

Государственное учреждение высшего  
профессионального образования  
«Белорусско-Российский университет»

На правах рукописи



**ЯКИМОВ АНАТОЛИЙ ИВАНОВИЧ**

**ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ТЕХНОЛОГИИ  
ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ И ПРИНЯТИЯ  
РЕШЕНИЙ В ИНФОРМАЦИОННЫХ СИСТЕМАХ  
ПРОМЫШЛЕННЫХ ПРЕДПРИЯТИЙ**

Специальность 05.13.10 – Управление  
в социальных и экономических системах

Диссертация на соискание ученой степени  
доктора технических наук

*Научный консультант  
д-р техн. наук, проф. Аверченков В. И.*

Могилёв – 2017

## ОГЛАВЛЕНИЕ

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ .....	6
ВВЕДЕНИЕ.....	7
ГЛАВА 1. МЕТОДЫ И СРЕДСТВА МОДЕЛИРОВАНИЯ И ПРИНЯТИЯ РЕШЕНИЙ В УПРАВЛЕНИИ ПРОМЫШЛЕННЫМ ПРЕДПРИЯТИЕМ.....	17
1.1 Анализ современных концепций управления промышленным предприятием.....	17
1.2 Информационные технологии в корпоративной информационной системе промышленного предприятия .....	22
1.3 Математическая постановка системной задачи управления сложным иерархическим объектом.....	28
1.4 Аналитический обзор моделей и способов моделирования промышленного предприятия.....	33
1.5 Методы и средства имитационного моделирования сложных систем .....	41
1.6 Применение методов оптимизации в имитационном моделировании .....	46
1.7 Выводы по первой главе.....	53
ГЛАВА 2. РАЗРАБОТКА МЕТОДА ПОИСКА РАЦИОНАЛЬНЫХ РЕШЕНИЙ В ИНФОРМАЦИОННОЙ СИСТЕМЕ УПРАВЛЕНИЯ ПРОМЫШЛЕННЫМ ПРЕДПРИЯТИЕМ .....	55
2.1 Формальное описание многоуровневой иерархической системы промышленного предприятия.....	55
2.2 Идея метода поиска рациональных решений в информационной системе управления предприятием .....	59
2.3 Формализация процесса оптимизации системы управления предприятием на основе имитационной модели .....	63

2.4 Методика решения задачи рационального выбора алгоритма оптимизации систем управления предприятием .....	66
2.5 Построение целевой функции оценки эффективности взаимодействия компонентов предприятия .....	71
2.6 Выводы по второй главе.....	84
<b>ГЛАВА 3. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРОЕКТИРОВАНИЯ ИМИТАЦИОННЫХ МОДЕЛЕЙ ВНУТРИФИРМЕННОГО УПРАВЛЕНИЯ В ИНФОРМАЦИОННОЙ СИСТЕМЕ ПРЕДПРИЯТИЯ .....</b>	
<b>86</b>	
3.1 Разработка метода построения имитационной модели функционирования предприятия.....	86
3.2 Оптимизация процесса управления построением имитационной модели в корпоративной информационной системе .....	95
3.3 Вербальная модель структурных компонентов промышленного предприятия .....	105
3.4 Концептуальные модели функционирования компонентов промышленного предприятия.....	111
3.5 Формализация основных компонентов в базовой имитационной модели.....	123
3.6 Построение имитационной модели промышленного предприятия..	130
3.7 Выводы по третьей главе .....	144
<b>ГЛАВА 4. РАЗРАБОТКА ПРОГРАММНО-ТЕХНОЛОГИЧЕСКОГО КОМПЛЕКСА ИМИТАЦИИ СЛОЖНЫХ СИСТЕМ .....</b>	
<b>146</b>	
4.1 Состав и структура комплекса.....	146
4.2 Система имитационного моделирования BelSim Simulator Core.....	150
4.3 Технология алгоритмизации имитационной модели в BelSim Simulator Core .....	155
4.4 Технология разработки программных модулей имитационной модели в BelSim Simulator Core .....	159
4.5 Программная реализация базовой имитационной модели промышленного предприятия в BelSim IDE .....	162

4.6 Автоматизация планирования, проведения и обработки результатов экспериментов в BelSim Experimenter.....	165
4.7 Программное приложение BelSim Data Integrator для интеграции ПТКИ BelSim с корпоративной информационной системой .....	176
4.8 Организация распределенных вычислений в BelSim Experimenter..	181
4.9 Применение подсистемы обработки данных BelSim Data Processing при эксплуатации имитационной модели в системе управления .....	184
4.10 Выводы по четвертой главе .....	187
<b>ГЛАВА 5. ИССЛЕДОВАНИЕ ПРОГРАММНЫХ СРЕДСТВ ДЛЯ РЕАЛИЗАЦИИ МЕТОДА ПОИСКА РАЦИОНАЛЬНЫХ РЕШЕНИЙ НА ЭТАПЕ ЭКСПЛУАТАЦИИ ИМИТАЦИОННОЙ МОДЕЛИ ПРОМЫШЛЕННОГО ПРЕДПРИЯТИЯ.....</b>	<b>189</b>
5.1 Система поиска рациональных решений BelSim Optimizer .....	189
5.2 Кластеризация целевых функций.....	198
5.3 Сингулярный спектральный анализ последовательностей данных на этапе эксплуатации имитационной модели.....	208
5.4 Выводы по пятой главе.....	216
<b>ГЛАВА 6. АПРОБАЦИЯ МЕТОДА ПОИСКА РАЦИОНАЛЬНЫХ РЕШЕНИЙ В ИНФОРМАЦИОННОЙ СИСТЕМЕ ПРОМЫШЛЕННОГО ПРЕДПРИЯТИЯ С ПРИМЕНЕНИЕМ ПРОГРАММНО-ТЕХНОЛОГИЧЕСКОГО КОМПЛЕКСА BELSIM .....</b>	<b>218</b>
6.1 Состав задач, выбранных на апробацию .....	218
6.2 Решение задачи оптимизации производственного процесса .....	220
6.3 Апробация средств оптимизации и распределенных вычислений для решения задачи о грузоперевозках на предприятии.....	245
6.4 Поиск оптимальных решений при заключении контрактов в маркетинговой деятельности промышленного предприятия .....	248
6.5 Выбор типа координации процесса планирования производственной программы .....	253
6.6 Выводы по шестой главе .....	284

ЗАКЛЮЧЕНИЕ .....	287
Основные выводы и результаты диссертации .....	287
Список использованных источников .....	290
ПРИЛОЖЕНИЕ А. Аксиоматический принцип Парето для оценки эффективности принятия решений.....	315
ПРИЛОЖЕНИЕ Б. Алгоритмы компонентов базовой имитационной модели.....	320
ПРИЛОЖЕНИЕ В. Глобальные данные и классы базовой имитационной модели .....	332
ПРИЛОЖЕНИЕ Г. Структура файла данных модели и данных эксперимента.....	336
ПРИЛОЖЕНИЕ Д. Дополнение программы BelSim Experimenter для распределения вычислений .....	339
ПРИЛОЖЕНИЕ Е. Документы об апробации, внедрении и эффективности использования .....	341

## ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

АО – алгоритм оптимизации.

ГА – генетический алгоритм.

ДЭ – дифференциальная эволюция.

ИМ – имитационная модель.

ИО – имитация отжига.

ИС – информационная система.

ИЭ – имитационный эксперимент.

КИС – корпоративная информационная система.

ЛВС – локальная вычислительная сеть.

ПМ – программный модуль.

ПО – программное обеспечение.

ПП – промышленное предприятие.

ПрО – производственная операция.

ПрП – производственный процесс.

ПТКИ – программно-технологический комплекс имитации.

РЧ – рой частиц.

СМ – система моделирования.

СПЭС – сложная производственно-экономическая система.

ССА – сингулярный спектральный анализ.

СЭС – социально-экономическая система.

ТП – технологический процесс.

ТР – технологический режим.

УПМ – управляющая программа моделирования.

## ВВЕДЕНИЕ

**Актуальность темы исследования.** Внутрифирменное управление промышленным предприятием является одним из важнейших объектов исследования в теории управления социально-экономическими системами (СЭС). Повышение эффективности управления в СЭС достигается, в частности, использованием информационных систем, построенных на современных компьютерных технологиях. Корпоративная информационная система (КИС) промышленного предприятия, являясь средой информационной поддержки целенаправленной коллективной деятельности сотрудников предприятия, решает задачу эффективного управления материально-техническими, финансовыми, технологическими, трудовыми и интеллектуальными ресурсами предприятия для достижения заданных показателей производственно-экономической деятельности [28]. Она включает в себя различные программно-аппаратные платформы, универсальные и предметно-ориентированные программные приложения, интегрированные в единую информационно-однородную систему.

Существенными проблемами для повышения эффективности механизмов управления и качества принимаемых решений является обеспечение оперативности и достоверности информации в условиях структурных и параметрических изменений процессов функционирования предприятия и его окружающей среды.

Разный характер хозяйственной деятельности позволяет разделить фирмы на промышленные, торговые, транспортные, страховые, инжиниринговые, туристские и др. В последующем будет рассматриваться внутрифирменное управление промышленными фирмами (промышленными предприятиями), основным видом деятельности которых является производство продукции.

При решении задач внутрифирменного управления для каждого подразделения фирмы определяются показатели, позволяющие оценивать экономические результаты его хозяйственной деятельности. В основном это стоимостные или объемные показатели выполняемых работ и, соответственно, затраты на испол-

нение работ (на оплату труда сотрудников подразделения, материальные и энергетические затраты, накладные расходы).

Эффективность управленческих решений значительно повышается, если используется имитационное моделирование производственной деятельности, позволяющее, с одной стороны, оценить на модели качество принятого решения, а с другой, является средством обучения сотрудников [101].

Оптимизация процессов функционирования производственной фирмы (предприятия) осуществляется путем декомпозиции из этих процессов отдельных компонентов и связанных с ними функций структурных подразделений, позволяющих выполнить формальное описание и построить модели оптимизации, именуемые оптимизационными моделями внутрифирменного управления. Задачи математического моделирования и оптимизации процессов деятельности предприятий в условиях структурных и параметрических изменений производственной и внешней среды, дальнейшего совершенствования методов имитационного моделирования процессов функционирования промышленного предприятия для их системного представления с учетом принятия решений представляются весьма актуальными при внедрении корпоративных информационных систем и дальнейшем развитии информационных технологий.

Ввиду актуальности темы, исследования проводились в Республике Беларусь по Государственной программе фундаментальных исследований «Исследование основных математических структур и проблем математического моделирования» (шифр «Математические структуры») в качестве научного руководителя задания (№ ГР 20021541, 2002–2005 год исполнения) при выполнении темы «Разработка качественных и конструктивных методов синтеза оптимальных систем управления в условиях неопределенности и их приложение» (шифр «Математические структуры–16»); по гранту Ф08Р–231 РФФИ–БРФФИ совместного белорусско-российского проекта «Интеграция систем имитационного моделирования и информационных систем на основе онтологий и многоуровневых моделей предметных областей» (№ ГР 20082063, 2008–2009 год исполнения) в качестве научного руководителя; по государственной программе научных исследо-



ваний «Информатика и космос, научное обеспечение безопасности и защиты от чрезвычайных ситуаций» в рамках задания 1.3.02 подпрограммы 1 «Научные основы информационных технологий» в качестве научного руководителя темы «Разработка методов и программных средств интеллектуальной информационно-аналитической системы поддержки принятия решений, интегрированной в комплексную информационную систему промышленного предприятия» (№ГР20110379, 2011–2013 год исполнения). Кроме того, автор принимал участие в качестве научного руководителя раздела Государственной программы прикладных исследований (шифр «Инфотех») при выполнении задания «Разработка методологии, средств и технологии имитационного моделирования вероятностных технологических процессов дискретного производства» (шифр «Инфотех–44») на кафедре математических проблем управления УО «Гомельский государственный университет им. Ф. Скорины» (№ ГР 20061846, 2006–2007 год исполнения).

**Степень разработанности темы.** Разработке теоретических основ управления производственными организациями, как активными социально-экономическими системами, посвящены работы многих российских и зарубежных ученых. Значительный вклад внесли А. А. Ашимов, В. Н. Бурков, В. Н. Волкова, В. Л. Волкович, И. А. Горгидзе, М. В. Губко, А. А. Емельянов, Дж. Клир, О. И. Ларичев, Д. Мако, М. Месарович, В. С. Михалевич, С. П. Мишин, Д. А. Новиков, Ф. И. Перегудов, Г. Саймон, И. М. Смирнов, И. Такахара, Ф. П. Тарасенко, Н. П. Федоренко, Дж. Форрестер, Т. Е. Шохина, В. З. Ямпольский и др.

Разработке методов и моделей внутрифирменного управления на основе применения современных информационных технологий уделено достаточно внимания многими учеными. Большой вклад в разработку этих проблем внесли Р. Л. Акофф, В. Г. Алиев, М. Ю. Афанасьев, И. В. Афонин, М. И. Баканов, В. И. Воропаев, Д. И. Галенко, Е. С. Гламаздин, С. Д. Ильенкова, В. В. Ковалев, Д. А. Новиков, У. У. Оливер, Г. Ортнер, Э. А. Уткин, А. В. Цветков и др.

Проблеме исследования оптимизационных моделей внутрифирменного управления на основе имитационного моделирования, задачам оптимального

планирования и распределения ресурсов промышленного предприятия посвящены работы П. В. Авдулова, И. В. Бурковой, А. Г. Гранберга, А. Ю. Заложнева, В. С. Иозайтиса, А. А. Кугаенко, И. В. Максимея, А. А. Спирина, В. И. Терехина, Л. Л. Терехова, Я. М. Уринсона и др.

Однако, несмотря на большое число опубликованных работ, в указанной области все еще остаются нерешенные проблемы, касающиеся вопросов совершенствования механизмов принятия решений по повышению эффективности управления промышленными предприятиями на базе применения современных информационных технологий, прежде всего, вопросов системного имитационного моделирования процессов функционирования производственной фирмы.

**Объектом** диссертационного исследования является система управления промышленного предприятия с корпоративной информационной системой.

**Предметом** исследования являются оптимизационные модели распределения финансовых, материальных и трудовых ресурсов промышленного предприятия на основе имитационного моделирования с применением комплекса информационных технологий.

**Целью** диссертационной работы является разработка теоретических положений и новых подходов к технологии имитационного моделирования и принятию решений в информационных системах управления промышленного предприятия.

Для достижения поставленной цели исследования в диссертации были поставлены и решены следующие основные **задачи**:

1. Разработать метод поиска рациональных параметров процессов управления производственно-экономической деятельностью предприятия на основе имитационного моделирования (ИМ) с комплексным применением современных информационных технологий.

2. Предложить базовую имитационную модель типовых процессов управления для системного анализа производственно-экономической деятельности промышленного предприятия и технологию использования этой имитационной модели, обеспечивающую реализацию вышеуказанного метода.

3. Разработать метод решения многокритериальной задачи построения оптимизационных моделей системы управления промышленного предприятия с применением базовой имитационной модели в КИС.

4. Разработать программные средства для автоматизации метода поиска параметров рациональных процессов управления производственно-экономической деятельностью предприятия.

5. Создать специальный программно-технологический комплекс имитации сложных систем (ПТКИ), интегрированный в корпоративную информационную систему предприятия и позволяющий разрабатывать оптимизационные модели самими специалистами предприятия.

6. Предложить новые технологии использования программно-технологического комплекса для реализации методов построения имитационных моделей и поиска рациональных решений при управлении производственно-экономической деятельностью промышленного предприятия.

7. Провести апробацию разработанных методов, средств и технологий в ходе внедрения программно-технологических комплексов на промышленных предприятиях Республики Беларусь и Российской Федерации.

**Научная новизна** состоит в том, что впервые разработаны:

1. Теоретические положения технологии имитационного моделирования и принятия решений в информационных системах управления промышленного предприятия, основанные на представлении информационной системы в виде многоуровневой иерархической социально-экономической системы и введении понятия ресурсов  $f$ - и  $p$ -типа, решении многокритериальной оптимизационной задачи управления проектированием имитационных моделей внутрифирменного управления предприятием.

2. Метод построения имитационной модели функционирования промышленного предприятия, основанный на процессном способе моделирования, включающий новые подходы на основе информационных технологий КИС и координирующих элементов в технологии использования ИМ в структуре системы управления.

3. Метод многокритериального управления построением имитационных моделей процессов функционирования промышленного предприятия с использованием комплекса информационных технологий КИС путем применения ресурсов  $f$ - и  $p$ -типа, основанный на использовании принципа оптимальности Беллмана.

4. Метод поиска рациональных параметров процессов управления, основанный на имитации системы управления ресурсами производственной фирмы с помощью базовой имитационной модели, отличающийся использованием стохастических алгоритмов оптимизации и постулата совместимости координирующих решений.

5. Алгоритм количественной оценки координирующих действий в системе управления при решении многокритериальных задач на основе базовой имитационной модели функционирования предприятия с применением аксиоматического принципа Парето и постулата совместимости.

**Теоретическая значимость** работы состоит в разработке новых теоретических положений технологии имитационного моделирования и принятия решений в информационных системах управления промышленными предприятиями, основанных на представлении информационной системы в виде многоуровневой иерархической социально-экономической системы и введении понятий ресурсов  $f$ - и  $p$ -типа, решении многокритериальной оптимизационной задачи управления проектированием имитационных моделей внутрифирменного управления предприятием. Разработанные теоретические положения отличаются новым подходом к применению информационных технологий корпоративной информационной системы, использованием ее имитационного моделирования, как концептуального развития общего метода моделирования сложных систем на основе комплексного применения современных информационных технологий.

Значимость теоретических положений состоит в разработке нового метода поиска и принятия рациональных решений, основанного на моделировании системы управления ресурсами промышленного предприятия с помощью базовой имитационной модели, в которой созданы типовые компоненты, являющиеся

общей основой для комплексного анализа процессов функционирования реальных предприятий. Разработка новых теоретических положений в совокупности является существенным научным достижением в теории управления социально-экономическими системами и имеет важное социально-экономическое и хозяйственное значение для повышения эффективности управления предприятиями.

**Практическая значимость** полученных результатов связана с разработкой и внедрением в реальном производстве следующих научных результатов и программных комплексов:

1) базовой имитационной модели функционирования промышленного предприятия с КИС, которая ускоряет адаптацию программного обеспечения имитационной модели к условиям конкретных производственных организаций;

2) программно-технологического комплекса имитации сложных систем (ПТКИ) BelSim, который сокращает временные затраты на исследование имитационных моделей за счет интеграции в КИС и автоматизации получения актуальных входных данных;

3) программного средства поиска рациональных решений, реализованного в ПТКИ BelSim, которое расширяет область применения ПТКИ BelSim за счет рационального выбора параметров системы внутрифирменного управления на основе обеспечения разработки имитационных моделей самими специалистами промышленного предприятия;

4) разработанных проблемно-ориентированных систем поддержки принятия решений для промышленных предприятий: ОАО «Могилевхимволокно», ОАО «Моготекс», ОАО «Обувь» (Республика Беларусь) и ООО «СМИТ-Ярцево» (Российская Федерация).

**Методология и методы исследования.** Лично автором определена методология имитационного моделирования процессов управления промышленным предприятием с корпоративной информационной системой (КИС) и системного анализа результатов имитации. На основе анализа источников определены цели и задачи исследования. Сформулировано направление дальнейшего развития метода имитационного моделирования процессным способом применительно к

производственно-экономическим системам с КИС. В результате разработана базовая имитационная модель процессов производственно-экономической деятельности промышленного предприятия. Для ее реализации предложен программно-технологический комплекс имитации сложных систем со средствами интеграции в корпоративную информационную систему промышленного предприятия и ориентацией на распределенные вычисления.

**Положения и результаты, выносимые на защиту:**

1. Теоретические положения технологии имитационного моделирования и принятия решений в информационных системах управления промышленным предприятием, основанные на их представлении в виде многоуровневой иерархической социально-экономической системы и введении понятия ресурсов  $f$ - и  $p$ -типа (трудовых ресурсов и программных средств).

2. Метод построения имитационных моделей (ИМ), основанный на процессном способе моделирования системы управления ресурсами промышленного предприятия путем использования комплекса взаимосвязанных имитационных моделей с высоким уровнем детализации.

3. Метод решения задачи многокритериального управления построением имитационных моделей системы управления промышленного предприятия с использованием комплекса информационных технологий и ресурсов  $f$ - и  $p$ -типа (трудовых ресурсов и программных средств), основанный на использовании принципа оптимальности Беллмана.

4. Метод поиска рациональных решений по распределению финансовых, материальных и трудовых ресурсов промышленного предприятия, основанный на имитации системы управления ресурсами с помощью базовой имитационной модели, использовании постулата совместимости координирующих решений и специальной процедуры адаптации модели алгоритма случайного поиска оптимума целевой функции с применением кластерного анализа.

5. Алгоритм количественной оценки эффективности координирующих решений в системе управления промышленного предприятия, основанный на

использовании базовой имитационной модели, постулата совместимости координирующих решений и применении аксиоматического принципа Парето.

6. Программно-технологический комплекс имитации сложных систем (ПТКИ) BelSim для автоматизации задач проектного моделирования механизмов управления предприятиями, использующими корпоративную информационную систему, и программное средство реализации метода поиска рациональных решений BelSim Optimizer, реализованное в среде ПТКИ BelSim.

**Обоснованность и достоверность** разработанных научных положений, имитационных моделей и программных систем обеспечивается корректным применением математических методов и подтверждается результатами внедрения в производственных условиях с анализом реальных данных на заводе органического синтеза ОАО «Могилевхимволокно», в ОАО «Моготекс», в ОАО «Обувь» (Беларусь, Могилев), в ООО «СМИТ-Ярцево» (Россия, Смоленская область, Ярцево). Требуемая степень достоверности результатов подтверждена в ходе тестирования в учебном процессе ГУВПО «Белорусско-Российский университет» на кафедре автоматизированных систем управления (Беларусь, Могилев), Белорусского национального технического университета на кафедре управления научными исследованиями, проектированием, производством (Беларусь, Минск).

**Апробация результатов.** Основные результаты исследований докладывались, обсуждались и представлялись на следующих международных конференциях и выставках: IFIP Symposium on Large Scale Systems: Theory and Applications (Греция, Патры, 1998); Перспективные технологии и системы (Минск, 2002-2006); Международная специализированная выставка ТИБО (Минск, 2005-2007); IV Белорусско-Польский научно-практический семинар и выставка (Польша, Ольштын, 2004); Проблемы управления и приложения (Минск, 2005); Математичне та імітаційне моделювання (Украина, Киев, 2006-2012); Моделирование–2006 (Украина, Киев, 2006); Интеллектуальные системы и Интеллектуальные САПР (Дивноморское, 2006); Актуальные проблемы математики, механики, информатики (Пермь, 2006); Computer Data Analysis and Modeling: Com-

plex Stochastic Data and Systems (Минск, 2007); Системный анализ и информационные технологии (Украина, Киев, 2008-2011); Динамические системы: устойчивость, управление, оптимизация (DSSCO '08) (Минск, 2008); Информационные технологии в профессиональной деятельности и научной работе (Йошкар-Ола, 2008); Вычислительный интеллект (результаты, проблемы, перспективы) (Украина, Черкассы, 2011); Belarus-Korea science and technology seminar (Минск, 2011); Информационные технологии и системы (Минск, 2011); Creativity in intellectual technologies & data science (Волгоград, 2015); Инфраструктурное обеспечение бизнес-процессов в региональных социально-экономических системах (Кострома, 2016); Математические методы в технике и технологиях (Санкт-Петербург, 2016) и др.

**Опубликованность результатов.** Материалы диссертационной работы представлены в 112 научных публикациях, из которых: 2 монографии; 15 статей в научных журналах, рекомендованных ВАК Российской Федерации; 25 статей в изданиях, включенных в Перечень ВАК Республики Беларусь; 18 статей в других журналах и сборниках научных трудов; 20 статей в трудах конференций; 32 тезисов докладов семинаров, конференций, симпозиумов. Одна из научных публикаций включена в базу данных Scopus. Получены 4 свидетельства о регистрации компьютерных программ.

**Структура и объем диссертации.** Диссертационная работа состоит из введения, шести глав, заключения, библиографического списка из 225 наименований и 6 приложений. Полный объем диссертации составляет 350 страниц, включает 83 иллюстрации и 20 таблиц.



# **ГЛАВА 1. МЕТОДЫ И СРЕДСТВА МОДЕЛИРОВАНИЯ И ПРИНЯТИЯ РЕШЕНИЙ В УПРАВЛЕНИИ ПРОМЫШЛЕННЫМ ПРЕДПРИЯТИЕМ**

## **1.1 Анализ современных концепций управления промышленным предприятием**

Современная методология управления [196] рассматривает управление деятельностью организации, выделяя проектно-технологический тип деятельности и тип, основанный на знаниях. В методологии управления определена общая структура теории управления и ее компонентов, разработаны психологические, социологические, философские, этические и эстетические основы деятельности, что позволяет улучшить и оптимизировать деятельность организации.

Знание является стратегическим ресурсом организации, но информационные технологии (ИТ) позволяют улучшить обмен знаниями между членами организации. Проведенные исследования [193] показали эффективность влияния информационных технологий на процессы, основанные на знаниях, а именно на обмен знаниями и обобщение знаний в условиях неопределенности проектирования.

Информационные системы представляют собой сложное взаимодействие человека, социальных, информационных и технических компонентов, которые являются связующим звеном организационных процессов [28]. Известно, что информационная система (ИС) может играть важную роль в обеспечении стратегических преимуществ для организации. Однако известным является и тот факт, что эти преимущества не появляются автоматически: они требуют, чтобы ИС и организация развивались согласованно [188].

Информационная система (ИС) представляет собой инфраструктуру предприятия, используемую при управлении всеми информационными потоками, которая представлена информационной моделью, являющейся набором правил и описанием последовательности действий ИС (формы используе-

мых документов, структура нормативных справочников и данных и т. д.); порядок развития информационной модели и правила ее совершенствования; программное обеспечение в соответствии с требованиями информационной модели; регламент внесения изменений в источники данных, структуру программного обеспечения и описание его функциональных модулей; соответствующее техническое обеспечение, отвечающее требованиям по использованию программного обеспечения (компьютерные системы на рабочих местах и их программное обеспечение, системы управления базами данных, периферийные устройства, средства телекоммуникаций); правила сопровождения программного обеспечения и пользовательские инструкции, регламент сертификации пользователей; трудовые ресурсы для формирования и развития информационной модели, сопровождения программного обеспечения, обслуживания аппаратно-технической базы [28].

Развитие информационных систем можно рассматривать как деятельность, создающую новое знание [178]. Таким образом, развитие информационных систем может быть вполне самостоятельным методом исследования. В этом случае не только создается новое знание о самом процессе развития, но и более глубокое понимание по организационной проблеме, для решения которой предназначена информационная система.

Для эффективного развития ИС в теории управления социально-экономическими системами при разработке, например, имитационных моделей и их программной поддержки исследованы математические модели организации и алгоритмов функционирования команд – коллективов, создаваемых для достижения цели автономно и согласованно при наименьших затратах при управлении ими [65]. При этом выделены следующие характеристики команды: единство цели, совместная деятельность, непротиворечивость интересов, автономность деятельности, коллективная и взаимная ответственность за результаты совместной деятельности, оптимальное распределение функций и объемов работ.

При управлении ресурсами в информационной системе производственной фирмы используют метод, представляющий организацию моделью финансовых

ресурсов, материальных запасов, кадровых ресурсов. Основной целью управления при таком методе является своевременное обеспечение требуемыми ресурсами и организация системы контроля над их перемещением. Метод управления достаточно хорошо формализуется моделями, ставшими широко распространенными: модель бухгалтерского учета (GAAP), концепция планирования материальных ресурсов (MRP I) производственного предприятия, концепция планирования материальных и финансовых ресурсов предприятия (ERP) [66].

В конце 60-х годов для крупных производственных фирм, оснащенных электронно-вычислительными машинами, была разработана концепция управления *MRP (Material Requirements Planning)* производственными процессами, основанная на единой модели данных в масштабе всей организации. Концепция управления *MRP* решает задачу автоматизированного планирования потребности сырья и материалов для обеспечения производства. Основное достижение *MRP*-систем – получение значительного экономического эффекта за счет минимизации издержек, обусловленных складскими запасами. *MRP* позволяет планировать потребности в материальных ресурсах по замкнутому производственному циклу, составлять производственную программу и осуществлять ее контроль в производственном цехе. В процессе такого планирования формируются рекомендации о пополнении материальных ресурсов для своевременного выполнения заказов. В дальнейшем, поскольку процесс планируется по времени, выдаются рекомендации по перепланированию заказов, находящихся в производстве в случае, если даты фактического выполнения заказа и даты в соответствии с контрактом не совпадают. Управление по *MRP* на определенный период начинается с указания изделий, заданных в плане производства, затем определяются данные по всем материальным ресурсам для изготовления этих изделий.

Дальнейшим развитием является *MRP II (Manufacturing resource planning)* – планирование уже всех ресурсов производственного предприятия с использованием данных от поставщиков и потребителей; выполнение прогнозирования потребностей в материалах, планирования складских запасов и контроля за распределением ресурсов в производстве.

*MRP II* является развитием концепции *MRP* с планированием по замкнутому производственному циклу. Производители, внедрившие систему *MRP* или *MRP II*, имеют минимальные издержки складских запасов, короткие сроки выхода на рынок новых продуктов, повышение прибыльности [26].

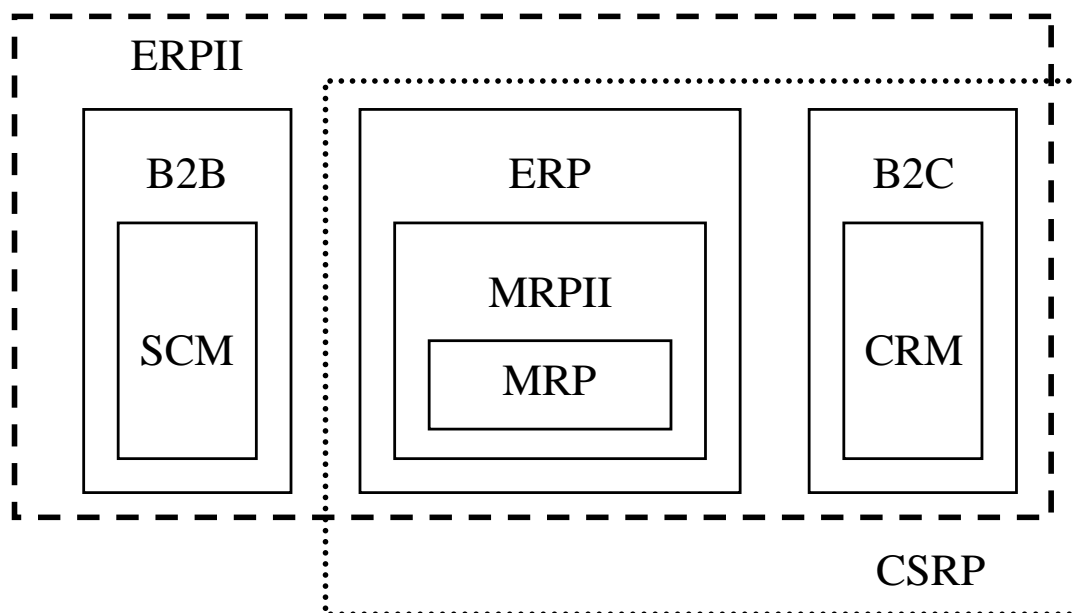
*ERP (Enterprise Resource Planning)* – планирование ресурсов в целом по предприятию. Если для *MRP* и *MRP II* имеется строгое формализованное описание, то класс *ERP* представлен только на концептуальном уровне. При планировании ресурсов в *ERP* добавлены планирование для распределения изготовленных изделий и финансовое планирование. В сравнении с *MRP II*, ориентированной на производство, концепция *ERP* ориентирована на бизнес. Например, бизнес-процесс отсрочки платежа потребителем готовой продукции принадлежит *ERP*, но отсутствует в *MRP II*. Так же инструментарий анализа данных *OLAP* [13] принадлежит *ERP*, но отсутствует в *MRP/MRP II*.

