередными поставками *T*п и значение точки подачи заказа *r*. Картина изменения уровня запаса в системе представлена на рис. 17.

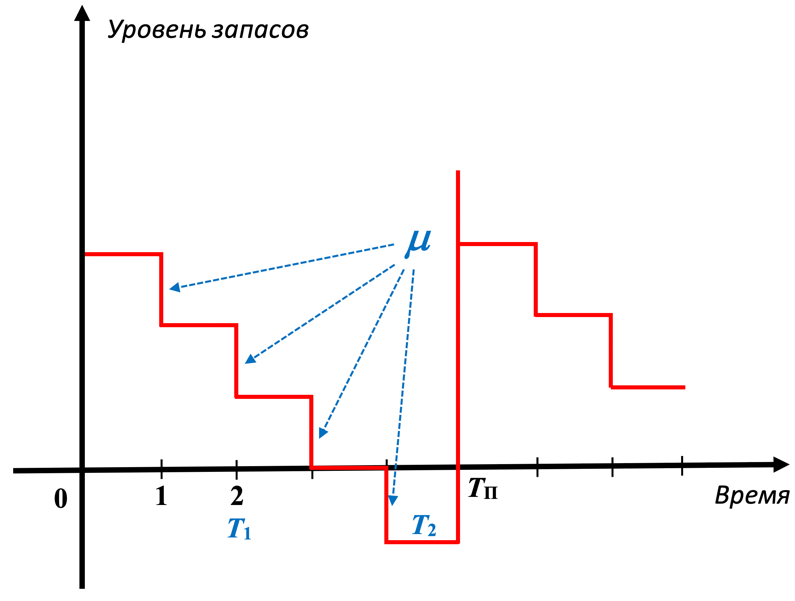


Рис. . График изменения уровня запасов в детерминированной дискретной модели.

Тогда, следуя картинке на рис. 17, где, через *T*1 обозначен интервал времени (между соседними поставками), когда запас положителен, а через *T*2 обозначен интервал времени (между соседними поставками), когда запас отрицателен, так что *T*п = *T*1+ *T*2, можно записать выражение для суммарных затрат в интервале между поставками:

|  |  |
| --- | --- |
|  | (29) |

Чтобы упростить запись, будем считать, что *Q* и *r* делятся без остатка на *μ*. Тогда, используя дополнительные выражения для целочисленных значений *T*п, *T*1и *T*2: *T*п = *K*, *T*1 = *I*, *T*2 = *J* и *K = I + J*, где *K*, *I* и *J* – натуральные числа, причем *K* = *Q*/*μ*, *I* = (*Q* – *r*)/*μ*, а *J = r/μ*. Осталось определить среднее значение затрат в единицу времени, разделив из формулы (29) на величину периода *T*п между поставками, определив тем самым минимизируемый функционал *C*1:

|  |  |
| --- | --- |
| = | (30) |

Минимизируя (30) по *Q* и *r*, можно получить оптимальное значение *Q* через связанное с ним целочисленное значение *K*. Оптимальное *K\** представляет собой наибольшее значение натурального числа *K*, которое удовлетворяет неравенству (31):

|  |  |
| --- | --- |
|  | (31) |

Нетрудно получить и выражение для оптимального значения *r*.

Что же следует из приведенных выкладок? Например, то, что с ростом *h* или какого-нибудь эконометрического параметра решения неравенства (31) изменяются скачкообразно, образуя «лестницу» с горизонтальными ступенями, причем высота ступени равна значению спроса на одном шаге. Тот же самой вывод можно было бы сделать и при учете отличного от 1 коэффициента дисконтирования *α*. Понятно, что, если спрос случаен, но масштабы этой случайности таковы, что рассматриваемая ситуация «почти» детерминистская (например, для гауссова закона, среднее значение спроса *m* за один шаг гораздо больше среднеквадратического отклонения *σ*), то ступеньки этой «лестницы» отклонятся от горизонтали. «Лестница» станет наклонной, что и подтверждается графиками на рис. 12 - рис. 16.

На рисунке рис. 18 приведён график зависимости *R*(𝛂) наглядно демонстрирующий поведение «лестницы» в зависимости от значения среднеквадратического отклонения 𝛿 для случая Гауссова распределения спроса. Видно, как с увеличением значения 𝛿 «лестница» распрямляется.

На рис. 19 рассмотрен случай равномерного распределения спроса. Мы можем наблюдать, что зависимости стремятся принять скачкообразную «лестничную» форму при стремлении параметров *a* и *b* равномерного распределения к их среднему значению (что в конечном счёте приведёт к вырождению функцию распределения спроса в чисто детерминистический скачок).

Рис. . Графики зависимостей оптимальных параметров *R*(𝛂) от среднеквадратического отклонения 𝛿 для гауссова распределения спроса. Значение параметров: *A* = 100; *c* = 5; *h* = 1; *d* = 20; 𝛂 = 0 ÷ 1; 𝜇 = 5.

Рис. . Графики зависимостей оптимальных параметров *R*(𝛂) от значений параметров *a* и *b* равномерного распределения спроса. Значение параметров: *A* = 100; *c* = 5; *h* = 1; *d* = 20; 𝛂 = 0 ÷ 1.

Рис. . Вид функций *С*(*x*) и *G*(*y*) для экспоненциального распределения (𝜆 = 0,2). Параметры модели: *A* = 100; *c* = 5; *h* = 1; *d* = 20; 𝛂 = 1. Номер шага *n* = 50.

Рис. . Вид функций *С*(*x*) и *G*(*y*) для нормального распределения (𝜇 = 5; 𝛿 = 0,05). Параметры модели: *A* = 100; *c* = 5; *h* = 1; *d* = 20; 𝛂 = 1. Номер шага *n* = 50.

На рис. 21 изображено «ступенчатое» поведение функций *C*(*x*) суммарных средних затрат и функции *G*(*y*) для случая нормального распределения спроса со средним значением 𝜇 = 5 и среднеквадратическим отклонением 𝛿 = 0,05. Аналогичное «ступенчатое» поведение функций будет при использовании равномерного закона для функции распределения спроса с параметрами *a* и *b*, находящимися близко к своему среднему значению. Именно такое ступенчатое поведение функции *G*(*y*) делает необходимым введение критерия *А*-выпуклости (помимо постоянства цены единицы товара) для того, чтобы двухуровневая стратегия управления запасами была оптимальной. На рис. 20 представлены функции *C*(*x*) и *G*(*y*) для случая экспоненциального распределения спроса.

Успехи в оптимизации скорости решения задачи динамического программирования для поиска значений оптимальных параметров *R* и *r*, достигнутые в процессе работы и описанные в разделах 5.2, 5.3 и 5.4, позволили модернизировать и расширить функционал программы. Новый алгоритм сделал возможным добавить дополнительную степень свободы – рассчитывать значения оптимальных параметров *R* и *r* как зависимость от двух эконометрических параметров. Если до этого момента на выходе программа выдавала двухмерные графики как функции зависимости *R* и *r* от одного эконометрического параметра, то теперь в результате работы программа строит трёхмерные поверхности *R* и *r* как функции зависимости от двух эконометрических параметров. Пример такой поверхности представлен на рис. 22.

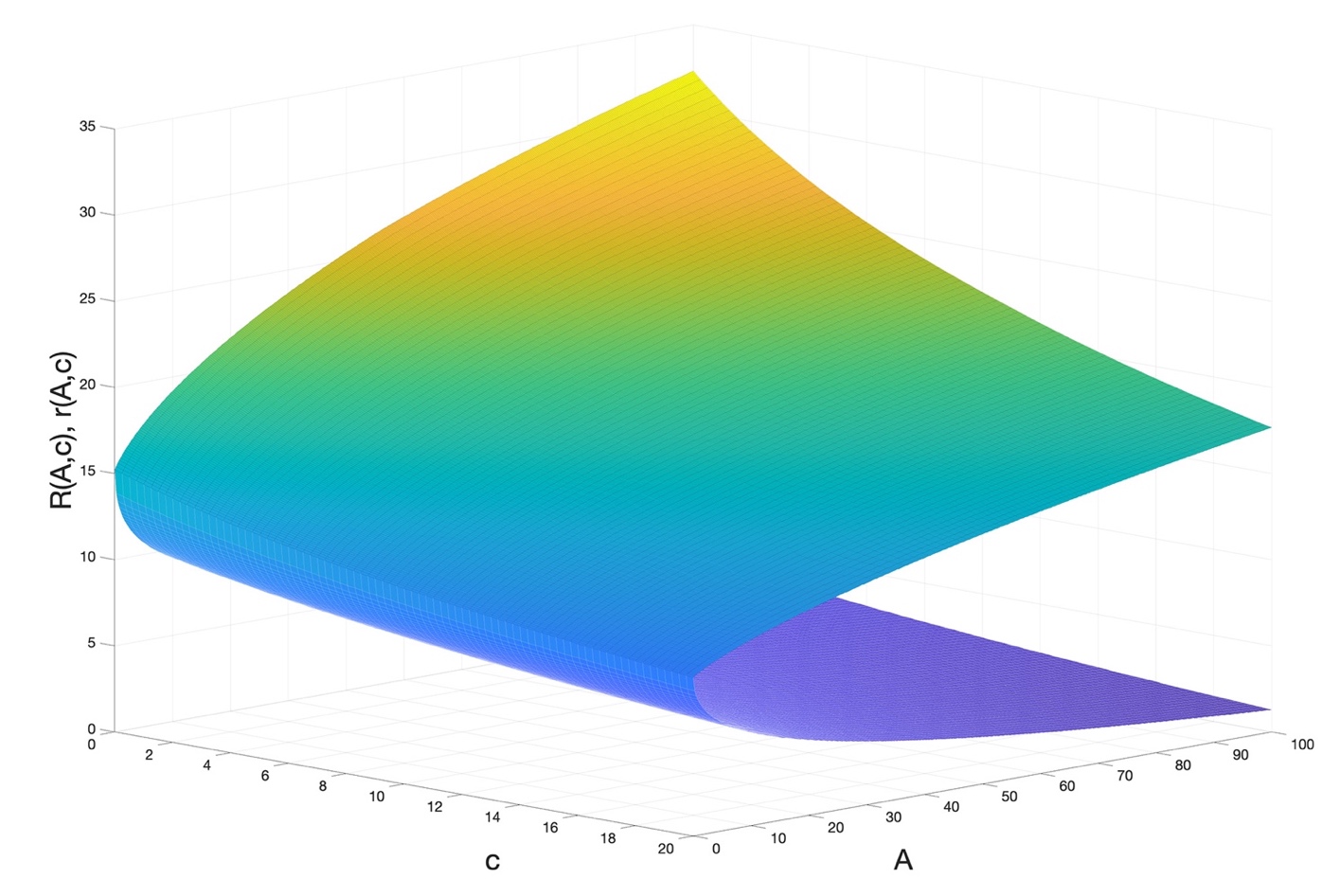


Рис. . Поверхности *R*(*A*,*c*) и *r*(*A*,*c*) для случая экспоненциального распределения спроса.

Параметры модели: *A* = 0 ÷ 100; *c* = 0 ÷ 20; *h* = 1; *d* = 20; 𝛂 = 0,9.

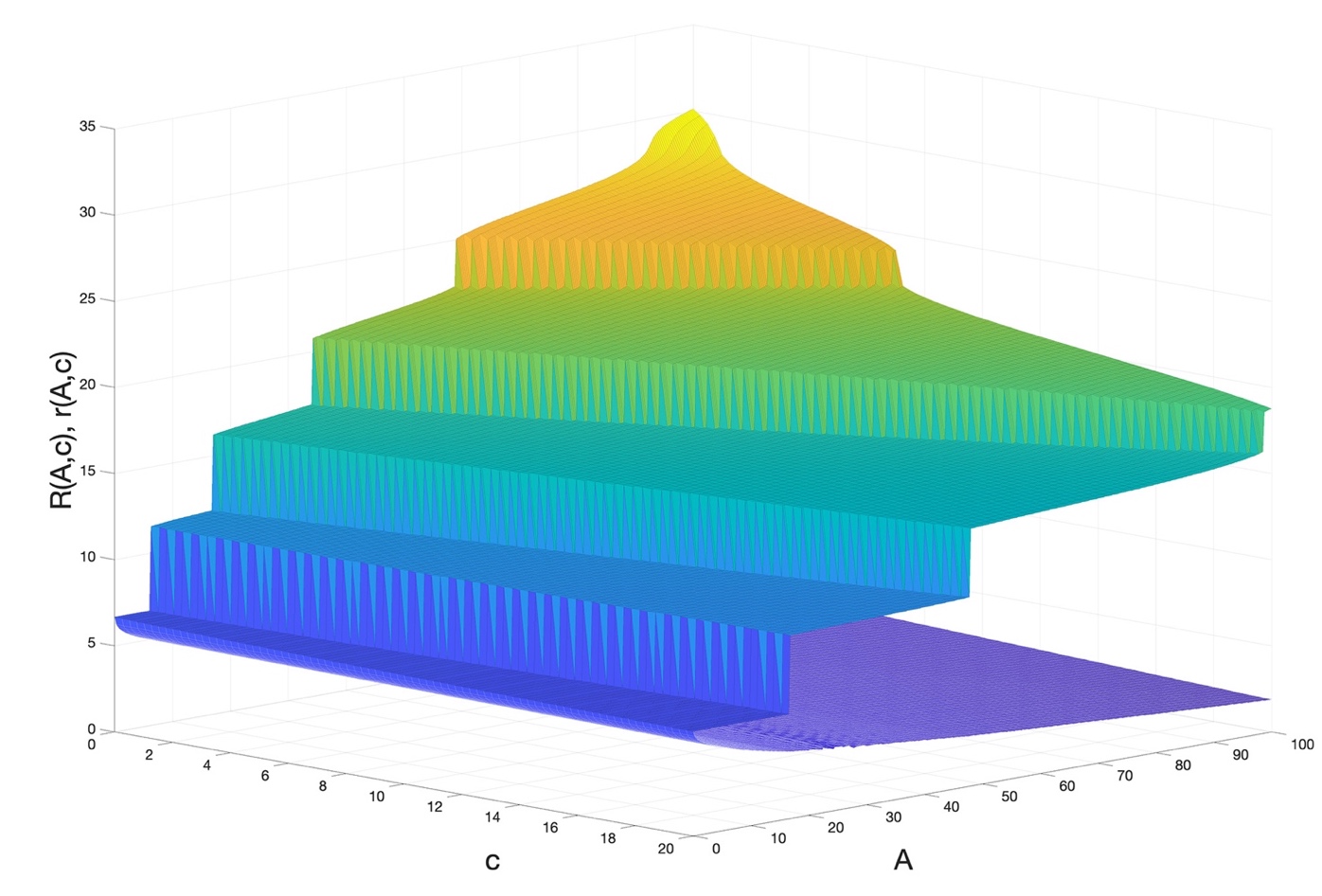


Рис. . Поверхности *R*(*A*,*c*) и *r*(*A*,*c*) для случая Гауссова распределения спроса.

Параметры модели: *A* = 0 ÷ 100; *c* = 0 ÷ 20; *h* = 1; *d* = 20; 𝛂 = 0,9.

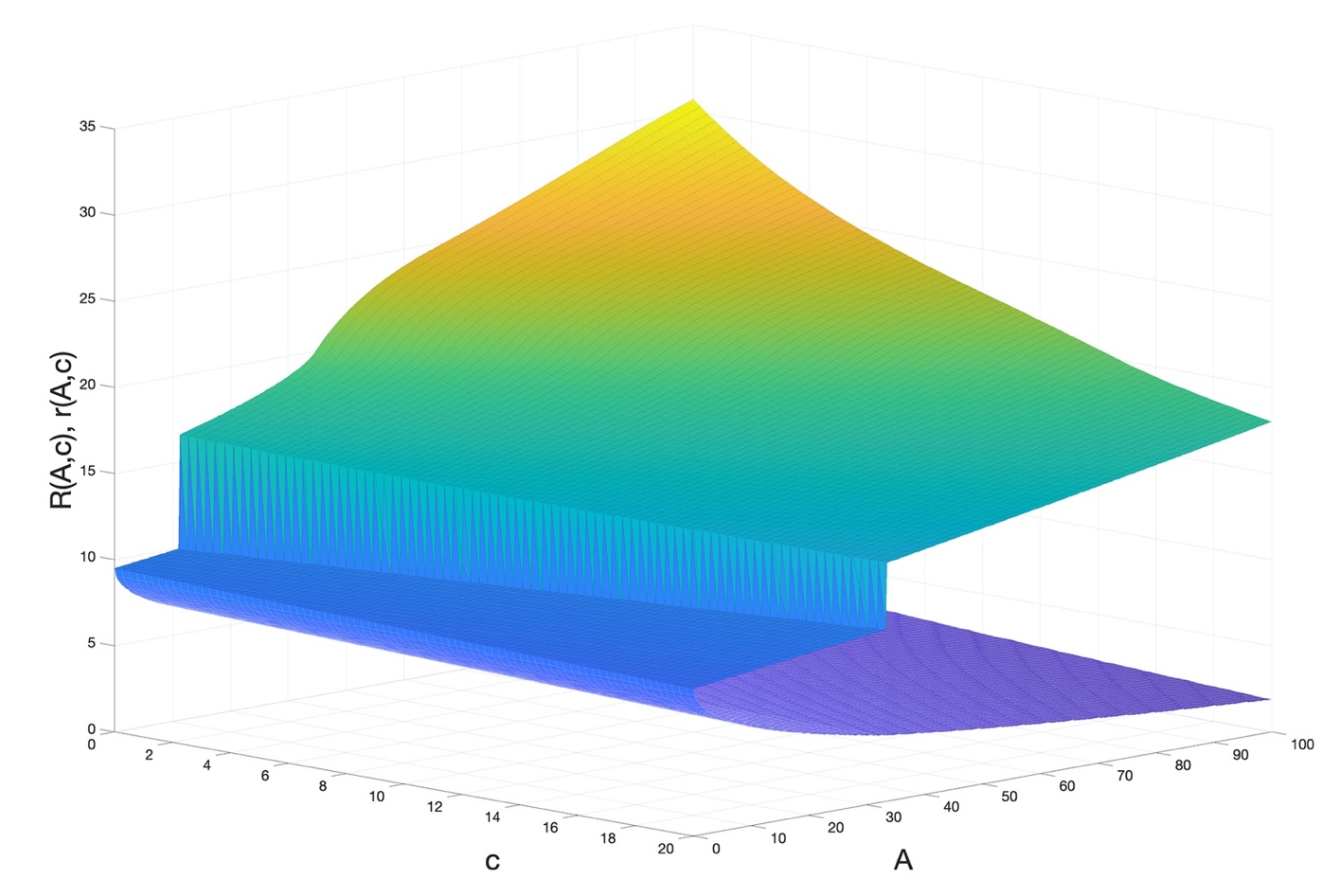


Рис. . Поверхности *R*(*A*,*c*) и r(*A*,*c*) для случая равномерного распределения спроса.

Параметры модели: *A* = 0 ÷ 100; *c* = 0 ÷ 20; *h* = 1; *d* = 20; 𝛂 = 0,9.

Поверхности, представленные на рис. 23 и рис. 24 демонстрируют ступенчатое поведение оптимальных параметров *R*,*r* задачи управления запасов для случая Гауссова и равномерного распределений, соответственно.

Такое представление результатов моделирования, ввиду своей наглядности, крайне полезно в исследовании задач управления запасами и позволяет строить предположения о поведении параметров модели при определённых соотношениях эконометрических параметров. В условиях нестационарности процессов, протекающих внутри системы управления запасами, крайне полезно иметь под рукой некий «атлас», который мог бы подсказать – как будут изменяться оптимальные параметры *R* и *r*. По своей сути, «атлас» – это *M*-мерные массив значений *R* и *r*, в зависимости от значений *M* эконометрических параметров модели, таких как: *A*, *c*, *h*, *d*, 𝛂 (в этот список можно включить параметры функций плотности вероятности спроса). Рассчитав эти массивы, их можно включать в различное программное обеспечение для предсказания поведения рынка или как прикладной инструмент для управляющих хозяйством. Исходный код модели в приложении А, благодаря оптимизации скорости вычислений, позволяет быстро находить оптимальные параметры *R*, *r* как зависимость от одного (график) или двух (поверхность) эконометрических параметров. Структура алгоритма позволяет легко добавить новые «степени свободы» в эту модель и рассчитывать «атласы», упомянутые выше.

## Скорость сходимости параметров *Rn* и *rn* к своим предельным значениям

Актуальный вопрос, возникающий при моделировании систем, основанных на двухуровневых стратегиях управления запасами, состоит в следующем – какое количество шагов *N* нужно выбрать, чтобы значения оптимальных параметров *Rn* и *rn* достаточно (в рамках конкретной задачи) приблизились к своим предельным значениям.

Ниже будут приведены графики зависимостей значений параметров *R* и *r* от номера шага *n*. Расчеты проводились для следующих значений параметров: коэффициент дисконтирования 𝛂 = 0,9; стоимость единичной поставки товара *А* = 100; цена единицы товара *c* = 10; стоимость хранения единицы товара *h* = 1; потери в следствии дефицита единицы товара *d* = 20. Рассматривались три основных распределения спроса: (а) Гауссово, (б) экспоненциальное и (в) равномерное. Для нормального распределения математическое ожидание равнялось 𝜇 = 5, а среднеквадратическое отклонение 𝛿 = 1. Для экспоненциального распределения параметр 𝜆= 0,2. Для равномерного распределения левая граница a = 0, права граница *b* = 10.

Были обнаружены следующие любопытные закономерности. Быстрее всего параметры *Rn* и *rn* сходились для равномерного распределения спроса – уже на 15-ом шаге. В случае с экспоненциальным законом распределения значение параметров приблизились к своим предельным значениям на 25-ом шаге, и дальше не сильно колебались до 40-го шага. Наиболее «турбулентно» вели себя параметры модели с нормальным распределением спроса. На графике на рис. 25 видны сильные колебания значения *Rn* вплоть до дватцатого шага и незначительные колебания вплоть до пятидесятого шага.

Рис. . Графики функций параметров *Rn* и *rn* для случаев экспоненциального, гауссова и равномерного распределений спроса. Параметры модели: *A* = 100; *c* = 10; *h* = 1; *d* = 20; 𝛂 = 0,9; *n* = 50.

Рис. . Графики функций параметров *Rn* и *rn* для случаев экспоненциального, гауссова и равномерного распределений спроса. Параметры модели: *A* = 100; *c* = 10; *h* = 1; *d* = 20; 𝛂 = 1; *n* = 50.

Так же было любопытно отметить, что параметры сходились тем медленнее, чем больше были значения стоимости единичной поставки *A* или коэффициента дисконтирования 𝛂. В случае больших *A* это объясняется тем, что с увеличением стоимости поставки единичного заказа система начинает делать выбор в пользу больших по размерам и более редким заказам. В случае с большим коэффициентом дисконтирования мы заглядываем всё дальше в будущее, что увеличивает значение параметра *Rn* и разность *Rn* - *rn* равную максимально-допустимому размеру заказа.

Практика моделирования показала, что *N* равное 50 будет достаточным для большинства случаев. В наших же моделях был реализован подход динамической остановки цикла по *n*, давший сокращение скорости вычислений в полтора раза, и подробно описанный в разделе 5.3.

## Выбор стратегии управления запасами при случайном спросе и изменении эконометрических параметров базовой модели

Содержание этого пункта следует результатам, полученным в п. 2.6, и развивает результаты из работ [3], [4], [6], [14] и [147].

Как видно из оценок, полученных в п. 2.6, если считать, что реальное распределение спроса за один шаг может быть произвольным, варьируя от гауссова через равномерное к экспоненциальному, то в любом случае примерно к 15-иу шагу оценки *Rn* и *rn* сходятся к своим предельным значениям *R* и *r*. Представляет интерес сравнить, насколько разойдутся значения средних затрат за период планирования продолжительностью в 30–180 шагов[[7]](#footnote-7), если мы будем использовать оптимальную динамическую и стационарную стратегии. Первые из них характеризуется тем, что используются динамические параметры *Rn* и *rn*, а вторая – стационарные *R* и *r*.

Из приведенных в разделе 2.2 графиков на рис. 7 - рис. 9 и табл. 2 следует, что можно заметно упростить практические правила выбора размера заказов, заменяя динамически изменяющиеся параметры на всего 2 стационарных параметра *R* и *r*, проиграв при этом оптимальной «дальнозоркой» (динамической) стратегии всего 10-15% от суммы затрат. Напомним, что выигрыш от использования стационарной «дальнозоркой» стратегии по сравнению с «близорукой» может составить порядка 25%. Это позволяет сделать вывод о том, что стационарные стратегии, как значительно более простые можно использовать с начала периода планирования до его конца, по-прежнему рассчитывая на существенный экономический эффект.

Итак, считая, что для управления при заданных значениях эконометрических параметров *A*, *c*, *h*, *d* и коэффициента дисконтирования *α* используются рассчитанные именно для них пары значений уровней *R* и *r*, обсудим, как придется поступить лицу, принимающему решения (ЛПР), если ему станет известно, что значения параметров *A*, *c*, *h*, *d* и коэффициента дисконтирования *α* в какой-то момент времени *t* (это *t*, как и номер шага *n*, является целочисленным) в будущем могут измениться.

Пока будем считать, что справедливо исходное предположение о том, что время запаздывания поставки *θ* = 0. В этом случае такое изменение параметров *A*, *c*, *h*, *d* и коэффициента дисконтирования *α*, которое приводит к увеличению значения *R*, не создает никакой экстремальной ситуации. В самом деле, если это значение *R* больше только что использовавшегося значения *R*, то в силу условий задачи мы сможем достичь его, мгновенно заменив старое значение на новое (как и значения *r*).

Другое дело, если новое *R* оказалось меньше (особенно, гораздо меньше), чем его прежнее значение. Тогда следует сопоставить потери от того, что мы войдем в новый интервал времени, стартующий в момент *t*, с остаточным запасом, являющийся производным от прежнего уровня поставки *R*, с убытками от того, что будут предприняты специальные меры по искусственному сокращению запаса к уровню, продиктованному новым значением *R*. Для этого воспользуемся результатами, полученными А.А. Первозванский в монографии [6]. Заданные значения *R* и *r* порождают в системе то, что А.А. Первозванский назвал циклом поставки. Цикл поставки – это интервал времени между соседними поставками. Естественно, что это величина случайная. При этом, как показано в [6], продолжительность цикла поставки является асимптотически нормальной случайной величиной с очень малым разбросом (дисперсия равна *σ*2/*m*2, где *σ* – среднеквадратическое отклонение спроса за один шаг, *m* – математическое ожидание спроса за один шаг, а – продолжительность цикла, см. ниже)[[8]](#footnote-8). Например, если для среднеквадратического отклонения спроса за один шаг *σ* и его математического ожидания *m* справедливо соотношение *σ*/*m* = 0,1, то вероятность того, что продолжительность цикла отклонится от его среднего значения больше, чем на 1, не превосходит 0,003. При этом средняя продолжительность цикла составляет с пренебрежимо малыми отклонениями для больших *A* и *d* величину, равную . Выражение в числителе этого соотношения – не что иное как знаменитая формула Уилсона [3]. Важно понимать, что постулированные выше большие значения *R* и *r* неминуемо, см. п. 2.4, приводят к большому (иногда, очень большому) значению *R*.

Из приведенных выше рассуждений следует, что ЛПРнужно сравнивать 2 варианта действий: (а) переход в следующий качественно отрезок времени, начиная с момента *t*, с остаточным (переходящим) запасом, продиктованным прежним уровнем поставки *R*, или (б) приостановка поставок за какое-то число шагов до момента *t*, чтобы к моменту *t* текущий уровень запасов упал до примерно нового значения *R*. И выбрать из этих вариантов тот, что приносит меньшие убытки.

Проиллюстрируем эту процедуру на конкретном примере. Представим, что мы – заведующий складом (лицо, принимающее решения) в городе, где оперирует единственный монополист поставщик логистических услуг. На текущий период планирования равный 30 дням (шагам) у нас выбрана стратегия управления запасами и рассчитаны оптимальные параметры *R1* и *r1* для набора эконометрических характеристик: *A1* = 300; *c1* = 10; *h1* = 1; *d1* = 60; 𝛂 = 0,99. В некоторый момент мы получаем инсайдерскую информацию, что одна крупная транснациональная логистическая корпорация в рамках политики расширения и захвата новых рынков откроет отделение в вашем городе в середине текущего периода. При этом у корпорации достаточно средств для выдавливания конкурентов с рынка путём демпинга цен на перевозки. Мы знаем, что через 15 дней, после начала периода планирования начнут действовать новые эконометрические параметры *A2* = 100; *c1* = 10; *h1* = 2; *d1* = 20. То есть, цена на единичную поставку упадёт в три раза (а вместе с ней и потери в следствии дефицита) и вырастет стоимость хранения единицы товара в единицу времени из-за хлынувшего в город потока товаров (новых складов при этом не было построено). Новые условия порождают новые оптимальные параметры *R2* и *r2* для второго полупериода планирования. При этом *R2* примерно в два раза меньше *R1*.

Рассмотрим несколько вариантов действий, доступных лицу, принимающему решения в рамках данной модели. Вариант 1 – использование стационарной стратегии управления запасами, где первые 15 дней склад использует значения *R1* и *r1*, а потом переходит на значения *R2* и *r2*. Вариант 2 - использование стационарной стратегии управления запасами, где за шагов до начала второго полупериода склад переходит на значения *R2* и *r2*. Моделирование показало, что: постепенный переход от значения *R1* к значению *R2* оказался более дорогим, а значение оказалось оптимальным. Вариант 3 – использование «дальнозоркой» стратегии управления запасами, где на каждом полупериоде используются свои значения *Rn* и *rn*. Вариант 4 – использование «дальнозоркой» стратегии управления запасами, где сначала используются значения *Rn* и *rn*, рассчитанные для первоначального набора параметров и за шага (значение оказалось оптимальным при моделировании) до конца первого полупериода склад переходит на динамические значения *Rn* и *rn*, рассчитанные для второго полупериода.

Результаты моделирования представлены на графиках на рис. 27 - рис. 29 и в табл. 4. Моделирование проводилось для трёх функций распределения с одним и тем же средним значением спроса равным 5.

Рис. . Функции суммарных средних затрат *C*(*x*) для четырёх вариантов стратегий управления запасами. Случай Гауссова распределения спроса.