Информационная система *ERP* ориентирована на бухгалтерский учет для планирования использования ресурсов по предприятию в целом, чтобы в кратчайшие сроки принимать, производить, отгружать и вести учет контрактов клиентов. Технические требования к системе *ERP* существенно отличаются от *MRP II* применением графического интерфейса пользователя, наличием базы данных реляционного типа, использованием объектно-ориентированных языков и интегрированных систем разработки программного обеспечения, архитектурой вычислительных средств и мобильностью открытой системы. Применение *ERP*-систем содействует уменьшению ошибок планирования и числа выполняемых операций, улучшает показатели качества прогноза и планирования, что в результате приводит к сокращению издержек и улучшению процесса производства.

*CRM (Customer Relations Management)* и *SCM (Supply Chain Management)* – управление бизнес-процессами, соответственно, с заказчиками продукции и поставщиками ресурсов. *CRM* – концепция управления бизнес-процессами продажи произведенной продукции и взаимоотношения с клиентами. Такие задачи решаются благодаря использованию браузера с поддержкой Java, система ориентирована на доступ пользователя к приложениям с использованием сете-

вых ресурсов предприятия и/или Интернет. Место *CRM* и *SCM* в концепции *ERP* II показано на рисунке 1.1 [109].

*WCM* (*World Class Manufacturing*) – идея глобального производства: предприятия, даже действующие строго на местном рынке, фактически конкурируют с компаниями по всему миру.



**Рисунок 1.1 – Схема основных концепций управления предприятием**

*MES* (*Manufacturing Execution Systems*) – исполнительные производственные системы: поддерживают интеграцию систем на заводе, концентрируясь на оперативной деятельности – включает центральную базу данных, систему качества, систему управления документооборотом, систему планирования.

*JIT* (*Just-In-Time*) – точно в срок, является попыткой определения и устранения потерь, увеличивающих цену без расширения функциональности: уменьшились складские запасы, изменилась планировка заводов, начата автоматизация процессов производства [11].

*B2C* (*Business to Customer*) и *B2B* (*Business to Business*) – группы программных продуктов для поддержки бизнес-процессов между предприятием и покупателями (*B2C*) и между предприятиями (*B2B*) [11].

*CSRP* (*Customer Synchronized Resource Planning*) [34] – концепция, ориентированная на повышение эффективности взаимодействия предприятия с его

клиентами. При использовании *CSRP*-системы обеспечивается проектирование будущего продукта с учетом особых требований клиента, обеспечивается гарантийное и поддерживается сервисное обслуживание [61].

В рассмотренных концепциях, направленных на повышение эффективности управления предприятиями важное место отводится моделированию. Например, В. Н. Бурков, Д. А. Новиков [23] выполнили обзор исследований по теории активных систем, в частности, сделан анализ механизмов распределения ресурса, приводится общая модель управления организационными системами и технология решения задач управления, в частности, механизмы планирования, показана целесообразность применения математических моделей в системе управления для эффективного функционирования предприятий [67]. Однако использование имитационного моделирования [35, 45, 135, 223] сдерживается трудностями построения модели сложной системы и отсутствием средств автоматизации построения таких моделей [110].

В работе С. А. Сенченко [88] отмечается необходимость информационного сопровождения управления. Представлен алгоритм принятия управленческих решений, построенный на основе работы Е. В. Попова, в котором применяется имитационное моделирование.

Однако практические рекомендации по применению имитационного моделирования в корпоративной информационной системе (КИС) предприятия отсутствуют, не развита целостная концепция управления экономической системой предприятия, основанная на эффективном формировании и использовании информационных ресурсов в управлении инновационными процессами [143].

## **1.2 Информационные технологии в корпоративной информационной системе промышленного предприятия**

Рыночные отношения, развиваемые в экономике, и непрерывный научно-технический прогресс позволили активно применять в социально-экономической деятельности последние достижения в области информатизации. Информатизация в системах управления социально-экономическими процессами

ми направлена на повышение качества управления, производительности управленческого труда и, как следствие, повышение эффективности работы организации и ее конкурентоспособности. Информационные технологии дают возможность рационализировать управленческую функцию при использовании новых средств сбора, хранения, передачи, преобразования и использования информации. Информационные системы и организации взаимно влияют друг на друга. Информационная система создается в организации, чтобы обеспечить необходимой информацией все ее уровни и группы, в то же время сама организация должна развиваться благодаря информационным системам с целью получения эффекта от использования новых информационных технологий [50, 81].

Проведен обзор и выполнен анализ современных информационных технологий, применяемых в корпоративных информационных системах [106].

**Информационные технологии для моделирования бизнес-процессов.** Составление содержательного описания, построение концептуальной и формальной моделей в имитационном моделировании могут быть выполнены с использованием стандартов моделирования бизнес-процессов, например, на основе IDEF, RUP (Rational Software), Catalysis (Computer Associates). Поддержка стандартов обеспечивается CASE-средствами: ErWin, S-Designer, Power Designer, BPWin, Design/IDEF, IDEF0/EMTool, Rational Rose и т. д. На промышленных предприятиях применяются стандарты группы IDEF для формализации системы управления качеством выпускаемой продукции при проверке на соответствие международным стандартам ISO9000, ISO9001 [82, 183].

**Информационные технологии для моделирования программных систем.** Поведение проектируемой или исследуемой системы обычно представляется не только процессом переходов между состояниями и их изменениями, но и детализацией особенностей алгоритмов и логики выполняемых системой операций. Такая детализация обеспечивается использованием блок-схем или структурных схем алгоритмов, в которых представлены логические последовательности определенных действий и/или элементарных операций, приводящих к получению желаемого результата.

UML (Unified Modeling Language) представляет собой язык моделирования, выполняющий функции графической интерпретации объектов моделирования при проектировании программного обеспечения [53]. UML обеспечивает поддержку взаимодействия между заказчиком и исполнителями, а также между самими исполнителями при обсуждении и разработке проекта программной системы. Практическое использование UML основывается на системных принципах моделирования сложных объектов и особенностях объектно-ориентированного подхода к проектированию.

Программная реализация конструкций языка UML поддерживается следующими CASE-средствами: Rational Software Architect (<https://www.ibm.com>); Micro Focus Together (<https://www.microfocus.com>); Gentleware Poseidon (<http://www.gentleware.com>); Microsoft Visio (<https://www.microsoft.com>); Enterprise Architect (<http://www.sparxsystems.com.au>) и др.

Моделирование процессов выполнения операций в UML реализуется диаграммами деятельности, которые позволяют глубже понять процессы реализации алгоритмов операций классов и потоков управления в моделируемой системе, обладают преимуществом в представлении параллельных процессов и их синхронизации по сравнению с традиционными блок-схемами алгоритмов [172, 202].

**Информационные технологии доступа к данным информационной системы.** Доступ к любому типу информации предприятия реализуется универсальным механизмом доступа к данным (УМДД), обеспечивающий эффективный доступ к различным источникам информации с помощью единой модели. Данные на предприятии могут храниться на мэйнфреймах, в электронной почте и файловой системе в виде текстового, графического, географического представления и др. Подобное разнообразие источников и вида представления данных характерно для многих современных приложений.

В практике использования систем управления базами данных (СУБД) широко распространены следующие УМДД: Open Database Connectivity (ODBC) для доступа к реляционным базам по стандарту CLI (Call Level Interface)



ANSI/ISO; OLE DB/ActiveX Data Objects (ADO), основанный на библиотеках OLE DB, предоставляющих интерфейс нижнего уровня для доступа к данным с помощью COM-интерфейсов; Java Database Connectivity (JDBC), использующий драйверы на языке Java и работающие на уровне API (Application Programming Interface), которые поддерживаются СУБД Oracle, Sybase, Informix, DB2 и т. п.; ADO.NET, обладающий функцией поддержки распределенных приложений [94].

**Программные средства статистической обработки данных.** Практически в каждом научном исследовании присутствует этап статистического анализа данных, который с появлением и совершенствованием современных программ обработки данных стал доступен исследователю без специальной математической подготовки. Достаточно знать основные статистические понятия и, самое главное, уметь правильно выбрать метод анализа.

Одним из наиболее доступных и широко используемых средств обработки данных остается MS Excel – электронная таблица, в которой некоторые статистические функции представлены встроенными формулами, разработан макрос-дополнение XLSTAT-Pro (<http://www.xlstat.com/>), в котором представлены более 50 статистических функций. STADIA (<http://www.protein.bio.msu.ru/~akula/index.htm>) – программа российских разработчиков с включением необходимых статистических функций. SPSS (Statistical Package for Social Science) – наиболее часто используемый статистический пакет обработки данных (<http://www.spss.com/>). STATA (<http://www.stata.com/>) – профессиональный статистический программный пакет. SYSTAT (<http://systat.com/>) – статистическая система для персональных компьютеров, обладает интуитивно понятным интерфейсом. NCSS (<http://www.ncss.com/>) – программа, ориентированная на пользователей, не являющихся специалистами в области статистической обработки данных. STATISTICA (<http://www.statsoft.com/>) – многофункциональный статистический пакет, содержит более 250 функций, сгруппированных в статистические модули. MINITAB 14 (<http://www.minitab.com/>) – программный пакет с хорошим интерфейсом пользователя.

Для извлечения дополнительной информации при статистической обработке данных при использовании имитационной модели могут применяться и современные подходы, например, анализ сингулярного спектра последовательностей числовых данных [150].

**Информационные технологии обмена данными между программными системами.** Формат обмена данными XML (Exchange Message Language) позволяет создавать стандартные документы с текстом и ссылками на графические объекты и внешние ресурсы; структурированные документы в форме HTML; компоненты Java или ActiveX; различные записи баз данных на web-странице и др.

XML – язык метауровня, подмножество известного языка SGML, а не конкретный язык, подобный HTML [4]. Возможность создания собственных тегов делает XML универсальным, имеется опыт интеграции XML-среды с объектными средами и имитационным моделированием [171, 210].

Кроме XML наиболее используемыми форматами обмена данными являются JSON, YAML. JSON (Java Script Object Notation) – компактный вариант формата XML обмена данными, его конструкции проще обрабатывать с помощью Java Script. В основном JSON применяется для разработки web-приложений. YAML – формат, предназначенный для сериализации данных, ориентирован на удобство ввода-вывода основных структур данных языков программирования [44].

**Объектно-ориентированное программирование (ООП).** ООП – набор концепций и идей для понимания задач при разработке программного приложения и определения путей их эффективного решения. Основные принципы ООП: инкапсуляция, наследование и полиморфизм.

Наиболее эффективными и часто используемыми являются следующие языки объектно-ориентированного программирования: C#, C++, Java, Delphi, Eiffel, Simula, VB.NET, Visual DataFlex, Objective-C, Object Pascal, Perl, Php, Scala, ActionScript 3.0, PowerBuilder, Python, JavaScript, JScript.NET, Ada, Xbase++, Ruby, Smalltalk и др. Нет самого лучшего языка программирования, т. к. каждый

из языков ООП поддерживает желаемый стиль программирования в своей области приложений [218].

**Интегрированные среды разработки программных систем.** Интегрированная среда разработки программного обеспечения (англ. IDE, Integrated development environment) – комплекс программных средств, предназначенных для разработки программного обеспечения (ПО), включающий в себя текстовый редактор программного кода, интерпретатор и/или компилятор, средства автоматизации сборки программных модулей и отладчик. Существуют среды разработки для нескольких языков, например Eclipse (особенность – абсолютная нейтральность относительно платформы и языка программирования, известны реализации Java, C/C++, Cobol, Python, Eiffel, PHP, Ruby и C# [57]) или Microsoft Visual Studio; Geany (с поддержкой синтаксиса основных языков: C/C++, Java, Perl, PHP, XML [174]); NetBeans (**бесплатная** интегрированная среда разработки приложений на языке программирования Java, Ruby и ряде других [63]). Многие среды разработки ориентированы на определённый язык программирования, например, Visual Basic. Примеры наиболее известных сред разработки – Borland C++, Sun Studio, Turbo Pascal, GNU toolchain, Dev-C++, DrPython, Borland Delphi, KDevelop, QDevelop, Lazarus.

**Информационные технологии распределения вычислений.** Для реализации технологий параллельных вычислений известны следующие схемы: (N)UMA, DSM, PVM и MPI. В схеме (N)UMA используется разделяемый доступ к памяти. DSM существует в программном и аппаратном виде с организацией абстрактного слоя для физически распределённой памяти. PVM (Parallel Virtual Machine) функционирует как пользовательская программа и в ядро системы вносить изменения не требуется. MPI – это открытая спецификация библиотеки передачи сообщений, свободными реализациями которой являются MPICH, LAM.

MOSIX, openMosix, Kerrighed, OpenSSI – полнофункциональные кластерные среды, встроенные в ядро, автоматически распределяющие задачи между однородными узлами. OpenSSI, OpenMosix и Kerrighed создают среду единой операционной системы между узлами [203]. Windows Compute Cluster Server

(CCS) был спроектирован для приложений, ориентированных на кластерные вычисления [106].

### 1.3 Математическая постановка системной задачи управления сложным иерархическим объектом

На развитом промышленном предприятии руководитель принимает решение по результатам анализа различной информации от других подразделений. При этом невозможно принять эффективное решение ввиду сложности и многообразия вопросов, подлежащих анализу. Декомпозиция функциональных задач приводит к созданию структуры управления, включающей различные отделы: производственный, отдел сбыта, финансовый, отдел кадров и пр. При этом отделы имеют разные *цели функционирования*, во многом *взаимно противоположные* [18]. Например, *производственный отдел* желает, чтобы продукция была однообразной (малономенклатурной) и, если даже нет сбыта, продолжался выпуск продукции. Цель – максимальный выпуск продукции с узкой номенклатурой, чтобы не перенастраивать станки для снижения затрат при переходе на новый ассортимент. *Отдел сбыта* требует широкого ассортимента продукции, чтобы легче ее реализовывать, чтобы были в запасе товары, даже редко пользующиеся спросом, т. к. они все равно могут понадобиться. Поэтому этот отдел не возражает против запасов, если даже нет производства. *Финансовый отдел*, однако, возражает против запасов, т. к. это связанные деньги, а его задача – минимизировать эти связанные деньги (деньги в запасах), что означает, в свою очередь, минимизацию запасов. Финансовый отдел требует производства продукции, даже если нет в настоящий момент продаж товара. *Отдел кадров* против сокращения производства, если отсутствуют продажи продукции, т. к. это связано с увольнением работников, а это всегда социально неприятная процедура.

Для производства и реализации продукции предприятие заключает множество контрактов. От эффективности этих контрактов напрямую зависят результаты деятельности предприятия. Это особенно актуально в условиях финансово-экономического кризиса, когда большинство промышленных предприятий

балансируют на грани выживания. При отсутствии резервов финансовых средств одно единственное неверное решение, один заключенный невыгодный контракт могут привести к банкротству. Поэтому требуется тщательная проработка и оценка последствий принимаемых решений [102].

Промышленные предприятия (ПП) имеют явно различимую иерархическую структуру распределения ресурсов. Поэтому предложена содержательная (вербальная) постановка системной задачи рационального управления функционированием предприятия, как сложным иерархическим объектом. *Дано:* для сложной иерархической системы определены структуры всех уровней, требования к основным характеристикам и показателям управления процессами функционирования системы, которые в заданных условиях отличаются неполнотой и нечеткостью. *Требуется:* определить структуру системы управления, разработать проектные решения всех иерархических уровней при условии достижения необходимого качества управления системой в прогнозируемых штатных и нештатных ситуациях.

*Математическая постановка задачи.* В общем случае при исследовании ПП в целом возникает потребность в определении его целей на разных уровнях иерархии. При этом для одних целей оптимальные решения соответствуют минимальному значению определенного критерия, а для других – максимальному. Однако заменой переменных эти задачи легко привести к единому типу критериев и одному типу задач оптимизации. Их можно рассматривать как задачу многокритериальной оптимизации:

$$f_1(\bar{x}) \rightarrow \max; f_2(\bar{x}) \rightarrow \max; \dots; f_m(\bar{x}) \rightarrow \max, \quad (1.1)$$

где  $\bar{x}$  – вектор принимаемого решения;

$D$  – область допустимых решений.

Очевидно, что наилучшим решением задачи раскрытия неопределенности будет такое значение  $\bar{x}$ , для которого условия (1.1) выполняются одновременно для всех целевых функций. Однако на практике это не выполняется, т. к. функции  $f_k(\bar{x})$ ,  $k=1, 2, \dots, m$  разные по природе. Поэтому экстремума функция до-

стигает при своем значении  $\bar{x}$  и зачастую невозможно найти такое значение  $\bar{x}^0$ , для которого условия (1.1) выполняются одновременно для всех целевых функций. Поэтому задача сводится к нахождению такого значения  $\bar{x}^0$ , для которого обеспечивается *рациональный* компромисс заданных целей [40].

Для сложной многоуровневой системы структурные отношения  $f_\ell$  и обратные отношения  $f_\ell^{-1}$  функциональных элементов (ФЭ) между разными иерархическими уровнями, а также структурные отношения  $f_{\ell p}$  и обратные отношения  $f_{\ell p}^{-1}$  внутри уровней определены в виде:

$$f_\ell : V_L \rightarrow \bigcup_{\ell=1}^L V_\ell; \quad f_{\ell p} : V_\ell \rightarrow \bigcup_{p=1}^{P_0} V_{\ell p}; \quad (1.2)$$

$$f_{\ell p}^{-1} : \bigcup_{p=1}^{P_0} V_{\ell p} \rightarrow V_\ell; \quad f_\ell^{-1} : \bigcup_{\ell=1}^L V_\ell \rightarrow V_L. \quad (1.3)$$

где  $V_L$  – множество ФЭ объекта в целом ( $L$  - количество уровней);

$V_\ell$  – множество ФЭ  $\ell$ -го иерархического уровня;

$V_{\ell p}$  –  $p$ -й функциональный элемент  $\ell$ -го иерархического уровня ( $P_0$  - количество ФЭ на  $\ell$ -м уровне).

Функциональная взаимосвязь показателей качества функционирования ПП в целом с параметрами ФЭ определена в виде функции с монотонным включением переменных:

$$\bar{K}_L = \tilde{F}_L(\bar{X}_L, \tilde{F}_{L-1}(\bar{X}_{L-1}, \tilde{F}_{L-2}(\dots \tilde{F}_\ell(\bar{X}_\ell, \tilde{F}_{\ell-1}(\dots, \tilde{F}_1(\bar{X}_1))\dots)))) \quad (1.4)$$

где  $\bar{K}_L$  – вектор показателей качества ПП;

$\bar{X}_L$  – вектор параметров функциональных элементов высшего уровня иерархической структуры системы (уровень ПП в целом);

$\bar{X}_\ell$  – вектор параметров функциональных элементов  $\ell$ -го иерархического уровня,  $\ell = 1, \dots, L$ ;

$\bar{X}_1$  – вектор параметров функциональных элементов нижнего иерархического уровня (например, уровень элементов технологического процесса);

$\tilde{F}_L, \dots, \tilde{F}_\ell, \dots, \tilde{F}_1$  – функции взаимосвязи показателей функциональных элементов иерархический уровней; эти функции определены неточно, нечетко, неполно.

*Требуется:* построить такую последовательность преобразований, которая определит структуру системы управления и параметры ее функциональных элементов с учетом достижения необходимого уровня работы ПП в некоторый момент времени  $t \in [0, T]$  в прогнозируемых ситуациях риска  $s_k \in S_0, k = 1, \dots, N$ :

$$\Phi_\ell : G_L \rightarrow \bigcup_{\ell=1}^{L_w} G_\ell; \quad \Phi_{\ell p} : G_\ell \rightarrow \bigcup_{p=1}^{P_w} G_{\ell p}; \quad (1.5)$$

$$\Phi_{\ell p}^{-1} : \bigcup_{p=1}^{P_w} G_{\ell p} \rightarrow G_\ell; \quad \Phi_\ell^{-1} : \bigcup_{\ell=1}^{L_w} G_\ell \rightarrow G_L, \quad (1.6)$$

где  $G_L$  – множество преобразований для ПП в целом;

$G_\ell$  – множество преобразований  $\ell$ -го иерархического уровня;

$G_{\ell p}$  –  $p$ -е преобразование  $\ell$ -го иерархического уровня.

Общие требования к ПП должны быть рационально трансформированы в требования к ФЭ каждого иерархического уровня системы. Для этого необходимо иметь описание взаимосвязи ФЭ в форме (1.2), которое также является основой для реализации процедуры декомпозиции. Необходимость взаимосвязи ФЭ в форме (1.3) вытекает из технологии проектирования (построения) реальных сложных иерархических систем, начиная с разработки ФЭ нижнего уровня. Также формула (1.3) является основой для реализации процедуры агрегирования в системном анализе. Для системы управления соотношения (1.5) аналогичны формуле (1.2), а (1.6) – аналогичны формуле (1.3) [40].

Принципиальная особенность этой системной задачи в том, что она имеет одновременно свойства задачи выбора структуры системы управления и задачи

системного анализа и минимизации рисков ситуации. Для определения факторов риска и управляющих воздействий связь вектора показателей качества с векторами параметров ФЭ, факторов риска и управляющих воздействий задается в общем виде

$$\bar{K}_L = \tilde{F}_L(\bar{\chi}_L, \tilde{F}_{L-1}(\bar{\chi}_{L-1}, \tilde{F}_{L-2}(\dots \tilde{F}_\ell(\bar{\chi}_\ell, \tilde{F}_{\ell-1}(\dots, \tilde{F}_1(\bar{\chi}_1))\dots))), \quad (1.7)$$

где  $\bar{\chi}_\ell$  – кортеж для  $\ell$ -го иерархического уровня сложной иерархической системы, определенный соотношением

$$\bar{\chi}_\ell = \langle \bar{X}_\ell, \bar{u}_\ell, \bar{p}_\ell \rangle, \quad (1.8)$$

где  $\bar{X}_\ell$  – вектор параметров функциональных элементов  $\ell$ -го иерархического уровня,  $\ell = 1, \dots, L$ ;  $\bar{u}_\ell$  – вектор управления;  $\bar{p}_\ell$  – вектор факторов риска.

В общем случае для реальных сложных систем задача управления как для объекта в целом, так и для каждого  $\ell$ -го уровня его иерархической структуры состоит в одновременном выполнении двух условий:

- 1) обеспечить стабильное значение некоторых показателей качества  $\bar{K}_{1\ell}$  в фиксированный момент времени  $t_r$  заданного интервала  $[0, T]$ ;
- 2) обеспечить изменение определенных показателей качества  $\bar{K}_{2\ell}$  по заданным программам.

Первое условие в идеальном случае означает

$$\bar{K}_{1\ell} = \text{const} \forall t_r \in [0, T]. \quad (1.9)$$

В реальных условиях допускается определенное изменение показателей при воздействии дестабилизирующих факторов риска на  $\ell$ -м уровне, что определяется соотношением

$$\bar{K}_{1\ell}(\bar{u}_\ell^r, \bar{p}_\ell^r, t_r) \in [\bar{K}_{1\ell}^+, \bar{K}_{1\ell}^-] \forall t_r \in [0, T], \ell = 1, \dots, L, \quad (1.10)$$

где  $\bar{K}_{1\ell}^+, \bar{K}_{1\ell}^-$  – заданные, соответственно, верхняя и нижняя границы показателей качества.



Другое условие в идеальном случае означает, что каждая компонента вектора  $\bar{K}_{2\ell}$  является заданной функцией времени. В реальных условиях из-за воздействия дестабилизирующих факторов риска и программных управляющих воздействий в момент времени  $t_r \in [0, T]$  требуется обеспечить выполнение условия

$$\bar{K}_{2\ell}(\bar{u}_\ell^r, \bar{p}_\ell^r, t_r) \in [\bar{K}_{2\ell}^+(t_r), \bar{K}_{2\ell}^-(t_r)] \forall t_r \in [0, T], \ell = 1, \dots, L, \quad (1.11)$$

где  $\bar{K}_{2\ell}^+(t_r), \bar{K}_{2\ell}^-(t_r)$  – заданные верхняя и нижняя границы, соответственно, допустимого интервала отклонения функции  $\bar{K}_{2\ell}(t_r)$  в момент  $t_r$ .

Многоуровневая иерархическая система управления требует рационального распределения функций между уровнями иерархии, вследствие чего на каждом уровне появляется иерархическая структура целей и задач управления. Отсюда вытекает необходимость такой иерархической системы моделей, на основе которой можно создать рациональную структуру управления и обеспечить достижение поставленных целей. Такая модель должна отображать не только свойства объекта, но и его структуру [40].

#### **1.4 Аналитический обзор моделей и способов моделирования промышленного предприятия**

Промышленные предприятия имеют явно различимую структуру ступенчатого разделения ресурсов, именуемую иерархией. Под иерархической системой, или иерархией, понимают систему, состоящую из взаимосвязанных подсистем, каждая из которых также является иерархической по своей структуре. Г. Саймон (H. Simon, 1972) выдвинул гипотезу, что сложные системы представляются как иерархические и путь эволюции сложных систем – через организацию устойчивых связей и образование подсистем на промежуточном уровне [86]. В иерархических системах Г. Саймон различает взаимодействия между подсистемами и взаимодействия внутри подсистем, т. е. взаимодействия между частями этих подсистем.

Появление иерархической структуры управления обусловлено возрастающей сложностью управляемых систем, создающей значительные трудности для центрального органа управления [17]. Понятно, что целесообразным является декомпозиция сложного процесса принятия решений на несколько уровней для обеспечения решения оптимизационной задачи без значительных трудностей. Однако многоуровневые иерархические системы управления поставили новую задачу, а именно согласование и координацию решений между отдельными уровнями управления [7].

Обычно функционирование производства рассматривается на больших временных интервалах (сутки, неделя и более), во время которых можно не учитывать протекание динамических процессов [33, 62]. Поэтому задача оптимального управления многоуровневой иерархической системой может быть поставлена как статическая оптимизационная задача. При этом полагают, что высокочастотные возмущения материальных потоков отрабатываются локальными системами автоматического регулирования отдельных установок и/или диспетчерскими подразделениями на нижнем уровне управления. Однако оптимальные решения с локальными критериями эффективности отдельных подсистем оказываются мало или совершенно неэффективными, т.к. не определены переменные, позволяющие согласовать режимы работ всех подсистем между собой и не скоординированы их критерии эффективности [7].

М. Модарес (M. Modarres, 2006) анализирует состояния статического и динамического моделирования сложных систем, представленных иерархическими структурами и отмечает множество ошибок при реинжиниринге бизнес-процессов [194]. Он также отмечает (со ссылкой на Г. Саймона [86]), что изменения отношений между подсистемами должны повлечь за собой изменения внутри подсистем и, наоборот, изменения внутри подсистем должны повлечь изменения на макроуровне. Стратегия изменений на предприятии должна проводиться в двух направлениях: административная реорганизация и реинжиниринг бизнес-процессов [91].

В теории управления социально-экономическими системами технология постановки и решения задачи управления включает имитационное моделирование [68]. Для создания эффективной системы управления организацией разработаны математические модели процедур принятия управленческих решений для планирования производственной программы, организации структуры системы, стимулирования исполнителей и контроля результатов управления составом и структурой социально-экономических систем, модели институционального, мотивационного, а также информационного управления [67].

В общей теории управления производством создаются математические модели оптимального планирования и управления для многоуровневых человеко-машинных производственных систем, решаются проблемы координации на различных уровнях и разрабатываются детальные математические модели планирования, контроля и оперативного управления производством [176]. Множество моделей управления различными человеко-машинными производственными системами основаны на приближительных или эвристических подходах и не предполагают точных оптимальных решений. Таким образом, они не обеспечивают оптимальное управление, особенно при случайных возмущениях. Тем не менее, такие модели реализуют современные концепции управления и исследования промышленного производства и, как правило, отвечают всем практическим требованиям [177].

Дж. Форрестер (Jay W. Forrester, 1971) предлагает метод моделирования предприятия как динамической экономической системы с целью выработки усовершенствованных форм организации и общего руководства [95]. В своей модели Дж. Форрестер представляет деятельность промышленного предприятия шестью взаимосвязанными потоками. Пять из них – это материальные потоки, потоки заказов на поставку ресурсов и реализацию продукции, потоки денежных средств, используемого оборудования и трудовых ресурсов. Шестым потоком является информационный поток для соединения других потоков в единую сеть. Для описания динамических процессов структура модели представлена резервуарами (уровнями), связанными между собой представленными выше управляе-

мыми потоками. В основе динамического моделирования лежат: теория информационных систем с обратной связью, исследование процессов принятия решений, экспериментальное моделирование сложных систем, ЭВМ как средство имитации реальных процессов на их математических моделях.

На основе системной динамики Г. Шульц (G. Zülch, 2006) представил четыре модели анализа решающих действий в иерархической структуре промышленного предприятия: с пассивным координатором, решающие действия на основе глобальной цели, локальной цели, решения на основе глобальной цели с учетом взаимодействия с другими координаторами, именуемых агентами [225].

С. А. Жданов [36] отмечает, что в литературе по рыночной экономике недостаточно уделено внимания вопросам автоматизации получения и использования экономической информации, применению математических моделей и алгоритмов для системного анализа экономической ситуации, их комплексному использованию для обоснования и принятия решений по объему выпускаемой предприятием продукции, ценам, прибыли, необходимым финансовым ресурсам, ускорению сбыта, сбалансированного использования оборудования.

На необходимость создания динамических моделей для прогнозирования развития экономических ситуаций, особенностей функционирования производства обращалось внимание в работах А. Г. Гранберга [31], А. А. Кугаенко [49], П. В. Авдулова [1], А. А. Спирина [96], В. С. Иозайтиса [43], Л. Л. Терехова [75], В. И. Терехина [93]. Этими авторами разработаны следующие модели: определения объема производства продукции предприятия; определения объема собственных и привлекаемых финансовых средств для обеспечения производства планируемых изделий; определения требуемой прибыли и цены для накопления планируемого объема собственных финансовых средств; для количественной оценки влияния ступенчатых изменений в издержках на объемы производства готовой продукции; ресурсного планирования объема выпускаемой продукции на промышленном предприятии; оптимальной загрузки производственных мощностей; выбора технологии изготовления заказной продукции, обеспечивающей выполнение заказа при

минимальных затратах; прогнозирования спроса на товары длительного использования с помощью логистической функции [36].

С. В. Сильновой и Г. Р. Валеевым [89] предложена модель реализации продукции в условиях рынка. Модель включает подмодели: распределения ресурсов между видами выпускаемой продукции; производственного процесса изготовления продукции всех видов; формирования затрат на производство и сбыт продукции; реализации готовой продукции; расчета выручки и прибыли по итогам производственно-экономической деятельности.

А. А. Григорьева, Ю. М. Осипов [32] для решения задачи определения конкурентоспособности продукции промышленного предприятия предложили аддитивно-мультипликативную модель, модель на основе метода порогов несравнимости, модель на основе теории нечетких множеств.

Математические методы и модели используются в планировании и управлении на металлургических предприятиях, при управлении предприятием и его развитием в условиях рыночной экономики на предприятиях легкой промышленности [72], при оптимизации производственных запасов в условиях перехода к рынку [103] и развитие модели запасов готовой продукции предприятия на складе при заключении договоров на изготовление продукции с субподрядчиками [155], при управлении сложными иерархическими системами в строительной промышленности [191], для оптимизации деятельности ремонтных предприятий [153]. Отмечается большое внимание разработчиков моделей для управления строительством и в финансовой сфере [159].

С. М. Саад (S. M. Saad, 2003) и др. предложили методику распределенного моделирования распределенных производств с решением вопросов интеграции моделей различных предприятий для анализа их совместной деятельности [211]. При этом использованы широко распространенные технологии DCOM и CORBA.

Л. Витман, К. Рамачандра, В. Кеткар (L. Whitman, K. Ramachandran, V. Ketkar, 2001) определили основные свойства действующей модели предприятия [219]:

- *работоспособность*: модель должна допускать внесение изменений в соответствии с изменениями на предприятии с течением времени;
- *динамичность*: в соответствии с изменениями предприятия как системы модель должна оценивать степень изменений и причины изменений;
- *расширяемость*: модель должна поддерживать включение новых подсистем на предприятии;
- *декомпозиция*: модель поддерживает множество уровней детализации и на каждом уровне обеспечивается не только понимание системы, но и принятие решений по управлению системой;
- *совместимость с ключевыми показателями предприятия*: оценивая основные показатели предприятия, сама модель становится неотъемлемой частью предприятия;
- *использование фактических данных предприятия*: на входе модели должны быть фактические данные предприятия. Модель должна сопровождать предприятие, и предприятие должно сопровождать модель. Это обеспечивает реальность и правдоподобность модели.