Рис. . Функции суммарных средних затрат *C*(*x*) для четырёх вариантов стратегий управления запасами. Случай экспоненциального распределения спроса.

Рис. . Функции суммарных средних затрат *C*(*x*) для четырёх вариантов стратегий управления запасами. Случай равномерного распределения спроса.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Гауссово** | **Экспон.** | **Равномерное** |
| **Вариант 1 → Вариант 2** | 1,6% | 2,2% | 1,7% |
| **Вариант 1 → Вариант 3** | 4,8% | 4,9% | 5,2% |
| **Вариант 1 → Вариант 4** | 6,5% | 8,6% | 7,1% |
| **Вариант 3 → Вариант 4** | 1,9% | 3,9% | 2,0% |

Табл. . Значения уменьшения суммарных средних затрат в процентах при переходе от использования одной стратегии управления запасами к другой для трёх функций распределения спроса.

Из результатов моделирования видно, что выгоднее заранее начать переход на новые параметры *R* и *r* (вариант 1 → вариант 2), правда экономия суммарных средних затрат составит всего 1,6% – 2,2%. При этом для «дальнозоркого» случая (вариант 3 → вариант 4) этот показатель составляет 1,9% – 3,9%. При этом стоит отметить, что использование более сложной «дальнозоркой» (динамической) стратегии управления запасами по сравнению со стационарной (вариант 1 → вариант 3) даст экономию суммарных средних затрат порядка 5%.

Что делать, если время поставки *θ* ≠ 0 и это время, по-прежнему, детерминировано (будем для простоты считать его целочисленным)? Тогда следуя [3], реальный момент подачи заказа отстоит от текущего момента времени *n* на величину *θ*. Потому текущие значения *R* и *r* надо запускать в действие за *θ* шагов до начала соответствующего отрезка времени, если текущее значение *R* больше предыдущего. А, если, напротив, меньше предыдущего, то ситуация снова сводится к принятию решений по результатам, описанной выше процедуре сравнительного имитационного моделирования.

Наконец, если время поставки случайно, то следует воспользоваться методикой, изложенной в работе [147], в которой введено понятие локального дефицита и на основе этого понятия сформированы соответствующие расчетные соотношения.

## Выводы по главе 2

1. Выполнены обсуждение и критический анализ базовой модели управления запасами и указаны области ее применения.
2. Выполнено сравнение эффективности использования «дальнозорких», стационарных и «близоруких» стратегий управления запасами. Анализ показал, что выигрыш по уровню средних суммарных затрат при использовании стационарных стратегий вместо «близоруких» (широко применяемых на практике) может составлять порядка 25% при типичных для практических проблем управления запасами соотношениях параметров затрат.
3. Разработан программный комплекс, позволяющий рассчитывать значения двух параметров стационарной стратегии оптимального управления запасами для трёх функций распределения спроса: (а) Гауссова распределения, (б) экспоненциального распределения и (в) равномерного распределения. Данный комплекс позволяет строить параметрические «атласы» для СППР в системах управления запасами и производством.
4. С использованием этого комплекса выполнено исследование зависимости значений параметров стационарных стратегий управления запасами от таких эконометрических характеристик системы управления запасами и внешней среды, как: коэффициент дисконтирования α, стоимость хранения единицы товара *h*, коэффициента потерь вследствие дефицита товара *d*, закупочной цены единицы товара *c*, фиксированных затрат на единичную поставку *A*.
5. Дано объяснение «лестничного» характера поведения зависимости верхнего уровня стационарной стратегии *R* от параметров системы и среды. Это объяснение опирается на выявленную аналогию между детерминированной моделью управления запасами и моделями управления запасами при случайном спросе. Представлены результаты моделирования системы управления запасами в дискретном времени. Моделирование позволило оценить роль случайных возмущений в эволюции системы и обнаружить, что в характере изменения параметров оптимальных стратегий возможно наличие флуктуаций, обусловленных наличием острых пиков у реальных вероятностных распределений спроса.
6. Проанализирована скорость сходимости параметров *R*, *r* к своим предельным значениям для трёх функций распределения спроса.
7. Для условий параметрической неопределенности даны рекомендации по выбору стратегии управления запасами при случайном спросе и изменениях эконометрических параметров базовой модели.

# Модели управления запасами при случайном спросе в условиях срывов в процессе поставок

## Проблематика управления запасами при случайном спросе и ненадежных поставщиках

В последнее десятилетие в области теории и приложений управления производственно-складскими системами все большее число работ посвящено решению проблем, собранных под общим наименованием «управление цепями поставок» (supply chain management) [148] – [149] [150] [151] [152]. Интерес к этому классу задач обусловлен прикладной актуальностью этой темы: все большее число предлагаемых моделей, алгоритмов и решений либо изначально разрабатывались для принятия решений в конкретных практических ситуациях, либо рано или поздно находят свои приложения.

Теоретические постановки этих задач, а также адекватный математический аппарат для их решения связаны с детальным исследованием ситуаций оперативного управления запасами, продиктованных различными сбоями (в том числе, и случайной природы) на запланированных ранее субъектах поставки. Одним из основных источников таких сбоев является ненадежность поставщиков, т.е. наличие ненулевой вероятности того, что поставщик сорвет заказанную ему поставку.

Последний класс задач весьма близок к классической проблематике современной теории управления запасами, в которой, начиная с 70-х годов прошлого века, исследовались задачи управления запасами с ненадёжными поставщиками. Пионером этих исследований был В.А. Лотоцкий, который в своей кандидатской диссертации 1972 г. первым в мире предложил рассматривать случай, когда величина поставки описывается условным распределением вероятностей при заданном размере заказа. Впрочем, вскоре стало понятным, что такое описание трудно выполнимо практически [153] и поэтому стали исследоваться системы с экстренными более дорогими поставками, см., например, [8]. По состоянию на начало 90-х годов прошлого века эти исследования подытожены в книге [15]. Затем, как отмечалось во введении к диссертации, это направление в связи со сдвигом интересов в сторону *ERP*-систем было «приброшено», а исследования на эту тему возобновилось только во второй декаде 2000-х годов [35] – [36] [37].

В данной главе, которая развивает результаты работ [35] – [36] [37], [154], рассматривается одна из таких задач, а именно задача управления запасами при случайном спросе при наличии альтернативных поставщиков с разными уровнями надёжности. Представляется, что именно такая модель с возможными ее модификациями может быть заложена с основу процедур проведения тендеров при выборе наилучших поставщиков, а также быть использована как теоретическое обоснование правил проведения тендеров и при оперативном управлении процессом поставок.

## Исходные положения для построения модели оптимального выбора стратегий управления запасами при случайном спросе и ненадежных поставщиках

Рассматривается однономенклатурная система управления запасами в дискретном времени на периоде планирования *T* = *Nτ*. Здесь *τ* – период контроля состояния запасов. Моментами принятия решений о размере заказов являются дискретные моменты времени *tk* = *kτ*, *k* = 1, 2, …, *N* – 1. Весь неудовлетворенный сразу потребительский спрос учитывается. В качестве правила оценки эффективности системы выбран критерий минимума суммарных средних затрат на периоде планирования *T*. Интервал длительности *τ* называется шагом процесса управления. В начальный момент каждого такого интервала, шага, измеряется значение фиктивного уровня запасов *x*, и на основании результатов измерения следует принять решение о необходимости подачи (или не подаче) заказа на пополнение запасов (который поступит в систему через случайное время *θ*) и размере заказа *u*.

В суммарные затраты входят затраты на пополнение запасов, затраты на хранение и потери вследствие дефицита. В силу дискретности измерения времени считается, что на очередном шаге периода планирования в конце его, завершаемом с уровнем фиктивного запаса *y*, выплачиваются затраты на хранение в размере *hy*, если *y* > 0, или несутся потери вследствие дефицита в размере – *dy*, если *y* < 0.

Спрос на *k*-м шаге (в интервале между (*k* – 1)-м и *k* –м моментами контроля описывается случайной величиной *zk*. Предполагается, что случайные величины не зависимы в совокупности и имеют одно и то же вероятностное распределение с функцией распределения *F*(*z*) с плотностью вероятности *f*(*z*).

Затраты на пополнение запасов в размере *u* связаны с подачей заказа одному из *M* поставщиков[[9]](#footnote-9). Эти затраты при подаче заказа *j*-му поставщику описываются функцией *Aj***1**(*u*) + *cju*, где *Aj* – фиксированная часть затрат на поставку у *j*-го поставщика, **1**(*u*) – функция Хэвисайда (единичного скачка), которая равна 1 для положительных *u* и 0 для неположительных *u*, а *cj* – закупочная цена ед. продукции у *j*-го поставщика. Предполагается, что поставщики ненадежны и для каждого из них существует вероятность *pj* того, что *j*-й поставщик полностью сорвет поставку (соответственно, с вероятностью 1 –*pj* поставка будет обеспечена). Считается, что сорвавший пополнение запасов поставщик не получает платы за поставку. При этом считается, что время запаздывания поставки *θ* у всех поставщиков равно 0, поэтому при срыве поставки одним из поставщиков заказ на поставку тут же передается другому поставщику. Таким образом, каждый из поставщиков характеризуется тройкой (*Aj*, *cj*, *pj*).

В дальнейшем будет рассмотрено 2 варианта предложенной постановки задачи: (а) когда все величины *Aj* равны между собой, т.е. *Aj* = *A*, и (б) когда все *Aj* – разные. Как будет ясно из дальнейшего, в варианте (а) удается получить гораздо более конструктивные результаты. Однако более глубокой причиной выделения варианта (а) является то, что он гораздо чаще встречается на практике. Дело в том, что, как правило, фиксированная часть затрат на пополнение запасов связана с необходимостью использования транспортных средств для перевозки на склад потребляемой продукции, которые заказываются ее получателем. При этом цена на транспортное средство оказывается для всех поставщиков одной и той же.

## Модель оптимального выбора размеров поставок при случайном спросе в условиях ненадежности поставщиков

Предположим, что за *n* шагов (длительности *τ*) до конца периода планирования фиктивный уровень запаса в системе равен *x* и принимается решение о подаче заказа размера *u*. Тогда средние одношаговые затраты на шаге между моментами *tn* и *tn*–1 (в обратном дискретном времени) составят величину

|  |  |
| --- | --- |
|  | (32) |

где *c* определяется по формуле (35).

Если ввести функцию минимально возможных средних затрат для *n*-шагового процесса, который начинается с фиктивного уровня запасов *x*, обозначив ее , то нетрудно убедиться в том, что эта функция удовлетворяет следующему уравнению дискретного динамического программирования:

|  |  |
| --- | --- |
| , | (33) |

с начальным условием:

|  |  |
| --- | --- |
| . | (34) |

Как следует из вида уравнений (33) – (34), для них справедливы все выводы теории управления запасами [3], и, в частности, факт оптимальности двухуровневых (*R*, *r*)-стратегий управления запасами. Иначе говоря, для каждого натурального *n* существуют такие 2 числа *rn* и *Rn*, что *rn* < *Rn* , и оптимальный размер заказа за *n* шагов до конца периода планирования при текущем фиктивном уровне запаса *x* вычисляется по формуле (1). При этом существуют пределы при *n* → ∞ монотонно возрастающих последовательностей {*Rn*} и {*rn*}. Обозначим эти пределы, параметры оптимальной стационарной управления запасами, *R* и *r*.

## Формирование правил рационального выбора поставщиков

**Вариант А: случай постоянства фиксированной части затрат на поставку у разных поставщиков**

В этом случае каждый из поставщиков характеризуется парой (*cj*, *pj*). Очевидно, что, если найдутся два поставщика с номерами *i* и *j*, для которых одновременно выполнены следующие два неравенства: *cj* ≥ *ci* и *pj* ≥ *pi*, то поставщик под номером *j* заведомо проигрывает поставщику *i* (он и дороже, и менее надежен). Таким образом, как отмечено в [154], если в качестве критерия выбран минимум суммарных средних затрат, то в множестве *M* поставщиков следует выделить подмножество Парето-оптимальных поставщиков, удалив из списка поставщиков всех поставщиков, у которых пара (*cj*, *pj*) обладает тем свойством, что найдется другой поставщик под номером *i*, для которого пара (*ci*, *pi*) обладает тем свойством, что *cj* ≥ *ci* и *pj* ≥ *pi*, причем хотя бы одно из этих неравенств строгое. Итак, в дальнейшем будем считать, что множество поставщиков *L* является подмножеством Парето. Иначе говоря, для всех *i*, *j*  {1, 2, …, *L*} справедливы пары неравенств *cj* ≥ *ci* и *pi* ≥ *pj* либо *cj* ≤ *ci* и *pi* ≤ *pj*. Причем будем считать, что номера поставщиков упорядочены по возрастанию цены ед. товара, т. е. . Будем также считать, что *pL* = 0, т.е. среди поставщиков имеется, по крайней мере, один абсолютно надежный (разумеется, с самой высокой ценой товара).

**Теорема** **2**. На каждом шаге оптимальная последовательность обращений к поставщикам совпадает с их нумерацией (т.е. сначала заказчик обращается к поставщику 1, затем при его неудаче – к поставщику 2,… вплоть до обращения к последнему поставщику под номеров *L* из подмножества Парето.

*Доказательство*.

При использовании критерия минимальных суммарных средних затрат последовательность обращений к поставщикам сказывается на каждом шаге на значении цены заказываемого товара. При том способе нумерации поставщиков, который назван оптимальным в формулировке теоремы 2, средняя цена поставляемого товара составит величину:

|  |  |
| --- | --- |
| . | (35) |

Докажем, что при любом другом упорядочении поставщиков (*i*1,*i*2,…,*iL*), где (*i*1,*i*2,…,*iL*) – произвольная перестановка множества номеров (1,2,…,*L*), для средней цены товара при его заказе может только возрасти, т.е. *с*(1,2,…,*L*) ≤ *с*(*i*1,*i*2,…,*iL*). Здесь *с*(*i*1,*i*2,…,*iL*) вычисляется по формуле, аналогичной (35):

|  |  |
| --- | --- |
| (*i*1,*i*2,…,*iL*) = (1 – *pi*1)*ci*1 + *pi*1(1 – *pi*2)*ci*2 + *pi*1*pi*2 (1 – *pi*3)*ci*3…+ *pi*2*pi*1… *piL*-1*сL* . | (36) |

Докажем, сначала более простой факт: если в перестановке (*i*1,*i*2,…,*iL*) встретились по соседству, начиная с произвольного места под номером *l*: 1 ≤ *l* ≤ *L*-1, два номера таких, что для них *cil* ≥ *cil+*1, то, поменяв местами этих поставщиков, среднюю цену ед. товара можно только уменьшить. Обозначим для простоты *cil* = *a*, *cil+*1 = *b*, *pi*2*pi*1… *pil-*1 = D, *pil* = *d* и *pil+*1 = *e*. При такой перемене мест двух соседних элементов в сумме, определяемой формулой (36) изменятся только два слагаемых – *l*-е по счету и (*l* + 1)-е. Используя введенные обозначения, запишем значения этих слагаемых до и после перемены мест. До перемены мест обозначим сумму этих двух слагаемых *Y*, а после перемены мест *Z*. Тогда

|  |  |
| --- | --- |
| *Y = D*(1 – *d*)*a* + *Dd*(1 – *e*)*b* | (37) |
| *Z = D*(1 – *e*)*b* + *De*(1 – *d*)*a* | (38) |

Подсчитаем разность *Z – Y*:

|  |  |
| --- | --- |
| *Z – Y = D*(1 – *e*)*b* + *De*(1 – *d*)*a – D*(1 – *d*)*a* – *Dd*(1 – *e*)*b =  D*(1 – *d*)(1 – *e*)(*b* – *a*) | (39) |

В силу того, что, по предположению, *b* ≤ *a*, разность *Z – Y* ≤ 0. Теорема доказана.

Итак, установлен следующий факт: любая перестановка двух соседних (по последовательности обращения к ним) поставщиков, у которых цены (по величине) «перепутаны» приводит к снижению средней цены товара. Естественно, верно, и обратное: если двух соседних поставщиков с «правильно» упорядоченными цена поменять местами, то средняя цена увеличиться.

Отсюда следует утверждение теоремы, поскольку произвольная индексация поставщиков в последовательности (*i*1,*i*2,…,*iL*) может быть получена из «правильной» индексации (1,2,…,*L*) посредством выполнения конечного числа «плохих» попарных перестановок. Действительно, сначала поставщик под номером *i*1 переставляется на первое место ((*i*1 – 1) «плохих» попарных перестановок), затем с помощью только «плохих» перестановок на второе место ставится поставщик под номером *i*2 и т. д., пока на последнем месте не окажется поставщик под номером *iL*.

**Вариант Б: случай изменения фиксированной части затрат на поставку у разных поставщиков**

В этом случае, как и в варианте (а), последовательность выбора поставщика обусловлена тем, что размер платежа поставщику должен быть как можно меньше. Однако та логика, которой можно было руководствоваться в варианте А (выделение подмножества Парето-оптимальных поставщиков и последующие действия), перестает работать. Правда, можно исключить поставщиков с характеристической тройкой параметров (*Ai*, *ci*, *pi*), для которых найдется такой поставщик под номером *j*, у которого *Aj*≤*Ai*, *cj*≤*ci* и *pj ≤ pi*. Считая, что подобное «прореживание» выполнено, воспользуемся результатом предыдущего раздела, который в данном случае будет использоваться как следующее эвристическое правило.

**Эвристическое правило:** последовательность обращения к поставщикам обуславливается их упорядочением по возрастанию затрат на поставку размера *u*, т.е. в данном случае последовательность обращений зависит от размера заказа *u*.

Итак, каждое упорядочение поставщиков (*i*1(*u*), *i*2(*u*), …, *iL*(*u*)) строится по возрастанию величины *Aj*1(*u*) + *cju*. При этом одношаговые затраты определяются не по формуле (32), а с помощью формулы, комбинирующей формулы (36) и (32):

|  |  |
| --- | --- |
|  | (40) |

где

|  |  |
| --- | --- |
| *A*(*u*) = (1 – *pi*1(*u*))*Ai*1(*u*) + *pi*1(*u*) (1 – *pi*2(*u*))*Ai*2 + *pi*1(*u*)*pi*2(*u*)(1 – *pi*3(*u*))*Ai*3+…  + *pi*1(*u*)*p*i2(*u*) … p*iL*-1(*u*)*AL*, | (41) |

а

|  |  |
| --- | --- |
| *с*(*u*) = (1 – *pi*1(*u*))*ci*1(*u*) + *pi*1(*u*) (1 – *pi*2(*u*))*ci*2 + *pi*1(*u*)*pi*2(*u*) (1 – *pi*3(*u*))*ci*3+…  + *pi*1(*u*)*p*i2(*u*) … p*iL*-1(*u*))с*L*. | (42) |

Функции *ϕ*(*x*,*u*) из формул (40) – (42) подставляются в алгоритм (33) – (34), который и позволяет рассчитать оптимальные размеры заказов.

## Компьютерное моделирование

Для моделирования были выбраны 4 поставщика, каждый из которых характеризуется комбинацией: *c* цены единицы товара и *p* вероятностью полностью сорвать поставку. Рассматривалось два случая – оптимальный и обратный оптимальному порядок выбора поставщиков. Оптимальный порядок выбора поставщиков: (0,5;0,8), (1;0,5), (2;0,1), (4;0). Обратный оптимальному порядок выбора поставщиков, соответственно: (4;0), (2;0,1), (1;0,5), (0,5;0,8). Средняя цена единицы товара по формуле (35) будет равна 1,38 для оптимального порядка и 4 для обратного оптимальному порядка выбора поставщиков. Для обратного оптимальному порядка это означает, что товар будет закупаться только у самого надёжного поставщика по самой высокой цене. Для моделирования использовались три функции распределения спроса: (a) экспоненциальное распределение с параметром 𝜆= 0,2; (б) нормальное распределение с математическим ожиданием 𝜇 = 5 и среднеквадратическим отклонением 𝛿 = 1; и (в) равномерное распределение на отрезке от *a* = 0 до *b* = 10.

Рассмотрим зависимостьпараметров *R*(𝛂), *r*(𝛂) оптимальной стратегии управления запасами от коэффициента дисконтирования 𝛂. Входные данные для этой модели были выбраны следующим образом: стоимость единичной поставки *A* = 20; стоимость хранения единицы товара *h* = 1; удельные потери вследствие дефицита товара *d* = 40; коэффициент дисконтирования 𝛂 = 0 ÷ 1. Ниже представлены результаты моделирования.

Рис. . Графики *R*(𝛂) и *r*(𝛂) для экспоненциального распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Рис. . Графики *R*(𝛂) и *r*(𝛂) для гауссова распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Рис. . Графики *R*(𝛂) и *r*(𝛂) для равномерного распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

На рис. 30 - рис. 31 можно наблюдать следующую картину – значения параметров *R*(𝛂) и *r*(𝛂) для оптимального порядка находятся выше, чем для обратного. По мере увеличения 𝛂 растут значения *r*(𝛂) и к 𝛂 = 1 они почти совпадают. Это говорит о том, что при малых 𝛂 экономически допустим больший уровень дефицита товара, при чём для обратного порядка выбора поставщиков дефицит более приемлем. Действительно, с ростом 𝛂 мы всё дальше «заглядываем» в будущее и не ограничиваемся ближайшим шагом принятия решения, это делает дефицит в системе менее рентабельным, так как излишки товара, которые не были реализованы на ближайшем шаге можно реализовать на следующих.

Рис. . Графики функций минимального размера заказа *umin*(𝛂) для трёх распределений спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

На рис. 33 для удобства были вынесены отдельно графики зависимостей *u*(𝛂) размера заказа. Хорошо видно, что размер заказа растёт с увеличением коэффициента дисконтирования, однако этот рост не монотонен. Рост размера заказа объясняется всё тем же «заглядыванием» дальше в будущее, что приводит к логическому выводу о необходимости рассчитывать размера заказа не только лишь на один ближайший шаг, а как минимум на несколько шагов вперёд. Интересно, что для «близорукого» случая (при 𝛂 = 0) оптимальный порядок поставщиков характерен меньшим размером заказа, чем обратный порядок. Далее с ростом 𝛂 размер заказа для обратного порядка поставщиков будет меньше, чем для оптимального и при 𝛂 = 1 они будут примерно совпадать.

Перейдём к рассмотрению зависимости параметров *R*(*A*), *r*(*A*) оптимальной стратегии управления запасами от стоимости единичной поставки *A*. Входные данные для этой модели были выбраны следующим образом: стоимость единичной поставки *A* = 0 ÷ 100; стоимость хранения единицы товара *h* = 1; удельные потери вследствие дефицита товара *d* = 40; коэффициент дисконтирования 𝛂 = 0,9.

Рис. . Графики *R*(*A*) и *r*(*A*) для экспоненциального распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Рис. . Графики *R*(*A*) и *r*(*A*) для гауссова распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Рис. . Графики *R*(*A*) и *r*(*A*) для равномерного распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

На рис. 34 - рис. 36 видно, что с ростом стоимости единичной поставки *A* уменьшается значение *r*(*A*), то есть сужается отрезок уровня запасок, на котором целесообразно подавать заказ и при достаточно больших A может даже уйти в отрицательную зону, позволяя тем самым небольшой дефицит товара в системе. Значения *R*(*A*) монотонно увеличиваются по мере роста значений *А*, что свидетельствует о том, что выгоднее делать заказы большего размера, так как стоимость единичной поставки постепенно перекрывает расходы на хранение товара. Оптимальный порядок поставщиков характерен большим размером заказа по сравнению с обратным оптимальному. В целом это логично, так как с увеличением цены при обратном порядке выбора поставщиков становиться экономически выгоднее делать более маленькие заказы.

Рис. . Графики функций минимального размера заказа *umin*(*A*) для трёх распределений спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Если сравнивать графики функций *umin* для оптимального и обратного оптимальному порядка выбора поставщиком на рис. 37 можно увидеть следующую картину: нижняя граница размера заказа находиться в среднем выше для оптимального порядка. По мере роста *A*, скорость роста *umin* для оптимального порядка выше, чем для обратного. Это можно объясняется более низкой ценой единицы товара для случая оптимального порядка выбора поставщиков.

Теперь рассмотрим зависимости параметров *R*(*h*), *r*(*h*) оптимальной стратегии управления запасами от стоимости хранения единицы товара *h*. Входные данные для этой модели были выбраны следующим образом: стоимость единичной поставки *A* = 100; стоимость хранения единицы товара *h* = 0 ÷ 20; удельные потери вследствие дефицита товара *d* = 40; коэффициент дисконтирования 𝛂 = 0,9.

Рис. . Графики *R*(*h*) и *r*(*h*) для экспоненциального распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Рис. . Графики *R*(*h*) и *r*(*h*) для гауссова распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Рис. . Графики *R*(*h*) и *r*(*h*) для равномерного распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

На рис. 38 - рис. 40 видно, что значения *r*(*h*) для оптимального порядка выбора поставщиков находятся в среднем выше, чем для обратного оптимальному порядка. Что объясняется более высокой ценой единицы товара для случая обратного оптимальному порядка. С ростом значения *h* уменьшается значение функции *r*(*h*), постепенно заходя в отрицательную зону, тем самым разрешая небольшой дефицит товара в системе, что вполне логично. Поведение функции нижней границы заказа *umin*(*h*) представлено на рис. 41.

Рис. . Графики функций минимального размера заказа *umin*(*h*) для трёх распределений спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Нижняя граница заказа находиться в среднем выше для оптимального порядка поставщиков, чем для обратного, что объясняется меньшей ценой за единицу товара. Однако эта разница постепенно уменьшается с ростом *h*, так как стоимость единицы товара вносит меньший вклад относительно других компонентов затрат – возросшей стоимостью хранения, в данном случае. Поведение серий оптимального и обратного порядков для случая нормального распределения спроса ведёт себя скачкообразно и не так однозначно из-за большей «детерминированности» этого конкретного распределения, что было описано в разделе 2.5.

Далее рассмотрим зависимости параметров *R*(*d*), *r*(*d*) оптимальной стратегии управления запасами от коэффициента *d* удельных потерь вследствие дефицита товара. Входные данные для этой модели были выбраны следующим образом: стоимость единичной поставки *A* = 20; стоимость хранения единицы товара *h* = 1; удельные потери вследствие дефицита товара *d* = 0 ÷ 100; коэффициент дисконтирования 𝛂 = 0,9.

Рис. . Графики *R*(*d*) и *r*(*d*) для экспоненциального распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Рис. . Графики *R*(*d*) и *r*(*d*) для гауссова распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Рис. . Графики *R*(*d*) и *r*(*d*) для равномерного распределения спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

Рис. . Графики функций минимального размера заказа *umin*(*d*) для трёх распределений спроса при оптимальном и обратном оптимальному порядке выбора поставщиков.

На рис. 42 - рис. 44 видно, что при значении *d* < 1 модель не может рассчитать значения *R*(*d*) и *r*(*d*), так как такая система не имеет смысла – дефицитом можно просто пренебречь. При небольших значениях *d* значения *r*(*d*) принимают резко отрицательные значения, тем самым в системе устанавливается режим, при котором экономически выгоднее поддерживать дефицит товара, чем платить за его хранение. По мере роста значения *d* растёт и значение *r*(*d*) постепенно запрещая дефицит.

При рассмотрении поведения функции *umin*(*d*) нижней границы заказа на рис. 45 можно убедиться, что в среднем значение *umin* выше для оптимального порядка выбора поставщиков, чем для обратного. Это, опять же, объясняется более низкой ценой единицы товара для оптимального порядка. В области, где значения *d* достаточно малы для допущения дефицита товара в системе можно наблюдать более резкие изменения соотношения значений *umin* для двух порядков выбора поставщиков, однако ситуация стабилизируется при больших значениях *d*. То есть, когда потери вследствие дефицита оказываются дороже расходов на хранение товаров.

## Выводы по главе 3

1. Рассмотрена задача управления запасами, связанная с оптимизацией процессов, которые возникают в системе управления запасами при случайном спросе и при наличии нескольких альтернативных поставщиков, обладающих разной степенью надежности и различными эконометрическими характеристиками. Исследовано два случая: (а) фиксированная часть затрат на поставку у всех поставщиков одна и та же при не совпадающих ценах на ед. товара и (б) все компоненты затрат у разных поставщиков различны. Для случая (а) предложены конструктивные алгоритмы построения оптимальных стратегий управления запасами и выбора поставщиков. В случае (б) предложена общая схема формирования оптимальных стратегий.
2. Сформулирована и доказана теорема о том, что при изменении порядка выбора поставщиков (относительно оптимального) будет возрастать средняя цена поставки, что в свою очередь будет вызывать уменьшение значений оптимальных параметров двухуровневой стратегии управления запасов *R* и *r* (а также снижение значения *R* ***–*** *r*), что на практике приводит к уменьшению размера заказов на каждом шаге (подтверждено при моделировании).

# Модели управления запасами при случайном спросе и отказах потребителей от поставленной им продукции

## Модель управления запасами с возвратами при случайном спросе

В последнее десятилетие было опубликовано несколько работ по управлению системами массового обслуживания (СМО) [38], [155] – [156] [157]. Рассматривались СМО, в которых в периодически расположенные моменты контроля можно было изменять число рабочих каналов обслуживания. Уже в работе [38] было подмечено, что модели оптимизации числа рабочих каналов обладают несомненным сходством с моделями теории управления запасами [3], [15]. В управляемых СМО выбираемые решения сводились к выбору между включением дополнительных каналов или отключением лишних каналов, то есть выбираемые решения были решениями двух типов. В отличие от них в рамках моделей теории управления запасами (ТУЗ) все решения оказывались «позитивными». Требовалось определить, каким должен быть размер заказа продукции для поставки на склад.

При этом математические модели ТУЗ и порождаемые ими стратегии управления обладали несомненной привлекательностью. Эти стратегии отказывались параметрическими и получили наименование (*R*, *r*)-стратегий (*R* > *r*). Здесь уровень *R* определяет размер заказа, а уровень *r*, получивший название точки заказа – уровень запаса, при достижении которого (в сторону снижения запаса), нужно подавать заказ на пополнение запасов.

Необходимость не только включать, но и отключать каналы в СМО была обусловлена тем, что входящий поток требований в СМО обладал интенсивностью, которая в моменты контроля совершала марковские скачки. Аналогов этому явлению в ТУЗ не рассматривалось. Соответствующие модели управления запасами были стационарными в том смысле, что спрос описывался последовательностью независимых в совокупности одинаково распределенных случайных величин.