Рассматривают *три категории* моделируемых процессов:

- а) процессы, преобразующие внешние ограничения в набор внутренних указаний;
- б) процессы, которые требуют ресурсы и готовят их для последующего использования;
- в) процессы, которые используют ресурсы для получения результатов производственной деятельности.

В модели предприятия могут быть представлены:

- информация (*Information*): определяет объекты, управляемые предприятием и правила управления их отношениями и взаимодействиями;
- деятельность (*Activity*): определяет функции, выполняемые предприятием (что делается);
- бизнес-процесс (*Process*), который определяет упорядоченный во времени набор бизнес-процессов;

- ресурс (*Resource*): определяет ресурсы и возможности, управляемые предприятием;

- организация (*Organization*): представляет структуру организации и набор ограничений и управляющих правил внутри предприятия.

Тогда модель  $Q$  предприятия представляется формулой:

$$Q = \langle A(S, D, E); P(S, D, E); I(S, D, E); R(S, D, E); O(S, D, E) \rangle, \quad (1.12)$$

где  $A$  – деятельность,  $P$  – бизнес-процесс,  $I$  – информация,  $R$  – ресурс,  $O$  – организация;  $S$  – возможности модели;  $D$  – динамичность модели;  $E$  – использование модели.

*Возможности  $S$*  модели классифицируются следующим образом: 5 – объединение предприятий (все три категории процессов смоделированы для объединения); 4 – множество участков предприятия (все три категории процессов смоделированы для участков предприятия); 3 – предприятие (все три категории процессов смоделированы для предприятия); 2 – система (модели становятся все более активными в деятельности предприятия и используются для отдельных важных решений); 1 – начальный уровень, на котором модели применяются однократно для решения отдельных локальных задач, но не используются активно при управлении процессами производственной деятельности предприятия.

*Динамичность  $D$*  модели классифицируют аналогично возможностям  $S$  модели.

*Использование  $E$*  модели классифицируют следующим образом: 5 – оптимизация (набор моделей активно используется и поддерживается предприятием); 4 – управление (формальное использование модели и поддержка ее предприятием); 3 – описание (формальное использование модели или поддержка ее предприятием); 2 – одномоментно (модели используются нечасто, менее одного раза в год); 1 – не узаконены (модели не используются в активной деятельности предприятия и применяются для незначительных решений, впоследствии не поддерживаются предприятием).

В системе промышленного предприятия не следует усматривать только организационный аспект, целесообразнее на первый план выдвинуть *функциональный*. Это даст возможность вариантного использования системного анализа в зависимости от способа и цели исследования. Г. Ортнер [74] отмечает, что в качестве целевой функции может быть задано следующее уравнение производственно-экономической оценки системы:

$$E_0 - C_0 \rightarrow \max, \quad (1.13)$$

где  $E_0$  – общий доход,  $C_0$  – общие затраты.

В преобразованном (в зависимости от иерархической организационной структуры системы) виде это уравнение представится следующим образом:

$$\sum_{i=1}^m \sum_{j=1}^n e_{ij} - \sum_{i=1}^m \sum_{j=1}^n c_{ij} \rightarrow \max, \quad (1.14)$$

где  $e_{ij}$  – доход от деятельности подсистемы;  $c_{ij}$  – сумма затрат подсистемы;  $m$  – число подсистем;  $n$  – число ступеней системы;  $i, j$  – номер подсистемы и номер ступени, соответственно.

На примере взаимоисключающих целевых функций подсистем глобальной системы «Предприятие» следует указать возможность возникновения ошибок при оценке системы с точки зрения максимизации разницы между доходами и затратами при несоответствии между собой целей оптимизации подсистем и общей цели оптимизации системы в целом. Для *производства* целевая функция – максимальная эластичность оснащения производства, отвечающего требованиям рынка при заданном уровне инвестиций; для *складирования* целевая функция – минимальный уровень складских запасов по отношению к формируемому спросу (поступающим заказам).

Таким образом, разработанные модели промышленного предприятия позволяют исследовать различные аспекты его функционирования [10, 12, 19]. Однако отсутствуют сведения об использовании комплексной модели предприятия,



включающей модели различных аспектов его функционирования на различных ступенях иерархической структуры (с учетом целевых функций подсистем и глобальной системы). Такая ситуация может объясняться сложным характером взаимодействия подсистем предприятия, особенно в условиях неопределенности и риска современного рынка. В этих условиях решением проблемы исследования функционирования промышленного предприятия является использование методов имитационного моделирования [30, 83].

### **1.5 Методы и средства имитационного моделирования сложных систем**

Наиболее полная классификация способов формализации и организации имитации приведена в работах [59]: имитация активностями, событийное моделирование, имитация транзактами, агрегатами и процессами. Не существует универсального способа имитации. Каждый из способов имеет свои преимущества и недостатки и ориентирован на создание моделей определенного класса систем. При построении имитационных моделей производственно-экономической деятельности предприятия требуется высокий уровень детализации выполнения функциональных действий, при этом в ходе эксплуатации имитационная модель часто используется для нахождения так называемых узких мест и слабых звеньев в системе. Метод имитационного моделирования на основе процессного способа формализации позволяет обеспечить очень важное в подобной ситуации сходство структуры модели и объекта исследования.

Основные подходы в имитационном моделировании – это использование системной динамики (СД), дискретно-событийное моделирование (ДС), агентное моделирование (АМ). Системная динамика и дискретно-событийное моделирование – традиционные устоявшиеся подходы, агентное моделирование – относительно новый подход. Математически системная динамика и динамические системы моделируют непрерывные во времени процессы. Дискретно-событийное моделирование и агентное моделирование – в основном дискретные процессы [213, 217, 182, 222].

Применение специальных систем имитационного моделирования обладает следующими преимуществами: сокращение временных затрат на программирование; применение эффективных методов обнаружения ошибок имитации; точность представления понятий, определяющих имитационные процессы; существует возможность для исследуемой предметной области заранее построить стандартные компоненты (библиотеки имитационных моделей); автоматическое построение типов данных в соответствии с используемым способом имитации и требуемых в ходе имитационного эксперимента; удобство накопления и представления выходных данных; возможность учета субъективных предпочтений эксперта и его опыта в процессе принятия решений; возможность динамического анализа сценария развития.

В настоящее время известен ряд систем имитационного моделирования, обладающих функциональными возможностями для моделирования деятельности промышленного предприятия. Например, *Simkit* является пакетом моделирования с открытым кодом, написанным на Java. Основан на графовом представлении дискретных событий, компоненто-ориентированном моделировании, является платформу-независимым [163]. Однако основным недостатком, ограничивающим его применение для моделирования сложного промышленного предприятия, является отсутствие средств коллективного управления процессом моделирования, что приводит к увеличению сроков проектирования и значительным ошибкам в проекте модели.

*Flexsim* - объектно-ориентированная среда для разработки, моделирования, исследования, визуализации и мониторинга динамических потоков. Использует C++ непосредственно или Flexs-скрипты. Вся мультипликация использует OpenGL. Результаты моделирования могут быть экспортированы во внешнее программное обеспечение через DDE, DLL, ODBC, SQL [195, 151]. Несмотря на значительные функциональные возможности, *Flexsim* ограничивает использование уже разработанных программных моделей.

*Pilgrim* – пакет имитационного моделирования для создания дискретно-непрерывных моделей, поддерживается коллективное управление процессом

моделирования, модели переносятся на любую платформу, если имеется компилятор языка программирования C++. Достоинствами пакета *Pilgrim* являются: комплексное моделирование материальных, «денежных», а также информационных процессов; имеется CASE-оболочка для проектирования многоуровневых моделей в режиме структурного системного анализа; наличие средств доступа к источникам данных [35]. Недостатки: исследование сложных имитационных моделей функционирования промышленного предприятия ограничивается отсутствием средств распределенных вычислений.

Наиболее известные системы имитационного моделирования и их основные характеристики представлены в таблице 1.1, взятой за основу из работы В. Б. Резникова [79].

Например, *РДО* (Ресурсы-Действия-Операции) – пакет имитационного моделирования, созданный в МГТУ им. Н. Э. Баумана, с развитыми средствами компьютерной графики, применяется при моделировании сложных технологий и производств. В нем, однако, моделирование функционирования промышленного предприятия ограничено отсутствием средств интеграции с корпоративной информационной системой предприятия [46].

*DESMO* относится к системам распределенного имитационного моделирования, использует генетический алгоритм для оптимизации. Разработанные модели являются Java программами, не зависят от используемой платформы. Эксперименты успешно проведены под Solaris, Linux, Windows и MacOS [175]. Недостатком является отсутствие средств автоматизации построения концептуальной модели исследуемого объекта, что является актуальным для комплексного моделирования подсистем в многоуровневой иерархической структуре промышленного предприятия.

*AweSim* включает встроенный язык моделирования Visual SLAM для сетей, дискретных и непрерывных систем. Сети моделируются в визуальном режиме. Другие системы могут быть смоделированы на Visual Basic, C или C++. Имеются встроенные средства мультипликации, иллюстрирующие работу модели на разных уровнях иерархии (декомпозиции).

**Таблица 1.1 – Сравнительный анализ систем имитационного моделирования и их основных характеристик**

Система имитационного моделирования	Визуальное программирование	Текстовое программирование	Непрерывные модели	Дискретные модели	Генерация кода исполнения	Отображение графиков функций	Поддержка анимации	Интеграция с информационной системой	Ориентация на распределенные вычисления	Встроенные средства оптимизации	Коллективное управление моделированием
2D-Sim	+	+	+	-	+	+	+	-	-	-	-
ACSL	-	+	+	-	+	+	-	-	-	-	-
AnyLogic	+	+	+	+	+	+	-	-	-	+	+
Awesim	+	+	+	+	+	+	+	+	-	-	-
BelSim2	-	+	+	+	+	+	-	+	+	+	+
Dymola	+	+	+	-	-	+	+	-	-	-	-
Dynast	+	+	+	-	-	+	-	-	-	-	-
Desmo	-	+	+	+	+	+	+	+	+	+	-
Easy-5	+	+	+	+	-	+	-	+	-	-	-
Flexsim	+	+	+	+	+	+	+	+	-	-	-
LabView	+	+	+	+	+	+	+	+	-	+	-
MICIC4	-	+	-	+	+	+	-	-	-	-	-
MVS	+	+	+	+	+	+	+	+	+	-	+
Pilgrim	+	+	+	+	+	+	+	+	-	-	-
Simulink	+	-	+	+	-	+	-	+	-	+	-
Simkit	+	+	-	+	+	+	-	-	-	-	-
SYMBOLS-2000	+	-	+	-	-	+	+	-	-	-	-
VisSim	+	-	+	+	-	+	-	-	-	+	-
VTB	+	-	+	-	-	+	+	+	-	+	-
РДО (МВТУ)	+	-	+	+	-	+	-	-	-	-	-
СимНП	+	+	+	+	+	+	+	-	+	-	-

Примечание. (+) означает есть; (-) означает нет

*AweSim* интегрируется с приложениями Windows. Использует реляционную базу данных, интегрируясь с Dbase, Access, FoxPro и Excel [199]. Недостаток: ориентирован на моделирование, в первую очередь, технических систем.

В Таганрогской научной школе вычислительных систем для моделирования и интеллектуального управления под руководством д.т.н., профессора В. Ф. Гузика и д.т.н., профессора В. Е. Золотовского реализована программно-аппаратная система структурного моделирования *СимНП* с использованием вычислительной сети и многопроцессорных акселераторов. Объекты моделирования описываются иерархическими структурными схемами, состоящими из компонент, задаваемых в форме математической нотации [79]. Несмотря на использование современных информационных технологий, его применение ограничено, в основном, техническими системами.

Таким образом, обзор методов и средств имитационного моделирования показывает, что известные системы имитационного моделирования по разным причинам не полностью отвечают требованиям моделирования функционирования промышленного предприятия в структуре корпоративной информационной системы.

С. Онго (S. Onggo, 2006) и др. отмечают, что в будущем моделирование будет обладать свойствами модульности, повторного использования моделей, иерархичностью структуры; системы имитационного моделирования должны быть масштабируемыми, портативными, взаимодействующими между собой; исполнение модели - в распределенном режиме, через Internet, удобным для использования [198].

Многообразие средств имитационного моделирования ставит перед исследователями задачу разработки стандартов для обеспечения взаимодействия программного обеспечения имитационного моделирования с другими программами, что позволит расширить область применения технологии моделирования [158, 192].

## 1.6 Применение методов оптимизации в имитационном моделировании

### 1.6.1 Анализ пакетов оптимизации в имитационном моделировании

В последние годы обращается внимание на решение оптимизационных задач в имитационном моделировании. На основании аналитических работ А. М. Лоу (A. M. Law) [186], Дж. Эприла (J. April) [156], М. С. Фу (M.C. Fu) [173] и др. построена таблица 1.2 с наиболее известными оптимизационными пакетами. Наиболее широко применяются в практике имитационного моделирования: *AutoStat*, *Evolutionary Optimizer*, *OptQuest*, *RISKOptimizer*, *WITNESS Optimizer*, построенные на основе эвристических процедур случайного поиска.

**Таблица 1.2 – Основные характеристики оптимизационных пакетов имитационного моделирования**

Пакет оптимизации	Поставщик (URL)	Используемые эвристические процедуры	Система имитационного моделирования
AutoStat	AutoSimulations, Inc. (www.autosim.com)	Эволюционные стратегии, генетические алгоритмы	AutoMod
Evolutionary Optimizer	ImagineThat, Inc. (www.imaginethatinc.com)	Эволюционные стратегии, генетические алгоритмы	Extend
OptQuest	OptTek Systems, Inc. (www.opttek.com)	Случайный поиск, поиск с запретами, нейронные сети	AnyLogic; Arena; Crystal Ball; CSIM19; Enterprise Dynamics; Micro Saint; ProModel; Quest; SimFlex; SIMPROCESS; SIMUL8; TERAS
RISKOptimizer	Palisade Corp. (www.palisade.com)	Генетические алгоритмы	@RISK
WITNESS Optimizer	Lanner Group, Inc. (www.lanner.com/corporate)	Моделируемый отжиг, поиск с запретами	WITNESS

Развитие эволюционных подходов, основанных на случайном поиске, к решению оптимизационных задач в значительной мере определяется работами Д. И. Батищева, И. Л. Букатовой, Л. А. Гладкова, В. В. Курейчика, В. М. Курейчика, Л. А. Растригина, Л. Фогеля (L.J. Fogel), Дж. Холланда (John H. Holland), М. Л. Цетлина, Г. Шефеля (H. Schwefel) и др. [24].

Преимущества эволюционных подходов к решению оптимизационных задач состоят в следующем:

- достаточно высокое быстродействие;
- относительно высокая надежность и помехоустойчивость;
- высокая робастность, т. е. малая чувствительность к нерегулярностям поведения функции полезности;
- достаточно простая внутренняя реализация;
- малая чувствительность к росту размерности задачи оптимизации;
- возможность ввода в процесс поиска процедур обучения и самообучения;
- на основе известных методов случайного поиска создаются новые алгоритмы с эвристическими процедурами адаптации.

*Недостатки* их использования такие:

- нет каких-либо гарантий качества получаемых решений;
- невозможность проверить на всех мыслимых классах задач;
- требуется предварительная настройка параметров алгоритма для его эффективной работы [143].

### **1.6.2 Основные алгоритмы случайного поиска, используемые при оптимизации в имитационном моделировании сложных систем**

Наиболее простым и распространенным алгоритмом случайного поиска является *метод Монте-Карло*, иногда именуемый *грубым случайным поиском*. Реализация метода Монте-Карло может быть представлена следующим образом:

$$G \rightarrow U \rightarrow X \rightarrow Y \rightarrow \hat{\theta}.$$

Каждая из четырех стрелок – детерминированная функция. Генератор случайных чисел  $G$  используется для получения множества случайных чисел  $U$ , которые формируют множество входных данных  $X$ , используемых для вычисления множества выходных данных  $Y$ , по которым определяется статистическая оценка  $\hat{\theta}$  [187].

Метод Монте-Карло – самый простой и известный из применяемых алгоритмов случайного поиска. Суть метода состоит в равномерном случайном «бросании» точек в пространство поиска, задаваемого условиями решаемой задачи. Ввиду его основного достоинства – простоты, применяется в теории глобальной оптимизации при сравнительном анализе алгоритмов и в качестве компонента алгоритмов глобального случайного поиска [168, 216].

Например, А. Шапиро (A. Shapiro, 2001) предлагает делать выборку и работать над ней с использованием детерминированных методов оптимизации. Затем множество подобных выборок усредняют, получая оптимальное решение [214].

*Генетические алгоритмы (ГА) (Genetic Algorithms)* сочетают в себе элементы детерминистического и стохастического подходов, поэтому их часто относят к группе адаптивных методов для решения задач поиска и оптимизации. ГА успешно применяются в сочетании с аналитическими методами оптимизации или другими алгоритмами случайного поиска [90].

Принцип работы генетических алгоритмов основан на моделировании заимствованных у природы процессов развития популяции биологической особи на генном уровне: перестраивание набора хромосом генотипа с наследованием участков хромосомных наборов родителей (кроссинговер), случайное изменение генотипа (мутация). Процедура естественного отбора, также заимствованная у природы, направлена на совершенствование при смене поколений приспособленности членов популяции к «выживанию», благодаря отбору особей с определенными признаками [70, 179, 189, 204, 205].



Применяются ГА достаточно широко: в автоматизированном решении комбинаторных задач, при проектировании и оптимизации нейронных сетей, в экспертных и обучающихся системах. Приводятся результаты совместного применения имитационного моделирования, генетических алгоритмов и целочисленного программирования для определения оптимальной сети поставки сырья и материалов; применения имитационного моделирования и генетического алгоритма для выбора поставщика с учетом рыночного спроса, параметров хранения запасов, характеристик поставщика и транспортных потоков [167, 200].

Метод *моделируемого отжига* (*Simulated Annealing*) был разработан С. Киркпатриком (S. Kirkpatrick, 1983), основан на аналогии термодинамического процесса нагревания и медленного охлаждения некоторой субстанции для получения кристаллической структуры.

Метод является существенно развитой версией метода наискорейшего спуска. Начинается реализация метода с выбора любой случайно выбранной точки в заданном пространстве поиска, затем двигаются в случайном направлении. Если это движение приводит в точку с более низким уровнем значения функции эффективности, то оно принимается. Если движение приводит в точку с большим значением функции эффективности, то оно принимается с вероятностью  $P(t)$ , где  $t$  – время. Функция  $P(t)$  сначала близка к единице, но затем постепенно уменьшается до нуля – по аналогии с процессом охлаждения твердого тела. Таким образом, в начале поиска принимаются любые перемещения, но, когда «температура» падает, вероятность совершения отрицательных шагов уменьшается [185]. Отрицательные перемещения иногда необходимы в том случае, когда требуется избежать локального экстремума. Метод активно исследуется, создавая новые модификации, например, быстрый «переотжиг», параллельный отжиг.

*Поиск с запретами* (*Tabu search*) предложен Ф. Гловером (Fred W. Glover, 1986), также является стохастическим методом глобального поиска, в основу которого положена модель человеческого поведения, т. е. в поведенческой схеме человека присутствуют случайные элементы, что означает разнообразное поведение в одной и той же ситуации. При поиске составляется и сохраняется список

запретов, например, уже рассмотренных точек пространства поиска (локальных экстремумов), чтобы препятствовать возврату к ранее посещенным решениям [156, 161, 173].

Д. Волперт, В. Макреди (D. H. Wolpert, W. G. Macready, 1996) опубликовали NFL (No Free Lunch) теорему, в которой утверждается, что для всех алгоритмов поиска экстремума функции качества (эффективности), при усреднении результатов по всем различным функциям эффективности, производительность одинакова [220, 224]. Значение теоремы для практики применения алгоритмов случайного поиска состоит в том, что не существует какого-либо наилучшего из оптимизационных методов, пригодного для всех областей их применения. Это означает, что для каждой предметной области для решения оптимизационных задач необходимо исследовать и выбирать свой метод оптимизации.

Эволюционные стратегии (Evolution Strategies) – эволюционный подход, созданный в начале 1960-х годов и развитый в 1970-х годах И. Рехенбергом (Ingo Rechenberg), Г.-П. Швэфелем (Hans-Paul Schwefel) и их сотрудниками. В стратегиях эволюции используют мутацию и выбор, как операторы поиска. Последовательность поколений продолжается до тех пор, пока не будет выполнен критерий завершения. Эволюционные стратегии применяются для решения задач комбинаторики, в экспертных системах и при машинном обучении [179].

Для решения задач глобальной оптимизации и построения оптимизатора при имитационном моделировании разработан алгоритм SPSA (Simultaneous Perturbation Stochastic Approximation), показывающий лучшие результаты в сравнении с генетическим алгоритмом и моделируемым отжигом, которые часто рекомендуются для глобальной оптимизации. Алгоритм SPSA основан на *стохастической аппроксимации SA (stochastic approximation)* с введением дополнительного шума. Использование SPSA часто приводит к рекурсии, что намного более экономично в терминах функции потерь, чем стандартная версия SA [190].

Р. Аль-Аомар (R. Al-Aomar, 2002) предложил *метод робастной многокритериальной оптимизации* на основе имитационного моделирования, который при реализации использует имитационную модель, генетический алгоритм, метод энтропии [48] и модуль робастности, дающий информационные оценки параметрам и откликам модели. Например, продукция производства может быть сделана робастной к изменению материалов, параметров производственного процесса, условий применения продукции. Производственные процессы также могут быть робастными к изменению материальных ресурсов, свойств окружающей среды, параметров используемого оборудования. Метод направлен на то, чтобы многоуровневые производства/бизнес-процессы сделать робастными к изменению системно-уровневых факторов, таких, как нормы поставок сырья и материалов, время обработки/обслуживания, отказы оборудования.

Для параметров и откликов модели введены три информационные оценки на основе отношения сигнал/шум. В *аддитивной функции эффективности* при многокритериальной оптимизации метод энтропии используется для оценки веса важности отдельного критерия. Критерии функции эффективности должны быть взаимно независимы, чтобы эффективность одного критерия не влияла на эффективность других критериев. В противном случае возможны ошибочные результаты и, соответственно, выводы по принятию решений по итогам исследований. Чтобы исключить подобную погрешность в принятом методе функция полезности имеет динамическую структуру. Структура изменяется на каждом шаге итерации ГА, изменяя величины весов. Чтобы функция полезности могла оценивать различные альтернативы, выполнено нормирование критериев [152].

Установлены шесть основных областей, в которых ожидается развитие методов оптимизации при имитационном моделировании: методы оптимизации, используемые в системах имитационного моделирования; интерфейсы между оптимизатором и пользователем и между оптимизатором и имитационной моделью; формализация проблемы, связанная с конструированием целевой функции и ограничений; классификация проблемы оптимизации для выбора метода оптимизации; стратегия и тактика использования оптимизации в имитационном

моделировании для эффективного использования вычислительных средств (выбор методов оптимизации для класса проблем, способа отображения показателей, способа имитационного моделирования, метода предварительной обработки данных, многочисленные сравнивающие тесты и др.); интеллектуализация (развитые логические возможности) оптимизатора для выбора стратегии и тактики использования различных методов [161].

Известен метод принятия решений, основанный на *визуальном интерактивном имитационном моделировании* с применением нейронной сети и экспертной системы. Метод реализуется следующими основными стадиями: понимание процесса принятия решения; сбор данных; определение стратегий принятия решений экспертами; вычисление результатов стратегии принятия решения; поиск усовершенствований. Апробирован метод на заводе по сборке двигателей [209].

В дополнение к хорошей модели требуется технология сжатия данных моделирования для принятия решения. Приводится алгоритм последовательного отбора с памятью *SSM (Sequential Selection with Memory)* [201].

Одна из трудностей использования методов оптимизации на основе имитационного моделирования состоит в том, что они достаточно сложны и требуют значительных знаний об объекте со стороны пользователя. Предложена структура оптимизации, которая позволяет эффективно решать сложные проблемы комбинаторного типа на основе имитационного моделирования в автоматизированном режиме. Для поиска оптимального решения, когда целевая функция не может быть задана аналитически, приводится алгоритм, в котором область решений разбивается на подмножества. Поиск проводится в наиболее многообещающих областях; выполняется случайная выборка из каждой подобласти для оценки всей области в целом; улучшаются оценки каждого подмножества, которые используются как отправные точки для поиска лучшего решения в пределах исследуемой области [197].

В литературе [169] предлагаются новые подходы к моделированию связанных между собой процессов в многоуровневых системах. Решение такой задачи представляется чрезвычайно сложным, хотя многие предметные области

извлекли бы выгоду. Рассмотрен метод решения задачи, состоящей из нескольких представлений описания. При этом находится решение для каждого из уровней, целевая функция представлена линейным уравнением с весовыми коэффициентами.

### 1.7 Выводы по первой главе

Проведенный обзор и анализ методов и средств моделирования и принятия решений в управлении промышленным предприятием и информационных технологий в корпоративных информационных системах позволяет сделать следующие выводы.

1. *Состояние промышленных предприятий* характеризуется большим количеством, интенсивностью и разнонаправленностью информационных связей, слабой формализацией механизмов управления, отсутствием методов и средств оптимизации систем управления бизнес-процессами и производственными процессами. Особенности *структуры управления* предприятием: сложность и многообразие вопросов, анализируемых при принятии решения; декомпозиция функциональных задач приводит к созданию структуры управления с разными уровнями, подсистемы уровней имеют *разные цели функционирования*, во многом *взаимно противоположные*.

2. Существующие концепции управления промышленным предприятием *MRP, MRP II, ERP* (планирование ресурсов предприятия) и др. направлены на повышение эффективности управления предприятием. Для их информационной поддержки используют программные средства – корпоративные информационные системы (КИС). В них важное место отводится моделированию. Однако существующие аналитические модели разработаны для отдельных подсистем, а использование имитационного моделирования в системах управления предприятием затруднено построением модели сложной системы и отсутствием средств автоматизации построения таких моделей. Поэтому нужна имитация деятельности предприятия, но для применения методов имитационного моделирования требуются новые информационные технологии.

3. Корпоративная информационная система располагает информационными технологиями моделирования бизнес-процессов, моделирования программных систем, технологиями доступа к базам данных информационной системы, программными средствами статистической обработки данных, технологиями обмена данными между программными системами, интегрированными средами разработки программных систем в соответствии с концепцией объектно-ориентированного программирования, имеются информационные технологии распределенных вычислений.

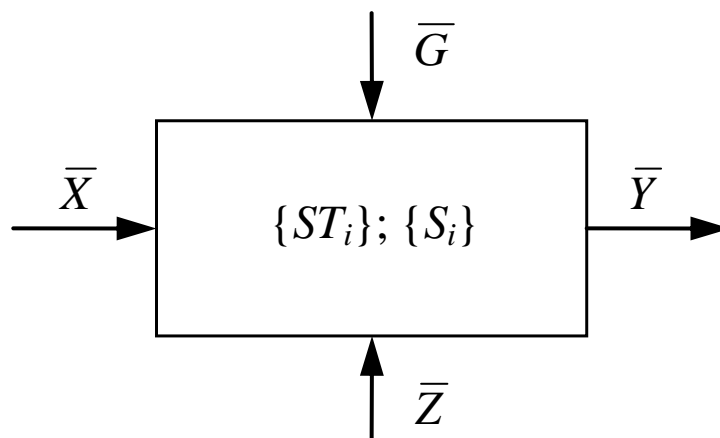
4. Анализ известных систем имитационного моделирования и их характеристик показывает непригодность их для применения в информационной системе промышленных предприятий из-за отсутствия полностью или в частичной мере средств распределенных вычислений, встроенных средств оптимизации, возможностей коллективного управления проектом имитационной модели сложной многоуровневой системы, отсутствием средств интеграции с корпоративной информационной ERP-системой управления ресурсами предприятия.

5. Современные системы имитационного моделирования применяют методы оптимизации в имитационных экспериментах. При этом наиболее широко используются эвристические процедуры, основанные на алгоритмах случайного поиска. Такие средства оптимизации, благодаря их универсальности и простоте использования, должны быть и в системах имитации для поиска рациональных решений на промышленном предприятии.

## ГЛАВА 2. РАЗРАБОТКА МЕТОДА ПОИСКА РАЦИОНАЛЬНЫХ РЕШЕНИЙ В ИНФОРМАЦИОННОЙ СИСТЕМЕ УПРАВЛЕНИЯ ПРОМЫШЛЕННЫМ ПРЕДПРИЯТИЕМ

### 2.1 Формальное описание многоуровневой иерархической системы промышленного предприятия

В практике имитационного моделирования [59] для построения концептуальной модели объекта исследования используют, как правило, графическое описание, представленное на рисунке 2.1. Однако для формализации промышленного предприятия, как многоуровневой иерархической системы, такое представление является недостаточным.



$\bar{X}$  – вектор управляемых параметров моделирования;  $\bar{G}$  – вектор задаваемых параметров,  $\bar{Z}$  – вектор неуправляемых параметров;  $\bar{Y} = \Phi(\bar{X}; \bar{G}; \bar{Z})$  – вектор выходных параметров (откликов), входящих в состав функции качества;  $\{ST_i\}$  – множество статистик моделирования;  $\{S_i\}$  – множество состояний объекта моделирования.

**Рисунок 2.1 – Графическое представление концептуальной модели**

В соответствии с формализацией сложной системы, предложенной в работе [180], любая многоуровневая система на  $\ell$ -м уровне  $S^\ell$  описывается следующей символьной конструкцией:

$$S^\ell \leftrightarrow \{w, S_0, \sigma\}^\ell, \quad (2.1)$$

где  $w^\ell$  – функциональные действия компонентов системы;

$\sigma^\ell$  – структура ( конструкция) многоуровневой системы на  $\ell$ -м уровне,

$S_0^\ell$  – координатор, определенный на  $\ell$ -м уровне,

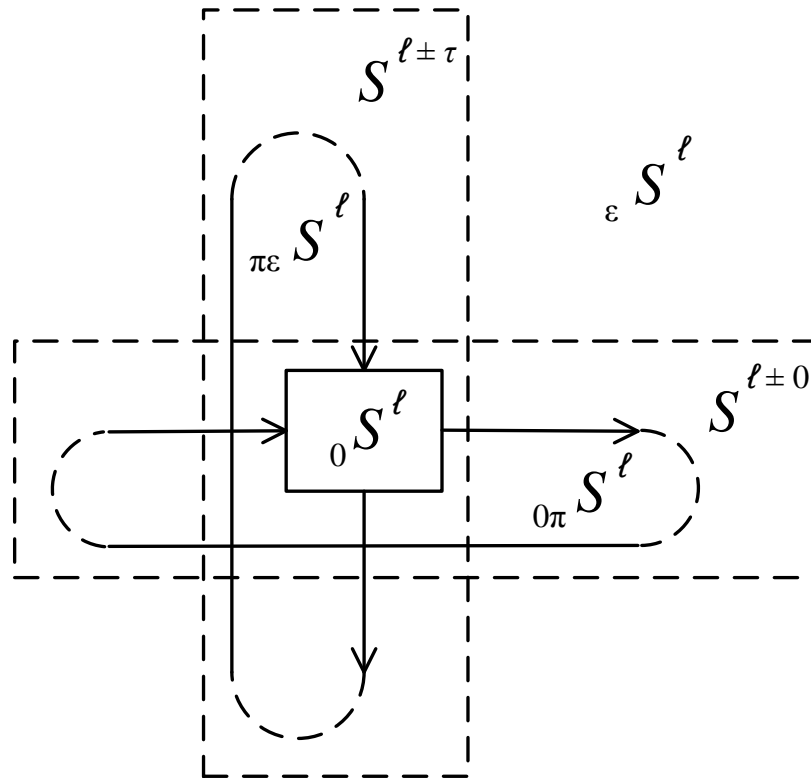
$\ell$  – индекс уровня,  $\ell \in L$ .

Для системы  $S^\ell$  с подсистемами  $\ell$ -го уровня функциональные действия и структура на  $\ell$ -м уровне представлены множествами, соответственно:

$$w^\ell \leftrightarrow \{\tilde{w}, S_0\}^\ell, \quad (2.2)$$

$$\sigma^\ell \leftrightarrow \{S_0, \tilde{\sigma}\}^\ell. \quad (2.3)$$

Координирующий элемент  $S_0^\ell$  определяется внутренней  ${}_0S^\ell$  и окружающей средой  ${}_\varepsilon S^\ell$ . При этом для  ${}_0S^\ell$  определены функциональные действия и структура  ${}_{0\pi}S^\ell$  внутриуровневых взаимодействий в окружающей среде  ${}_\varepsilon S^\ell$  и функциональные действия и структура  ${}_{\pi\varepsilon}S^\ell$  межуровневых взаимодействий в окружающей среде  ${}_\varepsilon S^\ell$  (рисунок 2.2).



**Рисунок 2.2 – Формализация многоуровневой системы**

Соответственно:



$$\{ {}_0S, {}_{0\pi}S \}^\ell \leftrightarrow S^{\ell \pm 0}; \quad (2.4)$$

$$\{ {}_{\pi\varepsilon}S, {}_\varepsilon S \}^\ell \leftrightarrow S^{\ell \pm \tau}, \quad (2.5)$$

где  $S^{\ell \pm 0}$  – система внутриуровневых отношений  $\ell$ -го уровня,  ${}_0S^\ell$  – внутренняя среда на  $\ell$ -м уровне;  ${}_{0\pi}S^\ell$  – структура внутриуровневых взаимодействий в окружающей среде  ${}_0S^\ell$ ;  $S^{\ell \pm \tau}$  – система межуровневых отношений;  ${}_{\pi\varepsilon}S^\ell$  – структура межуровневых взаимодействий с окружающей средой  ${}_\varepsilon S^\ell$ ,  $\tau$  – индекс уровня многоуровневой системы (рисунок 2.2) [124].

В системе на рисунке 2.2 функциональные действия и структура на  $\ell$ -уровне представляются взаимодействующими с окружающими уровнями:

$$\tilde{w}^\ell \leftrightarrow \{ w^{\ell \pm \tau}, {}_w\gamma \mid \tau \in L \}, \quad (2.6)$$

где  $w^{\ell \pm \tau}$  с учетом системных действий на  $\ell \pm \tau$ - уровне и отношений  ${}_w\gamma^\ell$  представлены в виде:

$$\begin{aligned} w^{\ell \pm \tau} &\leftrightarrow \{ \tilde{w}^{\ell \pm 0}, \tilde{w}^{\ell \pm \tau}, {}_w\gamma^\ell \mid \tau \in L, \tau \neq 0 \} \leftrightarrow \\ &\leftrightarrow \{ \{ {}_0w, {}_{0\pi}w, {}_w\gamma_0 \}^{\ell \pm 0}, \{ {}_{\pi\varepsilon}w, {}_\varepsilon w, {}_w\gamma_\varepsilon \}^{\ell \pm \tau}, {}_w\gamma^\ell \mid \tau \in L, \tau \neq 0 \}. \end{aligned} \quad (2.7)$$

Отношения  ${}_w\gamma^\ell$  определяются множествами состояний  ${}_kC^\ell$ , входов  ${}_kX^\ell$  и выходов  ${}_kY^\ell$ :

$${}_w\gamma^\ell \leftrightarrow \{ \{ X, C, Y \}_k \mid k \in {}_kL, {}_kL \leftrightarrow \{ 0, 0\pi, \pi\varepsilon, \varepsilon \} \}^\ell. \quad (2.8)$$

С учетом системных (внутриуровневых и межуровневых) взаимодействий динамика функциональных действий представлена следующими характеристическими функциями:

$${}_k w^\ell \leftrightarrow {}_k (\tilde{\rho}, \tilde{\delta})^\ell \mid k \in {}_kL, {}_kL \leftrightarrow \{ 0, 0\pi, \pi\varepsilon, \varepsilon \}. \quad (2.9)$$

Функция выходов:

$${}_k \tilde{\rho}^\ell \leftrightarrow {}_k \{ \rho_t : C_t \times X_t \rightarrow Y_t \wedge t \in T \}^\ell. \quad (2.10)$$

Функция переходов:

$${}_k \tilde{\delta}^\ell \leftrightarrow_k \{ \delta_{t'} : C_t \times X_t \rightarrow C_{t'} \wedge t, t' \in T \wedge t' > t \}^\ell. \quad (2.11)$$

В развитие работы [180] предлагается формализация структуры элементов  $\ell$ -го уровня с учетом структурных отношений  $\gamma_\sigma^\ell$ , позволяющих оценивать эффективность структуры системы:

$$\tilde{\sigma}^\ell \leftrightarrow \{ \sigma^{\ell \pm \tau}, \gamma_\sigma | \tau \in L \}. \quad (2.12)$$

Аналогично (2.6)  $\sigma^{\ell \pm \tau}$  представим в виде:

$$\begin{aligned} \sigma^{\ell \pm \tau} &\leftrightarrow \{ \tilde{\sigma}^{\ell \pm 0}, \tilde{\sigma}^{\ell \pm \tau}, \gamma_\sigma^\ell | \tau \in L, \tau \neq 0 \} \leftrightarrow \\ &\leftrightarrow \{ \{ {}_0\sigma w_i | i \in I, {}_0\gamma_\sigma \}^{\ell \pm 0}, \{ {}_\infty\sigma w_i | i \in I, {}_\infty\gamma_\sigma \}^{\ell \pm \tau}, \gamma_\sigma^\ell | \tau \in L, \tau \neq 0 \}. \end{aligned} \quad (2.13)$$

Структурные отношения могут быть представлены элементами матрицы для внутриуровневых отношений:

$$[{}_0\gamma_\sigma^{\ell \pm 0}] \leftrightarrow \begin{cases} 1, | ({}_0\sigma w_i, {}_0\sigma w_j) \in {}_0\gamma_\sigma^{\ell \pm 0}, \forall i, j \in I; \\ 0, | ({}_0\sigma w_i, {}_0\sigma w_j) \notin {}_0\gamma_\sigma^{\ell \pm 0}, \forall i, j \in I, \end{cases} \quad (2.14)$$

и аналогично для межуровневых отношений:

$$[{}_\infty\gamma_\sigma^{\ell \pm \tau}] \leftrightarrow \begin{cases} 1, | ({}_\infty\sigma w_i, {}_\infty\sigma w_j) \in {}_\infty\gamma_\sigma^{\ell \pm \tau}, \forall i, j \in I \wedge \tau \in L, \tau \neq 0; \\ 0, | ({}_\infty\sigma w_i, {}_\infty\sigma w_j) \notin {}_\infty\gamma_\sigma^{\ell \pm \tau}, \forall i, j \in I \wedge \tau \in L, \tau \neq 0, \end{cases} \quad (2.15)$$

Система  $S^{\ell \pm 0}$  имеет выходы  $Y^{\ell \rightarrow \ell \pm \tau}$ , направленные на вышестоящий уровень –  $Y^{\ell \rightarrow \ell + \tau}$  и, соответственно, на нижестоящий – выходы  $Y^{\ell \rightarrow \ell - \tau}$ . Система  $S^{\ell \pm 0}$  получает возмущения по входам  $X^{\ell \leftarrow \ell \pm \tau}$ : от вышестоящих уровней – по входам  $X^{\ell \leftarrow \ell + \tau}$  и от нижестоящих уровней – по входам  $X^{\ell \leftarrow \ell - \tau}$ . Состояние  $C^\ell$  системы  $\ell$ -уровня определяется собственными входами и выходами:

$$C^\ell \leftrightarrow \{ X^{\ell \leftarrow \ell}, Y^{\ell \rightarrow \ell} \} \quad (2.16)$$

Благодаря системным связям  ${}_w\gamma^\ell$  любое функциональное действие  $w^\ell$  может быть изменено другими элементами системы [92, 121, 157].

## 2.2 Идея метода поиска рациональных решений в информационной системе управления предприятием

В соответствии с представленной выше формализацией многоуровневой иерархической системы, использование имитационной модели для поиска рациональных решений в информационной системе управления промышленным предприятием осуществляется следующей последовательностью этапов [78]:

*Этап 1. Формулировка целей управления предприятием и подсистемой исследуемого уровня.* Выбирается цель или множество целей, которые должны быть достигнуты системой управления процессами функционирования предприятия и подсистемой на определенном уровне.

*Этап 2. Определение объекта управления.* Этот этап связан с выделением той подсистемы исследуемого объекта, состояние которой интересует для достижения сформулированных целей. При формальном описании объекта, проводя декомпозицию и выделяя отдельные подсистемы, всегда можно проверить на модели, достигаются ли цели управления объектом. Если не достигаются, то можно оценить степень неуправляемости этого объекта. На стадии определения объекта с помощью экспертов синтезируют несколько его вариантов, затем оценивают по одному или нескольким критериям и выбирают лучший.

*Этап 3. Структурный синтез модели.* Этап является сложным и включает следующие шаги [116]:

*Шаг 3.1.* Определение входных и выходных параметров объекта, т.е. выполняется синтез модели объекта в виде «черного ящика»;

*Шаг 3.2.* Экспертное ранжирование важности входных и выходных параметров объекта;

*Шаг 3.3.* Декомпозиция модели объекта на отдельные компоненты;

*Шаг 3.4.* Выбор структурных компонентов модели.

Экспертный метод решения задач, перечисленных на третьем этапе, чаще всего является основным.

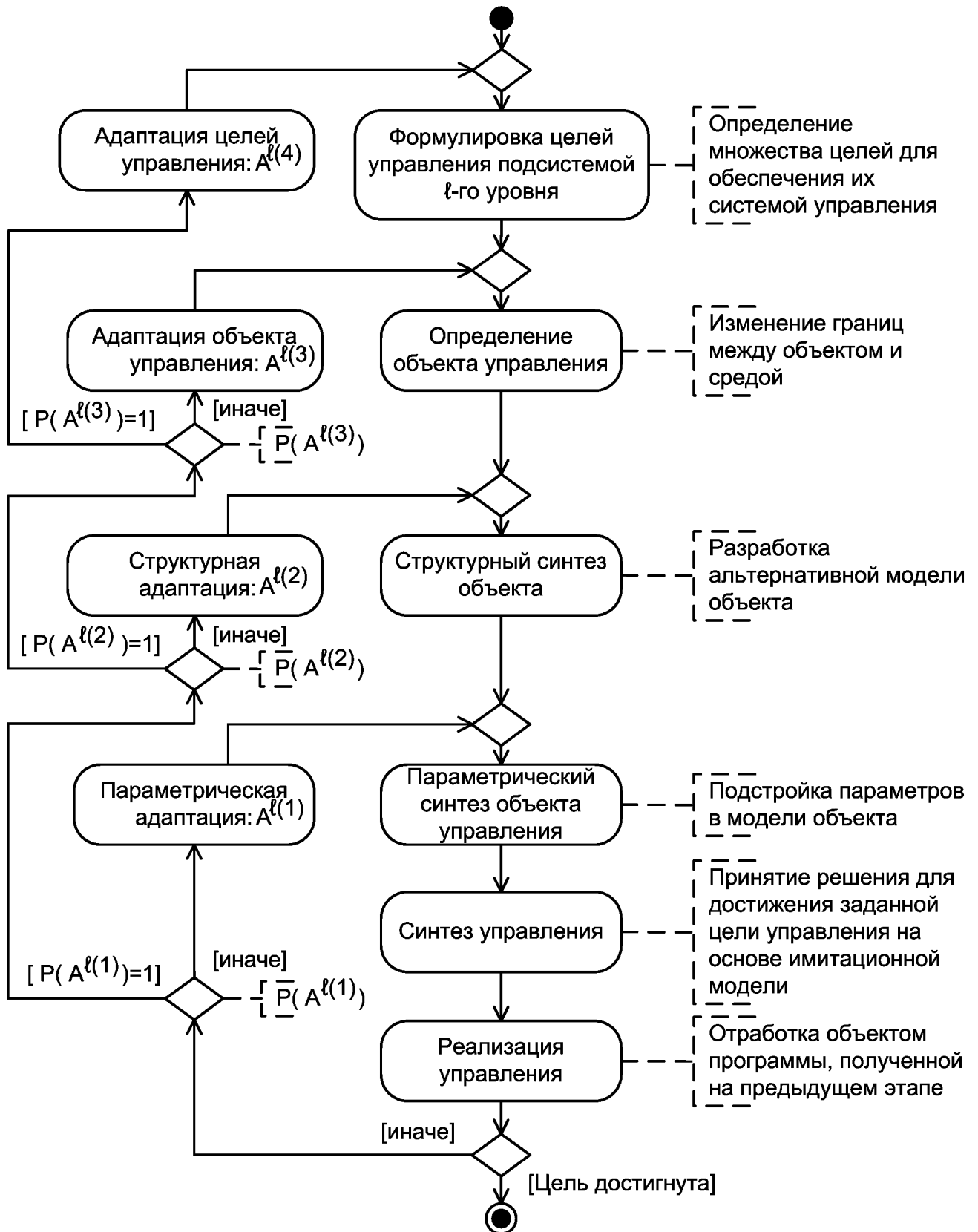
*Этап 4. Параметрический синтез модели* связан с определением значений ее параметров при выбранной структуре. Для определения значений параметров модели необходимо иметь информацию о характере изменения неуправляемых, управляемых параметров и состоянии объекта.

*Этап 5. Синтез управления* связан с принятием решения о механизме управления, позволяющим достигнуть выбранной цели управления в объекте. Это решение получают с помощью имеющейся модели объекта и преследуют заданную цель. При этом основываются также на полученной информации о состоянии внешней среды и объекта при выделенных ресурсах управления, которые необходимо учитывать как ограничения, накладываемые на управление в связи с особенностями объекта и возможностями системы управления. Полученное управление должно быть оптимальным в соответствии с критериями эффективности (целями) управления и представлять собой программу изменения управляемых параметров во времени.

*Этап 6. Реализация управления* требует разработки последовательности действий (мероприятий) для отработки объектом программы, полученной на предыдущем этапе. Такой процесс для неактивных элементов решается методом теории следящих систем, обрабатывающих заданную программу. Трудности анализа поведения активных элементов должны быть преодолены на стадии синтеза модели объекта управления.

*Этап 7. Адаптация* – коррекция всей системы управления, т.е. всех этапов управления. Адаптация системы управления является процессом приспособления к свойствам объекта и внешней среды (см. рисунок 2.2), при этом возможны четыре иерархических уровня адаптации модели системы управления [78], представленные на рисунке 2.3:

1) *Параметрическая адаптация*  $A^{\ell(1)}$ , связанная с коррекцией, подстройкой значений параметров в модели. Такая коррекция возникает ввиду изменения характеристик управляемого объекта.



**Рисунок 2.3 – Этапы поиска рациональных решений при эксплуатации имитационной модели в информационной системе предприятия**

2) Структурная адаптация  $A^{\ell(2)}$ , возникающая при несовпадении структур модели и объекта, устраняется, например, переходом к другой альтернатив-

ной модели. При этом альтернативные варианты модели могут различаться числом и характером входных и выходных параметров модели, вариантами декомпозиции и структурой компонентов модели.

3) *Адаптация объекта*  $A^{\ell(3)}$ , связанная с пересмотром границ, разделяющих исследуемый объект и внешнюю среду. Рассматриваются разные варианты расширения объекта (приводящие к повышению управляемости, но требующие дополнительных ресурсов для управления) и выбирается лучший из них, что и составляет основу адаптации объекта.

4) *Адаптация целей управления*  $A^{\ell(4)}$ . Определяются новые цели, достижение которых обеспечивается системой управления (см. рисунок 2.3, где  $P(A^{\ell(i)})$  – соответствующий предикат для  $A^{\ell(i)}$ ,  $i = 1, \dots, 4$ ).

Все четыре уровня адаптации системы управления предназначены для решения задачи достижения системой поставленных целей. При этом следует учитывать то обстоятельство, что верхние уровни адаптации имеют более высокие постоянные времени, т. е. работают значительно медленнее. Это означает, что верхние уровни адаптации используют лишь в том случае, когда нижние не отслеживают изменения в объекте с требуемой эффективностью.

Процедура поиска рациональных решений основана на исследовании базовой имитационной модели с формированием показателей функционирования предприятия как многоуровневой иерархической системы. При этом иерархическая структура предприятия обладает следующими особенностями, которые учитываются при проведении исследований:

- для высших уровней иерархии системы имеет место лучшее понимание целей, назначения, функций, возможностей системы, но и одновременно высший уровень неполноты, неопределенности, неточности, противоречивости исходной информации для принятия решения;

- при последовательном переходе к нижним уровням иерархии уменьшается уровень неопределенности и повышается возможность детального конкретного описания структуры подсистем и конкретизация различных видов деятельности и практических задач [40].

Специалистам промышленного предприятия в своей практической деятельности приходится принимать много технических решений, требующих научного обоснования. Сюда относятся, например, выбор структуры технологического процесса и оптимизация различных параметров технологических операций. При этом критериями оптимизации могут быть характеристики как самого технологического процесса (производительность, стабильность и др.), так и изготавливаемых изделий (себестоимость, точность, эксплуатационные показатели и т.д.). На этапе использования и исследования имитационной модели (объекта машиностроения, технологического процесса химического производства) возникает необходимость в изучении ее свойств, что выражается в нахождении значимых параметров модели, обеспечивающих эти свойства.

При решении класса подобных задач часто используются методы оптимизации, в основу которых положен случайный поиск [78, 162, 170, 181]. Важными признаками данного класса алгоритмов являются универсальность и простота применения к конкретной задаче [15, 141].

Решение задач оптимизации сложных систем для достижения заданного уровня затруднено тем, что трудно получить аналитическую зависимость между входными параметрами и откликами модели. Этим объясняется использование имитационных моделей для формализации таких систем. Методы оптимизации, основанные на случайном поиске обладают некоторыми недостатками: повышенные требования к вычислительным ресурсам по сравнению с методами, специально спроектированным для решения определенных проблем, отсутствие гарантии нахождения глобального оптимума, сложность реализации, – однако часто являются основным способом решения задачи.

### **2.3 Формализация процесса оптимизации системы управления предприятием на основе имитационной модели**

Пусть имитационная модель  $M$  задана набором [141]

$$M = \langle X_M, Y_M, A_M, Def_M \rangle, \quad (2.17)$$

где  $X_M$  – множество входных параметров модели,

$$X_M = \{x_{Mi} \mid x_{Mi} \in Def_M, i = 1, \dots, NX_M\}$$

и пространство значений входных параметров модели

$$SX_M = \{\overline{sx_M}\}, \overline{sx_M} = (x_{M1}, \dots, x_{Mn}), n = NX_M; \quad (2.18)$$

$Y_M$  – множество откликов модели,

$$Y_M = \{y_{Mi} \mid y_{Mi} \in R, i = 1, \dots, NY_M\}$$

и пространство значений откликов модели

$$SY_M = \{\overline{sy_M}\}, \overline{sy_M} = (y_{M1}, \dots, y_{Mn}), n = NY_M; \quad (2.19)$$

$A_M$  – алгоритм работы имитационной модели,

$$A_M: SX_M \rightarrow SY_M, \forall \overline{sx_M} \in SX_M \exists \overline{sy_M} \in SY_M \mid \overline{sy_M} = A_M(\overline{sx_M}). \quad (2.20)$$

В общем случае целевая функция задается набором

$$CF = \langle X_{CF}, Def_{CF}, F_{CF} \rangle, \quad (2.21)$$

где  $X_{CF}$  – множество параметров целевой функции,

$$X_{CF} = \{x_{CFi} \mid x_{CFi} \in Def_{CF}, i = 1, \dots, NX_{CF}\}$$

и пространство значений параметров целевой функции

$$SX_{CF} = \{\overline{sx_{CF}} \mid \overline{sx_{CF}} = (x_{CF1}, \dots, x_{CFn}), n = NX_{CF}\}. \quad (2.22)$$

Класс отображений целевой функции задается следующим образом:

$$CCF = \{F_{CF} \mid SX_{CF} \rightarrow R, \forall \overline{sx_{CF}} \in SX_{CF} \exists f \in R, f = F_{CF}(\overline{sx_{CF}})\}, \quad (2.23)$$

причем  $\forall \overline{sx_{CF1}}, \overline{sx_{CF2}} \in SX_{CF}$ , если решение  $\overline{sx_{CF1}}$  лучше в понятиях предметной области решения  $\overline{sx_{CF2}}$ , то  $F_{CF}(\overline{sx_{CF1}}) > F_{CF}(\overline{sx_{CF2}})$ .

Пусть  $P_{AO}$  – множество параметров, используемых для настройки алгорит-



ма оптимизации;  $X_{AO}$  – множество оптимизируемых параметров,

$$X_{AO} = \{x_{AOi}\}, x_{AOi} \in Def_{AO}, i = 1, \dots, NX_{AO}$$

и пространство поиска (решений) алгоритма

$$SX_{AO} = \overline{\{sx_{AO}\}}, \overline{sx_{AO}} = (x_{AO1}, \dots, x_{AO_n}), n = NX_{AO}. \quad (2.24)$$

Тогда целевая функция алгоритма оптимизации задается набором:

$$CF_{AO} = \langle X_{AO}, Def_{AO}, F_{AO} \rangle. \quad (2.25)$$

Таким образом, алгоритм оптимизации при заданном множестве параметров настройки  $P_{AO}$  и погрешности получаемого решения  $\varepsilon_{AO}$  задается как отображение:

$$A_{AO}: SX_{AO} \xrightarrow{P_{AO}} SX_{AO}^* / SX_{AO}^* \subseteq SX_{AO}, \forall \overline{sx_{AO}^*} \in SX_{AO}^*, \cdot$$

$$\max_{SX_{AO}} (F_{AO} - F_{AO}(\overline{sx_{AO}^*})) \leq \varepsilon_{AO} \quad (2.26)$$

Задача оптимизации задается следующим набором:

$$ZO = \langle M, X_{AO}, Def_{AO}, \overline{x_G}, CF_{ZO}, \varepsilon_{AO}, P_{AO} \rangle, \quad (2.27)$$

где  $X_{AO} \subseteq X_M, \forall (x_M \stackrel{def}{=} x_{AO}) \in X_{AO}, x_{AO} \in Def_{AO}(x_{AO}) \subseteq Def_M(x_M)$ ;

$\overline{x_G}$  – вектор условий, задающий значения постоянных параметров модели

$M$ ,

$$\overline{x_G} = (x_{G1}, \dots, x_{Gn}), x_{Gi} \in X_M \mid X_{AO} \stackrel{def}{=} X_G, i = 1, \dots, n,$$

$$n = NX_M - NX_{AO} \stackrel{def}{=} NX_G \quad (2.28)$$

$CF_{ZO}$  – целевая функция задачи оптимизации,

$$CF_{ZO} = \langle X_{ZO}, Def_{ZO}, F_{ZO} \rangle, \quad (2.29)$$

где  $X_{ZO}$  – множество параметров целевой функции в задаче оптимизации,

$$X_{ZO} = \{x_{ZOi}\} \subseteq (X_M \cup Y_M), x_{ZOi} \in Def_{ZO} \subseteq (Def_M \cup Def(Y_M)), i = 1, \dots, NX_{ZO}.$$

Решение  $SX_M^*$  задачи  $ZO$  оптимизации модели  $M$  с использованием алгоритма оптимизации  $A_{AO}$ , настроенного в соответствии с множеством  $P_{AO}$ , задается в виде

$$\begin{aligned} SX_M^* &= A_{AO}(SX_{ZO}) \mid SX_M^* \subseteq SX_M, \forall \overline{sx_M^*} \in SX_M^*, \\ \max_{SX_{ZO}} (F_{ZO} - \Phi_{ZO}^*(\overline{sx_M^*})) &\leq \varepsilon_{AO} \end{aligned} \quad (2.30)$$

при оптимальном значении целевой функции

$$\Phi_{ZO}^*(\overline{sx_M^*}) = \left[ A_M(\overline{sx_M^*}) = \overline{sy_M^*}, \overline{sx_{ZO}^*} = \underset{SX_{ZO}}{pr} (\overline{sx_M^*}, \overline{x_G}, \overline{sy_M^*}) \right] = F_{ZO}(\overline{sx_{ZO}^*}), \quad (2.31)$$

где  $\underset{SX}{pr} \overline{sx}$  - проекция вектора  $\overline{sx}$  на пространство  $SX$  [141].

Следует отметить, что для нахождения оптимального решения используется алгоритм оптимизации, в основу которого положен алгоритм случайного поиска.

## **2.4 Методика решения задачи рационального выбора алгоритма оптимизации систем управления предприятием**

### **2.4.1 Формальное описание задачи рационального выбора структуры и параметров алгоритма оптимизации**

Условно можно считать, что реализация алгоритма случайного поиска  $A_S$  определяется его структурой и параметрами:

$$A_S = \langle St, C \rangle, \quad (2.32)$$

где  $St$  – структура реализации алгоритма  $A_S$ ;

$C$  – параметры структуры  $St$  реализации алгоритма  $A_S$ ,  $C = (c_1, c_2, \dots, c_k)$ .

При проектировании реализации алгоритма случайного поиска выделены два этапа [16]:

– структурный синтез алгоритма, целью которого является определение структуры  $St$  реализации алгоритма;

– параметрический синтез алгоритма, целью которого является определение параметров  $C$  реализации алгоритма.

Обычно алгоритмы случайного поиска не имеют строго фиксированной структуры, а постоянно претерпевают модификации с целью улучшения их работы. Структура реализации алгоритма случайного поиска определяется типом используемых операторов и отношений между ними. Структурную реализацию генетического алгоритма определяют, например, операторы репродукции, к которым относятся: селекция на основе рулетки, заданной шкалы, элитная, турнирная. Существует множество операторов кроссинговера: простой, двухточечный, упорядоченный, частично-соответствующий, циклический, универсальный. К операторам мутации относятся: двухточечный, строительные блоки. Известны операторы транслокации, транспозиции, сегрегации, удаления, вставки, редукции, рекомбинации. Поэтому в ходе структурного синтеза исследователь получает не одну структуру реализации алгоритма, а некоторое их множество  $\{St_i\}$ . Проведя параметрический синтез для выделенных структур и определив множество параметров  $\{C_i\}$ , в соответствии с (2.32) получают некоторое множество  $A$  реализаций алгоритма случайного поиска  $A_S$ :

$$A = \{A_{Si}\}, A_{Si} = \langle S_{ii}, C_i \rangle, i = 1, \dots, NA, \quad (2.33)$$

где  $NA$  – число реализаций алгоритма  $A_S$ , выделенных при проектировании алгоритма.

Целевую функцию алгоритма случайного поиска можно задать следующим набором [16]:

$$CF_{AS} = \langle X_{AS}, Def_{AS}, F_{AS} \in CFF \rangle, \quad (2.34)$$

где  $X_{AS}$  – множество оптимизируемых параметров,

$$X_{AS} = \{x_{ASi}\}, x_{ASi} \in Def_{AS}, i = 1, \dots, NX_{AS};$$

$F_{AS} \in CFF$  – отображение из класса, заданного следующим образом:

$$CFF = \{F_{AS} \mid SX_{AS} \rightarrow R, \forall \overline{sx_{AS}} \in SX_{AS} \exists f \in R, f = F_{AS}(\overline{sx_{AS}})\}, \quad (2.35)$$

где  $SX_{AS}$  – пространство значений оптимизируемых параметров,

$$SX_{AS} = \{\overline{sx_{AS}}\}, \quad \overline{sx_{AS}} = (x_{AS1}, \dots, x_{ASn}), \quad n = NX_{AS},$$

причем  $\forall \overline{sx_{AS1}}, \overline{sx_{AS2}} \in SX_{AS}$ , если решение  $\overline{sx_{AS1}}$  лучше решения  $\overline{sx_{AS2}}$  (в понятиях предметной области):  $\overline{sx_{AS1}} \succ \overline{sx_{AS2}} \Rightarrow F_{AS}(\overline{sx_{AS1}}) > F_{AS}(\overline{sx_{AS2}})$ .

Тогда реализацию алгоритма  $A_S$  при заданной структуре  $St$ , множестве параметров  $C$  и погрешности получаемого решения  $\varepsilon_{AS}$  можно задать как отображение [16]:

$$A_{AS}: SX_{AS} \xrightarrow{\langle St, C \rangle} SX_{AS}^* \mid SX_{AS}^* \subseteq SX_{AS}, \quad \forall \overline{sx_{AS}^*} \in SX_{AS}^*,$$

$$\max_{SX_{AS}} (F_{AS} - F_{AS}(\overline{sx_{AS}^*})) \leq \varepsilon_{AS}. \quad (2.36)$$

Указанные выше информационные потоки более детально представлены в п. 5.1 при описании программной системы «Оптимизатор» для реализации метода поиска рациональных решений. При этом в качестве основного алгоритма оптимизации используется один из алгоритмов случайного поиска – генетический алгоритм [47].

#### 2.4.2 Решение задачи рационального выбора структуры и параметров алгоритма оптимизации

Пусть в ходе анализа алгоритма  $A_S$  (см. выражение (2.36)) выделено множество целевых функций  $CF$ , которые удовлетворяют выражениям (2.34)–(2.35) и на которых необходимо проверить работоспособность алгоритма  $A_S$ :

$$CF = \{CF_{ASi}\}, \quad i = 1, \dots, NCF. \quad (2.37)$$

Для каждой функции  $CF_{ASi}$  из множества  $CF$  проводится предварительное исследование  $RS_A^0$  с целью определения такой реализации алгоритма  $A_S$  из мно-

жества  $A$  (см. выражение (2.33)), которая находит лучшее решение целевой функции:

$$\begin{aligned} (CF_{AS_i}, \{A_{S_i}^*\}) | A_{S_i}^* = \langle St_i^*, C_i^* \rangle, A_{S_i}^* \in A, \forall A_{S_j} \in A; \\ j \neq i \rightarrow CF_{AS_i}(SX_{AS_j}^*) < CF_{AS_i}(SX_{AS_i}^*), \end{aligned} \quad (2.38)$$

где  $SX_{AS}^*$  – решения, найденные реализацией алгоритма в соответствии с выражением (2.36).

Используя статистические методы группировки, например кластерный анализ, множество функций  $CF$  разбивается на классы функций [123], для которых реализация алгоритма  $A_S^*$  дает наилучшее решение:

$$\begin{aligned} \Phi_1 = \langle (CF_{AS_1}, CF_{AS_2}, \dots, CF_{AS_{k_1}}), A_{S_1}^* \rangle; \\ \Phi_2 = \langle (CF_{AS_{k_1+1}}, CF_{AS_{k_1+2}}, \dots, CF_{AS_{k_2}}), A_{S_2}^* \rangle; \\ \dots \\ \Phi_{N\Phi} = \langle (CF_{AS_{k_{N\Phi-1}+1}}, CF_{AS_{k_{N\Phi-1}+2}}, \dots, CF_{AS_{k_{N\Phi}}}), A_{S_{N\Phi}}^* \rangle, \\ k_{N\Phi} = /CF / = NCF. \end{aligned} \quad (2.39)$$

Пусть на вход алгоритма случайного поиска поступает очередная  $(NCF+1)$  целевая функция  $CF_{AS(NCF+1)}$  [14]. Используя некоторый оператор  $R_X$  принадлежности, для произвольной функции определяется класс функций  $\Phi_i$ , к которому она принадлежит, или новый  $\Phi_{(N\Phi+1)}$ , если эта функция не принадлежит ни одному из классов, т. е.

$$R_X(CF_{AS}) = \begin{cases} i / CF_{AS} \in \Phi_i; \\ N\Phi + 1 / CF_{AS} \notin \Phi_i, i = 1, \dots, N\Phi. \end{cases} \quad (2.40)$$

Данный оператор  $R_X$  может быть построен на основе квалификационных признаков, которые можно будет получить после анализа множества классов функций  $\Phi_i$ . Также в основу оператора  $R_X$  может быть положена процедура исследования, подобная той, которая использовалась для получения пар (2.39).

Если оператор  $R_X$  находит класс функций  $\Phi_i$ , для функции  $CF_{AS(NCF+1)}$  используется соответствующая реализация алгоритма  $A_{Si}^*$  с уже известной структурой  $St_i^*$  и параметрами  $C_i^*$  [128]. В противном случае для функции  $CF_{AS(NCF+1)}$  необходимо провести предварительное исследование  $RS_A^0$  с целью определения из множества  $A$  такой реализации алгоритма  $A_{S(N\Phi+1)}^*$  (см. выражение (2.33)), которая находит лучшее решение целевой функции  $CF_{AS(NCF+1)}$ .