Выполненные нами в последние годы исследования показали, что этот «недостаток» моделей ТУЗ может быть «исправлен», если предложить принципиально новую модель (назовем ее модель управления запасами с возвратами). Иначе говоря, допустим, что потребитель может не только приобрести товар, хранящийся на складе или в магазине, но и вернуть его продавцу (возможно на условиях, отличных от условий приобретения). Точно также допускается, что и сам склад может не только подать заказ на пополнение запасов, но и вернуть ранее полученные партии товара тому же поставщику (и тоже на других условиях). Такие модели не так-то часто встречаются в практике управления складскими системами. Впрочем, можно отметить, что подобные ситуации нередко характерны для лизинговых схем. Таким образом, ниже мы рассматриваем модель управления запасами, в которой спрос может быть и отрицательным.

Основной мотивацией к исследованию этой новой модели управления запасами, которая впервые была описана в работе [158][[10]](#footnote-10) стала аналогия, выявленная между такой постановкой задачи об управлении запасами и задачами управления одним классом систем массового обслуживания, для которых стратегия управления состояла в том, чтобы включать дополнительные или отключать лишние рабочие каналы обслуживания [159], [160]. На основе этой аналогии для вполне прикладного класса задач теории управляемых систем массового обслуживания были построены реальные алгоритмы оптимального переключения каналов в системах массового обслуживания.

В отличие от задачи (19)-(20), будем считать, что областью определения функции распределения спроса на одном шаге *F*(*z*) является вся вещественная ось: – ∞ < *z* <+ ∞. Пусть также математическое ожидание этого распределения положительно. Возможность отрицательных значений спроса будем интерпретировать, как возвращение потребителем приобретенной на складе продукции. Будем также считать, что при возврате товара потребителю возвращается вся сумма, которую он заплатил при ее покупке. Склад тоже получает возможность вернуть товар своему поставщику, заплатив за это фиксированную сумму *A*2, но цена *c*2, по которой деньги возвращаются, меньше цены его приобретения: *c*2< *c*1*.*

Теперь вместо уравнений (20), (22) и (23) следует записать следующее уравнение динамического программирования:

|  |  |
| --- | --- |
| , | () |

где . Это уравнение можно переписать в следующем виде:

|  |  |
| --- | --- |
|  | (44) |

где функция , по-прежнему, задается формулой (22), а – формулой:

|  |  |
| --- | --- |
|  | () |

В рамках экономики подсчета затрат функция описывает минимальные затраты при принятии решения о подаче заказа (без учета фиксированной части ), включая случай неподачи заказа (*u* = 0), а функция минимальные затраты при принятии решения о возвращении товара поставщику (без учета фиксированной части ). Заметим также, что в силу вида функций и точка *Sn* абсолютного минимума функции находится правее точки *Rn* абсолютного минимума функции [[11]](#footnote-11)

На следующем рис. 46 показан гипотетический вид и взаимное расположение функций и .

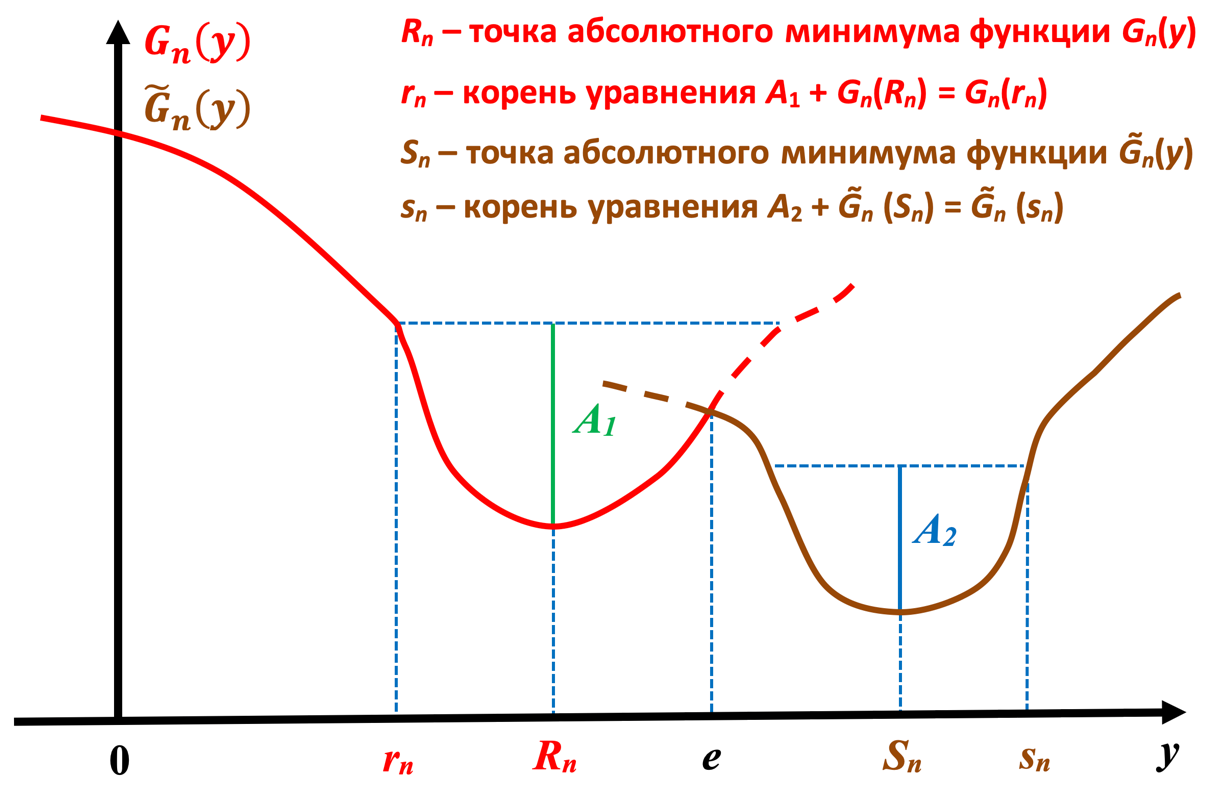


Рис. . Вид и взаиморасположение функцийи.

Для функций и , представленных на рис. 46, оптимальный закон управления запасами задается формулой:

|  |  |
| --- | --- |
|  | (46) |

Итак, если функции и для всех *n* обладают свойствами, которые качественно охарактеризованы рис. 46, то оптимальная стратегия управления запасами оказывается четырехуровневой (*Rn*, *rn*, *Sn*, *sn*)–стратегией (*rn* < *Rn* < *Sn* < *sn*), которая устроена так, что, если фиктивный уровень запасов *x* в момент принятия решений за *n* шагов до конца периода планирования меньше или равен *rn*, то подается заказа, пополняющий уровень запаса на складе до величины *Rn*. Если же фиктивный уровень запасов *x* в момент принятия решений за *n* шагов до конца периода планирования больше или равен *sn*, то часть запаса со склада возвращается поставщику так, чтобы довести уровень запасов на складе до величины *Sn*.

## Доказательство оптимальности (*Rn*, *rn*, *Sn*, *sn*)–стратегий

Преобразуем уравнение (44), введя вспомогательные функции и , смысл которых состоит в том, что в них подсчитываются средние затраты за *n* шагов до конца периода планирования при текущем уровне запасов *x* и при принятии решения о положительном размере заказа *u* (после чего уровень запасов становится равным *y* = *x + u*) и выборе оптимальной стратегии управления запасами на последующих (*n* – 1)-м шагах без учета постоянной составляющей затрат при подаче заказа (это нижний индекс «заказ») и средние затраты за *n* шагов до конца периода планирования при принятии решения об отрицательном размере заказа *u* (после чего уровень запасов также становится равным *y* = *x + u*) и выборе оптимальной стратегии управления запасами на последующих (*n* – 1)-м шагах без учета постоянной составляющей затрат при подаче заказа (это нижний индекс «возврат»). Пусть

|  |  |
| --- | --- |
| , | (47) |
| . | (48) |

Используя функции (47)–(48), можно переписать уравнения (44) в следующем виде:

|  |  |
| --- | --- |
| . | (49) |

*Утверждение 1. Rn* < *Sn*.

*Доказательство*. Используя формулы (47)–(48), нетрудно показать, что точки абсолютных минимумов по *u* функций и в силу того, что , приводят к таким значениям , что соответствующие значения это суммы, которые мы обозначим через *Rn* и *Sn* соответственно, (см. раздел 4.1) упорядочены в соответствии с неравенством *Rn* < *Sn*.

Для того, чтобы воспользоваться техникой доказательства оптимальности двухуровневых (*S*, *s*)-стратегий, предложенной в работах [3], [161], напомним базовое понятие *A*-выпуклости, на котором и было основано это доказательство.

*Определение* 1. Всюду дифференцируемая функция *f*(*x*) называется *A*-выпуклой (*A* ≥ 0), если для любого *a* ≥ 0 выполняется неравенство

|  |  |
| --- | --- |
| . | () |

Оказывается, что для доказательства оптимальности (*Rn*, *rn*, *Sn*, *sn*)–стратегий необходимо ввести обобщение этого понятия.

*Определение 2*. Пусть существует вещественное число *a*, которое обладает тем свойством, что оно находится правее точки абсолютного минимума всюду дифференцируемой функции *f*(*x*) и для всех вещественных *y* и *z* таких, что точки *y* и *y+z* находятся левее точки абсолютного минимума функции *f*(*x*) в интервале ] справедливо неравенство

|  |  |
| --- | --- |
| . | (51) |

Тогда функция *f*(*x*) называется *A*-выпуклой слева в интервале ].

*Определение 3*. Пусть существует вещественное число *b*, которое обладает тем свойством, что оно находится левее точки абсолютного минимума всюду дифференцируемой функции *f*(*x*) и для всех вещественных *y* и *z* таких, что точки *y* и *y*+*z* находятся правее точки абсолютного минимума функции *f*(*x*) в интервале ] справедливо неравенство (51). Тогда функция *f*(*x*) называется *A*-выпуклой справа в интервале ].

*Утверждение* 2. Все функции являются *A*1-выпуклыми слева и *A*2-выпуклыми справа.

*Доказательство*. Используя формулы (47)–(48) для случая *n* = 1, легко показать, что функции и являются просто выпуклыми, то есть 0-выпуклыми, и в силу характеристических свойств *A*2-выпуклости, соответственно, *A*1- и *A*2-выпуклыми. Отсюда так же, как и в работе [3], в силу формулы (49) для случая *n* = 1 устанавливается, что функция является одновременно *A*1-выпуклой слева и *A*2-выпуклой справа.

Выскажем предположение математической индукции о том, что для некоторого номера *n >*1 функции являются *A*1-выпуклыми слева и *A*2-выпуклыми справа. Дальнейшее продвижение заключается в том, что следует установить факт *A*1-выпуклости функции и *A*2-выпуклости функции . Но это следует из формул (47)–(48), а схема доказательства, как и в случае *n* = 1, повторяет схему доказательства, приведенного в работе [3]. Тогда из уравнения (49) и утверждения 1 будет следовать, что *A*1-выпуклой слева и *A*2-выпуклой справа будет функция . При этом роли точек *a* и *b* из определений 2 и 3 играет общая точка *e* пересечения графиков кривых (см. рис. 46). *Что и требовалось доказать*.

В результате оптимальная стратегия управления запасами будет определяться правилом, устанавливаемым формулой (46). При этом параметры *rn* и *sn* в формуле (46) представляют собой решения следующих уравнений (см. рис. 46):

|  |  |
| --- | --- |
| для , | () |
| для . | () |

Нетрудно видеть, что доказательство сформулированного выше факта математической индукции идейно повторяет доказательство аналогичного факта (для двухуровневых стратегий) в классической теории управления запасами [3] с поправкой на альтернативность (заказы и возвраты) целевого функционала.

## Модель управляемой системы массового обслуживания и ее связь с моделью управления запасами с возвратами

Рассматривается СМО, в которой число рабочих каналов обслуживания является управляемой величиной и может быть изменено в периодически (с шагом 1) разнесенные на оси времени моменты контроля за состоянием СМО. При этом, как в и работе [155], считается, что в СМО поступает простейший входящий поток, интенсивность которого *λ* в моменты контроля претерпевает скачкообразные изменения, принимая счетное или конечное число значений *λj*из дискретного множества *Λ*. Задана матрица вероятностей перехода соответствующей однородной марковской цепи , где –вероятность перехода (в момент контроля) от интенсивности *λi* на предыдущем шаге к интенсивности *λj*.

Считается, что величина шага контроля достаточна для того, чтобы в рассматриваемой СМО на каждом шаге устанавливался стационарный в вероятностном смысле режим функционирования. Если на данном шаге интенсивность входящего потока равна *λj*, а интенсивность обслуживания на одном рабочем канале составляет *μ*, то, очевидно [9] и [162], что число рабочих каналов в СМО должно выбираться удовлетворяющим следующему неравенству:

|  |  |
| --- | --- |
|  | (54) |

Введем эконометрические характеристики рассматриваемой СМО:

|  |  |  |
| --- | --- | --- |
| *c*1 | − | стоимость эксплуатации одного рабочего устройства обслуживания; |
| *c*2 | − | стоимость отключения одного рабочего устройства обслуживания (*c*1> *c*2); |
| *A*1 | − | фиксированная цена принятия решения о подключении новых рабочих устройств; |
| *A*2 | − | фиксированная цена принятия решения об отключении части рабочих устройств; |
| *d* | − | стоимость единицы времени пребывания одного требования в очереди на обслуживание; |
| *h* | − | доход, связанный с окончанием обслуживания одного требования; |
| *m*1 | − | текущее число рабочих каналов (до принятия управляющего решения); |
| *m* | − | выбираемое число рабочих каналов (управление). |

## Алгоритмы построения оптимальных стратегий переключения каналов

Зададим функционал качества управления, который при заданной интенсивности входящего потока *λj*, текущем числе рабочих устройств *m*1 и принятии решения о фактическом числе рабочих устройств *m* описывается величиной среднего суммарного одношагового дохода :

|  |  |
| --- | --- |
| , | (55) |

где при выполнении неравенства (54) – это выручка на 1 шаге, а – это средние затраты на 1 шаге, а

|  |  |
| --- | --- |
| . | (56) |

Здесь затраты на эксплуатацию рабочих устройств на одном шаге , затраты на очередь в стационарном режиме равны , где – это средняя длина очереди, а затраты на переключения можно представить в следующем виде:

|  |  |
| --- | --- |
|  | (57) |

Именно эта формула (57) устанавливает аналогию с моделью с возвратами ТУЗ (см. формулу (44)).

Воспользуемся классическими результатами [9], [162], чтобы записать

|  |  |
| --- | --- |
| , | () |

где . Нетрудно показать, что средняя длина очереди является выпуклой функцией от управления *m*.

Из формулы (55) очевидно, что в качестве критериальной функции можно выбрать функцию , которую в отличие от функции нужно минимизировать, а не максимизировать. Итак, для того чтобы построить оптимальную «близорукую» стратегию переключений необходимо найти

|  |  |
| --- | --- |
|  | (59) |

Формула (59) – прямой аналог формулы (44).

## Аналогии между задачами теории управляемых систем массового обслуживания и задачами теории управления запасами и производством и пороговый характер оптимальных стратегий переключения каналов

Исследуя решения оператора (59), в котором число рабочих каналов обслуживания является управляемой величиной и может быть изменено периодически (с шагом 1) в моменты контроля за состоянием СМО, можно выделить несколько случаев.

Случай *m*1 < *m*крит. В этом случае необходимо увеличить величину *m* (число включенных рабочих каналов), по крайней мере, до значения *m*крит. Итак, в этом случае приходится включать дополнительные каналы. Таким образом, реализуется процесс включения дополнительных каналов и требуется, как следует из формулы (59), минимизировать следующий функционал:

|  |  |
| --- | --- |
| . | () |

Введем обозначение (аналог функции из раздела 2.1). В зависимости от значений параметров возможны два варианта поведения функции в зависимости от значения *m*. Оба варианта показаны в левой половине рис. 47 (сплошной красной линией).

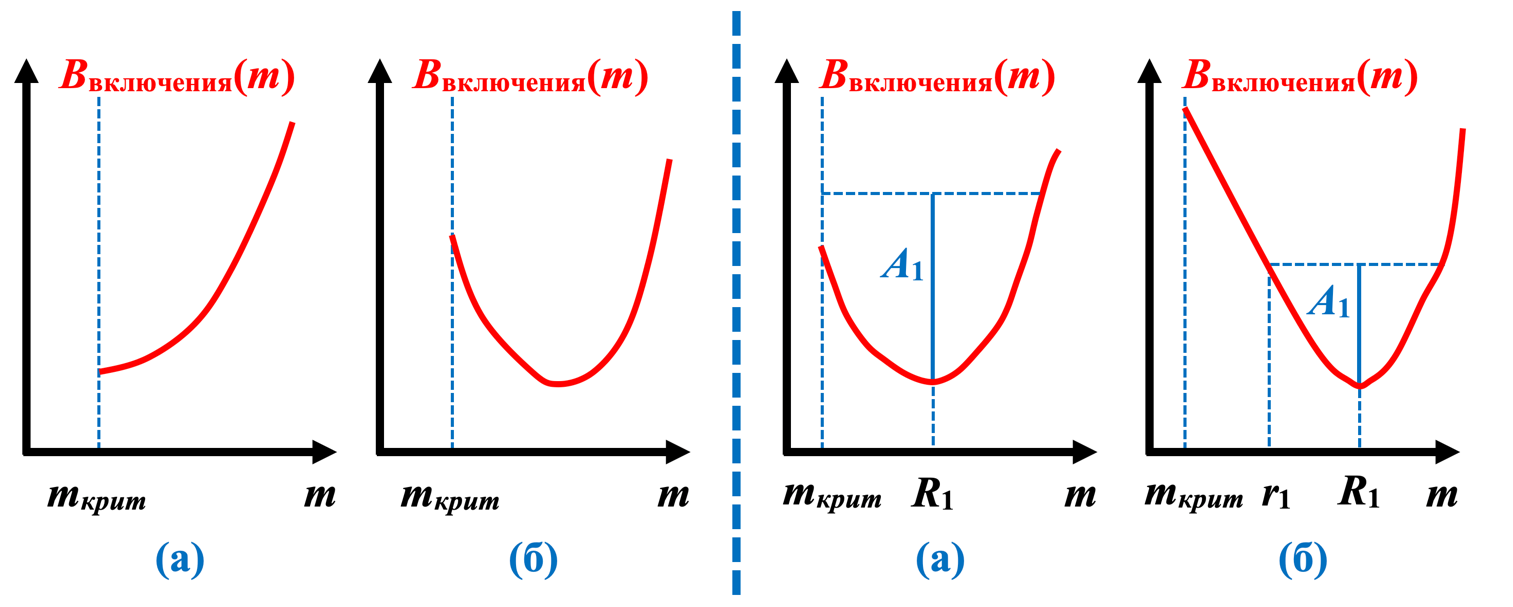


Рис. . Два варианта графиков и два подслучая функции.

В случае (а), слева в левой половине рис. 47, ответ очевиден: *m* = *m*крит. В свою очередь, случай (б), справа в левой половине рис. 47, разбивается на 2 подслучая, представленных в правой половине рис. 47.

В подслучае (a), слева в правой половине рис. 47, ответ очевиден: *m* = *R*1, где *R*1 – точка абсолютного минимума функции . В подслучае (б), справа в правой половине рис.  47, ответ усложняется. При этом удается проанализировать не только случай *m*1 < *m*крит, но и, отчасти, случай, когда *m*1 ≥ *m*крит. В самом деле, если *r*1 – точка пересечения слева от абсолютного минимума графика функции и горизонтали *A*1+, то при *m*крит ≤ *m* < *r*1 нужно выбирать *m* = *R*1. Если же *r*1 ≤ *m* < *R*1, то *m* = *m*1. Наконец, если *R*1 ≤ *m*, то следует думать, не стоит ли выключить лишние рабочие устройства.

Для этого исследуем поведение функционала (59) в случае отключения рабочих устройств. Действительно, в этом случае формула (59) приобретает вид:

|  |  |
| --- | --- |
| . | () |

По аналогии с функцией введем функцию (аналог функции из раздела 4.1). И снова возможны 2 варианта вида функции , показанные на рис. 48.

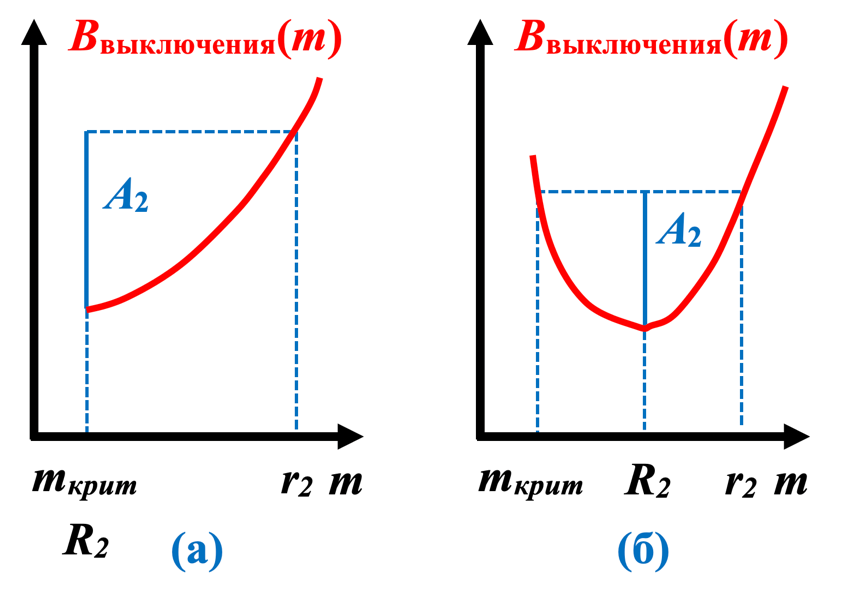


Рис. . Два варианта графиков функции *.*

Как нетрудно видеть из формул для функций и точка абсолютного минимума функции , которую мы обозначим *R*2 находится правее точки *R*1 абсолютного минимума функции . Случай, когда этот минимум достигается в точке *m*крит изображен в левой части рис. 48. Теперь вводится точка, которая отыскивается как абсцисса места пересечения графика функции и горизонтали *A*2+правее точки минимума *R*2. Аналог стратегии ТУЗ, задаваемой формулой (46).

Однако оба изображенных на рис. 48 варианта отличаются лишь тем, что в варианте (а) нужно осуществлять отключение до уровня *m*крит тогда, когда число включенных каналов *m*1 больше, чем *r*2. Если же число включенных каналов *m*1 меньше, чем *r*2 (или равно ему), то ничего отключать не нужно. Для случая, изображенного в правой половине рис. 48 (вариант (б)), при *m*1 > *r*2 число включенных каналов доводится за счет отключения до уровня *R*2. В противном случае отключение не производится.

## Компьютерное моделирование

Приведём несколько графиков, полученных в результате моделирования. При этом исходные параметры модели были выбраны следующим образом: интенсивность входящего потока 𝜆 = 299; интенсивность обслуживания на одном рабочем канале 𝜇 = 10; стоимость эксплуатации одного станка *c1* = 10; стоимость отключения одного станка *c2* = 9; цена принятия решения о подключении новых станков *A1* = 20; цена принятия решения об отключении станков *A2* = 15; стоимость пребывания требования в очереди *d* = 1 и доход от обслуживания требования *h* = 10. Рассматривалось восемь случаев для значений текущего числа рабочих каналов *m1* = 30 ÷ 37. Результаты моделирования представлены на рис. 50 ниже.

По формуле (54) значение *m*крит равняется 30. Отметим, что минимум функции *В*вкл. достигается в точке *m* = 32 (это и есть *R*1), а минимум функции *В*выкл. достигается в точке m = 35 (что в свою очередь соответствует значению *R*2). В случае, когда текущее число рабочих каналов равно *m*1 = 30 минимум функции средних затрат находиться в точке *m* = 32. Это означает, что необходимо подключить два дополнительных канала обслуживания входящих требований, так как это максимизирует функцию среднего одношагового дохода Для остальных случаев, представленных на рис. минимум функции достигается в точке *m* = *m*1. Это означает, что оптимально в этих ситуациях ничего не менять, так как это только повысит издержки по переключению каналов обслуживания.

|  |  |
| --- | --- |
| *m*1 = *m*крит. = 30 | *m*1 = 31 |

|  |  |
| --- | --- |
| *m*1 = 32 | *m*1 = 33 |

|  |  |
| --- | --- |
| *m*1 = 34 | *m*1 = 35 |

|  |  |
| --- | --- |
| *m*1 = 36 | *m*1 = 37 |

**Рис. 50. Графики функций: средних затрат на одном шаге *C1*, *B*вкл. и *B*выкл.**

## Выводы по главе 4

1. Предложена принципиально новая модель управления запасами при случайном спросе – модель с возвратами, описывающая системы управления запасами, в которых в условиях неопределенности спроса допускается отказ потребителей от уже приобретенной ими продукции. Непосредственное применение этой модели в практике возможно в лизинговых бизнес-схемах.
2. Для модели с возвратами доказана оптимальность четырехуровневой стратегии управления запасами, существенно обобщающей классическую двухуровневую стратегию.
3. Рассмотрена модель управляемой системы массового обслуживания с переключением каналов с широкими практическими применениями. Показано, что эта модель эквивалентна модели управления запасами с возвратами, поэтому результаты, полученные для модели с возвратами, переносятся и на рассмотренную модель теории массового обслуживания.
4. В результате выявлены аналогии между оптимизационными задачами теории управления запасами и теории управляемых систем массового обслуживания.
5. Для модели переключения каналов в системах массового обслуживания удалось также построить четырехуровневые стратегии переключения каналов.
6. Выполнено моделирование сформированных стратегий управления и дана оценка их эффективности.

# Особенности компьютерного моделирования

## Общие положения

Нельзя в полной мере понять результаты моделирования, а также достоинства и недостатки самой модели без подробного анализа, не только алгоритмов самой программы, но и всех релевантных сопутствующих факторов. Поэтому - несколько слов о компьютерной модели и пути её становления. Стоит начать с того, что в качестве среды разработки был сделан выбор в пользу *MATLAB* от *Mathworks* (ввиду удобства окружения). Была приобретена версия *MATLAB 2013a* с набором дополнительных модулей, необходимых для решения поставленной задачи (в дальнейшем было куплено обновление *2019b*, из-за чего пришлось переписывать часть кода, так как некоторые функции перестали работать в новой версии). Первая компьютерная модель начала разрабатываться сразу после моего поступления в магистратуру и была закончена перед защитой магистерской диссертации. Она моделировала задачу теории управления складскими запасами и вычисляла значения параметров *R* и *r* в зависимости от коэффициента дисконтирования α. Время расчёта одной серии составляло порядка одних суток, что накладывало конкретные ограничения на точность модели и на сам процесс её разработки (физически тяжело разрабатывать компьютерную модель, когда время тестового расчёта может занимать несколько часов). Помимо очевидных вещей, таких как увеличение времени разработки, существует ряд подводных камней, которые изначально были не так очевидны.

К примеру, тот факт, что процесс, ведущий расчёты мог «съесть» большую часть ресурсов центрального процессора, делал невозможным использование компьютера, так как это грозило потерей результатов расчётов текущей серии. Учитывая длительность расчётов крайне полезно использовать источник бесперебойного питания во избежание потери данных (а значит времени) в случае отключения электричества или перепада напряжения. Однако необходимо держать в уме то, что у всех подобных источников сильно ограниченное время работы, которое составляет порядка 5-10 минут для компьютера, находящегося в режиме пиковой нагрузки (не забываем про загруженный вычислениями центральный процессор), а источники с большим объёмом аккумулятора стоят дорого. К слову, значение размера шага параметров модели подбиралось таким образом, чтобы итоговое время расчёта серии не превышало одних суток.

После успешной защиты магистерской диссертации и поступления в аспирантуру встал вопрос о расширении модели на многомерный случай для построения «атласов» характеристик. Для этого было необходимо решить задачу ускорения вычислений, как минимум на 2 порядка. Первым делом был произведён рефакторинг (улучшение без добавление нового функционала кода, который позволил сократить исходный код в три раза, что в свою очередь сильно облегчило работу с моделью. Далее внимание было сконцентрировано на: оптимизации алгоритмов расчёта, сокращении числа операций и модернизации кода для перехода с последовательных вычислений на параллельные. Предпринятые меры позволили радикально сократить время расчётов серии с одних суток до нескольких минут, что позволило без проблем строить «атласы» - многомерные поверхности значений *R* и *r* в зависимости от различных эконометрических параметров модели.

Небольшая ремарка касательно аппаратной части. *MATLAB* в базовой конфигурации для проведения математических расчётов использует центральный процессор, поэтому крайне важен его грамотный выбор. Изначально я проводил расчёты на компьютере с четырёхъядерным процессором *Intel Core i7* четвёртого поколения и тактовой частотой 2,3 ГГц. В позапрошлом году в моём распоряжении оказался компьютер с четырёхъядерным процессором *Intel Core i7* седьмого поколения и тактовой частотой 4,2 ГГц. Если не вдаваться в маловажные в данном контексте подробности, то процессоры – идентичны, и поэтому – их корректно сравнивать друг с другом. Они оба четырёхъядерные, при этом каждое ядро способно обрабатывать одновременно два потока данных (2 логических ядра по технологии *Intel Hyper-Threading*). Тактовая частота ядер последнего примерно в два раза больше, чем у первого процессора, что на практике дало соответствующий прирост производительности при запуске одного и того же кода на двух разных машинах.

## Переход от последовательных к параллельные вычислениям

*MATLAB* в базовой конфигурации выполняет все вычисления на центральном процессоре в последовательном режиме, однако существует несколько способов сделать процесс расчёта модели параллельным. Первый – это использование библиотек *Nvidia CUDA* для высоко производительных вычислений на графической карте, второй – это покупка дополнительного модуля *Parallel Computing Toolbox*, который позволяет с минимальными изменениями исходного кода параллелизовать вычисления.

Рассмотрим первую технологию, а именно - *Nvidia CUDA*, позволяющую производить ресурсоёмкие вычисления на графической карте. Ввиду того, что *CUDA* – это проприетарная технология *Nvidia*, работать она будет только на компьютерах с дискретной графической картой от *Nvidia*. Для поддержки других видео карт, *Mathworks* следует встроить в *MATLAB* нативную поддержку открытых библиотек *OpenCL*, однако текущий уровень стандартизации *OpenCL* не позволяет это сделать так же просто, как это сделала *Nvidia* с программным обеспечением, оптимизированным под свои видео карты. При более подробном изучении вопроса выяснилось, что задача дискретного динамического программирования (для задачи управления запасами) слабо пересекается с областью применения *CUDA* (вычисление различных сеток и поверхностей).