Полученная реализация алгоритма случайного поиска используется для нахождения параметров исследуемой системы исходя из целевой функции  $CF_{AS(NCF+1)}$ .

В случае, когда оператор  $R_X$  (см. выражение (2.40)) не определил для новой функции  $CF_{AS(NCF+1)}$  класс функций (см. выражение (2.39)), необходимо проводить исследование  $RS_{AS}$ . Возникновение новых задач, появление новых ситуаций – основные причины, вызывающие необходимость использовать адаптацию алгоритмов поиска, которая позволит автоматизировать исследование  $RS_{AS}$ .

Предлагается следующая последовательность шагов применения адаптации реализации алгоритма случайного поиска к решаемой задаче [143].

*Шаг 1.* В ходе исследования алгоритма случайного поиска  $RS_A^0$  определяется множество реализаций  $\{A_{Si}^*\}$ , удовлетворяющих (2.38). Для увеличения пространства возможных реализаций алгоритма, которые будут использоваться при адаптации, предлагается расширить данное множество до следующего:

$$(CF_{ASi}, \{A_{Si}^*\}) | A_{Si}^* = \langle St_i^*, C_i^* \rangle, A_{Si}^* \in A, \forall A_{Sj} \in A, \\ j \neq i \rightarrow |CF_{ASi}(SX_{ASj}^*) - CF_{ASi}(SX_{ASi}^*)| < \varepsilon_{AS^*}. \quad (2.41)$$

где  $\varepsilon_{AS^*}$  – максимальное отклонение от лучшего решения, которое может быть найдено любой реализацией алгоритма из множества  $A$ .

*Шаг 2.* Полученные структуры  $\{St_i^*\}$  являются исходными данными для *первого этапа адаптации*. На этом этапе параллельно запускаются реализации

алгоритмов с разными структурами  $St_i^*$ , требующими различные объемы ресурсов  $R$  для выполнения одной итерации поиска. Поэтому сравнение результатов их работы необходимо проводить при исчерпании одинакового количества ресурса  $R$ . В качестве ресурса  $R$  выбирается число запусков целевой функции, т. к. для ее вычисления обычно используются результаты работы имитационной модели. После сравнения результатов работы различных реализаций алгоритма для дальнейшей адаптации выбирается  $k$  ( $k < |\{ St_i^* \}|$ ) реализаций.

*Шаг 3. На втором этапе* создаются алгоритмы адаптации для выделенных параметров на полученной для них области значений. Схема адаптации параметра в простейшем случае заключается в выборе шага или коэффициента изменения значения параметра, с которым параллельно запускается новая нить алгоритма.

Критерием останова адаптации алгоритма может быть достижение состояния системы, удовлетворяющего исследователя, или максимальное количество итераций адаптации [143].

## **2.5 Построение целевой функции оценки эффективности взаимодействия компонентов предприятия**

### **2.5.1 Конструирование многокритериальной целевой функции производственно-экономической деятельности предприятия**

Задача исследования имитационной модели процессов функционирования промышленного предприятия состоит в том, чтобы найти рациональное решение, позволяющее получить эффект для системы в целом, а не только его отдельным подразделениям. Производственно-экономическая деятельность предприятия оценивается многими показателями [29, 56, 84, 85], основные из которых представлены в таблице 2.1. Задачу поиска решений при анализе многих показателей относят, в этом случае, к классу многокритериальных.

**Таблица 2.1 – Основные показатели деятельности промышленного предприятия (система показателей)**

Наименование показателя	Формула расчета
1	2
<b>Y<sub>1</sub>: Показатели общей оценки финансового состояния предприятия</b>	
<b>Y<sub>1.1</sub>: Оценка неудовлетворительной структуры бухгалтерского баланса</b>	
Коэффициент текущей ликвидности K <sub>1</sub> на конец отчетного периода	$K_1 = \frac{OA}{KO}$
Коэффициент обеспеченности собственными оборотными средствами K <sub>2</sub> на конец отчетного периода	$K_2 = \frac{СК - ВА}{OA}$
<b>Y<sub>1.2</sub>: Определение устойчивой неплатежеспособности предприятия</b>	
Коэффициент обеспеченности финансовых обязательств активами	$K_3 = \frac{ДЗ + КО}{СА}$
<b>Y<sub>2</sub>: Показатели оценки устойчивости организации</b>	
<b>Y<sub>2.1</sub>: Показатели структуры капитала</b>	
Коэффициент собственности	$K_C = \frac{СК + ДЗ}{СА}$
<b>Y<sub>2.2</sub>: Показатели ликвидности</b>	
Коэффициент абсолютной ликвидности	$K_{АЛ} = \frac{ДС + KB}{КО}$
Коэффициент критической (быстрой) ликвидности	$K_{БЛ} = \frac{OA - ПЗ}{КО}$
<b>Y<sub>3</sub>: Показатели роста и развития предприятия</b>	
Темп роста продаж (нетто)	$T_{РП} = \frac{BP1}{BP0}$
Темп роста себестоимости	$T_{РС} = \frac{(СП + КР + УР)1}{(СП + КР + УР)0}$
Темп роста прибыли от реализации продукции	$T_{РПР} = \frac{ПВПН1}{ПВПН0}$
Темп роста чистой прибыли	$T_{РЧП} = \frac{ЧП1}{ЧП0}$
Темп роста стоимости активов	$T_{РСА} = \frac{СА1}{СА0}$
<b>Y<sub>4</sub>: Показатели рентабельности</b>	
Рентабельность продукции по затратам	$R_{ПЗ} = \frac{ПВПН}{СП + КР + УР}$
Рентабельность выручки от продаж	$R_{ВП} = \frac{ПВПН}{BP}$
Рентабельность активов предприятия	$R_A = \frac{ПВПН}{ср.СА}$



## Продолжение таблицы 2.1

1	2
Рентабельность основного капитала	$R_{OK} = \frac{ПВН}{ср. OC}$
Рентабельность собственного капитала	$R_{СК} = \frac{ЧП}{ср. OC}$
<b>У<sub>5</sub>: Показатели эффективности использования основных производственных фондов</b>	
Коэффициент фондоотдачи	$K_{\Phi O} = \frac{ВР}{ср. OC}$
Период окупаемости собственного капитала	$K_{ОСК} = \frac{ср. СК}{ЧП}$
<b>У<sub>6</sub>: Показатели эффективности использования оборотных активов</b>	
Коэффициент оборачиваемости оборотных активов	$K_{OOA} = \frac{ВР}{360ср. OA}$
Продолжительность оборота оборотных активов, дн.	$\Pi_{OOA} = \frac{360ср. OA}{ВР}$
Продолжительность оборота производственных запасов, дн.	$\Pi_{ОПЗ} = \frac{360ср. ПЗ}{ВР}$
Продолжительность оборота незавершенного производства, дн.	$\Pi_{ОНП} = \frac{360ср. НП}{ВР}$
Продолжительность оборота готовой продукции, дн.	$\Pi_{ОГП} = \frac{360ср. ГП}{ВР}$
Продолжительность оборота дебиторской задолженности, дн.	$\Pi_{ОДЗ} = \frac{360ср. ЗД}{ВР}$
Условные обозначения в формулах: ОА, ВА, СА – оборотные, внеоборотные и совокупные активы, соответственно; СК, ДЗ, КО – собственный капитал, долгосрочные заимствования, краткосрочные обязательства; ДС, КВ, ПЗ – денежные средства, краткосрочные финансовые вложения, стоимость производственных запасов; ВР, ВР1, ВР0 – выручка (нетто) от реализации продукции, в том числе за отчетный и базисный (предшествующий) год; СП, КР, УР – себестоимость продаж продукции, коммерческие расходы, управленческие расходы; ПВПН, ПВН, ЧП – прибыль до выплаты процентов и налогов, прибыль до выплаты налогов, чистая прибыль; ср. ОА, ср. ПЗ, ср. НП, ср. ГП, ср. ЗД – средние запасы оборотных активов, производственных запасов, незавершенного производства, готовой продукции на складе, дебиторской задолженности.	

Основные показатели процессов функционирования предприятия (см. таблицу 2.1) в формальном виде представлены ниже [119]:

– Показатели  $Y_1$  и  $Y_2$  определяют, соответственно, финансовое состояние и устойчивость предприятия:

$$\begin{aligned} Y_1 &= Y_{1.1} \cup Y_{1.2} / Y_{1.1} = \{K_1, K_2\}; Y_{1.2} = \{K_3\}; \\ Y_2 &= Y_{2.1} \cup Y_{2.2} / Y_{2.1} = \{K_C\}; Y_{2.2} = \{K_{АЛ}, K_{БЛ}\}. \end{aligned} \quad (2.42)$$

– Показатели  $Y_3$  и  $Y_4$  определяют, соответственно, развитие и рентабельность предприятия:

$$\begin{aligned} Y_3 &= \{T_{РП}, T_{РС}, T_{ТПР}, T_{РЧП}, T_{РСА}\}; \\ Y_4 &= \{R_{ПЗ}, R_{ВП}, R_A, R_{ОК}, R_{СК}\}. \end{aligned} \quad (2.43)$$

– Показатели  $Y_5$  и  $Y_6$  определяют, соответственно, эффективность основных фондов и оборотных активов (см. таблицу 2.1):

$$\begin{aligned} Y_5 &= \{K_{ФО}, K_{ОСК}\}; \\ Y_6 &= \{K_{ООА}, П_{ООА}, П_{ОПЗ}, П_{ОНП}, П_{ОГП}, П_{ОДЗ}\}. \end{aligned} \quad (2.44)$$

Многокритериальная целевая функция  $CF$  производственно-экономической деятельности предприятия в общем случае имеет вид:

$$CF = F(Y_1, Y_2, Y_3, Y_4, Y_5, Y_6) \quad (2.45)$$

или (см. таблицу 2.1) с учетом (2.42) – (2.44):

$$CF = F(\langle K_1, K_2 \rangle, \langle K_3 \rangle, \langle K_C \rangle, \langle K_{АЛ}, K_{БЛ} \rangle, \langle T_{РП}, T_{РС}, T_{ТПР}, T_{РЧП}, T_{РСА} \rangle, \langle R_{ПЗ}, R_{ВП}, R_A, R_{ОК}, R_{СК} \rangle, \langle K_{ФО}, K_{ОСК} \rangle, \langle K_{ООА}, П_{ООА}, П_{ОПЗ}, П_{ОНП}, П_{ОГП}, П_{ОДЗ} \rangle).$$

В условиях многокритериальности интегральная целевая функция (ИЦФ) представляется, в частности, линейной сверткой частных критериев в один [98] в следующем виде:

$$\begin{aligned}
 CF(Y_1, Y_2, \dots, Y_m) &= w_1 CF_1(Y_1) + w_2 CF_2(Y_2) + \dots + w_m CF_m(Y_m) = \\
 &= \sum_{i=1}^m w_i CF(Y_i),
 \end{aligned}
 \tag{2.46}$$

где  $w_1, w_2, \dots, w_m$  – весовые коэффициенты, оценивающие относительную значимость частных критериев ( $i = 1, 2, \dots, m$ ) ИЦФ. Для весовых коэффициентов  $w_i, i = 1, 2, \dots, m$ , выполняется следующее ограничение для удобства вычислений и анализа вариантов решений:

$$\sum_{i=1}^m w_i = 1.
 \tag{2.47}$$

Применение ИЦФ в виде (2.46) является корректным при следующих условиях [152]:

- линейная форма суммы критериев ИЦФ не накладывает ограничений на вид функции отдельного критерия  $w_1 CF_1(Y_1), w_2 CF_2(Y_2), \dots, w_m CF_m(Y_m)$ , которая в общем случае может быть нелинейной;
- критерии ИЦФ следует выбирать взаимно независимыми, чтобы эффективность одного критерия не влияла на эффективность других критериев;
- критерии ИЦФ обладают разной размерностью и для оценивания альтернативных вариантов необходимо выполнить нормирование критериев (делением абсолютных значений  $CF_i(Y_i)$  на их максимальное значение  $CF_i(Y_i)_{max}$ ) [54].

В информационной теории используют понятие энтропии как меру неопределенности [48]. Рассматривают энтропию событий с вероятностями  $p_1, p_2, \dots, p_n$ . Энтропия  $S(p_1, p_2, \dots, p_n)$  принимает максимальное значение, когда неопределенность появления событий максимальна, т. е.  $p_i = 1/n$ .

Идея энтропии особенно полезна при исследовании контрастов между наборами данных. Эти наборы данных могут быть представлены как альтернативные решения в матрице решений (МР), где каждая альтернатива оценена в терминах набора критериев полезности решения [152].

Запись МР с  $l$  альтернативными решениями, оцениваемых значениями  $m$  критериев имеет вид:

$$\text{MP} = \begin{bmatrix} y_{11} & y_{12} & \dots & y_{1m} \\ y_{21} & y_{22} & \dots & y_{2m} \\ \dots & \dots & \dots & \dots \\ y_{l1} & y_{l2} & \dots & y_{lm} \end{bmatrix}. \quad (2.48)$$

Альтернативные решения по Р. Аль-Аомару (R. Al-Aomar, 2002) [152] могут быть представлены распределением вероятностей  $p_{kj}$ , где  $k=1, 2, \dots, l$  и  $j=1, 2, \dots, m$ . Каждое значение  $p_{kj}$  содержит информацию, которая может быть измерена значением энтропии. При этом  $p_{kj}$  определяется по формуле

$$p_{kj} = y_{kj} / \sum_{k=1}^l y_k, \forall k, l. \quad (2.49)$$

где  $y_k$  – значение ИЦФ для  $k$ -го альтернативного решения.

Матрица вероятностей  $p_{kj}$  сформируется следующим образом:

$$\| p_{kj} \| = \begin{bmatrix} p_{11} & p_{12} & \dots & p_{1m} \\ p_{21} & p_{22} & \dots & p_{2m} \\ \dots & \dots & \dots & \dots \\ p_{l1} & p_{l2} & \dots & p_{lm} \end{bmatrix}. \quad (2.50)$$

Энтропия  $S_j$  для набора значений критериев  $j$  в  $l$  альтернативных решениях определяется по формуле

$$S_j = -K \sum_{k=1}^l p_{kj} \ln p_{kj}, \forall j, \quad (2.51)$$

где  $K$  представлено постоянной величиной со значением  $1/\ln l$  в  $S_{max}$ , которое гарантирует, что  $0 \leq S_j \leq 1$ .

При этом степень разнообразия  $d_j$  значений  $j$ -го критерия в  $l$  альтернативных решениях определяется по формуле

$$d_j = 1 - S_j, \forall j. \quad (2.52)$$

Тогда вес важности каждого  $j$ -го критерия в ИЦФ определяется по формуле:

$$w_j = d_j / \sum_{j=1}^m d_j, \forall j. \quad (2.53)$$

Если экспертом определены субъективные оценки  $(\lambda_1, \lambda_2, \dots, \lambda_m)$  каждого из  $m$  критериев, то, используя оценки весов  $(w_1, w_2, \dots, w_m)$ , получают адаптированные веса по формуле

$$w_j^0 = w_j \lambda_j / \sum_{j=1}^m w_j \lambda_j, \forall j. \quad (2.54)$$

Таким образом, метод энтропии позволяет оценивать относительный вес значений критериев в линейной ИЦФ. При использовании генетического алгоритма [143] в качестве метода оптимизации параметров в альтернативных решениях обеспечивается наличие обобщенного критерия для различения проектных альтернатив.

### 2.5.2 Координация многокритериальных задач в иерархической структуре управления предприятием

Деятельность промышленных предприятий М. Месарович и др. представляют многоуровневой иерархической структурой, в которой вводится понятие координации [60].

Координация подсистем означает такое воздействие на подсистемы, которое заставляет действовать их согласованно. Вводятся два понятия координируемости на примере двухуровневой системы. Первое – это координируемость по отношению к задаче вышестоящей системы, второе – координируемость по отношению к решаемой в настоящий момент глобальной задаче.

Пусть определен предикат [131]

$$\forall(x, D) P(x, D) \equiv x \text{ есть решение } D, \quad (2.55)$$

где  $D$  – произвольная решаемая задача. Следовательно, предикат  $P(x, D)$  является истинным тогда и только тогда, когда  $D$  – решаемая задача, а  $x$  – одно из ее решений.

Пусть  $D_0$  – конкретная задача, решаемая вышестоящей системой, и каждый координирующий сигнал  $g \in G$  уточняет задачу  $D(g)$ , которую будет решать координатор на  $\ell$ -уровне. Задачи, решаемые нижестоящими элементами, координируемы тогда и только тогда, когда истинным является следующий предикат:

$$\exists g \exists x P(x, D(g)) \wedge P(g, D_0), \quad (2.56)$$

т.е. координируемость требует, чтобы задача  $D_0$  имела решение  $g$  и для этого координирующего воздействия множество  $D(g)$  задач нижестоящих элементов также имело решение  $x$ .

Ни один из координаторов внутри иерархии не облечен специально полномочиями решать глобальную задачу и тем самым преследовать общую (глобальную) цель. Задачи, которые решаются нижестоящими решаемыми элементами, координируемы относительно глобальной задачи  $D^*$ , если справедливо следующее предложение [131]:

$$\exists g \exists x P(x, D(g)) \wedge P(g, D^*). \quad (2.57)$$

М. Месарович ввел постулат совместимости [60], в соответствии с которым для совместимости (согласования) решаемых задач, а тем самым и целей внутри двухуровневой системы, координация задач, решаемых нижестоящими элементами относительно задачи вышестоящего координатора, должна быть соответствующим образом связана с подлежащей решению глобальной задачей.

Высокий уровень абстракции введенного предложения не дает конкретных рекомендаций для применения его на практике. Были сделаны попытки разработки подобных рекомендаций [6]. Например, А. Е. Алтуниным и др. на основе теории нечетких множеств разработаны рекомендации для иерархических технологических объектов [7]. Однако в целом задача осталась нерешенной.

Постановка задачи многокритериального выбора должна содержать три объекта: множество вариантов решений, векторный критерий эффективности и отношение предпочтения координатора. Решение такой задачи – множество выбираемых решений на основе векторного критерия эффективности и отношения предпочтения.

В рассматриваемой модели принцип Эджворта-Парето (принцип Парето) формулируется в виде утверждения о том, что множество выбираемых решений содержится в множестве Парето, т.е. каждое выбираемое решение является Парето-оптимальным [76].

На основании работы В. Д. Ногина [69] определен класс задач, для которых применение принципа Парето является обоснованным (приложение А).

Если рассматривается множество возможных решений  $X$  (вариантов), из которого следует осуществлять выбор, то подмножества возможных решений связаны следующими включениями:

$$\text{Sel } X \subset \text{Ndom } X \subset P_f(X) \subset X, \quad (2.58)$$

где  $\text{Sel } X$  – множество выбираемых решений;  $\text{Ndom } X$  – множество недоминируемых решений;  $P_f(X)$  – множество Парето-оптимальных решений.

В терминах множества  $Y = f(X)$  векторов  $f = f_1, f_2, \dots, f_m$  критериев качества решений  $X$  эти включения имеют вид:

$$\text{Sel } Y \subset \text{Ndom } Y \subset P(Y) \subset Y, \quad (2.59)$$

где  $\text{Sel } Y$  – множество выбираемых векторов оценок качества решений  $X$ ;  $\text{Ndom } Y$  – множество недоминируемых векторов оценок;  $P(Y)$  – множество Парето-оптимальных векторов.

Для построения множества  $P(Y)$  (и  $P_f(X)$ ) при конечном множестве возможных векторов  $Y$  рекомендуется применять алгоритм, представленный на рисунке 2.4 [69].

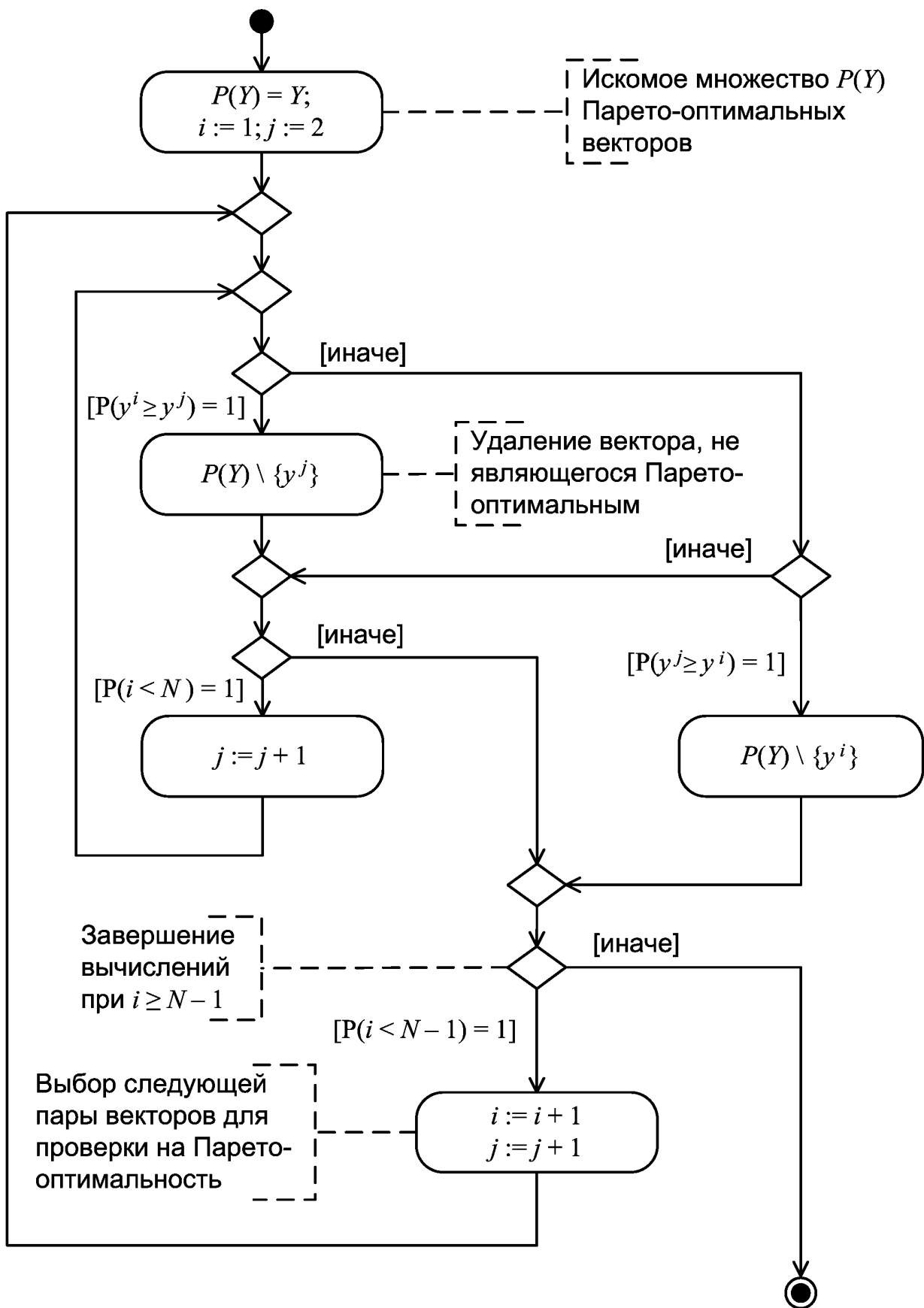


Рисунок 2.4 – Алгоритм построения множества Парето-оптимальных векторов  $P(Y)$



При выполнении аксиомы Парето (см приложение А) в терминах векторов для любой пары векторов  $y', y'' \in R^m$ , таких, что  $y' > y''$ , справедливо соотношение  $y' \succ y''$ , т. е.

$$y' > y'' \Rightarrow y' \succ y'' . \quad (2.60)$$

Полученные результаты инвариантны относительно уровней иерархической структуры предприятия. В этом случае имитационная модель позволяет оценивать решение ЛПР на  $\ell$ -уровне и вышестоящих уровнях, включая глобальную цель [122]:

$$x' \succ_X x'' \Rightarrow f(x') \geq f(x'' ) . \quad (2.61)$$

Вектор глобальной цели [131]:

$$(x')^\ell \xrightarrow{A_M} (y')^0; (x'')^\ell \xrightarrow{A_M} (y'')^0 , \quad (2.62)$$

где  $A_M$  – алгоритм модели. Соответственно,

$$(x' \succ_X x'')^\ell \Rightarrow (y' \geq y'')^0 . \quad (2.63)$$

Постулат совместимости М. Месаровича может быть представлен в виде

$$P[(x' \succ_{X_\ell} x'')^\ell \Rightarrow (y' \succ_{Y_\ell} y'')^\ell] \wedge (x' \succ_{X_\ell} x'')^\ell \Rightarrow (y' \succ_{Y_0} y'')^0] = 1 . \quad (2.64)$$

Получают два множества Парето-оптимальных векторов на  $\ell$ -уровне  $P(Y)^\ell$  и 0-уровне  $P(Y)^0$  глобальной цели. Понятно, что этим множествам в общем случае соответствуют два множества Парето-оптимальных решений на  $\ell$ -уровне:  $P_f(X)_\ell$  и  $P_f(X)_0$ .

Тогда возможны следующие варианты [131]:

а)  $P_f(X)_\ell \cap P_f(X)_0 = \emptyset$  (полная несовместимость);

б)  $P_f(X)_\ell \cap P_f(X)_0 \neq \emptyset$  (частичная совместимость);

в)  $P_f(X)_\ell \cap P_f(X)_0 = P_f(X)_\ell = P_f(X)_0$  (абсолютная совместимость при выполнении постулата совместимости).

Таким образом, появляется возможность количественной оценки координирующих действий в иерархической системе.

С. С. Макаров предлагает фундаментальный подход к задачам моделирования на примере медицинских учреждений, отличающийся формулировкой целей и задач моделирования; структурированием цели на основе системных методов с декомпозицией глобальной цели, позволяющий применять аналитические и эвристические методы для решения задач моделирования [58]. В алгоритме проектирования информационно-управляющей системы отмечается, что выбранные варианты оцениваются экспертами, результаты согласовываются и узакониваются с заказчиком. Применяют модели достижения согласия, на основании чего предложения проектировщиков и запросы заказчиков анализируются совместно, что разрешает конфликты [58].

### 2.5.3 Сокращение множества Парето с использованием информации об относительной значимости критериев

Пусть  $I$  – множество номеров критериев

$$I = \{1, 2, \dots, m\}. \quad (2.65)$$

Тогда  $i$ -й критерий предпочтительнее  $j$ -го критерия с определенными параметрами  $w_i^* \mid w_i^* > 0$  и  $w_j^* \mid w_j^* > 0$  для всех векторов  $y', y'' \in R^m$ , для которых выполняется следующее:

$$y'_i > y''_i, y''_j > y'_j, y'_s = y''_s \quad \forall s \in I \setminus \{i, j\}; \quad (2.66)$$

$$y'_i - y''_i = w_i^*, y''_j - y'_j = w_j^* \Rightarrow y' \succ y''. \quad (2.67)$$

Для ЛПР  $i$ -й критерий важнее  $j$ -го в том случае, если при выборе из двух векторов ЛПР готово пренебречь определенной величиной  $w_j^*$  по менее значимому  $j$ -му критерию для получения дополнительного количества  $w_i^*$  по более значимому  $i$ -му критерию. При этом должно быть выполнено условие сохранения всех остальных значений критериев. Соотношение между  $w_i^*$  и  $w_j^*$  позволяет количественно оценить указанную степень важности. Для пары критериев коэффициентом относительной значимости называют положительное число [122]

$$\theta_{ij} = \frac{w_j^*}{w_i^* + w_j^*}, \quad (2.68)$$

показывающее относительную потерю по менее значимому критерию.

Для относительной значимости критериев принято в качестве аксиомы утверждение об *инвариантности отношения предпочтения*, в соответствии с которым отношение предпочтения  $\succ$  является инвариантным для линейного положительного преобразования [69]. Признаком инвариантности отношения предпочтения  $\succ$  являются его свойства аддитивности:

$$\forall y', y'' \in R^m \quad \forall c \in R^m \quad y' \succ y'' \Rightarrow (y' + c) \succ (y'' + c) \quad (2.69)$$

и однородности:

$$\forall y', y'' \in R^m \quad \forall \alpha > 0 \quad y' \succ y'' \Rightarrow \alpha y' \succ \alpha y''. \quad (2.70)$$

Если отношение предпочтения  $\succ$  инвариантно, то можно полагать, что в определении (2.67) векторы  $y'$  и  $y''$  фиксированы. В частности, в этом определении можно положить

$$y'_i = w_i^*, y'_j = -w_j^*, y'_s = 0 \quad \forall s \in I \setminus \{i, j\}, y'' = 0_m, \quad (2.71)$$

или

$$y'_i = (1 - \theta_{ij}), y'_j = -\theta_{ij}, y'_s = 0 \quad \forall s \in I \setminus \{i, j\}, y'' = 0_m. \quad (2.72)$$

Если отношение предпочтения  $\succ$  удовлетворяет аксиомам 1 – 3 (см. приложение А), является инвариантным и  $i$ -й критерий важнее  $j$ -го с коэффициентом относительной значимости  $\theta_{ij} \in (0, 1)$ , тогда для любого непустого множества выбираемых оценок  $\text{Sel } Y$  справедливы включения [69]:

$$\text{Sel } Y \subset P(\hat{Y}) \subset P(Y), \quad (2.73)$$

где  $P(\hat{Y})$  – множество Парето-оптимальных оценок в многокритериальной задаче с возможными решениями  $X$  и новым векторным критерием

$\hat{f} = (\hat{f}_1, \hat{f}_2, \dots, \hat{f}_m)$ , т. е.  $\hat{Y} = \hat{f}(X)$ , компоненты которого вычисляются по формулам

$$\hat{f}_j = \theta_{ij} f_i + (1 - \theta_{ij}) f_j. \quad (2.74)$$

$$\hat{f}_s = f_s \quad \forall s \in I \setminus \{j\} \quad (2.75)$$

На основе информации о большей значимости одного критерия по сравнению с другим, множество Парето может быть сокращено без потери выбираемых векторов. По формуле (2.74) векторный критерий  $\hat{f}$  получается из прежнего критерия  $f$  заменой менее значимого критерия  $f_j$  на комбинацию критериев  $f_i$  и  $f_j$  с коэффициентом относительной значимости  $\theta_{ij}$ . Все остальные критерии  $f_s$  при этом сохраняются. Такой пересчет позволяет сохранить многие полезные для оптимизации свойства критериев  $f_i$  и  $f_j$ . Например, если  $f_i$  и  $f_j$  являются непрерывными, вогнутыми, выпуклыми или линейными, то критерий  $\hat{f}$  также будет обладать соответствующими свойствами [69].

## 2.6 Выводы по второй главе

Во второй главе диссертационной работы получены следующие основные результаты.

1. Для эффективного использования имитационной модели в контуре управления предприятием предложен метод *поиска рациональных решений в информационной системе управления промышленным предприятием*. Метод *поиска рациональных решений* реализуется последовательностью следующих этапов эксплуатации базовой имитационной модели: *формулировка целей* управления предприятием; *определение объекта* управления; *структурный синтез* имитационной модели; *параметрический синтез* имитационной модели; выбор алгоритма принятия решений; *коррекция* всей системы управления предприятием.

2. Предложена технология решения задач рационального выбора параметров систем с помощью алгоритма случайного поиска. При проектировании ре-

лизации алгоритма случайного поиска рациональных решений выделен этап структурного синтеза, целью которого является определение структуры реализации алгоритма, и этап параметрического синтеза, целью которого является определение вектора параметров реализации алгоритма. Структурная и параметрическая декомпозиция алгоритма случайного поиска позволяет создать базу знаний алгоритмов с лучшей структурой и лучшими значениями параметров при решении задачи построения интеллектуальной системы поддержки принятия решений.

3. Исследована проблема выбора набора функций для исследования генетического алгоритма. Предложено использовать специально разработанные функции, имитирующие наиболее сложные ситуации для генетического алгоритма при поиске оптимума, для которых в качестве параметров выступает бинарная строка.

4. Сформулирована целевая функция оценки эффективности взаимодействия компонентов предприятия и определены основные показатели деятельности промышленного предприятия, которые позволяют дать общую оценку состояния предприятия. При апробации технологии поиска рациональных решений многокритериальная целевая функция представлена линейной сверткой частных критериев с весовыми коэффициентами.

5. Предложена технология количественной оценки координирующих действий в иерархической системе при решении многокритериальных задач на основе аксиоматического принципа Парето и постулата совместимости М. Месаровича. При этом на каждом уровне решение полагают оптимальным, если оно удовлетворяет не только целевой функции исследуемого объекта, но и согласуется с глобальной целевой функцией системы.

### ГЛАВА 3. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРОЕКТИРОВАНИЯ ИМИТАЦИОННЫХ МОДЕЛЕЙ ВНУТРИФИРМЕННОГО УПРАВЛЕНИЯ В ИНФОРМАЦИОННОЙ СИСТЕМЕ ПРЕДПРИЯТИЯ

#### 3.1 Разработка метода построения имитационной модели функционирования предприятия

Промышленное предприятие (ПП) представляется многоуровневой иерархической системой с координирующими элементами, принимающими решения на каждом из уровней в соответствии со своим положением в иерархии управления [136]. Предлагаемый метод имитационного моделирования системы управления ПП в условиях неопределенности и риска выбора нерациональных ресурсов использует декомпозицию системы на подсистемы, процессы и активности; позволяет обеспечить важное сходство структуры модели и объекта исследования на основе процессного способа имитации.

В основу метода имитационного моделирования системы управления положено следующее:

1) представление промышленного предприятия в виде *многоуровневой иерархической системы* с распределением компонентов системы по уровням с активными элементами принятия решений на каждом из уровней иерархии [136];

2) использование информационных технологий КИС (см. п. 1.2): методологии *UML* и проведение функционального анализа на основе методологии *SADT* для построения вербальной и концептуальной модели объекта исследования [109]; применение концепции ООП для разработки программы имитационной модели на основе процессного способа имитации [109]; использование технологии *XML* для ввода данных в модель из корпоративной информационной системы [143];

3) создание и использование *базовой имитационной модели* (БИМ) промышленного предприятия в системе управления [143];

4) выбор *варианта решения* на соответствующем уровне системы управления промышленным предприятием [131];

5) использование иерархических *уровней адаптации* системы управления для приспособления ее к специфическим свойствам объекта и внешней среды [116].

**Предлагается метод построения имитационных моделей в корпоративных информационных системах**, состоящий в выполнении последовательности этапов (рисунок 3.1) [106]. Будем рассматривать функции  $F_i$ , определяемые на каждом из этапов (см. ниже) при построении и эксплуатации имитационной модели ( $i = 1, \dots, N$ , где  $N$  – общее число этапов,  $N = 12$ ). Например, функции  $F_1, \dots, F_3$ , соответственно на трех на начальных этапах проектирования ИМ, могут быть реализованы в автоматизированном режиме с применением методологий *IDEF*, *DFD*, *RUP* и CASE-средств *AllFusion Modeler*, *ERWin*, *S-Designer* и др.

*Этап 1.* Построение вербальной модели исследуемых объектов системы с помощью UML-диаграмм анализа уровней и методологии SADT (Structured Analysis and Design Technique). Формируется представление о структуре системы управления. Содержательное описание (вербальная модель) автоматизируется с применением информационной технологии IDEF0 и CASE-средства *AllFusion Modeler*. Следует отметить, что на этом этапе благодаря автоматизации выполняется контроль за ходом работ и управление проектом на ЭВМ, что сокращает длительность проектирования имитационной модели (см. рисунок 3.1).

*Этап 2.* Построение концептуальных моделей каждого уровня детализации с использованием методологии функционального моделирования SADT и применением CASE-средства *AllFusion Modeler*. На этом этапе формируются представления системы для последующего использования объектно-ориентированного программирования.

*Этап 3.* Проверка правильности концептуальной модели; если для какого-либо из уровней концептуальная модель не прошла валидацию (соответствие модели реальному объекту), то возвращаются на предыдущий этап проектирования. SADT в нотации IDEF0 используется для изучения и анализа функций системы, позволяет документировать результаты исследований, что является необходимым при работе в команде проектировщиков [143].

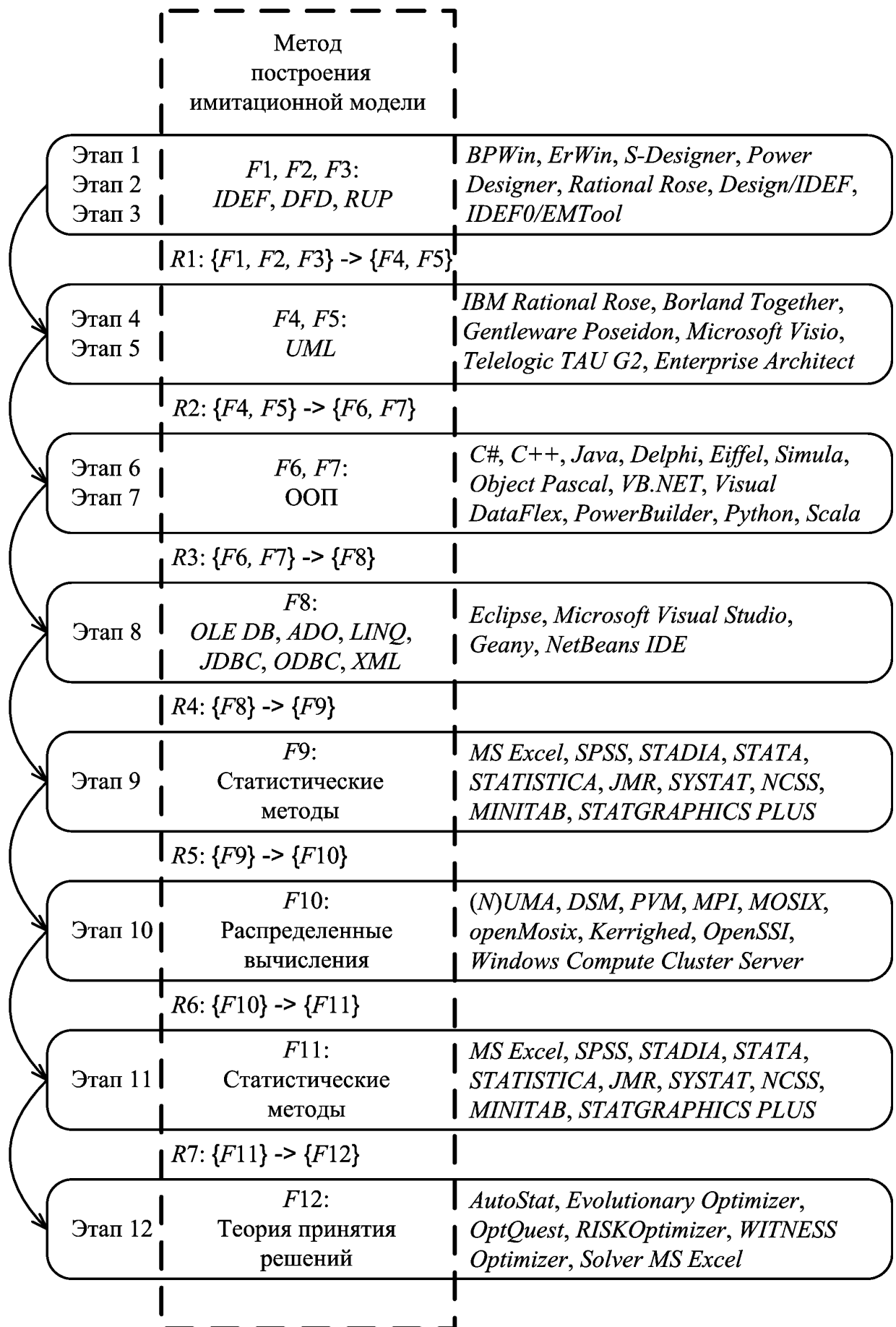


Рисунок 3.1 – Этапы реализации метода построения имитационной модели



*Этап 4.* Создание формальной модели исследуемой системы с применением процессного способа имитации, реализуемого составлением и описанием алгоритмов проектируемых процессов и отношений между ними с использованием *UML*-технологии. При этом решаются задачи синхронизации процессов между собой и управляющей программой моделирования (УПМ), задания начальных условий, организации сбора статистики и окончания имитации. Универсальный характер процессного способа имитации позволяет моделировать различные процессы в системе, сохраняя особенности ее структуры.

*Этап 5.* Построение имитационной модели (ИМ) системы с добавлением алгоритмов сбора статистики имитации и построением алгоритмов обработки статистических записей имитации динамики взаимодействия процессов ИМ с применением диаграмм деятельности *UML* и диаграммы классов. Программная поддержка *UML*-технологии обеспечивается *CASE*-средством *Enterprise Architect*.

*Этап 6.* Программирование на каждом уровне детализации ИМ с применением методологии ООП, что позволяет создавать библиотеку моделей объектов системы, расширяя их функциональность при эксплуатации базовой ИМ. Используется интегрированная система разработки программных приложений *Visual Studio.NET*. Основным является командный режим проектирования программного обеспечения ИМ с автоматизированным контролем версий в системе управления проектом.

*Этап 7.* Верификация программы имитационной модели, состоящая в проверке правильности функционирования алгоритмов программы замыслу разработчика. Проектируются тестовые примеры для верификации отдельных компонентов имитационной модели и для контроля за качеством реализации проекта. Функции этапа реализуются с использованием ООП, среды разработки *Visual Studio.NET* и программных средств поддержки автоматизированного тестирования.

*Этап 8.* Мониторинг прототипов объекта имитации для получения входных данных модели. Реализуется с применением *XML*-технологии для чтения информации из источников корпоративной информационной системы предприятия и технологии *ADO* совместно с динамической библиотекой *msado15.dll*.

*Этап 9.* Испытание ИМ и технологических характеристик программной реализации модели системы с использованием статистических пакетов (например, *Statistica*, *SPSS*) для обработки выходных данных имитационных экспериментов.

*Этап 10.* Проведение серий имитационных экспериментов (ИЭ) по методу Монте-Карло и усреднение значений откликов при каждой исследуемой комбинации параметров ИМ. Сокращение длительности экспериментов, не менее чем в 2,5 раза, достигается с помощью *MPI*-технологии распределенных вычислений и программного средства *MPICH*.

*Этап 11.* Анализ оперативной статистики ИЭ в пакете *Statistica* и модификация комбинации значений параметров ИМ согласно планов (полного или дробного факторного) постановки экспериментов.

*Этап 12.* Формирование матриц вариантов решений, выбор критериев оценки рациональных параметров ИМ и определение оптимального варианта организации моделируемой системы. Выбирается рациональный состав ресурсов предприятия для эксплуатации имитационной модели. На этом этапе используются программные реализации алгоритмов оптимизации на основе случайного поиска, в частности, генетический алгоритм.

Для разработки моделей исследуемых объектов и проведения имитационных экспериментов создаются специализированные программные средства (см. п. 1.5), при работе с которыми требуется специальная подготовка пользователей. В связи с этим возникает актуальная задача создания программного комплекса имитационного моделирования, удобного для эксплуатации лицу, принимающему решения при управлении деятельностью предприятия, однако не являющемуся специалистом в области информационных технологий, но обладающему и хорошо владеющему знаниями о предметной области.

Оценивание эффективности построения и эксплуатации имитационной модели (ИМ) производственной системы с учетом сложности используемых программных средств и уровня квалификации пользователей можно выполнить, используя информационные оценки, в частности, логарифм многообразия в качестве единицы количества информации [99].

**Основные теоретические положения.** Пусть на отдельных этапах проектирования и эксплуатации ИМ выполняются определенные функции операторов  $F_i$  ( $i = 1, \dots, N$ , где  $N$  – общее число этапов):

$$F_i = \{f_{ik} \mid f_{ik} \in F_i, k = 1, \dots, |F_i|\},$$

где  $f_{ik}$  – функции оператора  $F_i$ .

Широкое применение различных информационных технологий в КИС (см. п. 1.2) дает возможность предложить новый итерационный метод распределенного проектирования и эксплуатации имитационных моделей [106], в соответствии с которым на  $i$ -м этапе проектирования используют коммерческие программные средства с функциональностью, реализующей функции оператора  $F_i$ ,  $i = 1, \dots, N$ . Например, для этапа 11 (Анализ оперативной статистики ИЭ) могут применяться различные коммерческие статистические программные продукты (см. п. 1.2). Выбор программного продукта на  $i$ -м этапе обусловлен требованиями эффективной реализации функций оператора  $F_i$ ,  $i = 1, \dots, N$ .

В последовательности операторов  $F_i$ ,  $i = 1, \dots, N$  выделен этап эксплуатации ИМ, представленной в виде программного приложения, и определены два типа ресурсов – человеческий ресурс (ресурс  $f$ -типа) и программный ресурс (ресурс  $p$ -типа).

Количество информации в операторе  $F_N$  на этапе эксплуатации ИМ оценивается принятой логарифмической комбинаторной мерой Хартли [99]

$$H(F_N) = \log_2 |\{f_{Nk} \mid f_{Nk} \in F_N, k = 1, \dots, |F_N|\}|.$$

Пусть специалист предметной области (ресурс  $f$ -типа) имеет возможность перерабатывать количество информации  $H(f_N)$  такое, что

$$H(f_N)^{inf} \leq H(f_N) \leq H(f_N)^{sup},$$

где  $H(f_N)^{inf}$ ,  $H(f_N)^{sup}$  – соответственно нижняя и верхняя границы количества перерабатываемой информации, определяемые квалификационными возможностями специалиста, т. е. разнообразием знаний, умений, навыков в предметной области [65].

Будем полагать, что многообразие функций  $f_{Nk} \in F_N, k = 1, \dots, |F_N|$ , оцениваемое количеством информации  $H(F_N)$ , не превышает информационных возможностей ресурса  $f$ -типа по переработке определенного количества информации:

$$H(F_N) \in [H(f_N)^{inf}, H(f_N)^{sup}].$$

Сложность (многообразие) программного средства (ресурса  $p$ -типа) оценивается функционально-ориентированными метриками ( $FP$ -оценки). При этом определяют количество внешних вводов, выводов и запросов, внутренних логических и внешних интерфейсных файлов. Для программных продуктов с высокой алгоритмической сложностью, к которым могут быть отнесены имитационной модели, оценивают количество алгоритмов [73]. Тогда для ресурса  $p$ -типа логарифм многообразия (комбинаторная мера Хартли):

$$H(P) = \log_2 |FP|,$$

где  $|FP|$  – мощность множества  $FP$ -оценок.

**Постановка задачи.** Пусть на этапе проектирования имитационной модели для выполнения функций оператора  $F_N$  с количеством информации  $H(F_N)$  требуется программное средство (ресурс  $p$ -типа) с информационной оценкой  $H(P_N)$ . Для реализуемых функций будем полагать, что выполняется условие  $H(F_N) \leq H(P_N)$ , т. е. программное средство (ресурс  $p$ -типа) поддерживает вы-

полнение всех функций оператора  $F_N$ . Требуется определить условия для выполнения функций оператора  $F_N$  за минимальное время при известной информационной оценке  $H(f_N)$  ресурсов  $f$ -типа и информационной оценке  $H(p_N)$   $p$ -ресурсов.

**Соотношение 1.** Функции оператора  $F_N$  (в дальнейшем индекс  $N$  этапа эксплуатации имитационной модели будем опускать) реализуются с использованием ресурсов таких, что  $H(f) = H(F)$  и  $H(p) = H(P)$  при условии  $H(F) = H(P)$ . Тогда  $H(p)/H(f) = H(P)/H(F) = 1 \Rightarrow t_b = T_F$ , т. е. функции оператора  $F$  выполняются за определенное время  $T_F$ .

**Соотношение 2.** Функции оператора  $F$  реализуются с использованием ресурсов таких, что  $H(f) \leq H(f)^{sup} \mid H(F) \geq H(f)^{inf}$ ,  $H(p) = H(P)$  и  $H(F) = H(P)$ . Тогда

$$H(p)/H(f) = H(P)/(H(F) + (H(f)^{sup} - H(F))) \leq 1 \Rightarrow t_b \leq T_F,$$

т. е. функции оператора  $F$  выполняются за время, меньшее  $T_F$  с учетом квалификации (многообразия знаний, навыков, умений) ресурса  $f$ -типа.

**Утверждение 1.** Если для выполнения функций оператора  $F$  используются ресурсы  $f$ -типа и  $p$ -типа такие, что  $H(f) \leq H(f)^{sup} \mid H(F) \geq H(f)^{inf}$  и  $H(p) \geq H(P)$  при условии  $H(F) = H(P)$ , тогда

$$\frac{H(p)}{H(f)} = \frac{H(P) + (H(p) - H(P))}{H(F) + (H(f)^{sup} - H(F))} \Rightarrow t_b \leq T_F \mid H(p) \leq H(f)^{sup}. \quad (3.1)$$

**Доказательство.** Покажем, что в выражении (3.1) условие  $H(p) \leq H(f)^{sup}$  обеспечивает время выполнения функций оператора  $F$  такое, что  $t_b \leq T_F$ . Для этого в выражении (3.1) числитель и знаменатель разделим на  $H(F) \mid H(F) > 0$  по смыслу задачи. Используя дополнительно соотношение 1, получим:

$$\frac{H(p)}{H(f)} = \frac{H(p)}{H(f)^{sup}} . \quad (3.2)$$

Тогда с учетом соотношения 1 и соотношения 2 следует, что  $t_B \leq T_F$  при условии  $H(p) \leq H(f)^{sup}$ .

**Доказательство закончено** [113].

Доказанное утверждение 1 позволяет определить требования к проектированию имитационной модели с применением программных средств и информационных технологий на каждом из этапов. В качестве примера представлен программно-технологический комплекс имитации сложных систем (ПТКИ) *BelSim* [143]. Например, этапы с первого по девятый используют следующие информационные технологии:

$$IDEF0 \xrightarrow{R1} UML \xrightarrow{R2} C++ \xrightarrow{R3} ADO \xrightarrow{R4} STATISTICA \quad (3.3)$$

На этапах проведения исследований имитационных моделей и их эксплуатации используются следующие программные средства и информационные технологии:

$$STATISTICA \xrightarrow{R5} MPI \xrightarrow{R6} STATISTICA \xrightarrow{R7} Solver MS Excel \quad (3.4)$$

В преобразованиях (3.3) и (3.4)  $R_k$  ( $k = 1, \dots, 7$ ) – отношения между этапами, реализуемые информационными технологиями и соответствующими CASE-средствами. Например,  $R1: \{F1, F2, F3\} \rightarrow \{F4, F5\}$ ;  $R2: \{F4, F5\} \rightarrow \{F6, F7\}$  и др. (рисунок 3.1). Для автоматизации построения и эксплуатации ИМ могут быть предложены и другие варианты. Для выбора наилучшего из них может быть сформулирована оптимизационная задача.

Реализация отношений  $R_k$  требует разработки оригинальных технологий и средств их поддержки. Например,  $R1$  представлена технологией построения имитационных моделей на основе *IDEF0*-диаграмм. На этапе 10 в ПТКИ исполь-

зуется библиотека функций *MPI*, *R6* реализована программным интерфейсом, использующим *XML*-технологии для ввода результатов прогона ИМ в пакет *STATISTICA*.

Таким образом, предложено концептуальное развитие актуального научного направления, основанного на универсальном подходе к проблеме имитационного моделирования в информационных системах управления промышленным предприятием.

### 3.2 Оптимизация процесса управления построением имитационной модели в корпоративной информационной системе

В процессе выполнения функций операторов  $F_i$   $i = 1, \dots, N$  (см. п. 3.1) будем рассматривать два типа ресурсов – человеческий ресурс (ресурс  $f$ -типа) и программный ресурс (ресурс  $p$ -типа), используемых на этапах построения имитационной модели (создания программного средства). В связи с неоднозначностью выбора таких ресурсов ставится оптимизационная задача.

**Постановка задачи.** Пусть итерационный процесс проектирования, разработки и эксплуатации имитационной модели (программного средства)  $ItPr$  состоит из  $n$  фиксированных этапов:

$$ItPr \stackrel{def}{=} \{St_i / i = 1, \dots, n\}. \quad (3.5)$$

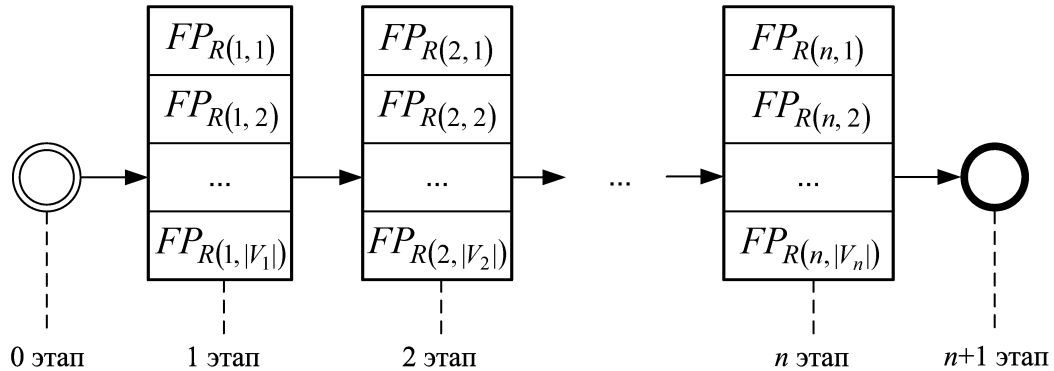
Этапы  $St_i$ ,  $i = 1, \dots, n$ , выполняются последовательно.

Пусть каждый из этапов  $St_i$  допускает выбор одного из допустимых состояний  $FP_{R(i,k)}$ , определяемых использованием человеческого ресурса (ресурса  $f$ -типа) и одного из видов допустимого программного обеспечения (ресурса  $p$ -типа) на  $i$ -м этапе (рисунок 3.2):

$$St_i \stackrel{def}{=} \{FP_{R(i,k)} \mid k = 1, \dots, |V_i|\}, \quad i = 1, \dots, n, \quad (3.6)$$

$$R(i,k) \stackrel{def}{=} (i, f_{ik}, p_{ik}, k),$$

где  $R(i, k)$  в (3.6) – индекс состояния в виде вектора, который включает порядковый номер этапа  $St_i: i, i \in \{1, \dots, n\}$ ; код используемого ресурса  $f$ -типа  $f_{ik}$ ,  $f_{ik} \in \{1, \dots, |f|\}$ ; порядковый номер используемого ресурса  $p$ -типа  $p_{ik}$ ,  $p_{ik} \in \{1, \dots, |p|\}$ ; порядковый номер состояния в пределах нумерации состояний этапа  $St_i: k, k \in \{1, \dots, |V_i|\}$ , в свою очередь  $|V_i|$  – мощность множества номеров состояний на  $i$ -м этапе  $ItPr$ .



**Рисунок 3.2 – Последовательность этапов проектирования, разработки и эксплуатации имитационной модели**

Себестоимости отдельных состояний складываются из стоимостей используемых ресурсов. Временные затраты на обработку информации в отдельных состояниях зависят в частности от знаний, умений, навыков (квалификации) человеческого ресурса (ресурса  $f$ -типа) и функциональных возможностей программных средств (ресурсов  $p$ -типа) [113].

Пусть  $Res$ ,  $Cost$  – множества ресурсов  $f$ -,  $p$ -типа и стоимостей соответствующих ресурсов:

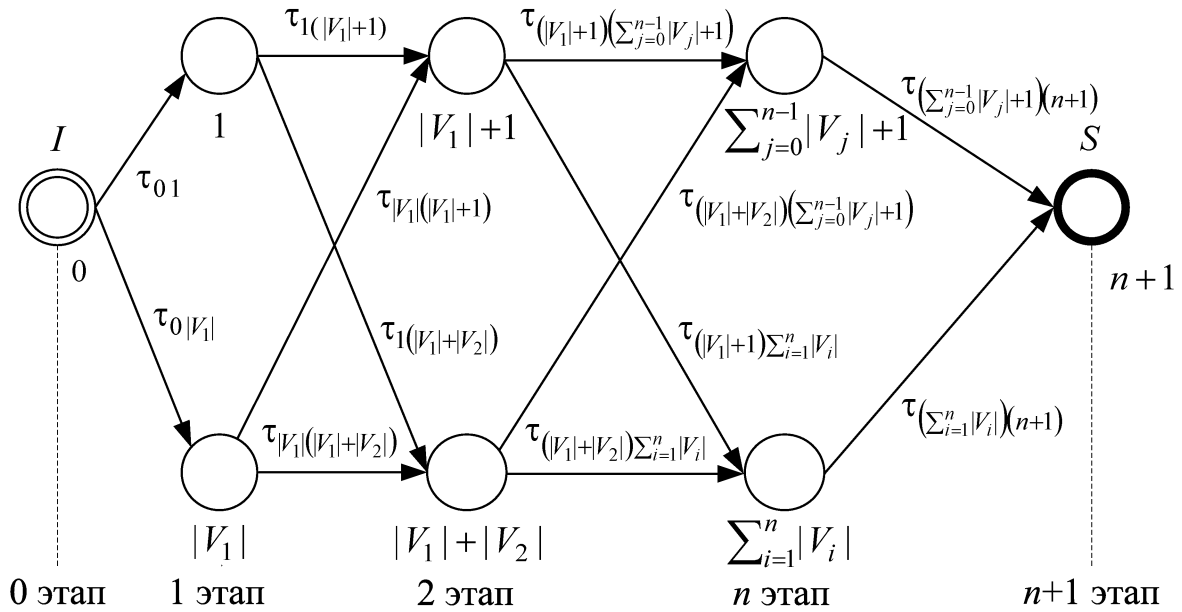
$$Res \stackrel{def}{=} \{Res_r / r = 1, \dots, |Res|\}, \quad (3.7)$$

$$Cost \stackrel{def}{=} \{Cost_r / Cost_r \in R, r = 1, \dots, |Res|\}. \quad (3.8)$$

Итерационный процесс (3.5) проектирования, разработки и эксплуатации имитационной модели может быть представлен ациклическим оргграфом с нагрузками на дугах в виде стоимостных и/или временных затрат (рисунок 3.3),



в котором состояния (3.6) – вершины графа. В таком графе *ItPr* (3.5) разделяется на  $n$  этапов, истоком является виртуальный нулевой этап. Вершины последовательно нумеруются от истока (0°этап) к стоку (  $n+1$  этап), соединяются дугами согласно последовательности выполнения этапов проектирования, разработки и эксплуатации имитационной модели (см. п. 3.1, рисунок 3.2).



**Рисунок 3.3 – Нагрузки на дуги ориентированного графа**

Обозначим через  $\tau_{uv}$  нагрузку на дугу графа (см. рисунок 3.3), инцидентную вершине с номером  $u \in V_{i-1}$  и вершине с номером  $v \in V_i$ , где  $V_i$  – множество номеров вершин для  $i$ -го яруса графа (этапа *ItPr* (3.5)):

$$V_i \equiv \left\{ \sum_{j=0}^{i-1} |V_j| + 1, \sum_{j=0}^{i-1} |V_j| + 2, \dots, \sum_{j=0}^i |V_j| \right\}, i = 1, \dots, n. \quad (3.9)$$

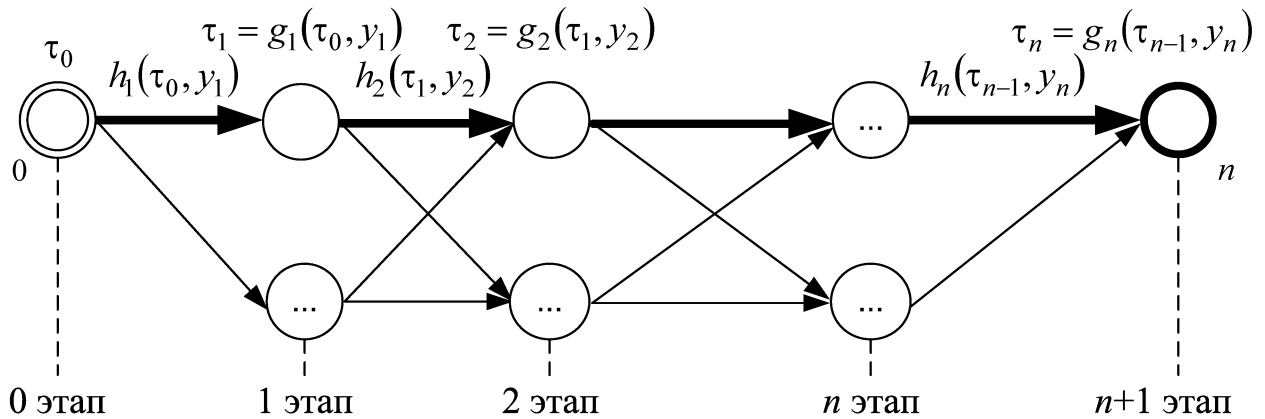
Пусть нагрузки  $\tau_{uv}$  представляют собой значения временных затрат и/или стоимостей затрат ресурсов на обработку информации при решении одной имитационной задачи в соответствии с  $FP_{R(u,v)}$  (3.6) *ItPr* (3.5).

**Решение задачи многокритериального управления построением имитационной модели.** Для поиска последовательных оптимальных решений

(оптимальных состояний  $FP_{R(u,v)}$  (3.6)) в задачах с многошаговой (многоэтапной) структурой ( $ItPr$  (3.5) с множеством последовательных этапов) используется метод динамического программирования [3].

Рассматриваемый итерационный процесс построения имитационной модели (создания программного средства) является управляемой системой, в которой состояния изменяются на каждом этапе  $ItPr$  в результате управляющего воздействия (управления). При этом эффективность управления характеризуется многомерной целевой функцией (ЦФ), определяемой текущим состоянием системы и управляющим воздействием.

Пусть в начальный момент времени (0-й этап  $ItPr$  (3.5)) система находится в исходном состоянии  $\tau_0$  (рисунок 3.4).



**Рисунок 3.4 – Последовательность состояний и управлений**

На следующем шаге в результате управления  $y_1$  система из состояния  $\tau_0$  переходит в состояние  $\tau_1 = g_1(\tau_0, y_1)$  – переход с 0-го этапа  $ItPr$  (3.5) на 1-й этап  $ItPr$  (рисунок 3.4). При этом достигается эффект  $h_1(\tau_0, y_1)$ .

Далее управление  $y_2$  переводит систему из состояния  $\tau_1$  в состояние  $\tau_2 = g_2(\tau_1, y_2)$  (см. рисунок 3.4) и достигается эффект  $h_2(\tau_1, y_2)$ .

Таким образом, за  $n$  этапов  $ItPr$  строится последовательность состояний  $\tau_0, \tau_1, \dots, \tau_n$  и последовательность управлений  $y_1, y_2, \dots, y_n$  таких, что

$$\tau_{i+1} = g_{i+1}(\tau_i, y_{i+1}), \quad i = 1, \dots, n.$$

Общий эффект  $J$  по всем этапам вычисляется по формуле:

$$J(\tau_0, y_1, y_2, \dots, y_i) = h_1(\tau_0, y_1) + h_2(\tau_1, y_2) + \dots + h_i(\tau_{i-1}, y_i), \quad i = 1, \dots, n. \quad (3.10)$$

Необходимо отыскать такую последовательность управлений  $y_1^*, y_2^*, \dots, y_i^*$ , при которой целевая функция (ЦФ)  $J$  (3.10) достигает  $\max(\min)$ , т. е. выполняется условие:

$$J(\tau_0, y_1^*, y_2^*, \dots, y_i^*) = \max(\min)(J(\tau_0, y_1, y_2, \dots, y_i)), \quad i = 1, \dots, n. \quad (3.11)$$

$y_1, y_2, \dots, y_i$

Согласно принципу оптимальности Беллмана на последовательности оптимальных управлений  $y_1^*, y_2^*, \dots, y_i^*$ ,  $i = 1, \dots, n$  должна достигать  $\max(\min)$  и каждая из функций:

$$J_i(\tau_{i-1}, y_i, y_{i+1}, \dots, y_n) = h_i(\tau_{i-1}, y_i) + h_{i+1}(\tau_i, y_{i+1}) + \dots + h_n(\tau_{n-1}, y_n), \quad i = 1, \dots, n. \quad (3.12)$$

Введем следующее обозначение:

$$\varphi_i(\tau_{i-1}) \stackrel{def}{=} \max(\min)(J_i(\tau_{i-1}, y_i, y_{i+1}, \dots, y_n)), \quad i = 1, \dots, n. \quad (3.13)$$

$y_i, y_{i+1}, \dots, y_n$

Тогда из (3.12) и (3.13) следуют функциональные уравнения, именуемые функциональными уравнениями Беллмана:

$$\varphi_i(\tau_{i-1}) = \max(\min)_{y_i}(\varphi_{i+1}(g_i(\tau_{i-1}, y_i)) + h_i(\tau_{i-1}, y_i)), \quad i = 1, \dots, n. \quad (3.14)$$

Решение уравнений Беллмана (3.14) находит последовательность оптимальных управлений и оптимальных значений ЦФ.