Вторая технология – это *Parallel Computing Toolbox*. Вычислительное ядро *MATLAB* построено по принципу последовательных вычислений – это значит, что в базовой версии программа не может разбить вычисления на несколько отдельных потоков и отправить их на центральный процессор. Вместо этого *Mathworks* предлагают купить данный модуль, который создаёт множество независимых «работников» - процессов внутри операционной системы, которые последовательно выполняют задачи, поступающие из общей очереди. Максимальное количество «работников» определяется центральным процессором. В нашем случае используется четырёхядерный процессор с 4 физическими ядрами (или 8 логическими), что даёт нам максимальное количество «работников», равняющееся восьми. Чтобы начать использовать *Parallel Computing Toolbox* необходимо провести ряд работ по модернизации исходного кода модели. Необходимо причислить переменные внутри параллельного цикла к одному из четырёх типов. Сложнее всего определить временные переменные, которые «затираются» внутри параллельного цикла и переменными, которые передают данные наружу из параллельных циклов (передача данных от материнской программы к «работникам» и обратно съедает большое количество ресурсов). Восьмикратный теоретических прирост производительности от инициации восьми параллельных «работников» оказался на практике 3,55-кратным ускорением вычислений.

## Динамическая остановка цикла по *n* при стабилизации значений *Rn*

Из теории известно, что с ростом номера шага *n* (при достаточно больших *N*) *Rn* и *rn* стремятся к своим предельным значениям. В программе был создан специальный массив, позволяющий отследить изменения *Rn* и *rn*. При моделировании было обнаружено следующее. Если стояла задача найти зависимости *R(A)*, *r(A)* или *R(α), r(α)*, то значение *R* стабилизировалось тем раньше, тем меньше было значение *A* (или *α*). На практике *R* сходилось к своему предельному значению (в рамках нашей дискретной модели) за *n* = 5 ÷ 40 шагов. На основании богатого опыта моделирования было реализовано эвристическое правило, которое заключалось в следующем: цикл по *n* прерывался в случае, если несколько предыдущих значений *R* были равны. То есть, по факту модель выполняла *n* = 13 ÷ 48 шагов прежде, чем остановить цикл. Данное правило помогло радикально сократить время расчётов примерно в 1,5 - 2 раза только за счёт отсечения лишних операций.

Стоит отметить, что в случае, когда не нужна большая точность, возможно останавливать цикл, если соседние значения *Rn-1* и *Rn* отличаются на какое-то заданное значение в процентном соотношении. Это позволит ещё больше «сэкономить» на лишних операциях.

## Предварительный расчёт и «кэширование» значений избранных функций

В современных веб браузерах давно используется принцип кэширования (от фр. *cacher* - «прятать»), который заключается в том, что некоторые данные, которые зачастую представляют собой файлы (как часть общей программы) или результат работы скрипта, сохраняются в локальном хранилище (на компьютере пользователя) и используются в дальнейшем, вместо повторных скачиваний этих данных с сервера. Этот не сложный, по сути, трюк помогает экономить интернет трафик, уменьшает нагрузку на сервера и ускоряет загрузку страниц. Аналогичные принципы используются повсеместно в *IT* индустрии, начиная от встроенной кэш памяти процессоров и заканчивая кэшированием результатов поиска наиболее популярных запросов в поисковых машинах.

Однако стоит помнить, что широкая известность данного принципа не означает его тривиальность. Возможность внедрить кэширование зависит от конкретной задачи. После длительной работы с компьютерной моделью задачи теории управления запасами в алгоритме были выявлены блоки, потенциально пригодные для усовершенствования. Данные блоки объединяла одна особенность – частое использование результатов их деятельности в общем теле алгоритма. Практика показала, что наиболее пригодны (в плане сохранения универсальности кода программы) оказалась функции вычисления одношаговых средних затрат на хранения товара и потерь вследствие его дефицита. Было установлено, что вычисления значений этих функций можно вынести из общего цикла расчёта значения функции *G*(*y*). В таком случае мы будем вычислять их итоговое значение один раз, запишем в память, к которой и будет обращаться общий алгоритм по мере необходимости. Данный принцип показал колоссальный прирост скорости вычислений.

## Дальнейшая возможная оптимизация скорости вычислений

Вышеперечисленные методы позволили сократить время обсчёта модели с первоначального значения, которое составляло одни сутки до нескольких минут. Это позволило повысить темп разработки, сделать модель точнее и совершить качественный переход – строить многомерные «атласы» оптимальных параметров.

Однако стоит помнить, что любая программа создаётся и настраивается под конкретно поставленную задачу. В нашем случае – это изучение поведения модели, что позволяет остановиться при достижении определённой комбинация точности результатов и времени расчётов. В случае, если компании понадобиться внедрить данную модель на реальном производстве, необходимо будет сделать сначала тестовую (*Development*) версию для обкатки, а затем, изучив и настроив модель необходимым образом перейти к созданию *Production* версии (то есть, довести программу до уровня, необходимого для выхода на рынок). На любом из этапов разработки крайне полезно сократить длительное время расчётов, так как оно затягивают проведение тестов, удлиняет процесс разработки, а следовательно – стоимость. Самая дорогая статья расходов при создании программного обеспечения – это время (стоимость) разработчика. Средняя зарплата программиста в США составляет порядка $5000 в месяц, но США в данном случае - не показатель, так как там созданы уникальные финансовые условия, для привлечения лучших «мозгов» планеты. Более показателен пример Европы, где средняя зарплата программиста варьируется в диапазоне €2000–5000 в месяц. На такой серьезный проект нужно будет, как минимум 3 работника - специалист по теории управления запасами, программист *MATLAB*, программист, знающий один из низкоуровневых языков (*C++* например) и 6 месяцев на разработку и внедрение. В теории с данной задачей мог бы справиться и один человек-универсал, но таких людей на рынке труда мало и в большинстве случаев они, либо уже заняты в крупных компаниях, либо развивают своё дело.

Первый наиболее простой и дешёвый способ сокращения времени вычислений – это использование более мощного железа. Для данной задачи первоочередную важность играет мощный центральный процессор с максимально доступным числом ядер (физических и логических). В текущий момент лидером на рынке микропроцессоров выступает *Intel*, которая много лет выпускает специальную линейку процессоров для рабочих станций и серверов. Ключевыми моментами этой линейки являются многопоточность и надёжность, достигающиеся отбором лучших кристаллов и поддержкой дорогой оперативной памяти с коррекцией ошибок (error-correcting code memory). В текущий момент лидером на рынке является процессор *Intel Xeon W-3175X* с 28 физическими ядрами и 56 потоками. Данная модель обойдётся конечному пользователю примерно в $3000. Есть вариант от компании *AMD* – это *Ryzen Threadripper 2950X* с 16 физическими ядрами и 32 потоками за $1000. Важно помнить, что скорость вычислений зависит линейно от тактовой частоты ядер процессора, а не от количество логических ядер.

Другой путь оптимизации вычислений (наиболее правильный) – это перенос доведённой до ума *Development* версии кода с *MATLAB* на более низкоуровневый язык программирования, такой как: *Java*, *Python*, *C*, *C#* и *C++*, к примеру.

Как язык программирования, *Java* хорош простотой подготовки среды и разработки (по сравнению с другими вариантами) и кроссплатформенностью (один и тот же код с минимальными изменениями можно будет запустить на различных операционных системах). Главным минусом *Java* является его низкая скорость и требовательность к оперативной памяти (из-за этого сам язык предполагает наличие «сборщиков мусора», которые выгружают из памяти что-то, на их взгляд, лишнее). *MATLAB*, кстати, написан на *Java*, что объясняет его не высокую скорость (по сравнению с *C++*, к примеру), но для тех задач, для которых он используется - а именно быстрая разработка и модификация новых алгоритмов - отлично подходит, так как нет необходимости точно задавать наперёд типы переменных и выделять под них память, к примеру.

Ещё один популярный вариант – это *Python*. Своими преимуществами и недостатками во многом похож на *Java*. Главное отличие – это стилистка написания кода. В *Python* принято использовать табуляцию для выделения различных семантических конструкций, что для многих программистов выглядит дико, особенно после языков с более строгим использованием фигурных скобок и точки с запятой. И *Python,* и *Java* поддерживается огромным сообщество по всему миру, с единственным отличием, что поддержка *Java* находится в руках одной или нескольких крупных коммерческих организаций, которые занимаются его развитием и стандартизацией, в то время как сообщество *Python* более равномерно распределено среди мирового сообщества программистов. Оба данных языка имеют большое количество высокоуровневых библиотек, выполняющих практически любые задачи. Ключевой момент, который надо учитывать при выборе – это скорость *Python*, которая в некоторых прикладных задачах может оказаться даже ниже, чем у *Java*.

Язык *C* - один из базовых языков по доступу к аппаратной части и уступает разве что языку ассемблера. Главные преимущества языка *C* – это скорость его работы и отсутствия непонятного мусора в памяти, как в случае с *Java*. Недостатки же состоят из строгого и сложного синтаксиса, где простейшие вещи и циклы могут занимать десятки строк кода, против однострочных конструкций языков более высокого уровня, а также необходимость чётко знать и задавать наперёд типы переменных. Язык *C++* стал логическим продолжением и развитием языка *C*, сохранив синтаксис написания кода и скорость работы, при этом расширив функции стандартной библиотеки и добавив поддержку объектно-ориентированного программирования. *C++* является одним из самых распространённых языков программирования, используемого в наши дня для всего - от написания операционных систем до видео игр, так как сочетает в себе простоту в разработке (по сравнению с языком *C*) и непревзойдённую производительность. Язык *C#* был разработан группой инженеров *Microsoft* как язык разработки практических приложений для *Microsoft .NET Framework*. Он относиться к семейству языков с *C*-подобным синтаксисом (наиболее близок к *C++*). Главным его преимуществом было упрощение разработки практических приложений для платформы *Windows* (и, соответственно, ускорение получения готового программного продукта). Недостатком, в первую очередь, является закрытость и проприетарность языка (и компилятора), а также возможность работать только на операционной системе *Windows*. Важно помнить, что производительность программ на *C#* гораздо ниже, производительности программ, написанных на языке *C++*.

Существует ещё один занятный путь решения поставленной задачи перехода от *Development* версии программы к *Production*. Возможно использование встроенных средств *MATLAB*, а именно – *Coder*, который позволяет переводить готовый *MATLAB* код в *C*. На бумаге всё выглядит хорошо – одним нажатием клавиши возможно перевести код из высокоуровневого языка *MATLAB* в низкоуровневый язык *C*, но по факту исполнитель столкнется с массой проблем. Ему придётся перепроверить (и вероятнее всего переделать) логику всей программы, а также точно определить типы всех переменных (для выделения памяти, так как язык *MATLAB* интерпретируемый, а не компилируемый). В любом случае, данный подход может быть интересен только для крупных проектов созданных в *MATLAB*, когда чисто с экономической точки зрения будет выгоднее осуществить данный переход, чтобы не создавать новую программу.

## Методика работы с компьютерными моделями

Учитывая, что расчёт моделей, основанных на алгоритмах динамического программирования, занимает большое время (даже на производительных машинах), то вопрос об оптимальной методике работы с компьютерной моделью встаёт во главу угла.

В приложениях *А*, *Б,* *В* и *Г* представлен исходный код математических моделей, использовавшихся при написании данной работы и предназначенный для исполнения в *MATLAB*. Приложение *А* содержит код программы для поиска оптимальных параметров задачи управления запасами для трёх функций распределения спроса. Программа позволяет осуществлять поиск значений *R*, *r* от одного набора эконометрических характеристик или от промежутка значений одного или двух параметров. При этом можно получить рассчитанные значения функции суммарных средних затрат и рассмотреть случай альтернативных поставщиков, обладающих разной надёжностью и различными ценами на товар. Приложение *Б* содержит исходный код программы для расчёта суммарных средних затрат методом Монте-Карло при сравнении «близоруких» и «дальнозорких» стратегий управления запасами. В приложении *В* содержится исходный код модели для сравнения затрат при использовании различных стратегий управления в случае оперативного изменения эконометрических параметров. Приложение *Г* – это исходный код программы для моделирования системы переключения каналов обслуживания. Наиболее ресурсоёмкие вычисления происходят в моделях из приложений *А*, *Б* и *В*.

Правильный метод работы с первой моделью заключается в следующем – сначала делается расчёт для большого отрезка значений *y* (чтобы наверняка попасть в нужную область) и с грубым значением шага *y* (т.к. цель тестовой серии – это «пристрелка»). После получения результатов предварительного расчёта отрезок значений *y* можно подкорректировать и выбрать необходимый размер шага (разумный предел *-* 0,01). При этом необходимо помнить, что продолжительность вычислений линейно зависит от величины шага *y* и может составлять от 5 минут (для случая Гауссова или равномерного распределения) до 50 минут (для случая экспоненциального распределения) для поиска значений *R*(𝛂)*, r*(𝛂), к примеру. Расчёт поверхностей *R*(*A*,*c*) и *r*(*A*,*c*) занял 3,5 часа (равномерное распределение), 4,5 часа (Гауссово распределение) и 15,7 часов (экспоненциальное распределение). При этом шаг *y* был равен 0,025.

Время обсчёта модели из приложения *Б* или *В* зависит от количества экспериментов по методу Монте-Карло. В результате тестирования это значение было выбрано равное 105. Время расчётов составило 3,8 часа (Гауссово распределение), 4,1 часа (равномерное распределение) и 11,3 часов (экспоненциальное распределение). Такая разница в продолжительности расчётов связана с тем, что при генерации случайного числа использовался метод усечения и алгоритм находил это значение для Гауссова и равномерного распределений раньше, чем для экспоненциального. Необходимо отметить, что уже при значении количества экспериментов равном 103 итоговый график функции суммарных средних затрат был достаточно гладок.

Приведённые выше данные, относящиеся к продолжительности расчётов, были получены в результате моделирования на четырёхъядерном процессоре *Intel Core i*7 4,2 ГГц. Результаты моделирования были представлены в виде двухмерных графиков или трёхмерных поверхностей.

Стоит отметить несколько простых, но крайне полезных вещей, которые облегчат жизнь исследователю, занимающемуся данной проблематикой. Во-первых – была сделана индикацию с выводом в консоль времени после расчёта каждой пары *R*, *r* внутри серии, чтобы получить представление о текущем прогрессе. Во-вторых – был написан дополнительный код, который из входных данных модели автоматически формирует название файла, куда сохраняются все выходные данные. И в-третьих – крайне полезно использовать возможность *MATLAB* запускать следующую серию после окончания предыдущей, что позволяет в автоматическом режиме выполнять «цепи» последовательных расчётов.

## Выводы по главе 5

1. Решение задачи дискретного динамического программирования сопряжено времяёмкими вычислениями. Этот факт накладывает серьёзные ограничения на процесс компьютерного моделирования задач из данной области.
2. Выбор среды разработки на первичном этапе сопряжён в основном с удобством того или иного программного пакета. «Коробочную» версию программы, необходимо делать на низкоуровневом (по доступу к аппаратной части) языке программирования, таким как *C*++, к примеру.
3. Скорость расчётов можно повысить, как за счёт установки более производительной аппаратной части, так и чисто алгоритмическим путём. Выше были описаны приёмы по оптимизации процесса вычислений, которые позволили быстро строить параметрические «атласы» для СППР в системах управления запасами и производством.

Заключение

В диссертационной работе получены следующие результаты:

* Выполнено сравнение эффективности использования «дальнозорких», стационарных и «близоруких» стратегий управления запасами. Анализ показал, что выигрыш по уровню средних суммарных затрат при использовании стационарных стратегий вместо «близоруких» (широко применяемых на практике) может составлять порядка 25% при типичных для практических проблем управления запасами соотношениях параметров затрат.
* Для условий параметрической неопределенности даны рекомендации по выбору стратегии управления запасами при случайном спросе и изменениях эконометрических параметров базовой модели.
* Дано объяснение «лестничного» характера поведения зависимости верхнего уровня стационарной стратегии *R* от параметров системы и состояния рынка. Это объяснение опирается на выявленную аналогию между детерминированной моделью управления запасами и моделями управления запасами при случайном спросе. Моделирование системы управления запасами позволило оценить роль случайных возмущений в эволюции системы и обнаружить, что в характере изменения параметров оптимальных стратегий возможно наличие флуктуаций, обусловленных наличием острых пиков у реальных вероятностных распределений спроса.
* Разработан программный комплекс, позволяющий рассчитывать значения двух параметров стационарной стратегии оптимального управления запасами для трёх функций распределения спроса: (а) Гауссова распределения, (б) экспоненциального распределения и (в) равномерного распределения. Данный комплекс позволяет строить параметрические «атласы» для СППР в системах управления запасами и производством.
* В условиях сбоев в действиях поставщиков поставлена и решена задача управления запасами при случайном спросе при наличии альтернативных поставщиков, обладающих различной степенью надёжности и разными эконометрическими характеристиками.
* Предложена принципиально новая модель управления запасами при случайном спросе – модель с возвратами, описывающая системы управления запасами, в которых в условиях неопределенности спроса допускается отказ потребителей от уже приобретенной ими продукции. Для этой модели доказана оптимальность четырехуровневой стратегии управления запасами, существенно обобщающей базовую двухуровневую стратегию.
* Рассмотрена модель управляемой системы массового обслуживания с переключением каналов с широкими практическими применениями. Показано, что эта модель эквивалентна модели управления запасами с возвратами, поэтому результаты, полученные для модели с возвратами, переносятся и на рассмотренную модель теории массового обслуживания.