С учетом введенных обозначений нагрузок на дуги графа  $\tau_{uv}$  (см. рисунок 3.3) функциональное уравнение (3.14) принимает вид:

$$\tau_v = \min(\max)_u \{ \tau_u + \tau_{uv} \}, \quad \tau_0 = 0, \quad (3.15)$$

где  $u \in V_{i-1}$  – номера вершин, инцидентных исходящим дугам графа  $(i-1)$ -го этапа  $ItPr$ ;  $v \in V_i$  – номера вершин, инцидентных входящим дугам графа  $i$ -го этапа  $ItPr$ ;  $V_i$  – множество номеров вершин вида (3.9) для  $i$ -го этапа  $ItPr$ .

Нагруженные исходящие дуги, инцидентные одной вершине, эквивалентны по интерпретации их характеристик: временных и/или стоимостных затрат для одного и того же состояния на  $i$ -м этапе  $ItPr$  (3.5) построения имитационной модели (создания программного средства), поэтому выполняются следующие соотношения.

**Соотношение 1.** Пусть  $\tau_{uv} \in R$ , тогда для  $ItPr$  (3.5) верны равенства вида [133]:

$$\forall u \in V_{i-1} \forall v_1 v_2 \in V_i [\tau_{uv_1} = \tau_{uv_2}], i = 1, \dots, n. \quad (3.16)$$

**Соотношение 2.** Пусть  $\tau_{uv} = (\tau_{uv}^{(1)}, \tau_{uv}^{(2)}) \in R^2$ ,  $\tau_{uv_1} = (\tau_{uv_1}^{(1)}, \tau_{uv_1}^{(2)})$ ,  $\tau_{uv_2} = (\tau_{uv_2}^{(1)}, \tau_{uv_2}^{(2)})$ . Тогда для  $TPr$  (3.5) верны равенства:

$$\begin{aligned} \forall u \in V_{i-1} \forall v_1 v_2 \in V_i [(\tau_{uv_1}^{(1)} = \tau_{uv_2}^{(1)}) \wedge (\tau_{uv_1}^{(2)} = \tau_{uv_2}^{(2)})] &\Leftrightarrow \\ \Leftrightarrow (\tau_{uv_1} = \tau_{uv_2}), i = 1, \dots, n. & \end{aligned} \quad (3.17)$$

**Утверждение 1.** Пусть для  $\tau_{uv} \in R$  выполняются соотношения вида (3.16), тогда для рассматриваемой задачи функциональное уравнение Беллмана (3.15) принимает вид:

$$\tau_v = L_i = L_{i-1} + \min_{u \in V_{i-1}} (\max) \{ \tau_{uv} \}, \quad (3.18)$$

где  $L_i$  – оптимальное состояние на  $i$ -м этапе,  $L_i \in R$ ,  $L_0 = \tau_0 = 0, u \in V_{i-1}$ ,  $v \in V_i, i = 1, \dots, n$ .

**Доказательство** проводится методом математической индукции.

Этап 1.  $u \in V_0 \equiv \{0\}, v \in V_1 \equiv \{1, \dots, |V_1|\}$ .

$$\tau_v = \tau_0 + \tau_{0v} \stackrel{(3.18)}{=} [\tau_0 = 0] = \tau_{0v}, v \in V_1, \quad (3.19)$$

$$\left( \tau_{01} \stackrel{(3.16)}{=} \tau_{02} \stackrel{(3.16)}{=} \dots \stackrel{(3.16)}{=} \tau_{0|V_1|} \right) \Leftrightarrow (\tau_1 = \tau_2 = \dots = \tau_{|V_1|} = L_1). \quad (3.20)$$

Этап 2.  $u \in V_1 \equiv \{1, \dots, |V_1|\}, v \in V_2 \equiv \{|V_1|+1, \dots, |V_1|+|V_2|\}$ .

$$\tau_v = \min_{u \in V_1} (\max) \{ \tau_u + \tau_{uv} \} \stackrel{(3.20)}{=} \min_{u \in V_1} (\max) \{ L_1 + \tau_{uv} \} = L_1 + \min_{u \in V_1} (\max) \{ \tau_{uv} \}, \quad (3.21)$$

$$\begin{aligned} & \left( \tau_{1(|V_1|+1)} \stackrel{(3.16)}{=} \dots \stackrel{(3.16)}{=} \tau_{1(|V_1|+|V_2|)} \right) \wedge \left( \tau_{2(|V_1|+1)} \stackrel{(3.16)}{=} \dots \stackrel{(3.16)}{=} \tau_{2(|V_1|+|V_2|)} \right) \wedge \dots \\ & \dots \wedge \left( \tau_{|V_1|/(|V_1|+1)} \stackrel{(3.16)}{=} \dots \stackrel{(3.16)}{=} \tau_{|V_1|/(|V_1|+|V_2|)} \right) \Rightarrow \\ & \Rightarrow \left( \begin{array}{l} \min_{u \in V_1} (\max) \{ \tau_u (|V_1|+1) \} = \min_{u \in M_B(1)} (\max) \{ \tau_u (|V_1|+2) \} = \dots \\ \dots = \min_{u \in V_1} (\max) \{ \tau_u (|V_1|+|V_2|) \} \end{array} \right) \Rightarrow \\ & \Rightarrow \left( \tau_{|V_1|+1} = \tau_{|V_1|+2} = \dots = \tau_{|V_1|+|V_2|} = L_2 \right) \\ & \tau_v = L_1 + \min_{u \in V_1} (\max) \{ \tau_{uv} \} = L_2, \quad v \in V_2. \end{aligned}$$

Этап  $i$ . Пусть для любого  $i \in \{1, \dots, n\}$  выполняется условие:

$$\tau_v = L_i = L_{i-1} + \min_{u \in V_{i-1}} (\max) \{ \tau_{uv} \}, \quad v \in V_i \stackrel{(3.9)}{\equiv} \{ \sum_{j=0}^{i-1} |V_j| + 1, \dots, \sum_{j=0}^i |V_j| \}. \quad (3.22)$$

$$\text{Этап } i+1. \quad u \in V_i, \quad v \in V_{i+1} \stackrel{(3.22)}{\equiv} \{ \sum_{j=0}^i |V_j| + 1, \dots, \sum_{j=0}^{i+1} |V_j| \}.$$

Докажем равенство:  $\tau_{|V_i|+1} = \tau_{|V_i|+2} = \dots = \tau_{|V_i|+|V_{i+1}|}$ .

$$\tau_v = \min_{u \in V_i} (\max) \{ \tau_u + \tau_{uv} \} \stackrel{(3.22)}{=} L_i + \min_{u \in V_i} (\max) \{ \tau_{uv} \}, \quad (3.23)$$

$$\begin{aligned} & \left( \tau_{(|V_{i-1}|+1)(|V_i|+1)} \stackrel{def}{=} \dots \stackrel{def}{=} \tau_{(|V_{i-1}|+1)(|V_i|+|V_{i+1}|)} \right) \wedge \\ & \wedge \left( \tau_{(|V_{i-1}|+2)(|V_i|+1)} \stackrel{def}{=} \dots \stackrel{def}{=} \tau_{(|V_{i-1}|+2)(|V_i|+|V_{i+1}|)} \right) \wedge \dots \\ & \dots \wedge \left( \tau_{(|V_{i-1}|+|V_i|)(|V_i|+1)} \stackrel{def}{=} \dots \stackrel{def}{=} \tau_{(|V_{i-1}|+|V_i|)(|V_i|+|V_{i+1}|)} \right) \Rightarrow \\ & \Rightarrow \left( \begin{array}{l} \min_{u \in V_i} (\max) \{ \tau_u (|V_i|+1) \} = \min_{u \in V_i} (\max) \{ \tau_u (|V_i|+2) \} = \\ = \dots = \min_{u \in V_i} (\max) \{ \tau_u (|V_i|+|V_{i+1}|) \} \end{array} \right) \Rightarrow \\ & \Rightarrow \left( \tau_{|V_i|+1} = \dots = \tau_{|V_i|+|V_{i+1}|} = L_{i+1} \right) \end{aligned}$$

Равенство доказано, из чего следует соотношение:

$$\tau_v = L_{i+1} = L_i + \min_{u \in V_i} (\max_{v \in V_{i+1}}) \{\tau_{uv}\}, \quad u \in V_i, \quad v \in V_{i+1}. \quad (3.24)$$

Следовательно, соотношение (3.18) имеет место для любого  $i \in \{1, \dots, n\}$ .

**Доказательство закончено.**

**Утверждение 2.** Пусть для  $\tau_{uv} = (\tau_{uv}^{(1)}, \tau_{uv}^{(2)}) \in R^2$  выполняются соотношения (3.17), тогда для рассматриваемой задачи из уравнения Беллмана (3.15) следует:

$$L_i = \begin{cases} \left( \begin{array}{l} L_{(i-1)1} + Fm_1 \left\{ \tau_{uv}^{(1)} \right\}_{u \in V_{i-1}} \\ L_{(i-1)2} + Fm_2 \left\{ \tau_{u_i v}^{(2)} \mid \tau_{u_i v}^{(1)} = Fm_1 \left( \left\{ \tau_{uv}^{(1)} \right\} \right) \right\}_{u \in V_{i-1}} \end{array} \right) \Big|_{pOpt = 1, 2}; \\ \left( \begin{array}{l} L_{(i-1)1} + Fm_1 \left\{ \tau_{u_i v}^{(1)} \mid \tau_{u_i v}^{(2)} = Fm_2 \left( \left\{ \tau_{uv}^{(2)} \right\} \right) \right\}_{u \in V_{i-1}} \\ L_{(i-1)2} + Fm_2 \left\{ \tau_{uv}^{(2)} \right\}_{u \in V_{i-1}} \end{array} \right) \Big|_{pOpt = 2, 1}; \end{cases}, \quad (3.25)$$

где  $L_i = (L_{i1} \quad L_{i2})^T$ ,  $L_0 = (0 \quad 0)^T$ ,  $u \in V_{i-1}$ ,  $v \in V_i$ ,  $i = 1, \dots, n$ ;  $pOpt$  – параметр наивысшего приоритета критериев оптимизации  $ItPr$  (3.5) по 1-му (времени), либо 2-му (стоимости ресурсов) измерениям векторов  $\tau_{uv} \in R^2$ ;  $Fm_1, Fm_2 \in \{\min, \max\}$  – лексикографически упорядоченные параметром  $pOpt$  критерии оптимизации  $ItPr$  по 1-му, 2-му измерениям векторов состояний на этапах ТП, соответственно.

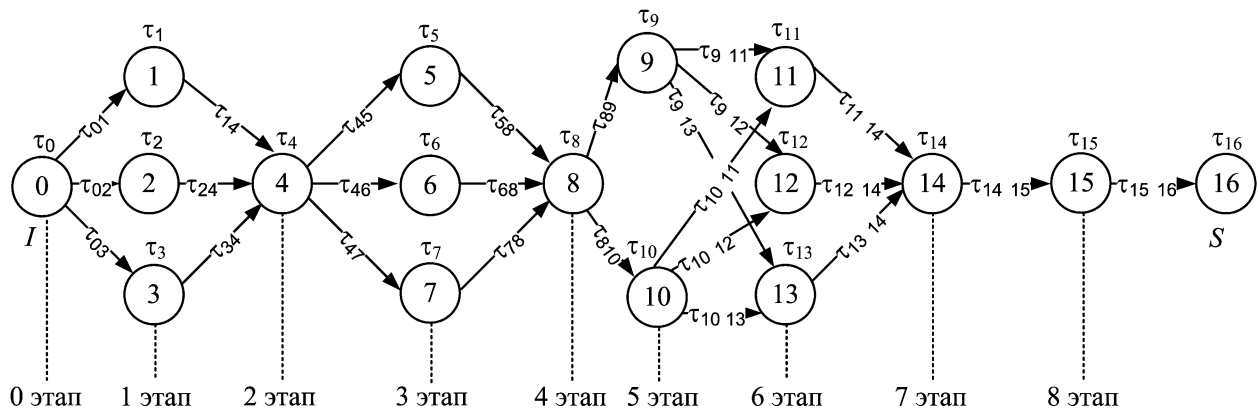
**Замечание.** Доказательство Утверждения 2 проводится с использованием метода математической индукции аналогично доказательству Утверждения 1 с учетом значений измерений векторов  $\tau_{uv}^{(pOpt)} \in R$ . При этом оптимизация  $ItPr$  (3.5) методом динамического программирования (в указанной многокритериальной задаче выбора предпочитаемых состояний рассматриваемого  $ItPr$ ) в первую очередь проводится по наиболее важному критерию (измерению  $pOpt$  векторов  $\tau_{uv}^{(pOpt)} \in R$ ). В случае совпадения оптимальных по наиболее важному критерию значений для нескольких

состояний рассматриваемого оптимизируемого этапа  $ItPr$ , проводится оптимизация по другому – менее значимому из измерений векторов  $\tau_{uv}^{(pOpt)} \in R$ , по которому и выбираются наиболее предпочтительные состояния (3.6)  $ItPr$ .

Выбор последовательности оптимальных управлений (оптимальных состояний для  $ItPr$ , либо оптимального маршрута по графу) определяется последовательностью предикатов вида:

$$Fl \equiv \left\{ fl_{ij} \mid fl_{ij} \in \{True, False\}, i = 1, \dots, n, j = 1, \dots, /V_i | \right\}. \quad (3.26)$$

**Пример оптимизационной задачи управления построением имитационной модели.** Рассматривается выбранный для исследования  $ItPr$  построения имитационной модели для системы управления производственно-экономической деятельностью промышленного предприятия (рисунок 3.5).



**Рисунок 3.5 – Ориентированный граф с состояниями**

Пусть итерационный процесс  $ItPr$  (3.5) построения имитационной модели (изготовления программного средства) состоит из  $n = 8$  этапов  $St_i$ ,  $i = 1, \dots, 8$ :  $St_1$  – вербальное моделирование;  $St_2$  – концептуальное моделирование;  $St_3$  – разработка формальной модели;  $St_4$  – программирование;  $St_5$  – верификация;  $St_6$  – проведение имитационных экспериментов;  $St_7$  – обработка статистики имитации;  $St_8$  – выбор решения.

При этом на каждом из этапов допускается выбор одного из следующего количества допустимых состояний – 1 этап:  $|V_1| = 3$ ; 2 этап:  $|V_2| = 1$ ; 3 этап:

$|V_3|=3$ ; 4 этап:  $|V_4|=1$ ; 5 этап:  $|V_5|=2$ ; 6 этап:  $|V_6|=3$ ; 7 этап:  $|V_7|=1$ ; 8 этап:  $|V_8|=1$ .

Пусть определены временные затраты для каждого из состояний  $ItPr$ . Ниже представлены временные затраты  $T_i$  (час.),  $i=1,\dots,8$  для обработки информации при решении одной имитационной задачи на каждом из этапов  $ItPr$  (3.5): 1)  $T_1=(6,8; 6,8; 5,8)$ ; 2)  $T_2=(7,8)$ ; 3)  $T_3=(13,1;18,2;16,4)$ ; 4)  $T_4=(11,3)$ ; 5)  $T_5=(13,9;13,1)$ ; 6)  $T_6=(11,2;11,2;5,6)$ ; 7)  $T_7=(24,7)$ ; 8)  $T_8=(16,6)$ .

Пусть известны стоимости ресурсов  $S_i$  (руб.),  $i=1,\dots,8$  для обработки информации при решении одной задачи на соответствующих этапах  $ItPr$ :

1)  $S_1=(1948,7; 2005,9; 1930,5)$  ; 2)  $S_2=(7714,2)$  ; 3)  $S_3=(12840,1; 12477,4; 12257,7)$  ; 4)  $S_4=(3160,3)$  ; 5)  $S_5=(7250,1; 6786,0)$  ; 6)  $S_6=(11159,2; 11107,2; 2973,1)$ ; 7)  $S_7=(3865,03)$ ; 8)  $S_8=(169,0)$ .

Соответственно, с учетом  $T_i=(t_{i1},\dots,t_{i|V_i|})$ ,  $S_i=(s_{i1},\dots,s_{i|V_i|})$ ,  $i=1,\dots,8$ , нагрузки на дуги графа  $\tau_{ij} \in R^2$  принимают значения:  $\tau_{01}=\tau_{02}=\tau_{03}=(0,0)$ ;  $\tau_{14}=(t_{11},s_{11})$ ,  $\tau_{24}=(t_{12},s_{12})$ ,  $\tau_{34}=(t_{13},s_{13})$ ;  $\tau_{45}=\tau_{46}=\tau_{47}=(t_{21},s_{21})$ ;  $\tau_{58}=(t_{31},s_{31})$ ,  $\tau_{68}=(t_{32},s_{32})$ ,  $\tau_{78}=(t_{33},s_{33})$ ;  $\tau_{89}=\tau_{810}=(t_{41},s_{41})$ ;  $\tau_{911}=\tau_{912}=\tau_{913}=(t_{51},s_{51})$ ,  $\tau_{1011}=\tau_{1012}=\tau_{1013}=(t_{52},s_{52})$ ;  $\tau_{1114}=(t_{61},s_{61})$ ,  $\tau_{1214}=(t_{62},s_{62})$ ,  $\tau_{1314}=(t_{63},s_{63})$ ;  $\tau_{1415}=(t_{71},s_{71})$ ,  $\tau_{1516}=(t_{81},s_{81})$ .

Определение оптимальных состояний  $ItPr$  (3.5) производится согласно принципу поэтапной оптимизации Беллмана с использованием построенных функций вида (3.26). При этом может быть построена следующая последовательность двумерных векторов:

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \xrightarrow{(3.26)} \begin{pmatrix} L\_1[1] \\ L\_1[2] \end{pmatrix} \xrightarrow{(3.26)} \begin{pmatrix} L\_2[1] \\ L\_2[2] \end{pmatrix} \xrightarrow{(3.26)} \dots \xrightarrow{(3.26)} \begin{pmatrix} L\_8[1] \\ L\_8[2] \end{pmatrix},$$

где  $L\_8[1]$ ,  $L\_8[2]$  –оптимальное время выполнения всех этапов  $ItPr$  и стоимость расхода ресурсов, соответственно.



Из последовательности предикатов (3.26) находится оптимальный путь по графу:

$$\begin{pmatrix} fl_{11} \\ fl_{12} \\ fl_{13} \end{pmatrix} \rightarrow (fl_{21}) \rightarrow \begin{pmatrix} fl_{31} \\ fl_{32} \\ fl_{33} \end{pmatrix} \rightarrow (fl_{41}) \rightarrow \begin{pmatrix} fl_{51} \\ fl_{52} \end{pmatrix} \rightarrow \begin{pmatrix} fl_{61} \\ fl_{62} \\ fl_{63} \end{pmatrix} \rightarrow (fl_{71}) \rightarrow (fl_{81}).$$

В результате решения оптимизационной задачи с заданными исходными данными получены следующие оптимальные состояния выбранного *ItPr*:

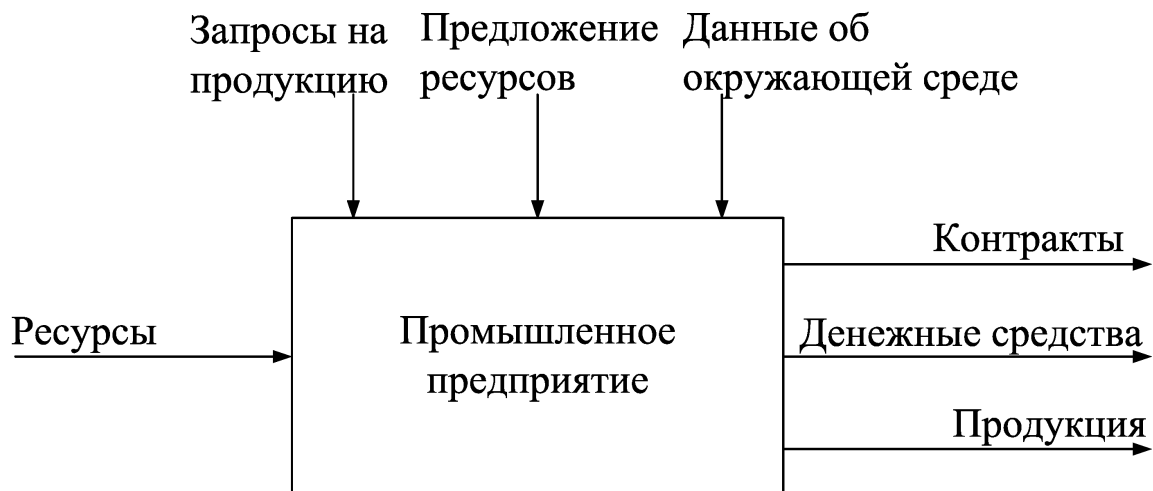
1) с минимальной суммарной стоимостью затрат ресурсов и времени ( $pOpt = 2$  – оптимизация в первую очередь по стоимости ресурсов (2-я координата), во вторую – по времени;  $Fm_1, Fm_2 = \min$ ):  $L_{\min}^* = L_{-8} = (102,3; 56492)$ , где время составило 102,3 час., стоимость ресурсов 56492 руб.; оптимальный путь по графу (для выбранного критерия):  $\langle 0-3-4-7-8-10-13-14-15 \rangle$ ;

2) с минимальными затратами времени и стоимости ресурсов ( $pOpt = 1$  – оптимизация в первую очередь по времени (1-я координата), во вторую – по стоимости ресурсов;  $Fm_1, Fm_2 = \min$ ):  $L_{\min}^* = L_{-8} = (99,1; 57074)$ , где время составило 99,1 час., стоимость ресурсов 57074 руб.; оптимальный путь по графу для выбранного критерия:  $\langle 0-3-4-5-8-10-13-14-15 \rangle$ .

### 3.3 Вербальная модель структурных компонентов промышленного предприятия

В контекстной диаграмме (диаграмма А-0) функционирования предприятия на рисунке 3.6 в качестве функции верхнего уровня рассматривается деятельность всего предприятия. На вход поступают, в общем случае, ресурсы, например, денежные средства (поступают отдельными платежами, характеризуются суммой платежа, датой поступления и источником денежных средств). Поток денежных средств определяется условиями заключенных контрактов на реализацию продукции (например, с требованием предварительной оплаты), полу-

чением займов. Другие ресурсы включают: материальные ресурсы (сырье, материалы); энергетические ресурсы (электроэнергия, тепловая энергия).



**Рисунок 3.6 – IDEF0-диаграмма функционирования промышленного предприятия**

Материальные ресурсы на предприятие поступают отдельными партиями в определенном количестве. Энергетические ресурсы для выпуска продукции поступают по мере их потребления в производстве и не запасаются. Результатом функционирования производственно-экономической системы являются *денежные средства и продукция*. Денежные средства направляются поставщику необходимых ресурсов в соответствии с условиями заключенных контрактов, на оплату налогов по их видам и ставкам, выплачивается установленный размер заработной платы, осуществляются платежи по условиям получения банковских кредитов. Продукция отгружается отдельными партиями в объеме, ассортименте и сроках в соответствии с условиями контрактов, заключенных на реализацию готовой продукции [109].

*Запросы на продукцию и предложение ресурсов* являются управляющими воздействиями. Запросы на продукцию определяются требуемым ассортиментом и объемами продукции, сроками отгрузки и формой оплаты. Предложение ресурсов на рынке определяет их возможные цены, объемы и сроки поставки необходимых ресурсов, условия оплаты за их поставку. Контракты заключаются с учетом имеющихся запросов на продукцию и предложений ресурсов.

Помимо вышеуказанных управляющих воздействий, используются также *данные об окружающей среде* предприятия: ставки, объекты и сроки уплаты

налогов, процентная ставка кредитования, уровень инфляции, нормы амортизационных отчислений, минимальный размер заработной платы и др.

Декомпозиция контекстной диаграммы (диаграмма А1) представлена на рисунке 3.7. Выделены функции, соответствующие процессам планирования объема реализации продукции, планирования производства, заключения и выполнения контрактов, управления финансами, ведения бухгалтерского учета, ведения складского хозяйства, производства продукции.

*Планирование объема реализации продукции* включает в себя планирование спроса – процесс прогнозирования спроса на продукцию в определенном периоде, и планирование продаж и производства – процесс преобразования бизнес-плана и плана спроса в планы продаж основных видов продукции. Производственные мощности при планировании могут не учитываться или учитываться обобщенно. В дальнейшем план продаж по видам продукции преобразуется в объемный или объемно-календарный план производства видов продукции (семейств однородной продукции). В этом плане в качестве планово-учетных единиц выступают изделия, но представления о них носят усредненный характер. Исходными данными для планирования производства продукции являются объемы текущих заказов и итоги реализации продукции за прошедшие периоды [109].

*Процесс планирования производства* состоит из трех основных этапов: составление плана производства, планирование потребностей в ресурсах и мощностях (рисунок 3.8).

*На этапе составления плана производства* укрупненный план производства преобразуется в график изготовления продукции – план-график производства. Это среднесрочный объемно-календарный план, задающий объемы и сроки изготовления конкретных изделий. Исходными данными выступают план продаж (объемно-календарный план для семейств однородной продукции), объемы запасов готовой продукции на складе, объемно-календарный график отгрузок продукции по текущим контрактам, сведения об обеспеченности ресурсами и мощностями, поступающие в виде обратной связи с этапов планирования потребностей в ресурсах и мощностях.

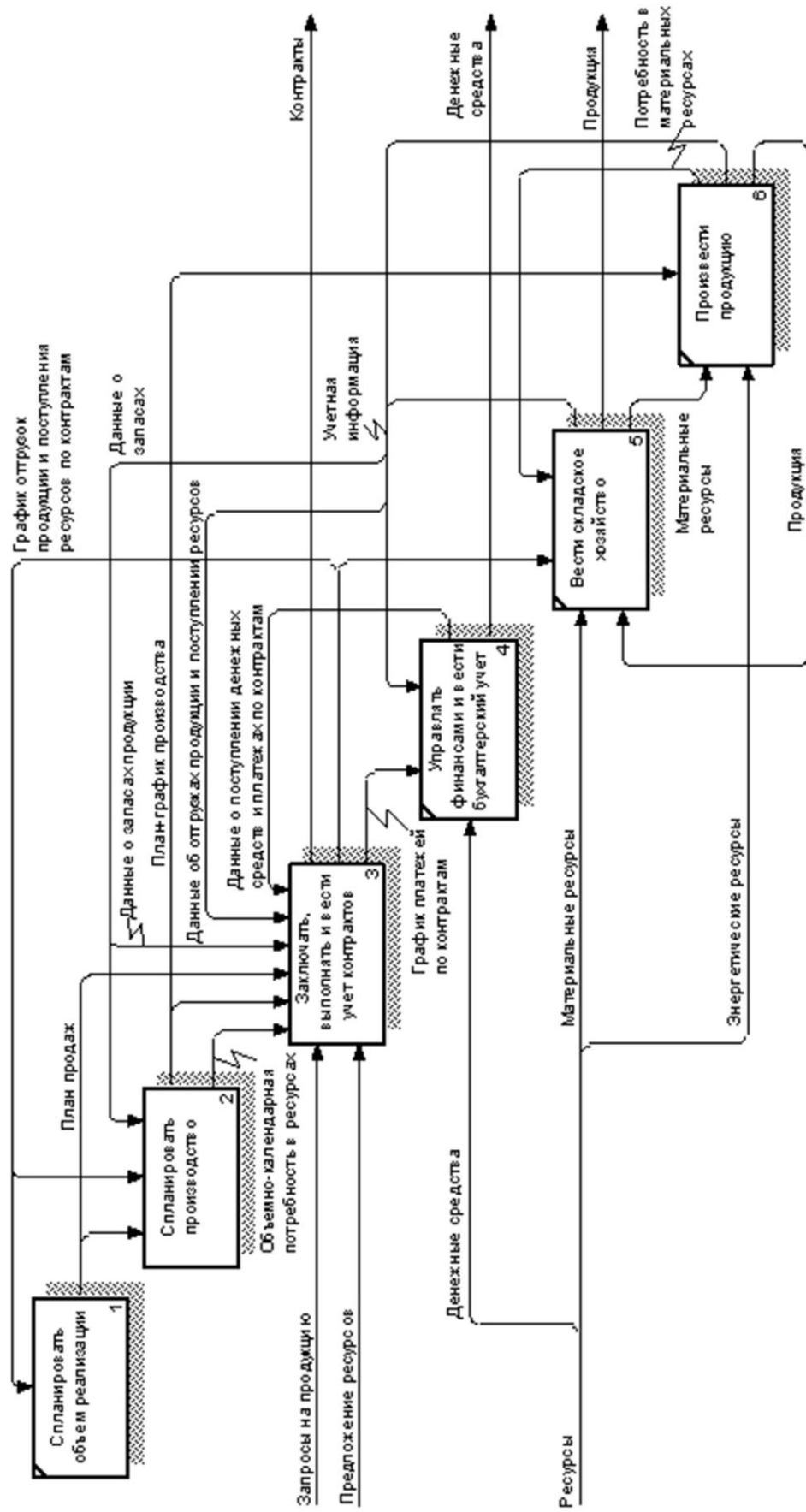


Рисунок 3.7 – Функциональная схема деятельности промышленного предприятия. Диаграмма А1

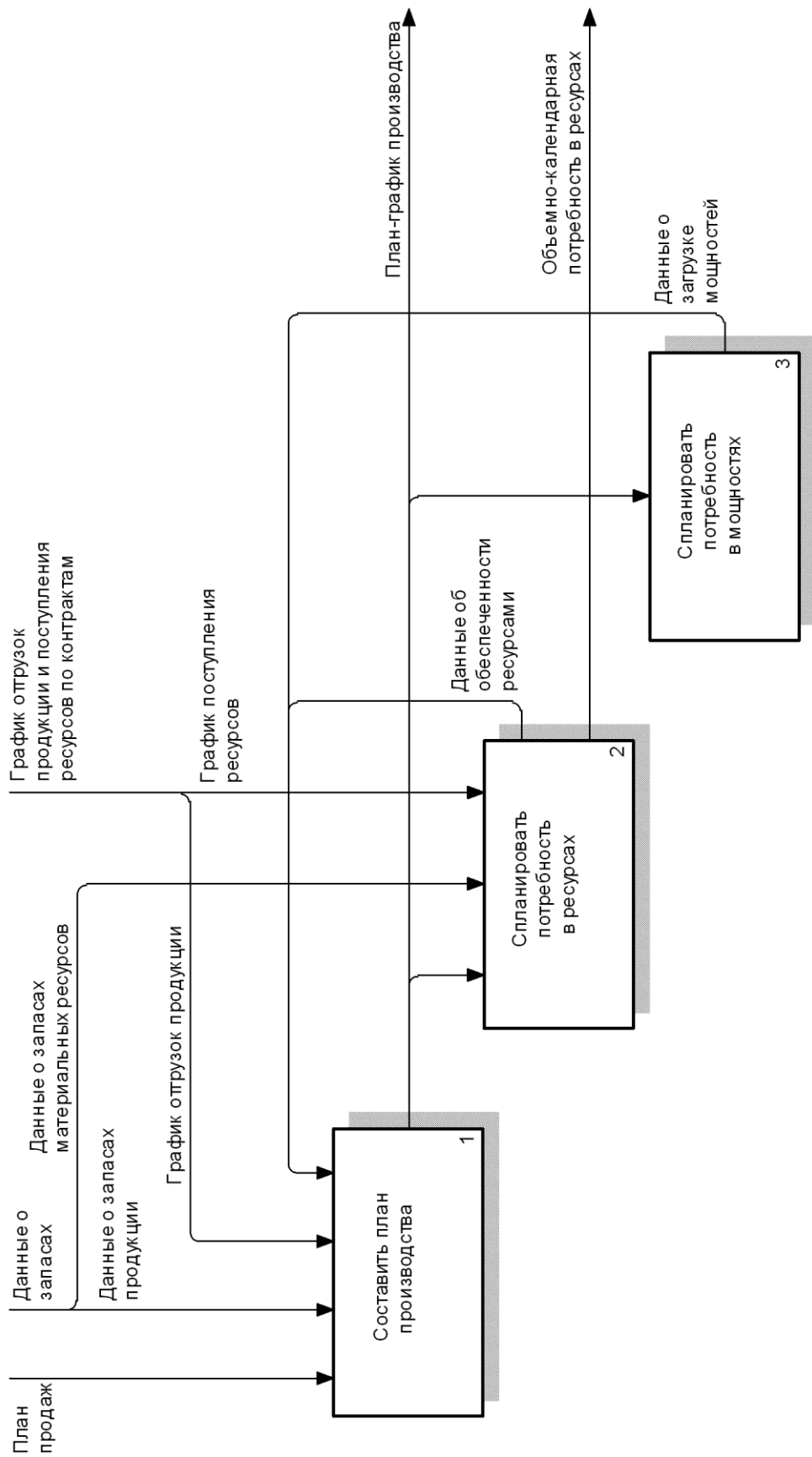


Рисунок 3.8 – Функциональная схема деятельности промышленного предприятия.