Литература

1. Arrow K., Karlin S. and Scarf H. Mathematical Theory of Inventory and Production. - California: Stanford University Press, 1958.
2. Hanssmann F. Operations Research in Production and Inventory Control. - London: John Wiley & Sons, Inc., 1962, - p. 254.
3. Hadley G. and Whitin Т. M. Analysis of Inventory Systems. - Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1969.
4. Прабху Н. Методы теории массового обслуживания и управления запасами. - Москва: Машиностроение, 1969.
5. Рыжиков Ю.И. Управление запасами. - Москва: Наука, 1969.
6. Первозванский А. А. Математические модели в управлении производством. - Москва: Наука, 1975. - стр. 616.
7. Булинская Е. В. Некоторые задачи оптимального управления запасами // Теория вероятностей и ее применения. - 1964 г., - 3: Т. IX. - стр. 431-447.
8. Афанасьева Л. Г. и Булинская Е. В. Случайные процессы в теории массового обслуживания и управления запасами. - Москва: Издательство МГУ, 1980. - стр. 110.
9. Гнеденко Б.В. и Коваленко И.Н. Введение в теорию массового обслуживания. - Москва: Наука, 1966. - стр. 432.
10. Саати Т. Элементы теории массового обслуживания и ее приложения. - Москва: Сов. Радио, 1971. - стр. 520.
11. Цыпкин Я.З. Адаптация и обучение в автоматических системах. - Москва: Наука, 1968.
12. Цыпкин Я.З. Основы теории обучающихся систем. - Москва: Наука, 1977.
13. Цыпкин Я.З. Информационная теория идентификации. - Москва: Наука, 1995. - стр. 336.
14. Прабху Н. Стохастические процессы теории запасов. - Москва: Мир, 1984. - стр. 186.
15. Лотоцкий В.А. и Мандель А.С. Модели и методы управления запасами. - Москва: Наука, 1991. - стр. 192.
16. Саати Т. Принятие решений при зависимостях и обратных связях: Аналитические сети. - Москва: Издательство ЛКИ, 2008. - стр. 360.
17. Гаджинский А.М. Логистика. - Москва: Изд.-торг. корпорация «Данилов и Ко», 2012.
18. Шапиро Дж. Моделирование цепи поставок. - СПб.: Питер, 2006. - стр. 720.
19. Ivanov D. and Sokolov B. Adaptive Supply Chain Management. - London: Springer, 2010. - p. 296.
20. Ronzoni C., Ferrara A. and Grassi A. A Stochastic Methodology for the Optimal Management of Infrequent Demand Spare Parts in the Automotive Industry // Preprints of the 15th IFAC Symposium on Information Control Problems in Manufacturing. - Ottawa: 2015. - pp. 1446-1451.
21. Yadollahi E., Aghezzaf El-H. and Raa B. A Statistical Comparison of Two Safety Stock Replenishment Mechanisms in a Cyclic Stochastic IRP // Preprints of IFAC Conference on Manufacturing Modelling, Management and Control. - Troyes: 2016. - pp. 1566-1571.
22. Sazvar Z., Sepehri M. and Baboli A. A Multi-objective Multi-Supplier Sustainable Supply Chain with Deteriorating Products, Case of Cut Flowers // Preprints of IFAC Conference on Manufacturing Modelling, Management and Control. - Troyes: 2016. - pp. 1696-1701.
23. Razmi J., Moghadam A. T. and Jolai F. An Evaluative Continuous Time Markov Chain Model for a Three Echelon Supply Chain with Stochastic Demand and Lead Time // Preprints of the 15th IFAC Symposium on Information Control Problems in Manufacturing. - Ottawa: 2015. - pp. 269-274.
24. Dotoli M., Epicoco N. and Falagario M. A Technique for Supply Chain Network Design under Uncertainty using Cross-Efficiency Fuzzy Data Envelopment Analysis // Preprints of the 15th IFAC Symposium on Information Control Problems in Manufacturing. - Ottawa: 2015. - pp. 668-673.
25. Ammar O., Marian H. and Dolgui A. Supply planning for multi-levels assembly system under random lead times // Preprints of the 15th IFAC Symposium on Information Control Problems in Manufacturing. - Ottawa: 2015. - pp. 276-280.
26. Habibi M. K., Battaia O., Cung V. and Dolgui A. Coordination of Collection and Disassembly Planning for End-of-Life Product // Preprints of the 15th IFAC Symposium on Information Control Problems in Manufacturing. - Ottawa: 2015. - pp. 76-80.
27. Беляков А.Г., Лапин А.В. и Мандель А.С. Управление запасами товаров ажиотажного спроса // Проблемы управления. - 2005 г. - стр. 40-45.
28. Мандель А.С. Экспертно-статистические методы обработки информации в интегрированных системах управления производством и технологическими процессами // Проблемы управления. - 2006 г. - стр. 55-59.
29. Барладян И.И., Кузнецов А.В. и Мандель А.С. Анализ критических значений параметров и моделирование управляемой системы массового обслуживания // Проблемы управления. - 2007 г. - стр. 21-25.
30. Мандель А.С. и Семенов Д.А. Адаптивные алгоритмы оценки параметров оптимальных стратегий управления запасами при ограниченном дефиците // Автоматика и телемеханика. - 2008 г. - 6. - стр. 117-128.
31. Mandel А. Models and Algorithms of Inventory Control in Case of Uncertainty // Preprint of 13th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2009, Moscow). - Москва: ИПУ РАН, 2009. - pp. 223-228.
32. Mandel А. Multi-Item Inventory Control under Uncertainty and Non-Stationarity // Proceedings of the 14th IFAC Symposium on Information Control Problems in Manufacturing (INCOM’12, Bucharest). - Bucharest: IFAC Publication, 2012. - Vol. F. - pp. 344-350.
33. Mandel А. Renewal theory methods to compute stationary inventory control strategy parameters (for lot-sizing) // Proceedings of the 20th Conference of the International Federation of Operational Research Societies (IFORS 2014, Barcelona). - Barcelona: IFORS Publications, 2014. - p. 174.
34. Гранин С.С. и Мандель А.С. Стационарные стратегии управления запасами в системах снабжения в условиях инфляции // Проблемы управления. - 2015 г. - 4. - стр. 19-24.
35. Вильмс М. А., А. Мандель А. С., Барладян И. И. и Токмакова А. Б. Локальная модель управления цепями поставок при ненадежных поставщиках // Управление большими системами: сборник трудов (электронный журнал). - Москва: ИПУ РАН, 2015. - Т. 59. - стр. 147-164.
36. Granin S. and Mandel А. Stationary Inventory Control Policies in Supply Systems under Inflation Condition // Automation and Remote Control. - 2016. - 8: Vol. 77. - pp. 1453-1460.
37. Mandel А. and Vilms М. Local Supply Chain Control Model with Unreliable Suppliers // Proceeding of the 8th IFAC Conference on Manufacturing Modelling, Management and Control (MIM 2016, Troyes, France). - Troyes: Труа: IFAC Publications, 2016. - pp. 465-470.
38. Mandel А. Econometric Models of Controllable Multiple Queuing Systems // Proceedings of the 18th International Conference, Distributed Computer and Communication Networks (DCCN 2015, Moscow, Russia). - Geneva: Springer, 2016. - pp. 296-304.
39. Mandel А., Granin S. and Vilms М. Simulation of inventory control process for supply chain with several suppliers // Proceedings of the 10th International Conference "Management of Large-Scale System Development" (MLSD). - Moscow: IEEE, 2017.
40. Mandel А. and Granin S. Optimization of Inventory Management Process // Proceedings of the 8th IEEE International Conference on Logistics, Informatics and Service Science (LISS2018). - Beejing: IEEE CFP18LIS-CDR, 2018. - pp. 178-182.
41. Harris F. Operations and Cost (Factory Management Series). - Chicago: A.W. Shaw Co., 1915. - pp. 48-52.
42. Raymond F. E. Quantity and Economic in Manufacturer. - New York: McGraw Hill Book Co., 1931.
43. Whitin T. W. The Theory of Inventory Management. - Princeton: Princeton University Press, 1953.
44. Arrow K. J., Harris T. and Marschak J. Optimal Inventory Policy // Econometrica. - 1951. - 3: Vol. 19. - pp. 250-272.
45. Boctor F. F. and Bolduc M.-C. Inventory replenishment planning and staggering // Preprint of 15th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2015, Ottawa). - Ottawa: 2015.
46. Ghorbel N., Addouche S-A. and El Mhamedi A. Forward management of spare parts stock shortages via causal reasoning using reinforcement learning // Preprint of 15th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2015, Ottawa). - Ottawa: 2015.
47. Sawik T. Integrated Supply Chain Scheduling under Multi-Level Disruptions // Preprint of 15th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2015, Ottawa). - Ottawa: 2015.
48. Scarf H., Arrow K. J., Karlin S. and Suppes P. The optimality of (S, s) policies in the dynamic inventory problem // Proc. of the first Stanford symposium, Stanford mathematical studies in the social sciences. - Stanford: Stanford University Press, 1959. - Vol. IV. - pp. 196-204.
49. Неруш Ю. М. Логистика: учебник. - Москва: Проспект; Велби, 2008. - стр. 517.
50. Щербаков В. В., Киппер И. Л., Мясникова Л. А. и и др. Основы логистики: теория и практика. - СПб.: Питер Пресс, 2009. - стр. 426.
51. Дыбская В. В. Логистика складирования: учебник. - Москва: Инфра-М, 2012. - стр. 557.
52. Мельников В. П., Схирладзе А. Г. и Антонюк А. К. Логистика. - Москва: Юрайт, 2014. - стр. 288.
53. Drezner Z. Facility Location: A Survey of Applications and Methods. - New York: Springer, 1995.
54. Daskin M. S. Network and Discrete Location: Models, Algorithms, and Applications. - New York: John Wiley, 1995.
55. Amiri A. Designing a distribution network in a supply chain system: Formulation and efficient solution procedure // European Journal of Operational Research. - 2006. - 2: Vol. 171. - pp. 567-576.
56. Melo M. T., Nickel S. and Saldanha-da-Gama F. Facility location and supply chain management - A review // European Journal of Operational Research. - 2009. - 2: Vol. 196. - pp. 401-412.
57. Georgiadis M. C., Tsiakis P., Longinidis P. and Longinidis S. Optimal design of supply chain networks under uncertain transient demand variations // Omega. - 2011. - 3: Vol. 39. - pp. 254-272.
58. Costantino A., Dotoli M., Falagario M., Fanti M. P. and Mangini A. M. A model for supply management of agile manufacturing supply chains // International Journal of Production Economics. - 2012. - Vol. 135. - pp. 451- 457.
59. Sourirajan K., Ozsen L. and Uzsoy R. A Genetic Algorithm for a Single Product Network Design Model with Lead Time and Safety Stock Considerations // European Journal of Operational Research. - 2009. - Vol. 197. - pp. 599-608.
60. Sadjady H. and Davoudpour H. Two-echelon, multi-commodity supply chain network design with mode selection, lead-times and inventory costs // Computers & Operations Research. - 2012. - Vol. 39. - pp. 1345-1354.
61. Kumar S. K. and Tiwari M. K. Supply chain system design integrated with risk pooling // Computers & Industrial Engineering. - 2013. - Vol. 64. - pp. 580-588.
62. Askin R. G., Baffo Ilaria and Xia Mingjun Multi-commodity warehouse location and distribution planning with inventory consideration // International Journal of Production Research. - 2014. - 7: Vol. 52. - pp. 1897-1910.
63. Chopra S., Reinhardt G. and Mohan U. The importance of decoupling recurrent and disruption risks in a supply chain // Naval Research Logistics. - 2007. - 5: Vol. 54. - pp. 544-555.
64. Wilson M. C. The impact of transportation disruptions on supply chain performance // Transportation Research Part E: Logistics and Transportation Review. - 2007. - Vol. 43. - pp. 295-320.
65. Singh A. R., Mishra P. K., Jain R. and Khurana M. Design of global supply chain network with operational risks // International Journal of Advanced Manufacturing Technology. - 2012. - 1-4: Vol. 60. - pp. 273-290.
66. Chopra S. and Sodhi M. S. Reducing the risk of supply chain disruptions // MIT Sloan Management Review. - 2014. - 3: Vol. 55. - pp. 73-80.
67. Ivanov D., Sokolov B. and Dolgui A. The Ripple effect in supply chains: trade-off ‘efficiency-flexibility-resilience’ in disruption management // International Journal of Production Research. - 2014a. - 7: Vol. 52. - pp. 2154-2172.
68. Sheffi Y. and Rice J. B. A supply chain view of the resilient enterprise // MIT Sloan Management Review. - 2005. - 1: Vol. 47. - pp. 41-48.
69. Tomlin B. On the Value of Mitigation and Contingency Strategies for Managing Supply Chain Disruption Risks // Management Science. - 2006. - Vol. 52. - pp. 639-657.
70. Bode C., Wagner S. M., Petersen K. J. and Ellram L. M. Understanding responses to supply chain disruptions: Insights from information processing and resource dependence perspectives // Academy of Management Journal. - 2011. - 4: Vol. 54. - pp. 833-856.
71. Ivanov D. and Sokolov B. Control and system-theoretic identification of the supply chain dynamics domain for planning, analysis, and adaptation of performance under uncertainty // European Journal of Operational Research. - 2013. - 2: Vol. 224. - pp. 313-323.
72. Kim S. H. and Tomlin B. Guilt by Association: Strategic Failure Prevention and Recovery Capacity Investments // Management Science. - 2013. - 7: Vol. 59. - pp. 1631-1649.
73. Knemeyer A. M., Zinn W. and Eroglu C. Proactive planning for catastrophic events in supply chains // Journal of Operations Management. - 2009. - 2: Vol. 27. - pp. 141-153.
74. Cui T., Ouyang Y. and Shen Z.-J. M. Reliable facility location design under the risk of disruptions // Operations Research. - 2010. - 4-part-1: Vol. 58. - pp. 998-1011.
75. Benyoucef L., Xie X. and Tanonkou G. A. Supply chain network design with unreliable suppliers: A lagrangian relaxation-based approach // International Journal of Production Research. - 2013. - 21: Vol. 51. - pp. 6435-6454.
76. Li Q., Zeng B. and Savachkin A. Reliable facility location design under disruptions // Computers & Operations Research. - 2013. - 4: Vol. 40. - pp. 901-909.
77. Simchi-Levi D., Schmidt W. and Wei Y. From superstorms to factory fires: Managing unpredictable supply chain disruptions // Havard Business Review. - February 2014.
78. Snyder L. V. and Daskin M. S. Reliability Models for Facility Location: The Expected Failure Cost Case // Transportation Science. - 2005. - Vol. 39. - pp. 400-416.
79. Klibi W., Martel A. and Guitouni A. The design of robust value-creating supply chain networks: A critical review // European Journal of Operational Research. - 2010. - 2: Vol. 203. - pp. 283-293.
80. Peng P., Snyder L. V., Lim A. and Liu Z. Reliable logistics networks design with facility disruptions // Transportation Research Part B: Methodological. - 2011. - 8: Vol. 45. - pp. 1190-1211.
81. Baghalian A., Rezapour S. and Farahani R. Z. Robust supply chain network design with service level against disruptions and demand uncertainties: A real-life case // European Journal of Operational Research. - 2013. - 1: Vol. 227. - pp. 199-215.
82. Qi L., Shen Z.-J. M. and Snyder L. V. The effect of supply disruptions on supply chain design decisions // Transportation Science. - 2010. - 2: Vol. 44. - pp. 274-289.
83. Schmitt A. J. and Singh M. A quantitative analysis of disruption risk in a multi-echelon supply chain // International Journal of Production Economics. - 2012. - 1: Vol. 139. - pp. 23-32.
84. Lim M., Daskin M. S., Bassamboo A. and Chopra S. A Facility Reliability Problem: Formulation, Properties and Algorithm // Naval Research Logistics. - 2010. - 1: Vol. 57. - pp. 58-70.
85. Lim M. K., Bassamboo A., Chopra S. and Daskin M. Facility location decisions with random disruptions and imperfect estimation // Manufacturing and Service Operations Management. - 2013. - 2: Vol. 15. - pp. 239-249.
86. Shen Z. M. Integrated supply chain design models: a survey and feature research directions // Journal of Industrial and Management Optimization. - 2007. - 1: Vol. 3. - pp. 1-27.
87. Chen Q., Li X. and Ouyang Y. Joint inventory-location problem under the risk of probabilistic facility disruptions // Transportation Research Part B: Methodological. - 2011. - 7: Vol. 45. - pp. 991-1003.
88. Mohebbi E. Supply interruptions in a lost-sales inventory system with random lead time // Computers & Operations Research. - 2003. - Vol. 30. - pp. 411-426.
89. Acar Y., Kadipasaoglu S. and Schipperijn P. A decision support framework for global supply chain modelling: An assessment of the impact of demand, supply and lead-time uncertainties on performance // International Journal of Production Research. - 2010. - 11: Vol. 48. - pp. 3245-3268.
90. Rafiei M., Mohammadi M. and Torabi S. Reliable multi period multi product supply chain design with facility disruption // Decision Science Letters. - 2013. - 2: Vol. 2. - pp. 81-94.
91. Tsiakis P., Shah N. and Pantelides C. C. Design of multi-echelon supply chain networks under demand uncertainty // Industrial & Engineering Chemistry Research. - 2001. - Vol. 40. - pp. 3585-3604.
92. Santoso T., Ahmed S., Goetschalckx G. and Shapiro A. A stochastic programming approach for supply chain network design under uncertainty // European Journal of Operational Research. - 2005. - Vol. 167. - pp. 96-115.
93. Goh M., Lim J.Y. S and Meng F. A stochastic model for risk management in global chain networks // European Journal of Operational Research. - 2007. - 1: Vol. 182. - pp. 164-173.
94. Azaron A., Brown K. N., Tarim S. A. and Modarres M. A multi-objective stochastic programming approach for supply chain design considering risk // International Journal of Production Economics. - 2008. - 1: Vol. 116. - pp. 129-138.
95. Schütz P., Tomasgard A. and Ahmed S. Supply chain design under uncertainty using sample average approximation and dual decomposition // European Journal of Operational Research. - 2009. - 2: Vol. 199. - pp. 409-419.
96. Iakovou E., Vlachos D. and Xanthopoulos A. A stochastic inventory management model for a dual sourcing supply chain with disruptions // International Journal of Systems Science. - 2010. - 3: Vol. 41. - pp. 315-324.
97. Sawik T. Integrated selection of suppliers and scheduling of customer orders in the presence of supply chain disruption risks // International Journal of Production Research. - 2013. - 23-24: Vol. 51. - pp. 7006-7022.
98. Petrovic D., Roy R. and Petrovic R. Modelling and simulation of a supply chain in an uncertain environment // European Journal of Operational Research. - 1998. - 2: Vol. 109. - pp. 299-309.
99. Aliev R. A., Fazlollahi B., Guirimov B. G. and Aliev R. R. Fuzzy-genetic approach to aggregate production-distribution planning in supply chain management // Information Sciences. - 2007. - 20: Vol. 177. - pp. 4241-4255.
100. Costantino A., Dotoli M., Falagario M., Fanti M. P., Mangini A. M., Sciancalepore F. and Ukovich W. A Fuzzy Programming Approach for the Strategic Design of Distribution Networks // Prodeedings of the IEEE International Conference on Automation Science and Engineering. - 2011. - pp. 66-71.
101. Gulpınar N., Pachamanova D. and Canakoglu E. Robust strategies for facility location under uncertainty // European Journal of Operational Research. - 2012. - 1: Vol. 225. - pp. 21-35.
102. Pishvaee M. S., Razmi J. and Torabi S. A. Robust possibilistic programming for socially responsible supply chain network design: a new approach // Fuzzy sets and systems. - 2012. - Vol. 206. - pp. 1-20.
103. Wu T., Blackhurst J. and O’grady P. Methodology for supply chain disruption analysis // International Journal of Production Research. - 2007. - 7: Vol. 45. - pp. 1665- 1682.
104. Tuncel G. and Alpan G. Risk assessment and management for supply chain networks - a case study // Computers in Industry. - 2010. - 3: Vol. 61. - pp. 250-259.
105. Deleris L. A. and Erhun F. Quantitative risk assessment in supply chains: A case study based on engineering risk analysis concepts // Planning Production and Inventories in the Extended Enter- prise, International Series in Operations Research & Management Science 152 / Kempf K. G., Keskinocak P. and Uzsoy R. - New York: Springer, 2011.
106. Zegordi S. H. and Davarzani H. Developing a supply chain disruption analysis model: Application of coloured petri-nets // Expert Systems with Applications. - 2012. - 2: Vol. 39. - pp. 2102-2111.
107. Lewis B. M., Erera A. L., Nowak M. A. and White I. C. Managing Inventory in Global Supply Chains Facing Port-of- Entry Disruption Risks // Transportation Science. - 2013. - 2: Vol. 47. - pp. 162-180.
108. Swaminathan J. M., Smith S. F. and Sadeh N. M. Modeling Supply Chain Dynamics: A Multiagent Approach // Decision Science. - 1998. - 3: Vol. 29. - pp. 607-632.
109. Surana A., Kumara S., Greaves M. and Greaves U. Supply-chain networks: a complex adaptive systems perspective // International Journal of Production Research. - 2005. - 20: Vol. 43. - pp. 4235-4265.
110. Kamath N. B. and Roy R. Capacity augmentation of a supply chain for a short lifecycle product: A system dynamics framework // European Journal of Operational Research. - 2007. - 2: Vol. 179. - pp. 334-351.
111. Xu M., Wang X. and Zhao L. Predicted supply chain resilience based on structural evolution against random supply disruptions // International Journal of Systems Science: Operations & Logistics. - 2014. - 2: Vol. 1. - pp. 105-117.
112. Schoenlein M., Makuschewitz T., Wirth F. and Schloz-Reiter B. Measurement and optimization of robust stability of multiclass queuing networks: Applications in dynamic supply chains // European Journal of Operational Research. - 2013. - Vol. 229. - pp. 179-189.
113. Ivanov D., Sokolov B. and Kaeschel J. A multi-structural framework for adaptive supply chain planning and operations with structure dynamics considerations // European Journal of Operational Research. - 2010. - Vol. 200. - pp. 409-420.
114. Teimoury E. and Fathi M. An integrated operations-marketing perspective for making decisions about order penetration point in multi-product supply chain: A queuing approach // International Journal of Production Research. - 2013. - 18: Vol. 51. - pp. 5776-5796.
115. Zhou W., Zhang R. and Zhou Y. A queuing model on supply chain with the form postponement strategy // Computers and Industrial Engineering. - 2013. - 4: Vol. 66. - pp. 643-652.
116. Meisel F. and Bierwirth C. The design of Make-to-Order supply networks under uncertainties using simulation and optimization // International Journal of Production Research. - 2014. - 22: Vol. 52. - pp. 6590- 6607.
117. Ivanov D., Sokolov B. and Pavlov A. Dual problem formulation and its application to optimal re-design of an integrated production-distribution network with structure dynamics and ripple effect considerations // International Journal of Production Research. - 2013. - 18: Vol. 51. - pp. 5386-5403.
118. Ivanov D., Sokolov B. and Pavlov A. Optimal distribution (re)planning in a centralized multi-stage network under conditions of ripple effect and structure dynamics // European Journal of Operational Research. - 2014. - 2: Vol. 237. - pp. 758-770.
119. Riddalls C. E., Bennett S. and Tipi N. S. Modelling the dynamics of supply chains // International Journal of Systems Science. - 2000. - 8: Vol. 31. - pp. 969-976.
120. Ivanov D. and Sokolov B. Structure dynamics control approach to supply chain planning and adaptation // International Journal of Production Research. - 2012. - 21: Vol. 50. - pp. 6133-6149.
121. Harjunkoski I., Maravelias C. T., Bongers P., Castro P. M., Engell S., Grossmann I. E., Hooker J., Méndez C., Sand G. and Wassick J. Scope for industrial applications of production scheduling models and solution methods // Computers and Chemical Engineering. - 2014. - Vol. 62. - pp. 161-193.
122. Carvalho H., Barroso A. P., Machado V. H., Azevedo S. and Cruz-Machado V. Supply chain redesign for resilience using simulation // Computers & Industrial Engineering. - 2012. - 1: Vol. 62. - pp. 329-341.
123. Bensoussan A., Çakanyildirim M. and Sethi S. Optimal ordering policies for inventory problems with dynamic information delays // Production & Operations Management. - 2007. - 2: Vol. 16. - pp. 241- 256.
124. Hwarng H. B. and Xie N. Understanding supply chain dynamics: A chaos perspective // European Journal of Operational Research. - 2008. - 3: Vol. 184. - pp. 1163-1178.
125. Schmitt A. J. Strategies for customer service level protection under multi-echelon supply chain disruption risk // Transportation Research Part B: Methodological. - 2011. - 8: Vol. 45. - pp. 1266-1283.
126. Wang D. and Ip W. H. Evaluation and analysis of logistic network resilience with application to aircraft servicing // IEEE Systems Journal. - 2009. - Vol. 3. - pp. 166-173.
127. Perea E., Grossmann I., Ydstie E. and Tahmassebi T. Dynamic modeling and classical control theory for supply chain management // Computers and Chemical Engineering. - 2000. - Vol. 24. - pp. 1143- 1149.
128. Braun M. W., Rivera D. E., Flores M. E., Carlyle W. M. and Kempf K. G. A model predictive control framework for robust management of multi-product, multi-echelon demand networks // Annual Reviews in Control. - 2003. - Vol. 27. - pp. 229-245.
129. Puigjaner L. and Lainez J. M. Capturing dynamics in integrated supply chain management // Computers and Chemical Engineering. - 2008. - Vol. 32. - pp. 2582-2605.
130. Mastragostino R., Patel S. and Swartz C. Robust decision making for hybrid process supply chain systems via model predictive control // Computers and Chemical Engineering. - 2014. - Vol. 62. - pp. 37- 55.
131. Unnikrishnan A. and Figliozzi M. Online Freight Network Assignment Model with Transportation Disruptions and Recourse // Transportation Research Record: Journal of the Transportation Research Board. - 2011. - 1: Vol. 2224. - pp. 17-25.
132. Vahdani B., Zandieh M. and Roshanaei V. A hybrid multistage predictive model for supply chain network collapse recovery analysis: A practical framework for effective supply chain network continuity management // International Journal of Production Research. - 2011. - 7: Vol. 49. - pp. 2035-2060.
133. Shao X. F. and Dong M. Supply disruption and reactive strategies in an assemble-to-order supply chain with time-sensitive demand // IEEE Transactions on engineering management. - 2012. - 2: Vol. 59. - pp. 201-212.
134. Paul S. K., Sarker R. and Essam D. Real time disruption management for a two-stage batch production-inventory system with reliability considerations // European Journal of Operational Research. - 2014. - Vol. 237. - pp. 113-128.
135. Soroor J., Tarokh M. J. and Keshtgary M. Preventing failure in IT-enabled systems for supply chain management // International Journal of Production Research. - 2009. - 23: Vol. 47. - pp. 6543-6557.
136. Tang O. and Musa S. N. Identifying Risk Issues and Research Advancements in Supply Chain Risk Management // International Journal of Production Economics. - 2011. - Vol. 133. - pp. 25-34.
137. van Delft Ch. and Vial J.-Ph. A practical implementation of stochastic programming: An application to the evaluation of option contracts in supply chains // Automatica. - 2004. - Vol. 40. - pp. 743-756.
138. Мандель А. С. О выборе критериев в задачах управления запасами в условиях неопределенности // Труды XII Всероссийского совещания по проблемам управления «ВСПУ-2014». - Москва: ИПУ РАН, 2014. - стр. 4212-4218.
139. Мандель А. С. Задача управления многономенклатурными запасами в условиях неопределенности и нестационарности. Часть I. Нормативная модель. // Проблемы управления. - 2011 г. - Т. 6. - стр. 47-51.
140. Мандель А. С. Управление многономенклатурными запасами в условиях неопределенности и нестационарности. Ч.II. Создание страховых запасов. // Проблемы управления. - 2012 г. - Т. 1. - стр. 42-46.
141. Коновалов А. С., Барладян И. И. и Токмакова А. Б. Сравнительный анализ эффективности различных методов кластеризации в задачах предпроектного обследования автоматизируемых систем управления снабжением и логистики // Теория активных систем. Труды международной научно-практической конференции. - Москва: ИПУ РАН, 2009. - Т. 1. - стр. 255-259.
142. Мандель А. С. Метод аналогов и прогнозирование коротких временных рядов. Экспертно-статистический подход // Автоматика и телемеханика. - 2004 г. - Т. 4. - стр. 84-93.
143. Беляков А. Г. и Мандель А. С. Предсказание временных рядов на основе метода аналогов (элементы теории экспертно-статистических систем). - Москва: ИПУ РАН, 2002. - стр. 60.
144. Барладян И. И., Борзенко Н. И., Лапин А. В. и и др. Сравнительный анализ «близоруких» и «дальнозорких» стратегий управления запасами в условиях неопределенности // Труды международной конференции по проблемам управления. - Москва: ИПУ РАН, 2006. - стр. 774-783.
145. Коновалов А. С. и Мандель А. С. Применение фильтра Калмана для прогнозирования спроса при решении задач управления запасами // Теория активных систем. Труды международной научно-практической конференции. - Москва: ИПУ РАН, 2009. - Т. 1. - стр. 259-263.
146. Лазарев А. А. и Гафаров Е. Р. Теория расписаний. Задачи и алгоритмы. Учебное пособие. - Москва: МГУ, 2011. - стр. 224.
147. Mandel А. Inventory Control Policies for Random Lead Times // Proceedings of the 10th International Conference "Management of Large-Scale System Development" (MLSD). - M.: IEEE, 2017.
148. Ettl M., Feigin G. E., Lin G. Y. and Yao D. D. A supply network model with base-stock control and service requirements // Operation Research. - 2000. - Vol. 48. - pp. 21-28.
149. Graves S. C. and Wilems S. P. Supply chain design safety stock placement and supply chain configuration // Handbook in OR & MS. - 2003. - Vol. 11. - pp. 95-132.
150. Justus N. and Meyr H. Designing a Planning System for Suppliers of the Machine Building Industry // Technical program of IFORS-2014 World Conference. - Barcelona: 2014. - p. 3.
151. Pishchulo G., Richter K. and Golesorkhi S. Supply Chain Contracting under Asymmetric Information and Partial Vertical Integration // Technical program of IFORS-2014 World Conference. - Barcelona: 2014. - p. 18.
152. Alegoz M. and Ozturk Z. K. A Goal Programming Approach to Design the Supply Chain Network // Technical program of IFORS-2014 World Conference. - Barcelona: 2014. - p. 27.
153. Лотоцкий В. А. Модели управления запасами при случайных поставках // Алгоритмы многомерного статистического анализа и их приложения. - Москва: ЦЭМИ, 1975.
154. Вильмс М. А., Гранин С. С. и Мандель А. С. Моделирование процесса управления запасами в цепи поставок при наличии нескольких поставщиков // Труды десятой международной конференции «Управление развитием крупномасштабных систем» (MLSD'2017). - Москва: ИПУ РАН, 2017. - Т. 1.
155. Мандель А. С., Барладян И. И. и Токмакова А. Б. Многолинейная СМО с изменением числа рабочих каналов: стационарный случай. Часть1 // Материалы 18-й Международной научной конференции «Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь» (DCCN-2015). - Москва: ИПУ РАН, 2015. - стр. 345-354.
156. Мандель А. С. и Махукова В. В. Многолинейная СМО с изменением числа рабочих каналов: нестационарный случай. Часть 2 // Материалы 18-й Международной научной конференции «Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь» (DCCN-2015). - Москва: ИПУ РАН, 2015. - стр. 355-361.
157. Mandel А. and Bakulin К. Models of controllable multiple queuing systems for channel switching myopic strategies // Proceedings of the 20th International Conference, Distributed Computer and Communication Networks (DCCN-2017). - Moscow: Technosphera, 2017. - pp. 534-542.
158. Мандель А. С. и Гранин С. С. Исследование аналогий между задачами теории управления запасами и задачами теории управляемых систем массового обслуживания // Материалы 11-й Международной конференции «Управление развитием крупномасштабных систем» (MLSD'2018). - Москва: ИПУ РАН, 2018. - Т. 1. - стр. 279-281.
159. Mandel А. and Laptin V. Myopic Channel Switching Strategies for Stationary Mode: Threshold Calculation Algorithms // Distributed Computer and Communication Networks. DCCN 2018. Communications in Computer and Information Science. - Geneva: Springer, 2018. - Vol. 919. - pp. 410-420.
160. Mandel А. and Laptin V. Channel Switching Threshold Strategies for Multichannel Controllable Queuing Systems // Distributed Computer and Communication Networks: Control, Computation, Communications. DCCN 2020. Communications in Computer and Information Science. - Geneva: Springer, 2020. - Vol. 1337. - pp. 259-270.
161. Karlin S. Mathematical Methods and Theory in Games, Programming, and Economics. - New York: Addison-Wesley Publishing Company, 1959.
162. Вишневский В. М. Теоретические основы проектирования компьютерных сетей. - Москва: Техносфера, 2003.
163. Кокс Д. и Смит Т. Теория восстановления. - Москва: Сов. Радио, 1967. - стр. 302.
164. Исходный код модели задачи управления запасами для расчёта оптимальных параметров, суммарных средних затрат, построения параметрического «атласа» и моделирования случая с альтернативными поставщиками

1 % очищаем рабочую область

2 clear variables;

3

4 % очищаем консоль

5 clc;

6

7 % перечисляем символьные параметры

8 syms y\_sym z\_sym lambda\_sym sigma\_sym myu\_sym pi\_sym;

9

10 % задаём формат вывода чисел в консоли

11 format longG;

12

13 % начинаем отсчёт времени

14 start\_time = clock;

15 tic;

16

17 % сохранять ли общие массивы Cn(x) и Gn(y)

18 save\_Gny\_and\_Cnx = 1;

19

20 % математическое ожидание и среднеквадратическое отклонение нормальной функции распределения

21 myu = 5;

22 sigma = 1;

23

24 % интенсивность экспоненциальной функции распределения

25 lambda = 0.2;

26

27 % минимум и максимум функции непрерывного равномерного распределения

28 uniform\_a = 0;

29 uniform\_b = 10;

30

31 % выбираем функцию распределения спроса (1 - нормальное, 2 - экспоненциальное, 3 - равномерное)

32 CDF = 1;

33

34 % коэффициент дисконтирования

35 alpha = 0.99;

36

37 % стоимость единичной поставки

38 A = 100;

39

40 % цена единицы товара

41 c = 10;

42

43 % стоимость хранение единицы товара

44 h = 1;

45

46 % потери от дефицита единицы товара

47 d = 80;

48

49 % количество шагов

50 N = 30 + 1;

51

52 % количество одинаковых значений R, необходимое для остановки цикла по n

53 stable\_R\_count = 8;

54

55 % минимальное гарантированное число выполненных шагов

56 if save\_Gny\_and\_Cnx == 1

57 N\_min = N;

58 else

59 N\_min = stable\_R\_count + 5;

60 end

61

62 % минимум y на последнем шаге

63 last\_step\_y\_min = -10;

64 % максимум y на последнем шаге

65 last\_step\_y\_max = 50;

66 % размер шага y

67 y\_step = 0.01;

68

69 % рассмотрение случая с ненадёжными поставщиками

70 unreliable = 0;

71 % обратный порядок пар поставщиков

72 c\_reverse = 0;

73 if unreliable == 1

74 % стоимости единицы товара и вероятность срыва поставки у различных поставщиков

75 c\_i = [0.5 1.0 2.0 4.0];

76 p\_i = [0.8 0.5 0.1 0.0];

77 if c\_reverse == 1

78 c\_i = fliplr(c\_i);

79 p\_i = fliplr(p\_i);

80 end

81 % средняя цена единицы товара

82 [m,n] = size(p\_i);

83 c\_avg = 0;

84 for i = 1:n

85 c\_avg\_part = 1;

86 for j = 1:i

87 if j == i

88 c\_avg\_part = c\_avg\_part \* (1 - p\_i(j)) \* c\_i(j);

89 c\_avg = c\_avg + c\_avg\_part;

90 else

91 c\_avg\_part = c\_avg\_part \* p\_i(j);

92 end

93 end

94 end

95 % присваиваем значение средней цены товара

96 c = c\_avg;

97 end

98

99 % переменная 1

100 v1\_name = 'Alpha';

101 % минимум переменной 1

102 v1\_min = alpha;

103 % максимум переменной 1

104 v1\_max = alpha;

105 % шаг переменной 1

106 v1\_step = 0;

107 % количество значений переменной 1

108 if v1\_step == 0

109 v1\_count = 1;

110 else

111 v1\_count = round((v1\_max - v1\_min) / v1\_step + 1);

112 end

113

114 % переменная 2

115 v2\_name = 'd';

116 % минимум переменной 2

117 v2\_min = d;

118 % максимум переменной 2

119 v2\_max = d;

120 % шаг переменной 2

121 v2\_step = 0;

122 % количество значений переменной 2

123 if v2\_step == 0

124 v2\_count = 1;

125 else

126 v2\_count = (v2\_max - v2\_min) / v2\_step + 1;

127 end

128

129 % нижний предел интегрирования интеграла при альфа

130 z\_min = 0;

131

132 % верхний предел интегрирования интеграла при альфа

133 if CDF == 1

134 z\_max = myu + 5 \* sigma;

135 elseif CDF == 2

136 z\_max = 1 / lambda \* 6;

137 elseif CDF == 3

138 z\_max = min(uniform\_b,last\_step\_y\_max);

139 uniform\_pdf = 1 / (uniform\_b - uniform\_a);

140 end

141

142 % размер шага z

143 z\_step = y\_step;

144 % количество значений z

145 z\_count = round((z\_max - z\_min) / z\_step + 1);

146

147 % минимум y на первом шаге

148 first\_step\_y\_min = last\_step\_y\_min - z\_max \* (N - 1);

149 % максимум y на первом шаге

150 first\_step\_y\_max = last\_step\_y\_max - z\_min \* (N - 1);

151 % количество значений y на первом шаге

152 first\_step\_y\_count = round((first\_step\_y\_max - first\_step\_y\_min) / y\_step + 1);

153

154 % значения переменной 1

155 v1\_values = NaN(1,v1\_count);

156 for v1 = 1:v1\_count

157 v1\_values(1,v1) = v1\_min + v1\_step \* (v1 - 1);

158 end

159

160 % значения переменной 2

161 v2\_values = NaN(1,v2\_count);

162 for v2 = 1:v2\_count

163 v2\_values(1,v2) = v2\_min + v2\_step \* (v2 - 1);

164 end

165

166 % подготовим массивы r(v1,v2) и R(v1,v2)

167 v1v2r = NaN(v1\_count+1,v2\_count+1);

168 v1v2R = NaN(v1\_count+1,v2\_count+1);

169 % первая строка - значения v2

170 for v2 = 1:v2\_count

171 v1v2r(1,v2+1) = v2\_values(v2);

172 v1v2R(1,v2+1) = v2\_values(v2);

173 end

174

175 % подготовим массив v1, R и r

176 if v2\_count == 1

177 v1Rr = NaN(v1\_count,3);

178 end

179

180 % количество работников MATLAB

181 workers\_count = 8;

182

183 % калибровочная модель (для экстраполяции ожидаемого времени вычислений)

184 c\_workers\_count = 8;

185 c\_v1\_min = 0;

186 c\_v1\_max = 100;

187 c\_v1\_step = 0.5;

188 if c\_v1\_step == 0

189 c\_v1\_count = 1;

190 else

191 c\_v1\_count = (c\_v1\_max - c\_v1\_min) / c\_v1\_step + 1;

192 end

193 c\_v2\_min = 1;

194 c\_v2\_max = 20;

195 c\_v2\_step = 0.5;

196 if c\_v2\_step == 0

197 c\_v2\_count = 1;

198 else

199 c\_v2\_count = (c\_v2\_max - c\_v2\_min) / c\_v2\_step + 1;

200 end

201 c\_last\_step\_y\_min = -5;

202 c\_last\_step\_y\_max = 30;

203 c\_y\_step = 0.05;

204 c\_z\_min = 0;

205 c\_z\_max = 9;

206 c\_N = 50;

207 c\_time\_s = 1142;

208

209 % количество операций

210 operations\_count = 0;

211 for n = 1:N

212 operations\_count = operations\_count + (((last\_step\_y\_max - z\_min \* (N - n)) - (last\_step\_y\_min - z\_max \* (N - n))) / y\_step + 1);

213 end

214 operations\_count = operations\_count \* ceil(v1\_count/workers\_count) \* v2\_count \* (N - 1);

215

216 % количество операций калибровочной модели

217 c\_operations\_count = 0;

218 for n = 1:c\_N

219 c\_operations\_count = c\_operations\_count + (((c\_last\_step\_y\_max - c\_z\_min \* (c\_N - n)) - (c\_last\_step\_y\_min - c\_z\_max \* (c\_N - n))) / c\_y\_step + 1);

220 end

221 c\_operations\_count = c\_operations\_count \* ceil(c\_v1\_count/c\_workers\_count) \* c\_v2\_count \* (c\_N - 1);

222

223 % ожидаемое время вычислений в секундах

224 eta\_s = c\_time\_s / c\_operations\_count \* operations\_count;

225

226 % ожидаемое время вычислений в часах

227 eta\_h = round((eta\_s / 3600),2);

228

229 % выведем в консоль ожидаемую длительность и ожидаемое окончание вычислений

230 fprintf('%s Started %.0fs %.2fh %s\n',datestr(now),eta\_s,eta\_h,datestr(datenum(clock + eta\_s \* [0 0 0 0 0 1])));

231

232 % выбираем функцию распределения

233 if CDF == 1

234

235 % нормальное распределение

236

237 % предварительный расчёт интеграла при h

238 int\_h\_sym = str2sym('(y\_sym - z\_sym) / (sigma\_sym \* (2 \* pi\_sym)^0.5) \* exp(((z\_sym-myu\_sym)^2) / (-2 \* sigma\_sym^2))');

239 int\_h\_sym = subs(int\_h\_sym,sigma\_sym,sigma);

240 int\_h\_sym = subs(int\_h\_sym,myu\_sym,myu);

241 int\_h\_sym = subs(int\_h\_sym,pi\_sym,pi);

242

243 % интеграл при h (от 0 до y)

244 int\_h\_0\_y = int(int\_h\_sym,z\_sym,0,y\_sym);

245

246 % предварительный расчёт интеграла при d

247 int\_d\_sym = str2sym('(z\_sym - y\_sym) / (sigma\_sym \* (2 \* pi\_sym)^0.5) \* exp(((z\_sym-myu\_sym)^2) / (-2 \* sigma\_sym^2))');

248 int\_d\_sym = subs(int\_d\_sym,sigma\_sym,sigma);

249 int\_d\_sym = subs(int\_d\_sym,myu\_sym,myu);

250 int\_d\_sym = subs(int\_d\_sym,pi\_sym,pi);

251

252 % интеграл при d (от y до бесконечности)

253 int\_d\_y\_inf = int(int\_d\_sym,z\_sym,y\_sym,inf);

254

255 % интеграл при d (от 0 до бесконечности)

256 int\_d\_0\_inf = int(int\_d\_sym,z\_sym,0,inf);

257

258 elseif CDF == 2

259

260 % экспоненциальное распределение

261

262 % предварительный расчёт интеграла при h

263 int\_h\_sym = str2sym('(y\_sym - z\_sym) \* lambda\_sym \* exp(-1 \* lambda\_sym \* z\_sym)');

264 int\_h\_sym = subs(int\_h\_sym,lambda\_sym,lambda);

265

266 % интеграл при h (от 0 до y)

267 int\_h\_0\_y = int(int\_h\_sym,z\_sym,0,y\_sym);

268

269 % предварительный расчёт интеграла при d

270 int\_d\_sym = str2sym('(z\_sym - y\_sym) \* lambda\_sym \* exp(-1 \* lambda\_sym \* z\_sym)');

271 int\_d\_sym = subs(int\_d\_sym,lambda\_sym,lambda);

272

273 % интеграл при d (от y до бесконечности)

274 int\_d\_y\_inf = int(int\_d\_sym,z\_sym,y\_sym,inf);

275

276 % интеграл при d (от 0 до бесконечности)

277 int\_d\_0\_inf = int(int\_d\_sym,z\_sym,0,inf);

278

279 end

280

281 % подготовим таблицу рассчитанных значений интегралов при h и d

282 int\_h\_d\_values = NaN(2,first\_step\_y\_count);

283

284 % заполним таблицу g(y)

285 for y = 1:first\_step\_y\_count

286

287 % значение y

288 y\_value = first\_step\_y\_min + y\_step \* (y - 1);

289

290 if CDF == 1 || CDF == 2

291

292 % определяем пределы интегрирования интегралов h и d

293 if y\_value <= 0

294 h\_int = 0;

295 d\_int = subs(int\_d\_0\_inf,y\_sym,y\_value);

296 else

297 h\_int = subs(int\_h\_0\_y,y\_sym,y\_value);

298 d\_int = subs(int\_d\_y\_inf,y\_sym,y\_value);

299 end

300

301 elseif CDF == 3

302

303 % определяем пределы интегрирования интегралов h и d

304 h\_int\_min = 0;

305 d\_int\_max = min(first\_step\_y\_max,uniform\_b+1);

306 if y\_value <= 0

307 h\_int\_max = 0;

308 d\_int\_min = 0;

309 else

310 h\_int\_max = y\_value;

311 d\_int\_min = y\_value;

312 end

313 h\_int\_count = (h\_int\_max - h\_int\_min) / z\_step + 1;

314 d\_int\_count = (d\_int\_max - d\_int\_min) / z\_step + 1;

315

316 % начальное значение интеграла при h

317 h\_int = 0;

318

319 if h\_int\_count > 1

320

321 % подсчитаем значение интеграла при h как среднее арифметическое верхней и нижней сумм Дарбу

322 for z = 1:h\_int\_count

323

324 % текущее значение z

325 z\_value = h\_int\_min + z\_step \* (z - 1);

326

327 % предыдущее значение подынтегральной функции

328 if z >= 2

329 h\_prev\_val = h\_this\_val;

330 end

331

332 % определим значение функции плотности вероятности в текущей точке z

333 if z\_value >= uniform\_a && z\_value <= uniform\_b

334 uniform\_pdf\_val = uniform\_pdf;

335 else

336 uniform\_pdf\_val = 0;

337 end

338

339 % значение подынтегральной функции данного среза

340 h\_this\_val = (y\_value - z\_value) \* uniform\_pdf\_val;

341

342 if z >= 2 && (z\_value > uniform\_a && z\_value <= uniform\_b)

343

344 % знаки текущего и предыдущего значений

345 h\_this\_sign = sign(h\_this\_val);

346 h\_prev\_sign = sign(h\_prev\_val);

347

348 % находим значение площади среза подынтегральной функции h

349 if (h\_this\_sign == 1 && h\_prev\_sign == -1) || (h\_this\_sign == -1 && h\_prev\_sign == 1)

350 h\_this\_step\_area = (abs(h\_this\_val) + abs(h\_prev\_val)) \* z\_step / 4;

351 else

352 h\_this\_step\_area = (min(abs(h\_this\_val),abs(h\_prev\_val)) + abs(abs(h\_this\_val) - abs(h\_prev\_val)) / 2) \* z\_step;

353 end

354

355 % добавим площадь среза к площади всего подграфика

356 h\_int = h\_int + h\_this\_step\_area;

357

358 end

359

360 end

361

362 end

363

364 % начальное значение интеграла при d

365 d\_int = 0;

366

367 if d\_int\_count > 1

368

369 % подсчитаем значение интеграла при h как среднее арифметическое верхней и нижней сумм Дарбу

370 for z = 1:d\_int\_count

371

372 % текущее значение z

373 z\_value = d\_int\_min + z\_step \* (z - 1);

374

375 % предыдущее значение подынтегральной функции

376 if z >= 2

377 d\_prev\_val = d\_this\_val;

378 end

379

380 % определим значение функции плотности вероятности в текущей точке z

381 if z\_value >= uniform\_a && z\_value <= uniform\_b

382 uniform\_pdf\_val = uniform\_pdf;

383 else

384 uniform\_pdf\_val = 0;

385 end

386

387 % значение подынтегральной функции данного среза

388 d\_this\_val = (z\_value - y\_value) \* uniform\_pdf\_val;

389

390 if z >= 2 && (z\_value > uniform\_a && z\_value <= uniform\_b)

391

392 % знаки текущего и предыдущего значений

393 d\_this\_sign = sign(d\_this\_val);

394 d\_prev\_sign = sign(d\_prev\_val);

395

396 % находим значение площади среза подынтегральной функции h

397 if (d\_this\_sign == 1 && d\_prev\_sign == -1) || (d\_this\_sign == -1 && d\_prev\_sign == 1)

398 d\_this\_step\_area = (abs(d\_this\_val) + abs(d\_prev\_val)) \* z\_step / 4;

399 else

400 d\_this\_step\_area = (min(abs(d\_this\_val),abs(d\_prev\_val)) + abs(abs(d\_this\_val) - abs(d\_prev\_val)) / 2) \* z\_step;

401 end

402

403 % добавим площадь среза к площади всего подграфика

404 d\_int = d\_int + d\_this\_step\_area;

405

406 end

407

408 end

409

410 end

411

412 end

413

414 int\_h\_d\_values(1,y) = h\_int;

415 int\_h\_d\_values(2,y) = d\_int;

416

417 end

418

419 % подготовим таблицу предварительно рассчитанных значений функции f(z) (кэш значений)

420 fz = NaN(1,z\_count);

421

422 for z = 1:z\_count

423 % значение z

424 z\_value = z\_min + z\_step \* (z - 1);

425 % выбираем одну из функций распределения

426 if CDF == 1

427 fz(1,z) = 1 / (sigma \* (2 \* pi)^0.5) \* exp(((z\_value - myu)^2) / (-2 \* sigma^2));

428 elseif CDF == 2

429 fz(1,z) = lambda \* exp(-1 \* lambda \* z\_value);

430 elseif CDF == 3

431 if z\_value >= uniform\_a && z\_value <= uniform\_b

432 uniform\_pdf\_val = uniform\_pdf;

433 else

434 uniform\_pdf\_val = 0;

435 end

436 fz(1,z) = uniform\_pdf\_val;

437 end

438 end

439

440 % таблицы значений Gn(y) и Cn(x) от n, v1, v2

441 if save\_Gny\_and\_Cnx == 1

442 Gny\_g = NaN(N+1,first\_step\_y\_count+1,v1\_count,v2\_count);

443 Cnx\_g = NaN(N+1,first\_step\_y\_count+1,v1\_count,v2\_count);

444 end

445

446 % подготовим таблицы значений R(n,v1,v2) и r(n,v1,v2)

447 Rn\_g = NaN(N,v1\_count,v2\_count);

448 rn\_g = NaN(N,v1\_count,v2\_count);

449

450 % параллельный цикл по переменной 1

451 parfor v1 = 1:v1\_count

452

453 % значение переменной 1

454 v1\_value = v1\_values(v1);

455 % присваиваем значение переменной 1 значению альфа

456 alpha = v1\_value;

457

458 % временные строки r и R для переменной 2

459 v1r = NaN(1,v2\_count+1);

460 v1R = NaN(1,v2\_count+1);

461 v1r(1,1) = v1\_value;

462 v1R(1,1) = v1\_value;

463

464 % строка v1, R, r

465 v1Rr\_row = NaN;

466 if v2\_count == 1

467 v1Rr\_row = NaN(1,3);

468 end

469

470 % цикл по переменной 2

471 for v2 = 1:v2\_count

472

473 % значение переменной 2

474 v2\_value = v2\_values(v2);

475 % присваиваем значение переменной 1 значению d

476 d = v2\_value;

477

478 % обозначим временные переменные параллельного цикла

479 this\_step\_y\_min = NaN;

480 this\_step\_y\_count = NaN;

481 this\_step\_y\_offset = NaN;

482 Gny\_min = NaN;

483 R = NaN;

484 r = NaN;

485 this\_val = NaN;

486 prev\_val = NaN;

487 r\_val = NaN;

488 r\_val\_prev = NaN;

489 delta\_val = NaN;

490 delta\_val\_prev = NaN;

491

492 % задаём рамки таблицы Gn(y), где первый столбец - это значения n, а первая строка - это значения y

493 Gny = NaN(N+1,first\_step\_y\_count+1);

494

495 % задаём рамки таблицы Cn(x), где первый столбец - это значения n, а первая строка - это значения x

496 Cnx = NaN(N+1,first\_step\_y\_count+1);

497

498 % первый столбец (значения n)

499 for n = 1:N

500 Gny(n+1,1) = n;

501 Cnx(n+1,1) = n;

502 end

503

504 % первый ряд (значения y/x)

505 for y = 1:first\_step\_y\_count

506 Gny(1,y+1) = first\_step\_y\_min + y\_step \* (y - 1);

507 Cnx(1,y+1) = first\_step\_y\_min + y\_step \* (y - 1);

508 end

509

510 % значения функции Cn(x) на первом шаге

511 Cnx(2,2:first\_step\_y\_count+1) = 0;

512

513 % подготовим столбец R(n) и r(n)

514 Rn = NaN(N,1);

515 rn = NaN(N,1);

516

517 % начинаем выполнять шаги

518 for n = 2:N

519

520 % минимум y текущего шага

521 this\_step\_y\_min = last\_step\_y\_min - z\_max \* (N - n);

522

523 % максимум y текущего шага

524 this\_step\_y\_max = last\_step\_y\_max - z\_min \* (N - n);

525

526 % количество значений y на текущем шаге

527 this\_step\_y\_count = round((this\_step\_y\_max - this\_step\_y\_min) / y\_step + 1);

528

529 % индекс смещения y для текущего шага, с учётом первой колонки Gn(y) и Cn(x)

530 this\_step\_y\_offset = round((this\_step\_y\_min - first\_step\_y\_min) / y\_step + 1);

531

532 % цикл по y - заполняем строку таблицы Gn(y)

533 for y = 1:this\_step\_y\_count

534

535 % текущее значение y

536 y\_value = Gny(1,this\_step\_y\_offset+y);

537

538 % вычислим значение интеграла при альфа

539 integral\_a = 0;

540

541 if alpha ~= 0 && n ~= 2

542

543 % цикл по z

544 for z = 1:z\_count

545

546 % предыдущее значение подынтегральной функции

547 if z >= 2

548 prev\_val = this\_val;

549 end

550

551 % текущее значение z

552 z\_value = z\_min + z\_step \* (z - 1);

553

554 % индекс значения x из таблицы Cn(x)

555 x\_index = round((y\_value - z\_value - first\_step\_y\_min) / y\_step + 1 + 1);

556

557 % текущее значение подынтегральной функции

558 this\_val = Cnx(n,x\_index) \* fz(1,z);

559

560 % площадь вписанного прямоугольника и треугольника на данном шаге z

561 if z >= 2

562

563 % знаки текущего и предыдущего значений

564 this\_sign = sign(this\_val);

565 prev\_sign = sign(prev\_val);

566

567 % находим значение площади среза z подынтегральной функции

568 if (this\_sign == 1 && prev\_sign == -1) || (this\_sign == -1 && prev\_sign == 1)

569 this\_step\_area = (abs(this\_val) + abs(prev\_val)) \* z\_step / 4;

570 else

571 this\_step\_area = (min(abs(this\_val),abs(prev\_val)) + abs(abs(this\_val) - abs(prev\_val)) / 2) \* z\_step;

572 end

573

574 % добавляем площадь среза к площади всего подграфика

575 integral\_a = integral\_a + this\_step\_area;

576

577 end

578

579 end

580

581 end

582

583 % текущее значение функции Gn(y)

584 Gny(n+1,this\_step\_y\_offset+y) = c \* y\_value + h \* int\_h\_d\_values(1,this\_step\_y\_offset-1+y) + d \* int\_h\_d\_values(2,this\_step\_y\_offset-1+y) + alpha \* integral\_a;

585

586 end

587

588 % найдём минимальное значение Gn(y) на текущем шаге и его индекс в таблице Gn(y)

589 [Gny\_min,Gny\_min\_i] = min(Gny(n+1,2:first\_step\_y\_count+1));

590

591 % и соответствующий y = R

592 R = Gny(1,Gny\_min\_i+1);

593

594 % найдём значение одношаговых затрат g(R)

595 gR = h \* int\_h\_d\_values(1,Gny\_min\_i) + d \* int\_h\_d\_values(2,Gny\_min\_i);

596

597 % запишем значение в столбец R(n)

598 Rn(n,1) = R;

599

600 % заполним строку таблицы Cn(x)

601 for y = 1:this\_step\_y\_count

602

603 % текущее значение x

604 x\_value = Gny(1,this\_step\_y\_offset+y);

605

606 if x\_value < R

607 Cnx\_value\_1 = -c \* x\_value + A + Gny\_min;

608 Cnx\_value\_2 = -c \* x\_value + Gny(n+1,this\_step\_y\_offset+y);

609 Cnx\_value = min(Cnx\_value\_1,Cnx\_value\_2);

610 else

611 Cnx\_value = -c \* x\_value + Gny(n+1,this\_step\_y\_offset+y);

612 end

613

614 % текущее значение функции Cn(x)

615 Cnx(n+1,this\_step\_y\_offset+y) = Cnx\_value;

616

617 end

618

619 % случай A = 0

620 if A == 0

621

622 r = R;

623 rn(n,1) = r;

624

625 else

626

627 % вычислим значение r (движение от R налево)

628 for i = Gny\_min\_i+1:-1:this\_step\_y\_offset+1

629

630 % предыдущее значение r

631 r\_val\_prev = r\_val;

632

633 % текущее значение r

634 r\_val = Gny(1,i);

635

636 % предыдущее значение выражения G(r) - G(R) - A

637 delta\_val\_prev = delta\_val;

638

639 % значение выражения G(r) - G(R) - A

640 delta\_val = Gny(n+1,i) - Gny\_min - A;

641

642 % ищем смену знака выражения G(r) - G(R) - A

643 if i <= Gny\_min\_i

644 if (delta\_val >= 0 && delta\_val\_prev < 0) || (delta\_val <= 0 && delta\_val\_prev > 0)

645 r = (r\_val + r\_val\_prev) / 2;

646 rn(n,1) = r;

647 break;

648 end

649 end

650

651 end

652

653 end

654

655 % останавливаем цикл для альфа = 0

656 if alpha == 0

657 break;

658 end

659

660 % сравним предыдущие значения R

661 if n >= N\_min

662

663 % останавливаем цикл при достижении заданного количества одинаковых значений R

664 if all(Rn(n-stable\_R\_count+1:n,1) == Rn(n,1))

665 break;

666 end

667

668 end

669

670 end % конец цикла по N

671

672 % записываем найденные значения R(v2) и r(v2)

673 v1R(1,v2+1) = R;

674 v1r(1,v2+1) = r;

675

676 % строка массива v1, R, r

677 if v2\_count == 1

678 v1Rr\_row(1,1) = v1\_value;

679 v1Rr\_row(1,2) = R;

680 v1Rr\_row(1,3) = r;

681 end

682

683 % вставим столбец значений R(n) в таблицу R(n,v1,v2)

684 Rn\_g(:,v1,v2) = Rn(:,1);

685 rn\_g(:,v1,v2) = rn(:,1);

686

687 % запишем таблицы Gn(y) и Cn(x) в общие таблицы

688 if save\_Gny\_and\_Cnx == 1

689 Gny\_g(:,:,v1,v2) = Gny;

690 Cnx\_g(:,:,v1,v2) = Cnx;

691 end

692

693 % выводим в консоль отчёт о проделанных операциях

694 fprintf('%s %d %d %.2f %.3f\n',datestr(now),v1,v2,R,r);

695

696 end

697

698 % записываем строки значений R и r

699 v1v2R(v1+1,:) = v1R;

700 v1v2r(v1+1,:) = v1r;

701

702 % заполним массив v1, R, r строкой R, r

703 if v2\_count == 1

704 v1Rr(v1,:) = v1Rr\_row;

705 end

706

707 end

708

709 % время вычислений в секундах, минутах, часах

710 time\_s = toc

711 time\_m = round((time\_s / 60),2)

712 time\_h = round((time\_s / 3600),2)

713

714 % выводим время исполнения в консоль

715 fprintf('%s Finished %.0fs %.2fh\n',datestr(now),time\_s,time\_h);

716

717 % выбираем что построить: график или поверхность

718 if v1\_count == 1 || v2\_count == 1

719

720 % выбираем x и y

721 if v1\_count == 1

722 xr = v1v2r(1,2:v2\_count);

723 yr = v1v2r(2,2:v2\_count);

724 xR = v1v2R(1,2:v2\_count);

725 yR = v1v2R(2,2:v2\_count);

726 else

727 xr = v1v2r(2:v1\_count,1);

728 yr = v1v2r(2:v1\_count,2);

729 xR = v1v2R(2:v1\_count,1);

730 yR = v1v2R(2:v1\_count,2);

731 end

732

733 % строим график

734 plot(xr,yr,xR,yR);

735

736 % указываем названия осей

737 if v1\_count == 1

738 xlabel(v2\_name);

739 y\_label\_str = sprintf('R(%s), r(%s)',v2\_name,v2\_name);

740 else

741 xlabel(v1\_name);

742 y\_label\_str = sprintf('R(%s), r(%s)',v1\_name,v1\_name);

743 end

744 ylabel(y\_label\_str);

745

746 % графики функций C(x) и G(y)

747 if save\_Gny\_and\_Cnx == 1

748

749 % промежуточные параметры для постройки массивов C(x) и G(y)

750 n = N;

751 v1\_n = v1\_count;

752 this\_step\_y\_min = last\_step\_y\_min - z\_max \* (N - n);

753 this\_step\_y\_offset = round((this\_step\_y\_min - first\_step\_y\_min) / y\_step + 1);

754

755 % создаём массив C(x)

756 C = [];

757 C(1,:) = Cnx\_g(1,this\_step\_y\_offset+1:first\_step\_y\_count+1,v1\_n);

758 C(2,:) = Cnx\_g(n+1,this\_step\_y\_offset+1:first\_step\_y\_count+1,v1\_n);

759 C = C';

760

761 % создаём массив G(y)

762 G = [];

763 G(1,:) = Gny\_g(1,this\_step\_y\_offset+1:first\_step\_y\_count+1,v1\_n);

764 G(2,:) = Gny\_g(n+1,this\_step\_y\_offset+1:first\_step\_y\_count+1,v1\_n);

765 G = G';

766

767 % создаём общий массив x, C(x) и G(y)

768 xCG(:,1) = C(:,1);

769 xCG(:,2) = C(:,2);

770 xCG(:,3) = G(:,2);

771

772 % строим график функции С(x)

773 xC = C(:,1);

774 yC = C(:,2);

775 plot(xC,yC);

776 xlabel('x');

777 ylabel('C(x)');

778 legend({'C(x)'},'Location','southwest');

779

780 end

781

782 else

783

784 % задаём сетку поверхностей R(v1,v2) и r(v1,v2)

785 x = v1v2r(1,2:v2\_count+1);

786 y = v1v2r(2:v1\_count+1,1);

787 [X,Y] = meshgrid(x,y);

788 Zr = v1v2r(2:v1\_count+1,2:v2\_count+1);

789

790 % поверхность r(v1,v2)

791 z\_label\_str = sprintf('R(%s,%s), r(%s,%s)',v1\_name,v2\_name,v1\_name,v2\_name);

792 figure('Name',z\_label\_str,'NumberTitle','off');

793 surf(X,Y,Zr,'FaceAlpha',1,'EdgeColor','white','EdgeAlpha',0.3);

794 xlabel(v2\_name,'FontSize',30);

795 ylabel(v1\_name,'FontSize',30);

796 zlabel(z\_label\_str,'FontSize',30);

797

798 % поверхность R(v1,v2)

799 hold on;

800 ZR = v1v2R(2:v1\_count+1,2:v2\_count+1);

801 surf(X,Y,ZR,'FaceAlpha',1,'EdgeAlpha',0.2);

802 hold off;

803 set(gca, 'CameraPosition', [130 -550 90]);

804 pbaspect([1 1 0.75]);

805

806 end

807

808 % соберём название файла для сохранения результатов

809 v1\_min\_str = strrep(num2str(v1\_min),'.','');

810 v1\_max\_str = strrep(num2str(v1\_max),'.','');

811 v1\_step\_str = strrep(num2str(v1\_step),'.','');

812 y\_step\_str = strrep(num2str(y\_step),'.','');

813

814 % запишем данные второй переменной

815 if v2\_count == 1

816 v2\_str = '';

817 else

818 v2\_min\_str = strrep(num2str(v2\_min),'.','');

819 v2\_max\_str = strrep(num2str(v2\_max),'.','');

820 v2\_step\_str = strrep(num2str(v2\_step),'.','');

821 v2\_str = sprintf('\_%s\_%s\_%s\_%s',v2\_name,v2\_min\_str,v2\_max\_str,v2\_step\_str);

822 end

823

824 % запишем распределение

825 if CDF == 1

826 cdf\_str = sprintf('gauss\_myu%s\_sigma%s',strrep(num2str(myu),'.',''),strrep(num2str(sigma),'.',''));

827 elseif CDF == 2

828 cdf\_str = sprintf('exp\_lambda%s',strrep(num2str(lambda),'.',''));

829 elseif CDF == 3

830 cdf\_str = sprintf('uniform\_a%s\_b%s',strrep(num2str(uniform\_a),'.',''),strrep(num2str(uniform\_b),'.',''));

831 end

832

833 % отметим случай ненадёжных поставщиков

834 if unreliable == 1

835 if c\_reverse == 1

836 u\_str = 'un\_r\_';

837 else

838 u\_str = 'un\_o\_';

839 end

840 else

841 u\_str = '';

842 end

843

844 % собираем итоговое название файла

845 %fname = sprintf('0\_im\_%s%s\_%s\_%s\_%s%s\_y%s\_%s.mat',u\_str,v1\_name,v1\_min\_str,v1\_max\_str,v1\_step\_str,v2\_str,y\_step\_str,cdf\_str);

846 fname = sprintf('0\_im\_CDF%.f.mat',CDF);

847

848 % сохраняем файл

849 save(fname);

850

851 % звуковой сигнал об окончании расчётов

852 load chirp;

853 sound(y,1\*Fs);

1. Исходный код модели сравнения затрат «близоруких», стационарных и «дальнозорких» стратегий управления методом Монте-Карло

1 % очищаем рабочую область

2 clear variables;

3

4 % очищаем консоль

5 clc;

6

7 % задаём формат вывода чисел в консоли

8 format longG;

9

10 % начинаем отсчёт времени

11 unix\_start = floor(posixtime(datetime('now','TimeZone','local')));

12

13 % выведем в консоль время начала вычислений

14 fprintf('%s -> %s.m start\n\n',datestr(now),mfilename);

15

16 % обозначим глобальные переменные

17 global myu sigma lambda uniform\_a uniform\_b CDF y\_step last\_step\_y\_min last\_step\_y\_max z\_min z\_max z\_step z\_count fz uniform\_pdf;

18

19 % математическое ожидание и среднеквадратическое отклонение нормальной функции распределения

20 myu = 5;

21 sigma = 1;

22 % интенсивность экспоненциальной функции распределения

23 lambda = 0.2;

24 % границы функции непрерывного равномерного распределения

25 uniform\_a = 0;

26 uniform\_b = 10;

27

28 % выбираем функцию распределения спроса (1 - нормальное, 2 - экспоненциальное, 3 - равномерное)

29 CDF = 1;

30 % коэффициент дисконтирования

31 alpha = 0.99;

32 % стоимость единичной поставки

33 A = 100;

34 % цена единицы товара

35 c = 10;

36 % стоимость хранение единицы товара

37 h = 1;

38 % потери от дефицита единицы товара

39 d = 11;

40 % количество шагов

41 N = 30;

42 % количество экспериментов по методу Монте-Карло

43 Nmc = 10^5;

44

45 % минимум y на последнем шаге

46 last\_step\_y\_min = -10;

47 % максимум y на последнем шаге

48 last\_step\_y\_max = 100;

49 % размер шага y

50 y\_step = 0.01;

51

52 % среднее значение спроса m

53 if CDF == 1

54 m = myu;

55 elseif CDF == 2

56 m = 1 / lambda;

57 elseif CDF == 3

58 m = (uniform\_b - uniform\_a) / 2;

59 end

60

61 % нижний предел интегрирования интеграла при альфа

62 z\_min = 0;

63 % верхний предел интегрирования интеграла при альфа

64 if CDF == 1

65 z\_max = myu + 5 \* sigma;

66 elseif CDF == 2

67 z\_max = 1 / lambda \* 6;

68 elseif CDF == 3

69 z\_max = min(uniform\_b,last\_step\_y\_max);

70 uniform\_pdf = 1 / (uniform\_b - uniform\_a);

71 end

72

73 % размер шага z

74 z\_step = y\_step;

75 % количество значений z

76 z\_count = round((z\_max - z\_min) / z\_step + 1);

77

78 % подготовим таблицу предварительно рассчитанных значений функции f(z) (кэш значений)

79 fz = NaN(1,z\_count);

80 for z = 1:z\_count

81 % значение z

82 z\_value = z\_min + z\_step \* (z - 1);

83 % выбираем одну из функций распределения

84 if CDF == 1

85 fz(1,z) = 1 / (sigma \* (2 \* pi)^0.5) \* exp(((z\_value - myu)^2) / (-2 \* sigma^2));

86 elseif CDF == 2

87 fz(1,z) = lambda \* exp(-1 \* lambda \* z\_value);

88 elseif CDF == 3

89 if z\_value >= uniform\_a && z\_value <= uniform\_b

90 fz(1,z) = uniform\_pdf;

91 else

92 fz(1,z) = 0;

93 end

94 end

95 end

96

97 % расчитаем значения оптимальной стратегии управления

98 [nRr, xC] = calc\_Rr(alpha,A,c,h,d,N);

99

100 % определим значения оптимальных параметров R и r

101 [nRr\_h,nRr\_w] = size(nRr);

102 R\_myopic = nRr(2,2);

103 r\_myopic = nRr(2,3);

104 R\_opt = mean(nRr(ceil(nRr\_h\*2/3):nRr\_h,2));

105 r\_opt = mean(nRr(ceil(nRr\_h\*2/3):nRr\_h,3));

106

107 % выведем значения в консоль

108 fprintf('R\_myopic = %.2f -> r\_myopic = %.3f\n',R\_myopic,r\_myopic);

109 fprintf('R\_opt = %.2f -> r\_opt = %.3f\n\n',R\_opt,r\_opt);

110

111 % выводим время исполнения в консоль

112 time\_im = floor(posixtime(datetime('now','TimeZone','local'))) - unix\_start;

113 time\_im = [time\_im, time\_im/60, time\_im/3600];

114 fprintf('%s -> %.0fs / %.2fm / %.2fh\n\n',datestr(now),time\_im(1),time\_im(2),time\_im(3));

115

116 % минимум фиктивного уровня запасов x

117 x\_min = last\_step\_y\_min;

118 % максимум фиктивного уровня запасов x

119 x\_max = last\_step\_y\_max;

120 % размер шага x

121 x\_step = y\_step;

122 % количество значений x

123 x\_count = (x\_max - x\_min) / x\_step + 1;

124

125 % количество знаков правее запятой в размере шага

126 z\_step = x\_step;

127 sp = 0;

128 while floor(z\_step \* 10^sp) ~= z\_step \* 10^sp

129 sp = sp + 1;

130 end

131

132 % таблица значений функции суммарных средних затрат C(x)

133 x\_Cmyopic\_Cstatic\_Cdynamic = NaN(x\_count,4);

134

135 % параллельный цикл по фиктивному уровню запасов x

136 parfor x = 1:x\_count

137

138 % начальное значение x фиктивного уровня запасов

139 x\_init = x\_min + x\_step \* (x - 1);

140

141 % значение суммарных средних затрат C(x)

142 C\_myopic = 0;

143 C\_static = 0;

144

145 % цикл экспериментов по методу Монте-Карло

146 for m = 1:Nmc

147

148 % значение x фиктивного уровня запасов

149 x\_myopic = x\_init;

150 x\_static = x\_init;

151

152 % цикл по шагам стратегии управления запасами

153 for n = 1:N

154

155 % очищаем значения переменных спроса и затрат на шаге

156 zn = 0;

157 Cn\_myopic = 0;

158 Cn\_static = 0;

159

160 % определим значение случайного спроса

161 zn\_ready = 0;

162 while zn\_ready == 0

163 % произвольные нормализованные координаты

164 rx = z\_min + (z\_max - z\_min) \* rand;

165 rx = round(rx,sp);

166 ry = rand;

167 % определим индекс значения rx в таблице значений функции плотности вероятности

168 z\_index = round((rx - z\_min) / z\_step + 1);

169 % если значение ry меньше значения функции плотности вероятности в точке rx => событие произошло (с вероятностью ry)

170 if fz(1,z\_index) > ry

171 % прерываем цикл

172 zn\_ready = 1;

173 % сохраняем значение спроса

174 zn = rx;

175 end

176 end

177

178 % значение x фиктивного уровня запасов после удовлетворения потребительского спроса на шаге

179 x\_myopic = x\_myopic - zn;

180 x\_static = x\_static - zn;

181

182 % определим размер заказа Un, затраты на размещение заказа и его стоимость

183 if x\_myopic < r\_myopic

184 Cn\_myopic = Cn\_myopic + A + c \* (R\_myopic - x\_myopic);

185 x\_myopic = R\_myopic;

186 end

187

188 % определим размер заказа Un, затраты на размещение заказа и его стоимость

189 if x\_static < r\_opt

190 Cn\_static = Cn\_static + A + c \* (R\_opt - x\_static);

191 x\_static = R\_opt;

192 end

193

194 % определим затраты на хранение и потери вследствие дефицита

195 if x\_myopic > 0

196 Cn\_myopic = Cn\_myopic + x\_myopic \* h;

197 else

198 Cn\_myopic = Cn\_myopic + abs(x\_myopic) \* d;

199 end

200

201 % определим затраты на хранение и потери вследствие дефицита

202 if x\_static > 0

203 Cn\_static = Cn\_static + x\_static \* h;

204 else

205 Cn\_static = Cn\_static + abs(x\_static) \* d;

206 end

207

208 % учтём инфляцию

209 Cn\_myopic = (alpha ^ (n - 1)) \* Cn\_myopic;

210 Cn\_static = (alpha ^ (n - 1)) \* Cn\_static;

211

212 % добавим затраты за шаг к общим затратам

213 C\_myopic = C\_myopic + Cn\_myopic;

214 C\_static = C\_static + Cn\_static;

215

216 end

217

218 end

219

220 % определим среднее значение суммарных средних затрат для всех экспериментов

221 C\_myopic = C\_myopic / Nmc;

222 C\_static = C\_static / Nmc;

223 C\_dynamic = xC(x,2);

224

225 % сохраним значения

226 x\_Cmyopic\_Cstatic\_Cdynamic(x,:) = [x\_init C\_myopic C\_static C\_dynamic];

227

228 end

229

230 % построим график суммарных средних затрат С(x)

231 x = x\_Cmyopic\_Cstatic\_Cdynamic(:,1);

232 y1 = x\_Cmyopic\_Cstatic\_Cdynamic(:,2);

233 y2 = x\_Cmyopic\_Cstatic\_Cdynamic(:,3);

234 y3 = x\_Cmyopic\_Cstatic\_Cdynamic(:,4);

235 plot(x,y1,x,y2,x,y3);

236 xlabel('x');

237 ylabel('C(x)');

238 legend({'Myopic','Static','Dynamic'},'Location','southwest');

239

240 % индексы

241 i\_min = round((-m - last\_step\_y\_min) / y\_step + 1);

242 i\_max = round((R\_opt\*1.05 - last\_step\_y\_min) / y\_step + 1);

243

244 % расчитаем суммарные средние затраты

245 Cmyopic\_sum = sum(x\_Cmyopic\_Cstatic\_Cdynamic(i\_min:i\_max,2));

246 Cstatic\_sum = sum(x\_Cmyopic\_Cstatic\_Cdynamic(i\_min:i\_max,3));

247 Cdynamic\_sum = sum(x\_Cmyopic\_Cstatic\_Cdynamic(i\_min:i\_max,4));

248

249 % расчитаем выигрыш от использования различных стратегий

250 gain\_myopic\_to\_dynamic = (Cmyopic\_sum - Cdynamic\_sum) / Cmyopic\_sum \* 100;

251 gain\_static\_to\_dynamic = (Cstatic\_sum - Cdynamic\_sum) / Cstatic\_sum \* 100;

252 gain\_myopic\_to\_static = (Cmyopic\_sum - Cstatic\_sum) / Cmyopic\_sum \* 100;

253

254 % выводим значения в консоль

255 fprintf('myopic -> dynamic = %.2f %%\n',gain\_myopic\_to\_dynamic);

256 fprintf('myopic -> static = %.2f %%\n',gain\_myopic\_to\_static);

257 fprintf('static -> dynamic = %.2f %%\n\n',gain\_static\_to\_dynamic);

258

259 % выберем значения для построение графика

260 graph\_step = 0.1;

261 if y\_step >= graph\_step

262 graph\_step = y\_step;

263 end

264 graph\_count = (x\_max - x\_min) / graph\_step + 1;

265 graph\_x\_Cmyopic\_Cstatic\_Cdynamic = NaN(graph\_count,4);

266 for g = 1:graph\_count

267 x = graph\_step / x\_step \* (g - 1) + 1;

268 graph\_x\_Cmyopic\_Cstatic\_Cdynamic(g,:) = x\_Cmyopic\_Cstatic\_Cdynamic(x,:);

269 end

270

271 % звуковой сигнал об окончании расчётов

272 end\_signal = audioplayer([sin(1:.6:400),sin(1:.7:400),sin(1:.4:400)],22050);

273 play(end\_signal);

274

275 % выводим время исполнения в консоль

276 time\_exec = floor(posixtime(datetime('now','TimeZone','local'))) - unix\_start;

277 time\_exec = [time\_exec, time\_exec/60, time\_exec/3600];

278 fprintf('%s -> %s.m ended -> %.0fs / %.2fm / %.2fh\n',datestr(now),mfilename,time\_exec(1),time\_exec(2),time\_exec(3));

279

280 % сохраняем файл с данными

281 fname = sprintf('2\_mf\_A%.fc%.fh%.fd%.fa%s\_N%.f\_CDF%.f\_step%s.mat',A,c,h,d,strrep(num2str(alpha),'.',''),N,CDF,strrep(num2str(y\_step),'.',''));

282 %save(fname);

283

284 %запускаем следующий расчёт

285 next\_script = strcat('mf\_',num2str(sscanf(mfilename(),'mf\_%d')+1));

286 if exist(next\_script,'file') == 2

287 run(next\_script);

288 end

289

290 % функция расчёта значений R, r и Cnx

291 function [nRr, xC] = calc\_Rr(alpha,A,c,h,d,N)

292

293 % обозначим глобальные и символьные переменные

294 global myu sigma lambda uniform\_a uniform\_b CDF y\_step last\_step\_y\_min last\_step\_y\_max z\_min z\_max z\_step z\_count fz uniform\_pdf;

295 syms y\_sym z\_sym lambda\_sym sigma\_sym myu\_sym pi\_sym;

296

297 % минимум y на первом шаге

298 first\_step\_y\_min = last\_step\_y\_min - z\_max \* (N - 1);

299 % максимум y на первом шаге

300 first\_step\_y\_max = last\_step\_y\_max - z\_min \* (N - 1);

301 % количество значений y на первом шаге

302 first\_step\_y\_count = round((first\_step\_y\_max - first\_step\_y\_min) / y\_step + 1);

303

304 % выбираем функцию распределения

305 if CDF == 1

306

307 % нормальное распределение

308

309 % предварительный расчёт интеграла при h

310 int\_h\_sym = str2sym('(y\_sym - z\_sym) / (sigma\_sym \* (2 \* pi\_sym)^0.5) \* exp(((z\_sym-myu\_sym)^2) / (-2 \* sigma\_sym^2))');

311 int\_h\_sym = subs(int\_h\_sym,sigma\_sym,sigma);

312 int\_h\_sym = subs(int\_h\_sym,myu\_sym,myu);

313 int\_h\_sym = subs(int\_h\_sym,pi\_sym,pi);

314

315 % интеграл при h (от 0 до y)

316 int\_h\_0\_y = int(int\_h\_sym,z\_sym,0,y\_sym);

317

318 % предварительный расчёт интеграла при d

319 int\_d\_sym = str2sym('(z\_sym - y\_sym) / (sigma\_sym \* (2 \* pi\_sym)^0.5) \* exp(((z\_sym-myu\_sym)^2) / (-2 \* sigma\_sym^2))');

320 int\_d\_sym = subs(int\_d\_sym,sigma\_sym,sigma);

321 int\_d\_sym = subs(int\_d\_sym,myu\_sym,myu);

322 int\_d\_sym = subs(int\_d\_sym,pi\_sym,pi);

323

324 % интеграл при d (от y до бесконечности)

325 int\_d\_y\_inf = int(int\_d\_sym,z\_sym,y\_sym,inf);

326

327 % интеграл при d (от 0 до бесконечности)

328 int\_d\_0\_inf = int(int\_d\_sym,z\_sym,0,inf);

329

330 elseif CDF == 2

331

332 % экспоненциальное распределение

333

334 % предварительный расчёт интеграла при h

335 int\_h\_sym = str2sym('(y\_sym - z\_sym) \* lambda\_sym \* exp(-1 \* lambda\_sym \* z\_sym)');

336 int\_h\_sym = subs(int\_h\_sym,lambda\_sym,lambda);

337

338 % интеграл при h (от 0 до y)

339 int\_h\_0\_y = int(int\_h\_sym,z\_sym,0,y\_sym);

340

341 % предварительный расчёт интеграла при d

342 int\_d\_sym = str2sym('(z\_sym - y\_sym) \* lambda\_sym \* exp(-1 \* lambda\_sym \* z\_sym)');

343 int\_d\_sym = subs(int\_d\_sym,lambda\_sym,lambda);

344

345 % интеграл при d (от y до бесконечности)

346 int\_d\_y\_inf = int(int\_d\_sym,z\_sym,y\_sym,inf);

347

348 % интеграл при d (от 0 до бесконечности)

349 int\_d\_0\_inf = int(int\_d\_sym,z\_sym,0,inf);

350

351 end

352

353 % подготовим таблицу рассчитанных значений интегралов при h и d

354 int\_h\_d\_values = NaN(2,first\_step\_y\_count);

355

356 % заполним таблицу g(y)

357 for y = 1:first\_step\_y\_count

358

359 % значение y

360 y\_value = first\_step\_y\_min + y\_step \* (y - 1);

361

362 if CDF == 1 || CDF == 2

363

364 % определяем пределы интегрирования интегралов h и d

365 if y\_value <= 0

366 h\_int = 0;

367 d\_int = subs(int\_d\_0\_inf,y\_sym,y\_value);

368 else

369 h\_int = subs(int\_h\_0\_y,y\_sym,y\_value);

370 d\_int = subs(int\_d\_y\_inf,y\_sym,y\_value);

371 end

372

373 elseif CDF == 3

374

375 % определяем пределы интегрирования интегралов h и d

376 h\_int\_min = 0;

377 d\_int\_max = min(first\_step\_y\_max,uniform\_b+1);

378 if y\_value <= 0

379 h\_int\_max = 0;

380 d\_int\_min = 0;

381 else

382 h\_int\_max = y\_value;

383 d\_int\_min = y\_value;

384 end

385 h\_int\_count = (h\_int\_max - h\_int\_min) / z\_step + 1;

386 d\_int\_count = (d\_int\_max - d\_int\_min) / z\_step + 1;

387

388 % начальное значение интеграла при h

389 h\_int = 0;

390

391 if h\_int\_count > 1

392

393 % подсчитаем значение интеграла при h как среднее арифметическое верхней и нижней сумм Дарбу

394 for z = 1:h\_int\_count

395

396 % текущее значение z

397 z\_value = h\_int\_min + z\_step \* (z - 1);

398

399 % предыдущее значение подынтегральной функции

400 if z >= 2

401 h\_prev\_val = h\_this\_val;

402 end

403

404 % определим значение функции плотности вероятности в текущей точке z

405 if z\_value >= uniform\_a && z\_value <= uniform\_b

406 uniform\_pdf\_val = uniform\_pdf;

407 else

408 uniform\_pdf\_val = 0;

409 end

410

411 % значение подынтегральной функции данного среза

412 h\_this\_val = (y\_value - z\_value) \* uniform\_pdf\_val;

413

414 if z >= 2 && (z\_value > uniform\_a && z\_value <= uniform\_b)

415

416 % знаки текущего и предыдущего значений

417 h\_this\_sign = sign(h\_this\_val);

418 h\_prev\_sign = sign(h\_prev\_val);

419

420 % находим значение площади среза подынтегральной функции h

421 if (h\_this\_sign == 1 && h\_prev\_sign == -1) || (h\_this\_sign == -1 && h\_prev\_sign == 1)

422 h\_this\_step\_area = (abs(h\_this\_val) + abs(h\_prev\_val)) \* z\_step / 4;

423 else

424 h\_this\_step\_area = (min(abs(h\_this\_val),abs(h\_prev\_val)) + abs(abs(h\_this\_val) - abs(h\_prev\_val)) / 2) \* z\_step;

425 end

426

427 % добавим площадь среза к площади всего подграфика

428 h\_int = h\_int + h\_this\_step\_area;

429

430 end

431

432 end

433

434 end

435

436 % начальное значение интеграла при d

437 d\_int = 0;

438

439 if d\_int\_count > 1

440

441 % подсчитаем значение интеграла при h как среднее арифметическое верхней и нижней сумм Дарбу

442 for z = 1:d\_int\_count

443

444 % текущее значение z

445 z\_value = d\_int\_min + z\_step \* (z - 1);

446

447 % предыдущее значение подынтегральной функции

448 if z >= 2

449 d\_prev\_val = d\_this\_val;

450 end

451

452 % определим значение функции плотности вероятности в текущей точке z

453 if z\_value >= uniform\_a && z\_value <= uniform\_b

454 uniform\_pdf\_val = uniform\_pdf;

455 else

456 uniform\_pdf\_val = 0;

457 end

458

459 % значение подынтегральной функции данного среза

460 d\_this\_val = (z\_value - y\_value) \* uniform\_pdf\_val;

461

462 if z >= 2 && (z\_value > uniform\_a && z\_value <= uniform\_b)

463

464 % знаки текущего и предыдущего значений

465 d\_this\_sign = sign(d\_this\_val);

466 d\_prev\_sign = sign(d\_prev\_val);

467

468 % находим значение площади среза подынтегральной функции h

469 if (d\_this\_sign == 1 && d\_prev\_sign == -1) || (d\_this\_sign == -1 && d\_prev\_sign == 1)

470 d\_this\_step\_area = (abs(d\_this\_val) + abs(d\_prev\_val)) \* z\_step / 4;

471 else

472 d\_this\_step\_area = (min(abs(d\_this\_val),abs(d\_prev\_val)) + abs(abs(d\_this\_val) - abs(d\_prev\_val)) / 2) \* z\_step;

473 end

474

475 % добавим площадь среза к площади всего подграфика

476 d\_int = d\_int + d\_this\_step\_area;

477

478 end

479

480 end

481

482 end

483

484 end

485

486 int\_h\_d\_values(1,y) = h\_int;

487 int\_h\_d\_values(2,y) = d\_int;

488

489 end

490

491 r\_val = NaN;

492 r\_val\_prev = NaN;

493 delta\_val = NaN;

494 delta\_val\_prev = NaN;

495

496 % задаём рамки таблицы Gn(y), где первый столбец - это значения n, а первая строка - это значения y

497 Gny = NaN(N+1,first\_step\_y\_count+1);

498

499 % задаём рамки таблицы Cn(x), где первый столбец - это значения n, а первая строка - это значения x

500 Cnx = NaN(N+1,first\_step\_y\_count+1);

501

502 % первый столбец (значения n)

503 for n = 1:N

504 Gny(n+1,1) = n;

505 Cnx(n+1,1) = n;

506 end

507

508 % первый ряд (значения y/x)

509 for y = 1:first\_step\_y\_count

510 Gny(1,y+1) = first\_step\_y\_min + y\_step \* (y - 1);

511 Cnx(1,y+1) = first\_step\_y\_min + y\_step \* (y - 1);

512 end

513

514 % значения функции Cn(x) на первом шаге

515 Cnx(2,2:first\_step\_y\_count+1) = 0;

516

517 % подготовим столбец R(n) и r(n)

518 Rn = NaN(N,1);

519 rn = NaN(N,1);

520

521 % начинаем выполнять шаги

522 for n = 2:N

523

524 % минимум y текущего шага

525 this\_step\_y\_min = last\_step\_y\_min - z\_max \* (N - n);

526

527 % максимум y текущего шага

528 this\_step\_y\_max = last\_step\_y\_max - z\_min \* (N - n);

529

530 % количество значений y на текущем шаге

531 this\_step\_y\_count = round((this\_step\_y\_max - this\_step\_y\_min) / y\_step + 1);

532

533 % индекс смещения y для текущего шага, с учётом первой колонки Gn(y) и Cn(x)

534 this\_step\_y\_offset = round((this\_step\_y\_min - first\_step\_y\_min) / y\_step + 1);

535

536 % цикл по y - заполняем строку таблицы Gn(y)

537 for y = 1:this\_step\_y\_count

538

539 % текущее значение y

540 y\_value = Gny(1,this\_step\_y\_offset+y);

541

542 % вычислим значение интеграла при альфа

543 integral\_a = 0;

544

545 if alpha ~= 0 && n ~= 2

546

547 % цикл по z

548 for z = 1:z\_count

549

550 % предыдущее значение подынтегральной функции

551 if z >= 2

552 prev\_val = this\_val;

553 end

554

555 % текущее значение z

556 z\_value = z\_min + z\_step \* (z - 1);

557

558 % индекс значения x из таблицы Cn(x)

559 x\_index = round((y\_value - z\_value - first\_step\_y\_min) / y\_step + 1 + 1);

560

561 % текущее значение подынтегральной функции

562 this\_val = Cnx(n,x\_index) \* fz(1,z);

563

564 % площадь вписанного прямоугольника и треугольника на данном шаге z

565 if z >= 2

566

567 % знаки текущего и предыдущего значений

568 this\_sign = sign(this\_val);

569 prev\_sign = sign(prev\_val);

570

571 % находим значение площади среза z подынтегральной функции

572 if (this\_sign == 1 && prev\_sign == -1) || (this\_sign == -1 && prev\_sign == 1)

573 this\_step\_area = (abs(this\_val) + abs(prev\_val)) \* z\_step / 4;

574 else

575 this\_step\_area = (min(abs(this\_val),abs(prev\_val)) + abs(abs(this\_val) - abs(prev\_val)) / 2) \* z\_step;

576 end

577

578 % добавляем площадь среза к площади всего подграфика

579 integral\_a = integral\_a + this\_step\_area;

580

581 end

582

583 end

584

585 end

586

587 % текущее значение функции Gn(y)

588 Gny(n+1,this\_step\_y\_offset+y) = c \* y\_value + h \* int\_h\_d\_values(1,this\_step\_y\_offset-1+y) + d \* int\_h\_d\_values(2,this\_step\_y\_offset-1+y) + alpha \* integral\_a;

589

590 end

591

592 % найдём минимальное значение Gn(y) на текущем шаге и его индекс в таблице Gn(y)

593 [Gny\_min,Gny\_min\_i] = min(Gny(n+1,2:first\_step\_y\_count+1));

594

595 % и соответствующий y = R

596 R = Gny(1,Gny\_min\_i+1);

597

598 % найдём значение одношаговых затрат g(R)

599 gR = h \* int\_h\_d\_values(1,Gny\_min\_i) + d \* int\_h\_d\_values(2,Gny\_min\_i);

600

601 % запишем значение в столбец R(n)

602 Rn(n,1) = R;

603

604 % заполним строку таблицы Cn(x)

605 for y = 1:this\_step\_y\_count

606

607 % текущее значение x

608 x\_value = Gny(1,this\_step\_y\_offset+y);

609

610 if x\_value < R

611 Cnx\_value\_1 = -c \* x\_value + A + Gny\_min;

612 Cnx\_value\_2 = -c \* x\_value + Gny(n+1,this\_step\_y\_offset+y);

613 Cnx\_value = min(Cnx\_value\_1,Cnx\_value\_2);

614 else

615 Cnx\_value = -c \* x\_value + Gny(n+1,this\_step\_y\_offset+y);

616 end

617

618 % текущее значение функции Cn(x)

619 Cnx(n+1,this\_step\_y\_offset+y) = Cnx\_value;

620

621 end

622

623 % случай A = 0

624 if A == 0

625

626 r = R;

627 rn(n,1) = r;

628

629 else

630

631 % вычислим значение r (движение от R налево)

632 for i = Gny\_min\_i+1:-1:this\_step\_y\_offset+1

633

634 % предыдущее значение r

635 r\_val\_prev = r\_val;

636

637 % текущее значение r

638 r\_val = Gny(1,i);

639

640 % предыдущее значение выражения G(r) - G(R) - A

641 delta\_val\_prev = delta\_val;

642

643 % значение выражения G(r) - G(R) - A

644 delta\_val = Gny(n+1,i) - Gny\_min - A;

645

646 % ищем смену знака выражения G(r) - G(R) - A

647 if i <= Gny\_min\_i

648 if (delta\_val >= 0 && delta\_val\_prev < 0) || (delta\_val <= 0 && delta\_val\_prev > 0)

649 r = (r\_val + r\_val\_prev) / 2;

650 rn(n,1) = r;

651 break;

652 end

653 end

654

655 end

656

657 end

658

659 % останавливаем цикл для альфа = 0

660 if alpha == 0

661 break;

662 end

663

664 end % конец цикла по N

665

666 for n = 1:N

667 n\_vals(n,1) = n;

668 end

669

670 % массив значений n, R и r

671 nRr = [n\_vals, Rn, rn];

672

673 % массив значений C(x)

674 this\_step\_y\_min = last\_step\_y\_min;

675 this\_step\_y\_offset = round((this\_step\_y\_min - first\_step\_y\_min) / y\_step + 1);

676 C = [];

677 C(1,:) = Cnx(1,this\_step\_y\_offset+1:first\_step\_y\_count+1);

678 C(2,:) = Cnx(N+1,this\_step\_y\_offset+1:first\_step\_y\_count+1);

679 C = C';

680 xC = C;

681

682 end

1. Исходный код модели сравнения затрат методом Монте-Карло для задачи управления запасами при изменении параметров

1 % очищаем рабочую область

2 clear variables;

3

4 % очищаем консоль

5 clc;

6

7 % задаём формат вывода чисел в консоли

8 format longG;

9

10 % начинаем отсчёт времени

11 unix\_start = floor(posixtime(datetime('now','TimeZone','local')));

12

13 % выведем в консоль ожидаемую длительность и ожидаемое окончание вычислений

14 fprintf('%s -> %s.m start\n\n',datestr(now),mfilename);

15

16 % обозначим глобальные переменные

17 global myu sigma lambda uniform\_a uniform\_b CDF y\_step last\_step\_y\_min last\_step\_y\_max z\_min z\_max z\_step z\_count fz uniform\_pdf;

18

19 % математическое ожидание и среднеквадратическое отклонение нормальной функции распределения

20 myu = 5;

21 sigma = 1;

22 % интенсивность экспоненциальной функции распределения

23 lambda = 0.2;

24 % границы функции непрерывного равномерного распределения

25 uniform\_a = 0;

26 uniform\_b = 10;

27

28 % выбираем функцию распределения спроса (1 - нормальное, 2 - экспоненциальное, 3 - равномерное)

29 CDF = 1;

30 % коэффициент дисконтирования

31 alpha1 = 0.99;

32 % стоимость единичной поставки

33 A1 = 300;

34 % цена единицы товара

35 c1 = 10;

36 % стоимость хранение единицы товара

37 h1 = 1;

38 % потери от дефицита единицы товара

39 d1 = 60;

40 % количество шагов

41 N1 = 30;

42 % количество экспериментов по методу Монте-Карло

43 Nmc = 10^5;

44

45 % минимум y на последнем шаге

46 last\_step\_y\_min = -10;

47 % максимум y на последнем шаге

48 last\_step\_y\_max = 100;

49 % размер шага y

50 y\_step = 0.1;

51

52 % эконометрические параметры после их изменения

53 alpha2 = 0.99;

54 A2 = 100;

55 c2 = 10;

56 h2 = 2;

57 d2 = 20;

58 N2 = floor(N1/2);

59

60 % среднее значение спроса m

61 if CDF == 1

62 m = myu;

63 elseif CDF == 2

64 m = 1 / lambda;

65 elseif CDF == 3

66 m = (uniform\_b - uniform\_a) / 2;

67 end

68

69 % среднее значение цикла пополнения

70 tau = ceil((2 \* A1 \* h1 / m)^0.5 / m);

71 tau\_shift = 1;

72 tau\_shift4 = 2;

73

74 % нижний предел интегрирования интеграла при альфа

75 z\_min = 0;

76 % верхний предел интегрирования интеграла при альфа

77 if CDF == 1

78 z\_max = myu + 5 \* sigma;

79 elseif CDF == 2

80 z\_max = 1 / lambda \* 6;

81 elseif CDF == 3

82 z\_max = min(uniform\_b,last\_step\_y\_max);

83 uniform\_pdf = 1 / (uniform\_b - uniform\_a);

84 end

85

86 % размер шага z

87 z\_step = y\_step;

88 % количество значений z

89 z\_count = round((z\_max - z\_min) / z\_step + 1);

90

91 % подготовим таблицу предварительно рассчитанных значений функции f(z) (кэш значений)

92 fz = NaN(1,z\_count);

93 for z = 1:z\_count

94 % значение z

95 z\_value = z\_min + z\_step \* (z - 1);

96 % выбираем одну из функций распределения

97 if CDF == 1

98 fz(1,z) = 1 / (sigma \* (2 \* pi)^0.5) \* exp(((z\_value - myu)^2) / (-2 \* sigma^2));

99 elseif CDF == 2

100 fz(1,z) = lambda \* exp(-1 \* lambda \* z\_value);

101 elseif CDF == 3

102 if z\_value >= uniform\_a && z\_value <= uniform\_b

103 fz(1,z) = uniform\_pdf;

104 else

105 fz(1,z) = 0;

106 end

107 end

108 end

109

110 % расчитаем значения оптимальной стратегии управления

111 [nRr1, xC1] = calc\_Rr(alpha1,A1,c1,h1,d1,N1+1);

112 [nRr2, xC2] = calc\_Rr(alpha2,A2,c2,h2,d2,N2+1);

113 [nRr3plus, xC3plus] = calc\_Rr(alpha2,A2,c2,h2,d2,N2+1+tau+tau\_shift4);

114

115 % создадим массив n, R и r для динамической стратегии

116 nRr3 = zeros(N1,3);

117 for n = 1:N1

118 if n <= N2

119 nRr3(n,:) = [n nRr2(n+1,2) nRr2(n+1,3)];

120 else

121 nRr3(n,:) = [n nRr1(n+1,2) nRr1(n+1,3)];

122 end

123 end

124 nRr3(:,:) = [nRr3(:,1) flip(nRr3(:,2)) flip(nRr3(:,3))];

125

126 % создадим массив n, R и r для динамической стратегии с более ранним переходом на вторые R, r

127 nRr4 = zeros(N1,3);

128 for n = 1:N1

129 if n <= N2+tau+tau\_shift4

130 nRr4(n,:) = [n nRr3plus(n+1,2) nRr3plus(n+1,3)];

131 else

132 nRr4(n,:) = [n nRr1(n+1,2) nRr1(n+1,3)];

133 end

134 end

135 nRr4(:,:) = [nRr4(:,1) flip(nRr4(:,2)) flip(nRr4(:,3))];

136

137 % определим значения оптимальных параметров R и r

138 [nRr1\_h,nRr1\_w] = size(nRr1);

139 R1 = mean(nRr1(ceil(nRr1\_h\*2/3):nRr1\_h,2));

140 r1 = mean(nRr1(ceil(nRr1\_h\*2/3):nRr1\_h,3));

141 [nRr2\_h,nRr2\_w] = size(nRr2);

142 R2 = mean(nRr2(ceil(nRr2\_h\*2/3):nRr2\_h,2));

143 r2 = mean(nRr2(ceil(nRr2\_h\*2/3):nRr2\_h,3));

144

145 % выведем значения в консоль

146 fprintf('R1 = %.2f -> r1 = %.3f\n',R1,r1);

147 fprintf('R2 = %.2f -> r2 = %.3f\n\n',R2,r2);

148

149 % выводим время исполнения в консоль

150 time\_im = floor(posixtime(datetime('now','TimeZone','local'))) - unix\_start;

151 time\_im = [time\_im, time\_im/60, time\_im/3600];

152 fprintf('%s -> %.0fs / %.2fm / %.2fh\n\n',datestr(now),time\_im(1),time\_im(2),time\_im(3));

153

154 % минимум фиктивного уровня запасов x

155 x\_min = last\_step\_y\_min;

156 % максимум фиктивного уровня запасов x

157 x\_max = last\_step\_y\_max;

158 % размер шага x

159 x\_step = y\_step;

160 % количество значений x

161 x\_count = (x\_max - x\_min) / x\_step + 1;

162

163 % количество знаков правее запятой в размере шага

164 z\_step = x\_step;

165 sp = 0;

166 while floor(z\_step \* 10^sp) ~= z\_step \* 10^sp

167 sp = sp + 1;

168 end

169

170 % момент смены параметров

171 Nswitch = floor(N1/2) + 1;

172

173 % момент остановки новых заказов в первый полупериод

174 Nstop = Nswitch - (tau + tau\_shift);

175

176 % параллельный цикл по фиктивному уровню запасов x

177 parfor x = 1:x\_count

178

179 % начальное значение x фиктивного уровня запасов

180 x\_init = x\_min + x\_step \* (x - 1);

181

182 % значение суммарных средних затрат C(x)

183 C1 = 0;

184 C2 = 0;

185 C3 = 0;

186 C4 = 0;

187

188 % цикл экспериментов по методу Монте-Карло

189 for m = 1:Nmc

190

191 % значение x фиктивного уровня запасов

192 x1 = x\_init;

193 x2 = x\_init;

194 x3 = x\_init;

195 x4 = x\_init;

196

197 % цикл по шагам стратегии управления запасами

198 for n = 1:N1

199

200 % очищаем значения переменных спроса и затрат на шаге

201 C1n = 0;

202 C2n = 0;

203 C3n = 0;

204 C4n = 0;

205

206 % момент смены параметров

207 if n >= Nswitch

208 alpha = alpha2;

209 A = A2;

210 c = c2;

211 h = h2;

212 d = d2;

213 R = R2;

214 r = r2;

215 else

216 alpha = alpha1;

217 A = A1;

218 c = c1;

219 h = h1;

220 d = d1;

221 R = R1;

222 r = r1;

223 end

224

225 % определим значение случайного спроса

226 zn = 0;

227 zn\_ready = 0;

228 while zn\_ready == 0

229 % произвольные нормализованные координаты

230 rx = z\_min + (z\_max - z\_min) \* rand;

231 rx = round(rx,sp);

232 ry = rand;

233 % определим индекс значения rx в таблице значений функции плотности вероятности

234 z\_index = round((rx - z\_min) / z\_step + 1);

235 % если значение ry меньше значения функции плотности вероятности в точке rx => событие произошло (с вероятностью ry)

236 if fz(1,z\_index) > ry

237 % прерываем цикл

238 zn\_ready = 1;

239 % сохраняем значение спроса

240 zn = rx;

241 end

242 end

243

244 % вариант 1 - стационарная стратегия

245 % значение x фиктивного уровня запасов после удовлетворения потребительского спроса на шаге

246 x1 = x1 - zn;

247 % определим размер заказа Un, затраты на размещение заказа и его стоимость

248 if x1 < r

249 un = R - x1;

250 x1 = x1 + un;

251 C1n = C1n + A + c \* un;

252 end

253 % определим затраты на хранение и потери вследствие дефицита

254 if x1 > 0

255 C1n = C1n + x1 \* h;

256 else

257 C1n = C1n - x1 \* d;

258 end

259 % учтём инфляцию

260 C1n = (alpha ^ (n - 1)) \* C1n;

261 % добавим затраты за шаг к общим затратам

262 C1 = C1 + C1n;

263

264 % вариант 2 - стационарная стратегия + переход на R2, r2 за тау+1 шаг до конца первого полупериода

265 % значение x фиктивного уровня запасов после удовлетворения потребительского спроса на шаге

266 x2 = x2 - zn;

267 % определим размер заказа Un, затраты на размещение заказа и его стоимость

268 if x2 < r

269 if n >= Nstop && n < Nswitch

270 % не подаём заказ, чтобы уровень запасов снизился ко второму полупериоду

271 un = R2 - x2;

272 x2 = x2 + un;

273 C2n = C2n + A + c \* un;

274 else

275 un = R - x2;

276 x2 = R;

277 C2n = C2n + A + c \* un;

278 end

279 end

280 % определим затраты на хранение и потери вследствие дефицита

281 if x2 > 0

282 C2n = C2n + x2 \* h;

283 else

284 C2n = C2n - x2 \* d;

285 end

286 % учтём инфляцию

287 C2n = (alpha ^ (n - 1)) \* C2n;

288 % добавим затраты за шаг к общим затратам

289 C2 = C2 + C2n;

290

291 % вариант 3 - динамическая стратегия

292 % значение x фиктивного уровня запасов после удовлетворения потребительского спроса на шаге

293 x3 = x3 - zn;

294 % определим размер заказа Un, затраты на размещение заказа и его стоимость

295 if x3 < nRr3(n,3)

296 un = nRr3(n,2) - x3;

297 x3 = x3 + un;

298 C3n = C3n + A + c \* un;

299 end

300 % определим затраты на хранение и потери вследствие дефицита

301 if x3 > 0

302 C3n = C3n + x3 \* h;

303 else

304 C3n = C3n - x3 \* d;

305 end

306 % учтём инфляцию

307 C3n = (alpha ^ (n - 1)) \* C3n;

308 % добавим затраты за шаг к общим затратам

309 C3 = C3 + C3n;

310

311 % вариант 4 - динамическая стратегия + переход на новые Rn и rn за тау+2 шага до конца первого полупериода

312 % значение x фиктивного уровня запасов после удовлетворения потребительского спроса на шаге

313 x4 = x4 - zn;

314 % определим размер заказа Un, затраты на размещение заказа и его стоимость

315 if x4 < nRr4(n,3)

316 un = nRr4(n,2) - x4;

317 x4 = x4 + un;

318 C4n = C4n + A + c \* un;

319 end

320 % определим затраты на хранение и потери вследствие дефицита

321 if x4 > 0

322 C4n = C4n + x4 \* h;

323 else

324 C4n = C4n - x4 \* d;

325 end

326 % учтём инфляцию

327 C4n = (alpha ^ (n - 1)) \* C4n;

328 % добавим затраты за шаг к общим затратам

329 C4 = C4 + C4n;

330

331 end

332

333 end

334

335 % определим среднее значение суммарных средних затрат для всех экспериментов

336 C1 = C1 / Nmc;

337 C2 = C2 / Nmc;

338 C3 = C3 / Nmc;

339 C4 = C4 / Nmc;

340

341 % сохраним значения

342 x\_C1\_C2\_C3\_C4(x,:) = [x\_init C1 C2 C3 C4];

343

344 end

345

346 % строим график суммарных средних затрат С(x)

347 x = x\_C1\_C2\_C3\_C4(:,1);

348 y1 = x\_C1\_C2\_C3\_C4(:,2);

349 y2 = x\_C1\_C2\_C3\_C4(:,3);

350 y3 = x\_C1\_C2\_C3\_C4(:,4);

351 y4 = x\_C1\_C2\_C3\_C4(:,5);

352 plot(x,y1,x,y2,x,y3,x,y4);

353 xlabel('x');

354 ylabel('C(x)');

355 legend({'Var 1','Var 2','Var 3','Var 4'},'Location','southwest');

356

357 % индексы

358 i\_min = round((-m - last\_step\_y\_min) / y\_step + 1);

359 i\_max = round((R1\*1.05 - last\_step\_y\_min) / y\_step + 1);

360

361 % расчитаем суммарные затраты

362 C1\_sum = sum(x\_C1\_C2\_C3\_C4(i\_min:i\_max,2));

363 C2\_sum = sum(x\_C1\_C2\_C3\_C4(i\_min:i\_max,3));

364 C3\_sum = sum(x\_C1\_C2\_C3\_C4(i\_min:i\_max,4));

365 C4\_sum = sum(x\_C1\_C2\_C3\_C4(i\_min:i\_max,5));

366

367 % расчитаем выигрыш от использования различных стратегий

368 gain\_1\_to\_2 = (C1\_sum - C2\_sum) / C1\_sum \* 100;

369 gain\_3\_to\_4 = (C3\_sum - C4\_sum) / C3\_sum \* 100;

370 gain\_1\_to\_3 = (C1\_sum - C3\_sum) / C1\_sum \* 100;

371 gain\_1\_to\_4 = (C1\_sum - C4\_sum) / C1\_sum \* 100;

372

373 % выводим значения в консоль

374 fprintf('Var 1 -> Var 2 = %.3f%%\n',gain\_1\_to\_2);

375 fprintf('Var 3 -> Var 4 = %.3f%%\n',gain\_3\_to\_4);

376 fprintf('Var 1 -> Var 3 = %.3f%%\n',gain\_1\_to\_3);

377 fprintf('Var 1 -> Var 4 = %.3f%%\n',gain\_1\_to\_4);

378 fprintf('\n');

379

380 % выберем значения для построение графика

381 graph\_step = 0.1;

382 if y\_step >= graph\_step

383 graph\_step = y\_step;

384 end

385 graph\_count = (x\_max - x\_min) / graph\_step + 1;

386 graph\_x\_C1\_C2\_C3\_C4 = NaN(graph\_count,5);

387 for g = 1:graph\_count

388 x = graph\_step / x\_step \* (g - 1) + 1;

389 graph\_x\_C1\_C2\_C3\_C4(g,:) = x\_C1\_C2\_C3\_C4(x,:);

390 end

391

392 % звуковой сигнал об окончании расчётов

393 end\_signal = audioplayer([sin(1:.6:400),sin(1:.7:400),sin(1:.4:400)],22050);

394 play(end\_signal);

395

396 % выводим время исполнения в консоль

397 time\_exec = floor(posixtime(datetime('now','TimeZone','local'))) - unix\_start;

398 time\_exec = [time\_exec, time\_exec/60, time\_exec/3600];

399 fprintf('%s -> %s.m ended -> %.0fs / %.2fm / %.2fh\n',datestr(now),mfilename,time\_exec(1),time\_exec(2),time\_exec(3));

400

401 % сохраняем файл с данными

402 fname = sprintf('3\_rc\_N%.f\_CDF%.f\_step%s.mat',N1,CDF,strrep(num2str(y\_step),'.',''));

403 save(fname);

404

405 %запускаем следующий расчёт

406 next\_script = strcat('rc\_',num2str(sscanf(mfilename(),'rc\_%d')+1));

407 if exist(next\_script,'file') == 2

408 run(next\_script);

409 end

410

411 % функция расчёта значений R, r и Cnx

412 function [nRr, xC] = calc\_Rr(alpha,A,c,h,d,N)

413

414 % обозначим глобальные и символьные переменные

415 global myu sigma lambda uniform\_a uniform\_b CDF y\_step last\_step\_y\_min last\_step\_y\_max z\_min z\_max z\_step z\_count fz uniform\_pdf;

416 syms y\_sym z\_sym lambda\_sym sigma\_sym myu\_sym pi\_sym;

417

418 % минимум y на первом шаге

419 first\_step\_y\_min = last\_step\_y\_min - z\_max \* (N - 1);

420 % максимум y на первом шаге

421 first\_step\_y\_max = last\_step\_y\_max - z\_min \* (N - 1);

422 % количество значений y на первом шаге

423 first\_step\_y\_count = round((first\_step\_y\_max - first\_step\_y\_min) / y\_step + 1);

424

425 % выбираем функцию распределения

426 if CDF == 1

427

428 % нормальное распределение

429

430 % предварительный расчёт интеграла при h

431 int\_h\_sym = str2sym('(y\_sym - z\_sym) / (sigma\_sym \* (2 \* pi\_sym)^0.5) \* exp(((z\_sym-myu\_sym)^2) / (-2 \* sigma\_sym^2))');

432 int\_h\_sym = subs(int\_h\_sym,sigma\_sym,sigma);

433 int\_h\_sym = subs(int\_h\_sym,myu\_sym,myu);

434 int\_h\_sym = subs(int\_h\_sym,pi\_sym,pi);

435

436 % интеграл при h (от 0 до y)

437 int\_h\_0\_y = int(int\_h\_sym,z\_sym,0,y\_sym);

438

439 % предварительный расчёт интеграла при d

440 int\_d\_sym = str2sym('(z\_sym - y\_sym) / (sigma\_sym \* (2 \* pi\_sym)^0.5) \* exp(((z\_sym-myu\_sym)^2) / (-2 \* sigma\_sym^2))');

441 int\_d\_sym = subs(int\_d\_sym,sigma\_sym,sigma);

442 int\_d\_sym = subs(int\_d\_sym,myu\_sym,myu);

443 int\_d\_sym = subs(int\_d\_sym,pi\_sym,pi);

444

445 % интеграл при d (от y до бесконечности)

446 int\_d\_y\_inf = int(int\_d\_sym,z\_sym,y\_sym,inf);

447

448 % интеграл при d (от 0 до бесконечности)

449 int\_d\_0\_inf = int(int\_d\_sym,z\_sym,0,inf);

450

451 elseif CDF == 2

452

453 % экспоненциальное распределение

454

455 % предварительный расчёт интеграла при h

456 int\_h\_sym = str2sym('(y\_sym - z\_sym) \* lambda\_sym \* exp(-1 \* lambda\_sym \* z\_sym)');

457 int\_h\_sym = subs(int\_h\_sym,lambda\_sym,lambda);

458

459 % интеграл при h (от 0 до y)

460 int\_h\_0\_y = int(int\_h\_sym,z\_sym,0,y\_sym);

461

462 % предварительный расчёт интеграла при d

463 int\_d\_sym = str2sym('(z\_sym - y\_sym) \* lambda\_sym \* exp(-1 \* lambda\_sym \* z\_sym)');

464 int\_d\_sym = subs(int\_d\_sym,lambda\_sym,lambda);

465

466 % интеграл при d (от y до бесконечности)

467 int\_d\_y\_inf = int(int\_d\_sym,z\_sym,y\_sym,inf);

468

469 % интеграл при d (от 0 до бесконечности)

470 int\_d\_0\_inf = int(int\_d\_sym,z\_sym,0,inf);

471

472 end

473

474 % подготовим таблицу рассчитанных значений интегралов при h и d

475 int\_h\_d\_values = NaN(2,first\_step\_y\_count);

476

477 % заполним таблицу g(y)

478 for y = 1:first\_step\_y\_count

479

480 % значение y

481 y\_value = first\_step\_y\_min + y\_step \* (y - 1);

482

483 if CDF == 1 || CDF == 2

484

485 % определяем пределы интегрирования интегралов h и d

486 if y\_value <= 0

487 h\_int = 0;

488 d\_int = subs(int\_d\_0\_inf,y\_sym,y\_value);

489 else

490 h\_int = subs(int\_h\_0\_y,y\_sym,y\_value);

491 d\_int = subs(int\_d\_y\_inf,y\_sym,y\_value);

492 end

493

494 elseif CDF == 3

495

496 % определяем пределы интегрирования интегралов h и d

497 h\_int\_min = 0;

498 d\_int\_max = min(first\_step\_y\_max,uniform\_b+1);

499 if y\_value <= 0

500 h\_int\_max = 0;

501 d\_int\_min = 0;

502 else

503 h\_int\_max = y\_value;

504 d\_int\_min = y\_value;

505 end

506 h\_int\_count = (h\_int\_max - h\_int\_min) / z\_step + 1;

507 d\_int\_count = (d\_int\_max - d\_int\_min) / z\_step + 1;

508

509 % начальное значение интеграла при h

510 h\_int = 0;

511

512 if h\_int\_count > 1

513

514 % подсчитаем значение интеграла при h как среднее арифметическое верхней и нижней сумм Дарбу

515 for z = 1:h\_int\_count

516

517 % текущее значение z

518 z\_value = h\_int\_min + z\_step \* (z - 1);

519

520 % предыдущее значение подынтегральной функции

521 if z >= 2

522 h\_prev\_val = h\_this\_val;

523 end

524

525 % определим значение функции плотности вероятности в текущей точке z

526 if z\_value >= uniform\_a && z\_value <= uniform\_b

527 uniform\_pdf\_val = uniform\_pdf;

528 else

529 uniform\_pdf\_val = 0;

530 end

531

532 % значение подынтегральной функции данного среза

533 h\_this\_val = (y\_value - z\_value) \* uniform\_pdf\_val;

534

535 if z >= 2 && (z\_value > uniform\_a && z\_value <= uniform\_b)

536

537 % знаки текущего и предыдущего значений

538 h\_this\_sign = sign(h\_this\_val);

539 h\_prev\_sign = sign(h\_prev\_val);

540

541 % находим значение площади среза подынтегральной функции h

542 if (h\_this\_sign == 1 && h\_prev\_sign == -1) || (h\_this\_sign == -1 && h\_prev\_sign == 1)

543 h\_this\_step\_area = (abs(h\_this\_val) + abs(h\_prev\_val)) \* z\_step / 4;

544 else

545 h\_this\_step\_area = (min(abs(h\_this\_val),abs(h\_prev\_val)) + abs(abs(h\_this\_val) - abs(h\_prev\_val)) / 2) \* z\_step;

546 end

547

548 % добавим площадь среза к площади всего подграфика

549 h\_int = h\_int + h\_this\_step\_area;

550

551 end

552

553 end

554

555 end

556

557 % начальное значение интеграла при d

558 d\_int = 0;

559

560 if d\_int\_count > 1

561

562 % подсчитаем значение интеграла при h как среднее арифметическое верхней и нижней сумм Дарбу

563 for z = 1:d\_int\_count

564

565 % текущее значение z

566 z\_value = d\_int\_min + z\_step \* (z - 1);

567

568 % предыдущее значение подынтегральной функции

569 if z >= 2

570 d\_prev\_val = d\_this\_val;

571 end

572

573 % определим значение функции плотности вероятности в текущей точке z

574 if z\_value >= uniform\_a && z\_value <= uniform\_b

575 uniform\_pdf\_val = uniform\_pdf;

576 else

577 uniform\_pdf\_val = 0;

578 end

579

580 % значение подынтегральной функции данного среза

581 d\_this\_val = (z\_value - y\_value) \* uniform\_pdf\_val;

582

583 if z >= 2 && (z\_value > uniform\_a && z\_value <= uniform\_b)

584

585 % знаки текущего и предыдущего значений

586 d\_this\_sign = sign(d\_this\_val);

587 d\_prev\_sign = sign(d\_prev\_val);

588

589 % находим значение площади среза подынтегральной функции h

590 if (d\_this\_sign == 1 && d\_prev\_sign == -1) || (d\_this\_sign == -1 && d\_prev\_sign == 1)

591 d\_this\_step\_area = (abs(d\_this\_val) + abs(d\_prev\_val)) \* z\_step / 4;

592 else

593 d\_this\_step\_area = (min(abs(d\_this\_val),abs(d\_prev\_val)) + abs(abs(d\_this\_val) - abs(d\_prev\_val)) / 2) \* z\_step;

594 end

595

596 % добавим площадь среза к площади всего подграфика

597 d\_int = d\_int + d\_this\_step\_area;

598

599 end

600

601 end

602

603 end

604

605 end

606

607 int\_h\_d\_values(1,y) = h\_int;

608 int\_h\_d\_values(2,y) = d\_int;

609

610 end

611

612 r\_val = NaN;

613 r\_val\_prev = NaN;

614 delta\_val = NaN;

615 delta\_val\_prev = NaN;

616

617 % задаём рамки таблицы Gn(y), где первый столбец - это значения n, а первая строка - это значения y

618 Gny = NaN(N+1,first\_step\_y\_count+1);

619

620 % задаём рамки таблицы Cn(x), где первый столбец - это значения n, а первая строка - это значения x

621 Cnx = NaN(N+1,first\_step\_y\_count+1);

622

623 % первый столбец (значения n)

624 for n = 1:N

625 Gny(n+1,1) = n;

626 Cnx(n+1,1) = n;

627 end

628

629 % первый ряд (значения y/x)

630 for y = 1:first\_step\_y\_count

631 Gny(1,y+1) = first\_step\_y\_min + y\_step \* (y - 1);

632 Cnx(1,y+1) = first\_step\_y\_min + y\_step \* (y - 1);

633 end

634

635 % значения функции Cn(x) на первом шаге

636 Cnx(2,2:first\_step\_y\_count+1) = 0;

637

638 % подготовим столбец R(n) и r(n)

639 Rn = NaN(N,1);

640 rn = NaN(N,1);

641

642 % начинаем выполнять шаги

643 for n = 2:N

644

645 % минимум y текущего шага

646 this\_step\_y\_min = last\_step\_y\_min - z\_max \* (N - n);

647

648 % максимум y текущего шага

649 this\_step\_y\_max = last\_step\_y\_max - z\_min \* (N - n);

650

651 % количество значений y на текущем шаге

652 this\_step\_y\_count = round((this\_step\_y\_max - this\_step\_y\_min) / y\_step + 1);

653

654 % индекс смещения y для текущего шага, с учётом первой колонки Gn(y) и Cn(x)

655 this\_step\_y\_offset = round((this\_step\_y\_min - first\_step\_y\_min) / y\_step + 1);

656

657 % цикл по y - заполняем строку таблицы Gn(y)

658 for y = 1:this\_step\_y\_count

659

660 % текущее значение y

661 y\_value = Gny(1,this\_step\_y\_offset+y);

662

663 % вычислим значение интеграла при альфа

664 integral\_a = 0;

665

666 if alpha ~= 0 && n ~= 2

667

668 % цикл по z

669 for z = 1:z\_count

670

671 % предыдущее значение подынтегральной функции

672 if z >= 2

673 prev\_val = this\_val;

674 end

675

676 % текущее значение z

677 z\_value = z\_min + z\_step \* (z - 1);

678

679 % индекс значения x из таблицы Cn(x)

680 x\_index = round((y\_value - z\_value - first\_step\_y\_min) / y\_step + 1 + 1);

681

682 % текущее значение подынтегральной функции

683 this\_val = Cnx(n,x\_index) \* fz(1,z);

684

685 % площадь вписанного прямоугольника и треугольника на данном шаге z

686 if z >= 2

687

688 % знаки текущего и предыдущего значений

689 this\_sign = sign(this\_val);

690 prev\_sign = sign(prev\_val);

691

692 % находим значение площади среза z подынтегральной функции

693 if (this\_sign == 1 && prev\_sign == -1) || (this\_sign == -1 && prev\_sign == 1)

694 this\_step\_area = (abs(this\_val) + abs(prev\_val)) \* z\_step / 4;

695 else

696 this\_step\_area = (min(abs(this\_val),abs(prev\_val)) + abs(abs(this\_val) - abs(prev\_val)) / 2) \* z\_step;

697 end

698

699 % добавляем площадь среза к площади всего подграфика

700 integral\_a = integral\_a + this\_step\_area;

701

702 end

703

704 end

705

706 end

707

708 % текущее значение функции Gn(y)

709 Gny(n+1,this\_step\_y\_offset+y) = c \* y\_value + h \* int\_h\_d\_values(1,this\_step\_y\_offset-1+y) + d \* int\_h\_d\_values(2,this\_step\_y\_offset-1+y) + alpha \* integral\_a;

710

711 end

712

713 % найдём минимальное значение Gn(y) на текущем шаге и его индекс в таблице Gn(y)

714 [Gny\_min,Gny\_min\_i] = min(Gny(n+1,2:first\_step\_y\_count+1));

715

716 % и соответствующий y = R

717 R = Gny(1,Gny\_min\_i+1);

718

719 % найдём значение одношаговых затрат g(R)

720 gR = h \* int\_h\_d\_values(1,Gny\_min\_i) + d \* int\_h\_d\_values(2,Gny\_min\_i);

721

722 % запишем значение в столбец R(n)

723 Rn(n,1) = R;

724

725 % заполним строку таблицы Cn(x)

726 for y = 1:this\_step\_y\_count

727

728 % текущее значение x

729 x\_value = Gny(1,this\_step\_y\_offset+y);

730

731 if x\_value < R

732 Cnx\_value\_1 = -c \* x\_value + A + Gny\_min;

733 Cnx\_value\_2 = -c \* x\_value + Gny(n+1,this\_step\_y\_offset+y);

734 Cnx\_value = min(Cnx\_value\_1,Cnx\_value\_2);

735 else

736 Cnx\_value = -c \* x\_value + Gny(n+1,this\_step\_y\_offset+y);

737 end

738

739 % текущее значение функции Cn(x)

740 Cnx(n+1,this\_step\_y\_offset+y) = Cnx\_value;

741

742 end

743

744 % случай A = 0

745 if A == 0

746

747 r = R;

748 rn(n,1) = r;

749

750 else

751

752 % вычислим значение r (движение от R налево)

753 for i = Gny\_min\_i+1:-1:this\_step\_y\_offset+1

754

755 % предыдущее значение r

756 r\_val\_prev = r\_val;

757

758 % текущее значение r

759 r\_val = Gny(1,i);

760

761 % предыдущее значение выражения G(r) - G(R) - A

762 delta\_val\_prev = delta\_val;

763

764 % значение выражения G(r) - G(R) - A

765 delta\_val = Gny(n+1,i) - Gny\_min - A;

766

767 % ищем смену знака выражения G(r) - G(R) - A

768 if i <= Gny\_min\_i

769 if (delta\_val >= 0 && delta\_val\_prev < 0) || (delta\_val <= 0 && delta\_val\_prev > 0)

770 r = (r\_val + r\_val\_prev) / 2;

771 rn(n,1) = r;

772 break;

773 end

774 end

775

776 end

777

778 end

779

780 % останавливаем цикл для альфа = 0

781 if alpha == 0

782 break;

783 end

784

785 end % конец цикла по N

786

787 for n = 1:N

788 n\_vals(n,1) = n;

789 end

790

791 % массив значений n, R и r

792 nRr = [n\_vals, Rn, rn];

793

794 % массив значений C(x)

795 this\_step\_y\_min = last\_step\_y\_min;

796 this\_step\_y\_offset = round((this\_step\_y\_min - first\_step\_y\_min) / y\_step + 1);

797 C = [];

798 C(1,:) = Cnx(1,this\_step\_y\_offset+1:first\_step\_y\_count+1);

799 C(2,:) = Cnx(N+1,this\_step\_y\_offset+1:first\_step\_y\_count+1);

800 C = C';

801 xC = C;

802

803 end

1. Исходный код модели переключения каналов обслуживания

1 % очищаем рабочую область

2 clear variables;

3

4 % очищаем консоль

5 clc;

6

7 % начинаем отсчёт времени

8 start\_time = clock;

9 tic;

10

11 % интенсивность входящего потока

12 Lambdaj = 299;

13

14 % интенсивность обслуживания на одном рабочем канале

15 Mu = 10;

16

17 % стоимость эксплуатации одного рабочего устройства обслуживания

18 c1 = 10;

19

20 % стоимость отключения одного рабочего устройства обслуживания

21 c2 = 9;

22

23 % фиксированная цена принятия решения о подключении новых рабочих устройств

24 A1 = 20;

25

26 % фиксированная цена принятия решения об отключении части рабочих устройств

27 A2 = 15;

28

29 % стоимость единицы времени пребывания одного требования в очереди на обслуживание

30 d = 1;

31

32 % доход, связанный с окончанием обслуживания одного требования

33 h = 10;

34

35 % текущее число рабочих каналов (до принятия управляющего решения)

36 m1 = 37;

37

38 % значение m критическое

39 m\_crit = floor(Lambdaj / Mu) + 1;

40

41 % задаём диапазон значений m

42 m\_range = 30;

43

44 % правая граница значений m

45 m\_end = m\_crit + m\_range;

46

47 % цикл по m

48 for m = m\_crit:m\_end

49

50 % pj

51 pj = Lambdaj / (m \* Mu);

52

53 % средняя длина очереди

54 sum = 0;

55 for i = 1:m-1

56 sum = sum + ((m \* pj) ^ i) / factorial(i);

57 end

58 Lqueue = ((sum + ((m \* pj) ^ m) / (factorial(m) \* (1 - pj))) ^ -1) \* (((m \* pj) ^ m) \* pj) / (factorial(m) \* (1 - pj) ^ 2);

59

60 % C переключения

61 if m > m1

62 C1\_switch = A1;

63 elseif m == m1

64 C1\_switch = 0;

65 elseif m < m1

66 C1\_switch = A2 + c2 \* (m1 - m);

67 end

68

69 % значение m

70 C1\_Bon\_Boff(m-m\_crit+1,1) = m;

71

72 % значение C1

73 C1\_Bon\_Boff(m-m\_crit+1,2) = c1 \* m + d \* Lqueue + C1\_switch;

74

75 % значение Bвкл.

76 C1\_Bon\_Boff(m-m\_crit+1,3) = c1 \* m + d \* Lqueue;

77

78 % значение Bвыкл.

79 C1\_Bon\_Boff(m-m\_crit+1,4) = (c1 - c2) \* m + d \* Lqueue + c2 \* m1;

80

81 end

82

83 % ищем минимум функции C1

84 [C1\_min,C1\_min\_i] = min(C1\_Bon\_Boff(:,2));

85

86 % значение m минимума функции C1

87 C1\_min\_m = C1\_Bon\_Boff(C1\_min\_i,1);

88

89 % ищем минимум функции Bвкл.

90 [Bon\_min,Bon\_min\_i] = min(C1\_Bon\_Boff(:,3));

91

92 % значение m минимума функции Bвкл.

93 Bon\_min\_m = C1\_Bon\_Boff(Bon\_min\_i,1);

94

95 % ищем минимум функции Bвыкл.

96 [Boff\_min,Boff\_min\_i] = min(C1\_Bon\_Boff(:,4));

97

98 % значение m минимума функции Bвыкл.

99 Boff\_min\_m = C1\_Bon\_Boff(Boff\_min\_i,1);

100

101 % выводим значения в консоль

102 fprintf('m\_crit = %d m1 = %d C1\_min\_m = %d Bon\_min\_m = %d Boff\_min\_m = %d\n',m\_crit,m1,C1\_min\_m,Bon\_min\_m,Boff\_min\_m);

103

104 % задаём координатную сетку для построения графиков

105 x = C1\_Bon\_Boff(:,1);

106 yC1 = C1\_Bon\_Boff(:,2);

107 yBon = C1\_Bon\_Boff(:,3);

108 yBoff = C1\_Bon\_Boff(:,4);

109

110 % строим графики функций C1, Bвкл. и Bвыкл.

111 plot(x,yC1,x,yBon,x,yBoff);

112

113 % подпишем ось x

114 xlabel('m');

115

116 % подпишем ось y

117 ylabel('C1, Bвкл., Bвыкл.');

118

119 % легенда

120 legend({'C1','Bвкл.','Bвыкл.'},'Location','southeast');

121

122 % время выполнения

123 execution\_time = toc;

1. Акт о внедрении результатов кандидатской диссертационной работы



1. А в России еще и перестройкой. И это при том, что пионером этих исследований в мире был В.А. Лотоцкий, который подытожил полученные им результаты в итоговой совместной монографии [14]. [↑](#footnote-ref-1)
2. Это означает [3], что при возникновении дефицита неудовлетворенный спрос регистрируется и удовлетворяется при поступлении очередной поставки. [↑](#footnote-ref-2)
3. В п. 2.7 мы рассматривается возможность отказа от этого предположения. [↑](#footnote-ref-3)
4. Под фиктивным уровнем запасом понимается сумма наличного запаса и запаса в пути (который в силу того, что θ = 0, также равен нулю) за вычетом уровня задолженного спроса [3]. Таким образом, фиктивный уровень запаса может быть и отрицательным. [↑](#footnote-ref-4)
5. Аналогичные рассуждения применимы и к интегралу . Еще одним аргументом в пользу введенного видоизменения записи интегралов служит то, что при компьютерном моделировании в качестве аппроксимирующей модели часто используют такие вероятностные распределения (гауссово и др.), плотность вероятности которых в качестве носителя захватывает и отрицательную полуось. [↑](#footnote-ref-5)
6. Еще раз отметим, что поскольку отсчет времени ведется от конца периода планирования, то стационарный режим связан с началом периода планирования. Иначе говоря, при *N* → ∞ стационарные управления используются сразу, с первых шагов периода планирования. [↑](#footnote-ref-6)
7. 30-180 шагов выбрано не случайно. Дело в том, что обычно в системах розничной торговли в роли шага принимается 1 день, а традиционным отрезком подведения итогов выбирается месяц – период планирования. [↑](#footnote-ref-7)
8. Полученные А.А. Первозванским в [6], результаты, см. также работу Н. Прабху [13], опирались на методы теории восстановления [158]. В области теории управления запасами эти результаты были развиты А.С. Манделем в работе [32]. [↑](#footnote-ref-8)
9. Предлагаемая модель применима только в тех случаев, когда не имеется серьезных правовых или административных помех при замене одних поставщиков на других. В системах розничной торговли это условие, как правило, выполнено. [↑](#footnote-ref-9)
10. В работе [158] модель управления запасами с возвратами была названа «фантазийной». [↑](#footnote-ref-10)
11. Это утверждение будет доказано в следующем разделе. [↑](#footnote-ref-11)