Диаграмма А2 – Спланировать производство

*Планирование потребностей в материальных ресурсах* – определение требуемых объемов в каждом интервале планирования материальных ресурсов, необходимых для своевременного обеспечения плана-графика производства продукции. Входными данными для планирования потребностей в материалах являются спецификации выпускаемой продукции (состав и количественные характеристики комплектующих конкретного вида продукции), размер текущих складских запасов материальных ресурсов, а также график поступления ресурсов по контрактам (в виде обратной связи от процесса заключения контрактов на поставку материальных ресурсов).

*Планирование потребностей в мощностях* – расчет располагаемых и требуемых производственных мощностей и других видов производственных ресурсов, оказывающих влияние на пропускную способность предприятия. Входными данными при оценке потребностей в производственных мощностях является план-график производства, а также маршрутизация выпускаемых изделий [109].

В функциональной модели процесса «*Заключать, выполнять и вести учет контрактов*» поступающие запросы клиентов сопоставляются с планами выпуска продукции с целью определения возможности их выполнения. Для этого используется информация о плане продаж, запасах готовой продукции на складе, план-график производства продукции, график отгрузок продукции по заключенным контрактам. Проработанные запросы с указанием возможных сроков их выполнения являются основой для заключения контрактов на отгрузку продукции. Производится оценка предложения ресурсов и на основании объемно-календарной потребности в ресурсах согласовываются условия и заключаются контракты на поставку материальных и энергетических ресурсов. Процесс контроля выполнения контрактов: на основе контрактов и данных об отгрузках продукции, поступлений ресурсов и денежных средств, платежах вырабатываются управляющие воздействия в виде графика платежей и отгрузок продукции, графика ожидаемых поступлений ресурсов и денежных средств [109].

*Управление финансами и ведение бухгалтерского учета* заключается в осуществлении необходимых платежей по контрактам, формировании бухгал-

терской отчетности, проведении расчетов по налогам, заработной плате, а также в формировании запросов на получение кредитов и займов в случае нехватки денежных средств для покрытия текущих расходов.

### 3.4 Концептуальные модели функционирования компонентов промышленного предприятия

**Структура базовой имитационной модели.** Для поддержки принятия решений в социально-экономических системах предприятий разработана базовая имитационная модель (БИМ) в соответствии со стандартом *MRP II*. БИМ реализует следующие функции: *Demand Management (DEM)* – Управление спросом; *Sales and Operation Planning (SOP)* – Планирование продаж и производства; *Master Production Scheduling (MPS)* – Составление плана производства; *Material Requirement Planning (MRP)* – Планирование потребностей в сырье и материалах; *Bill of Materials (BOM)* – Спецификации продукции; *Capacity Requirement Planning (CRP)* – Планирование производственных мощностей; *Shop Flow Control (SFC)* – Управление производством на цеховом уровне; *Scheduled Receipts Subsystem (SRS)* – Отгрузка готовой продукции; *Inventory Transaction Subsystem (ITS)* – Складская подсистема; *Purchasing (PUR)* – Материально-техническое снабжение; *Financial Planning (FPL)* – Финансовое планирование; *Performance Measurement (PEM)* – Оценка результатов деятельности (рисунок 3.9).

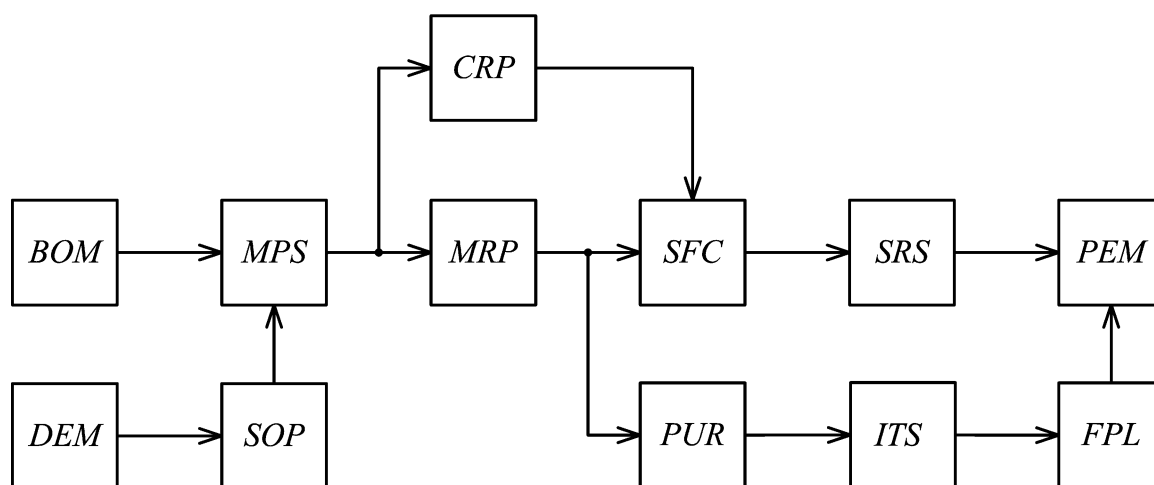


Рисунок 3.9 – Структура базовой имитационной модели

Ограничение функциональности БИМ в том, что в ней не реализован ряд функций *MRP II*: Моделирование; Планирование запасов сбытовой сети; Планирование и управление инструментальными средствами; Контроль входа/выхода.

Следует заметить, что в БИМ (см. рисунок 3.9) производственный процесс представлен укрупненно, поэтому на основе такой модели сложно управлять производственным цехом (*SFC*) и инструментальными средствами (*TPC*). Такое ограничение БИМ позволяет исследовать вопросы материально-технического снабжения (*PUR*) и планирование производственных мощностей (*CRP*) для производственного процесса в целом, что, в свою очередь, не позволяет смоделировать потребности в необходимых ресурсах и предотвратить возникающие проблемы при выполнении отдельных операций [120].

**Концептуальные модели структурных компонентов базовой имитационной модели.** *Параметры производственного процесса* включают: тип и метод организации производства; длительность технологического цикла; потребность в сырье, материалах и энергоресурсах в процессе изготовления продукции; трудоемкость изготовления единицы продукции; производительность оборудования; структуру затрат [109].

Множество параметров, характеризующих *текущее состояние* предприятия, соответствует набору коэффициентов, рассчитываемых для определения показателей, входящих в состав целевой функции:

- показателями общей оценки финансового состояния являются коэффициенты текущей ликвидности  $K_1$ , обеспеченности собственными оборотными средствами  $K_2$  и финансовых обязательств активами  $K_3$ ;

- показатели оценки финансовой устойчивости – коэффициенты абсолютной ликвидности  $K_{\text{АЛ}}$ , критической (быстрой) ликвидности  $K_{\text{БЛ}}$ , собственности  $K_{\text{С}}$ ;

- показатели роста и развития предприятия – темпы роста продаж (нетто)  $T_{\text{РП}}$ , себестоимости  $T_{\text{РС}}$ , прибыли от реализации продукции  $T_{\text{РПР}}$ , чистой прибыли  $T_{\text{РЧП}}$ , стоимости активов  $T_{\text{РСА}}$ ;



– показатели эффективности использования оборотных активов – коэффициент оборачиваемости  $K_{OOA}$ , продолжительности оборота производственных запасов  $П_{OPZ}$ , готовой продукции  $П_{OГП}$ , дебиторской задолженности  $П_{OДЗ}$ ;

– показатели рентабельности – рентабельность продукции по затратам  $R_{ПЗ}$ , выручки от продаж  $R_{ВП}$ , активов предприятия  $R_A$ , основного капитала  $R_{OK}$ , собственного капитала  $R_{СК}$ .

Алгоритм функционирования системы управления предприятием определяет способ управления запасом сырья и формирования производственной программы.

Параметры *трудовых ресурсов* и системы оплаты труда включают:

– показатели структуры трудовых ресурсов: численность основных и вспомогательных рабочих, специалистов, служащих, руководителей;

– показатели производительности труда;

– используемую форму оплаты труда;

– размер заработной платы: тарифные ставки для рабочих и должностные оклады для служащих; размер дополнительной заработной платы и других выплат.

В состав *характеристик производственных фондов* входят:

– видовой состав основных и оборотных производственных фондов;

– первоначальная, восстановительная, остаточная стоимость основных производственных фондов;

– изношенность, используемые виды и нормы амортизации основных производственных фондов;

– производственная мощность предприятия.

Параметры финансовых ресурсов:

– структура финансовых ресурсов по источникам формирования: размер собственного капитала, долгосрочных и краткосрочных обязательств (кредиты, займы, кредиторская задолженность);

– структура оборотных средств.

*Характеристики контракта* определяются в момент его заключения и в дальнейшем остаются неизменными. К ним относятся данные о видах продукции и объемах их отгрузки, ценах, формах и сроках оплаты, сроке действия контракта.

Параметры *налоговой системы* представляют собой множество видов налогов, для каждого элемента которого определена ставка и объект налогообложения, а также источник, из которого предприятие может его оплачивать [51]. Параметры *денежно-кредитной системы* определяются объемами, предполагаемыми сроками и процентными ставками банковских кредитов; показателями уровня инфляционных процессов.

В состав *характеристик спроса* входят параметры потока заказов на продукцию: интенсивность поступления заказов; параметры поступающих заказов: запрашиваемый ассортимент, объемы и сроки поставки продукции, формы и сроки оплаты.

Предложение материальных и энергетических ресурсов определяется объемом, сроками поставки, ценой, формой оплаты по каждому виду сырья, материалов, энергоресурсов. Параметры денежно-кредитной системы, показатели спроса на выпускаемую продукцию, показатели предложения используемых материальных и энергетических ресурсов характеризуют внешнюю среду предприятия, являются неуправляемыми и определяются измерениями или оценками. Параметры контрактов относятся к управляемым параметрам. Остальные параметры могут полагаться неизменными в ходе моделирования или изменяться исследователем. Представляется важным учитывать следующие основные аспекты деятельности промышленного предприятия, характеризующие функционирование и взаимодействие соответствующих реально существующих подсистем: планирование производства; заключение и контроль выполнения контрактов; управление финансами; бухгалтерский учет; ведение складского хозяйства; производство продукции.

*Планирование производства* рассматривается в виде обобщенного планирования объема продаж основных видов готовой продукции и составления плана-графика производства.

Обобщенное планирование объема продаж включает планирование спроса и планирование объемов продаж основных видов готовой продукции.

*Планирование спроса* описывается следующими переменными:

$D_{C_i}$  – спрос на  $i$ -й вид продукции (в стоимостном выражении);

$\tau_D$  – временной период, на котором определяется планирование спроса (в рабочих днях);

$D_{C\tau_D i}$  – объем спроса на временной период  $\tau_D$  по  $i$ -му виду продукции (в стоимостном выражении);

Для планирования объема продаж основных видов готовой продукции используются переменные:

$\tau_{SV}$  – временной период, в течение которого планируются продажи основных видов готовой продукции,  $\tau_{SV} \leq \tau_D$ ;

$R_{\Pi\Pi}$  – планируемая рентабельность предприятия (в относительных единицах);

$FC$  – постоянные затраты в целом по предприятию (в стоимостном выражении);

$(VC_{1i})$  – вектор переменных затрат на единицу  $i$ -го вида продукции (в стоимостном выражении);

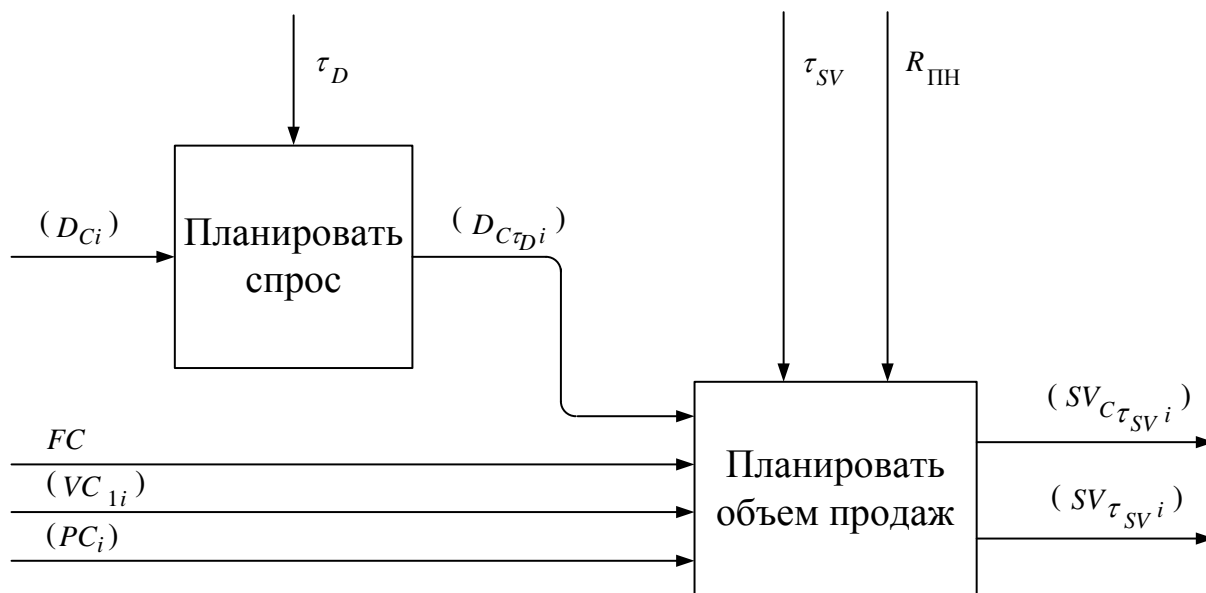
$(PC_i)$  – вектор производственных мощностей оборудования предприятия по  $i$ -му виду продукции (в количестве продукции за один рабочий день);

$SV_{C\tau_{SV} i}, i = \overline{1, n_{A\Sigma}}$  – объем продаж  $i$ -й группы продукции, запланированный на период  $\tau_{SV}$  (в стоимостном выражении);

$SV_{\tau_{SV} i}, i = \overline{1, n_{A\Sigma}}$  – объем продаж  $i$ -й группы продукции за период  $\tau_{SV}$  (в количественном выражении);

$n_{A\Sigma}$  – общее количество основных видов продукции, выпускаемой на предприятии.

Концептуальная модель обобщенного планирования объема продаж показана на рисунке 3.10.



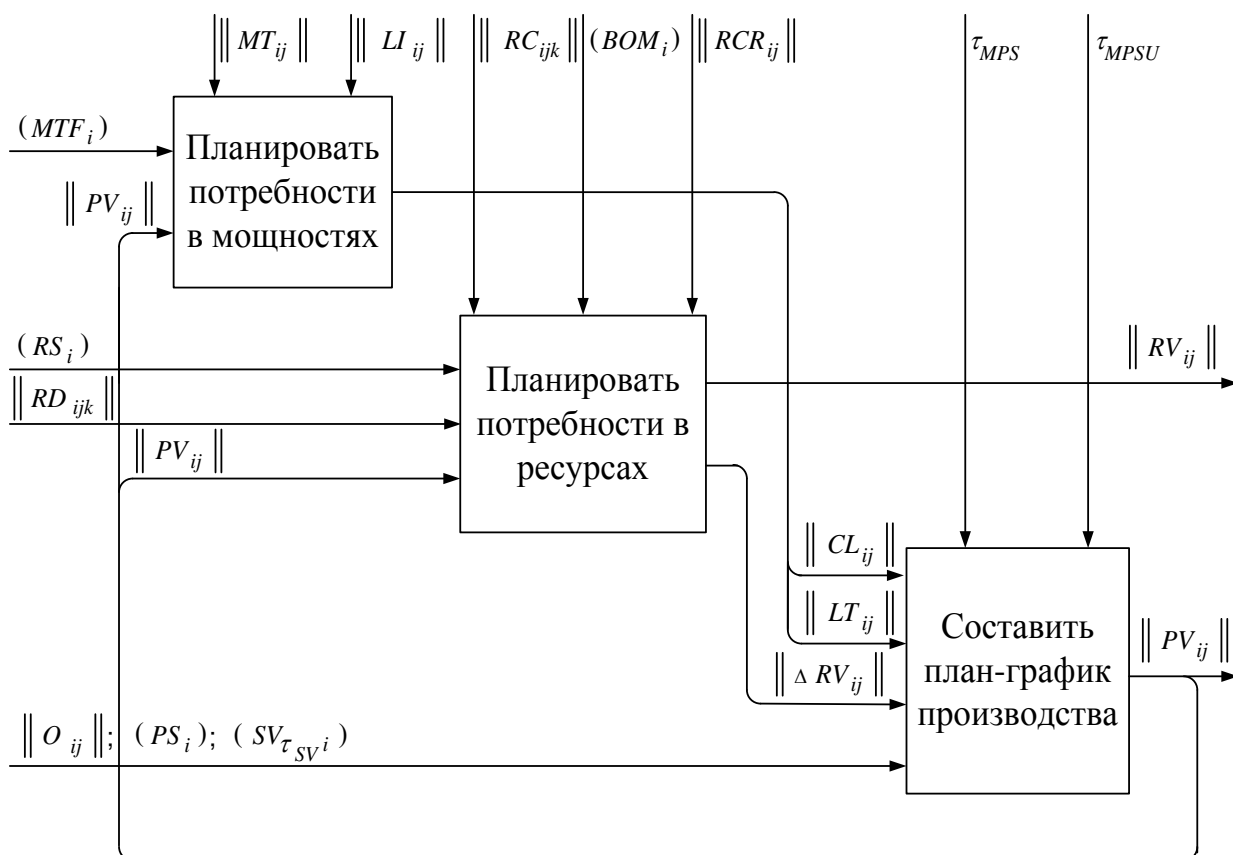
**Рисунок 3.10 – Концептуальная модель обобщенного планирования объема продаж**

Деятельность служб предприятия по составлению плана-графика производства включает: непосредственно составление плана-графика производства продукции; планирование потребности в ресурсах; планирование потребности в производственных мощностях. Составление плана-графика производства (рисунок 3.11) представлено переменными [109]:

$\tau_{MPS}$  – временной период, на который разрабатывается план-график производства (в рабочих днях);

$\tau_{MPSU}$  – временной интервал в периоде разработки плана-графика производства продукции (в рабочих днях);

$\|PV_{ij}\|$  – объемы выпуска  $i$ -го вида готовой продукции в  $j$ -м интервале планирования,  $i = \overline{1, n_A}$ ,  $j = \overline{1, \tau_{MPS}/\tau_{MPSU}}$  (в порядковом исчислении).



**Рисунок 3.11 – Концептуальная модель составления плана-графика производства**

Для формализации планирования потребности в ресурсах определены следующие переменные:

$\|RC_{ijk}\|$  – потребности в  $j$ -м материальном ресурсе для изготовления единицы  $i$ -го вида продукции,  $i = \overline{1, n_A}$ ,  $j = \overline{1, n_R}$  в  $k$ -ом интервале планирования (в стоимостном выражении);

$\|RV_{ij}\|$  – потребности в  $i$ -м материальном ресурсе в  $j$ -м интервале планирования,  $i = \overline{1, n_R}$ ,  $j = \overline{1, \tau_{MPS}/\tau_{MPSU}}$  (в количественном выражении).

$\|\Delta RV_{ij}\|$  – разности фактического и требуемого количества  $i$ -го материального ресурса в  $j$ -м интервале,  $i = \overline{1, n_R}$ ,  $j = \overline{1, \tau_{MPS}/\tau_{MPSU}}$  (в количественном выражении).

Планирование потребности в мощностях представляется использованием ресурсов предприятия:

$\|CL_{ij}\|$  – загрузки  $i$ -го оборудования в  $j$ -м интервале планирования;

$\|LT_{ij}\|$  – затраты  $i$ -го вида труда в  $j$ -м интервале планирования.

Процесс заключения и выполнения контрактов представлен следующим набором переменных [109]:

$\{PR_i\}$  – заявки на поставку продукции;

$\{PR_{\tau i}\}$  – множество проработанных заявок на поставку продукции, подлежащих выполнению в периоде времени  $\tau$ ;

$\{CS_i\}$  – условия заключения контрактов;

$\{C_i\}$  – множество заключенных контрактов;

$\|O_{ijk}\|$  – объемы отгрузок по  $k$ -м контрактам  $i$ -го вида продукции в  $j$ -м интервале планирования,  $i = \overline{1, n_A}$ ,  $j = \overline{1, \tau_{MPS} / \tau_{MPSU}}$ ;

$\|RD_{ijk}\|$  – объемы поставок по  $k$ -м контрактам  $i$ -го вида ресурса в  $j$ -м интервале,  $i = \overline{1, n_R}$ ,  $j = \overline{1, \tau_{MPS} / \tau_{MPSU}}$ ;

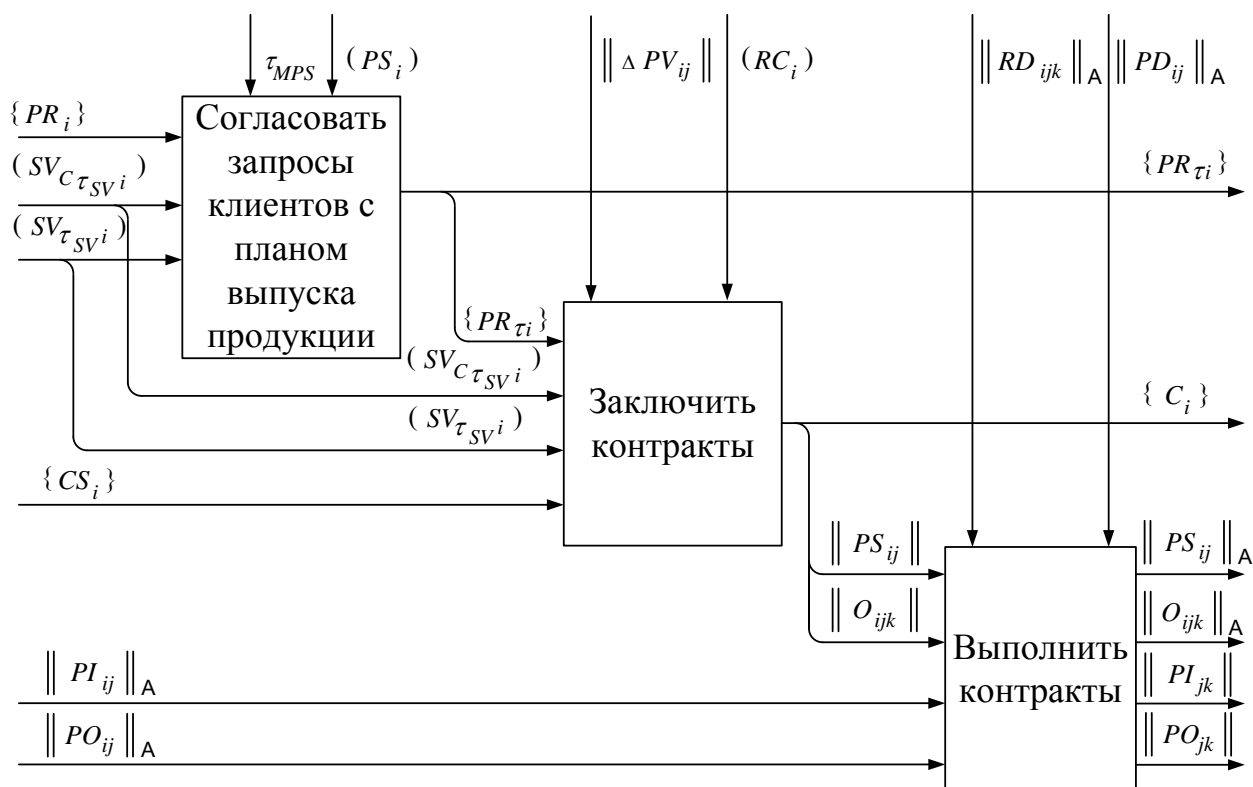
$RC_i$  – цена  $i$ -го ресурса,  $i = \overline{1, n_R}$ ;

$\|PI_{jk}\|$ ,  $\|PO_{jk}\|$  – матрицы входящих и исходящих платежей по контрактам в  $j$ -м интервале планирования,  $j = \overline{1, \tau_{MPS} / \tau_{MPSU}}$ .

$\|O_{ijk}\|_A$  – выполненные отгрузки  $i$ -го вида продукции в  $j$ -м периоде по  $k$ -ому контракту;

$\|RD_{ijk}\|_A$  – фактические объемы поставок по  $k$ -м контрактам  $i$ -го вида ресурса в  $j$ -м периоде;

$\|PI_{ij}\|_A$ ,  $\|PO_{ij}\|_A$  – матрицы поступивших денежных средств и произведенных платежей по контрактам в  $j$ -м периоде,  $i$  – номер контракта (рисунок 3.12).



**Рисунок 3.12 – Концептуальная модель заключения и выполнения контрактов**

Подсистемы управления финансами и бухгалтерского учета описываются переменными [109]:

$(B_i)$  – текущий бухгалтерский баланс,  $i$  – номер статьи баланса;

$(IS_i)_\tau$  – отчет о прибылях и убытках за период времени  $\tau$ ,  $i$  – номер статьи отчета;

$\|(S_v; S_c)_i\|_\tau$  – объемы реализации продукции в количественном ( $S_v$ ) и стоимостном ( $S_c$ ) выражении за период времени  $\tau$ ,  $i$  – вид продукции;

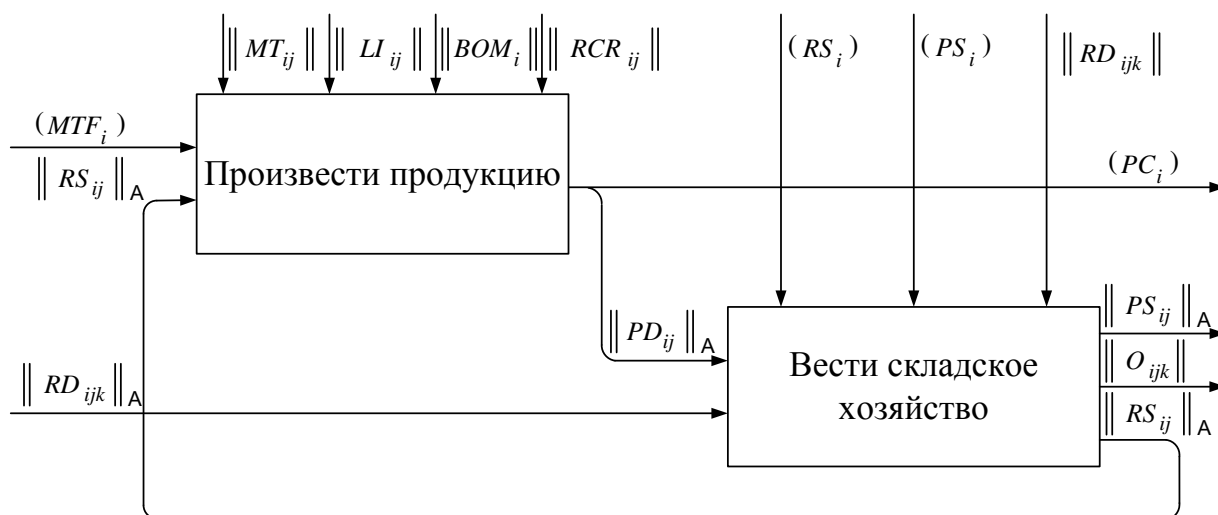
$\|BPAC_{io}; BPAC_{ir}; BPAC_{id}\|$  – первоначальная, восстановительная, остаточная стоимость основных производственных фондов  $i$ -го вида;

$DR_i$  – норма амортизации основных производственных фондов  $i$ -го вида;

$\{L_i\}$  – множество кредитов банков;  $L_i = (S_i; T_i; k_{\%i})$ , где  $S_i$  – сумма,  $T_i$  – срок,  $k_{\%i}$  – процентная ставка  $i$ -го кредита;

$\|PI_{ij}\|$ ,  $\|PO_{ij}\|$  – матрицы поступлений денежных средств и произведенных платежей по контрактам в  $j$ -м периоде,  $i$  – номер контракта.

Концептуальная модель складского хозяйства и производства продукции в графическом виде представлена на рисунке 3.13.



**Рисунок 3.13 – Концептуальная модель складского хозяйства и производства продукции**

Работа складского хозяйства представлена с помощью следующих переменных [109]:

$\|RD_{ijk}\|_A$  – поступления на склад  $i$ -го вида материальных ресурсов в  $j$ -м периоде по  $k$ -м контрактам;

$\|O_{ijk}\|_A$  – выполненные отгрузки готовой продукции  $i$ -го вида в  $j$ -м периоде по  $k$ -му контракту;

$\|PD_{ij}\|_A$  – поступившие на склад объемы готовой продукции  $i$ -го вида в  $j$ -м периоде;

$\|RS_{ij}\|_A$  – объемы отпущенных со склада материальных ресурсов  $i$ -го вида в  $j$ -м периоде;

$\|PS_{ij}\|_A$  – отгрузки со склада  $i$ -го вида готовой продукции в  $j$ -м периоде;



$\|O_{ijk}\|$  – объемы отгрузок по  $k$ -м контрактам  $i$ -го вида продукции в  $j$ -м интервале планирования;

$\|RD_{ijk}\|$  – объемы поставок по  $k$ -м контрактам  $i$ -го вида ресурса в  $j$ -м интервале планирования;

$(PS_i)$  – запасы готовой продукции на складе,  $i = \overline{1, n_A}$ ;

$(RS_i)$  – запасы материальных ресурсов на складе,  $i = \overline{1, n_R}$ .

Для характеристики производственного процесса определены следующие переменные:

$MTF_i$  – фонд машинного времени  $i$ -го вида;

$BOM_i$  – граф-спецификация  $i$ -го изделия (определяет состав и количество комплектующих  $i$ -го изделия);

$RCR_{ij}$  – нормативное значение потребления  $i$ -го материального ресурса на изготовление единицы  $j$ -го вида продукции;

$MT_{ij}$  – нормативное значение расхода фонда машинного времени  $i$ -го вида на изготовление единицы  $j$ -го вида продукции;

$LI_{ij}$  – норма затрат ()  $i$ -го вида труда (по профессиям) на изготовление единицы  $j$ -го вида продукции.

$(RC_i)$  – потребность в ресурсах в текущем периоде,  $i = \overline{1, n_R}$ .

Внешняя среда рассматривается состоящей из потребителей продукции предприятия; поставщиков материальных и энергетических ресурсов; денежно-кредитной и налоговой подсистемы государства.

Потребители продукции предприятия формируют поток заказов на продукцию предприятия со следующими характеристиками [109]:

$V_Z$  – интенсивность поступления заказов;

$P_{Ai}$ ,  $i = \overline{1, n_A}$  – вероятности запроса  $i$ -го наименования продукции;  $n_A$  – общее количество наименований продукции;

$\xi_{O_i}, \xi_{T_i}, i = \overline{1, n_A}$  – случайные величины, характеризующие соответственно запрашиваемые объемы и сроки поставки  $i$ -го наименования продукции;

*Характеристики потока заказов* определяются путем статистических расчетов реального потока заказов или соответствующих оценок.

*Поставщики материальных и энергетических ресурсов* определяют условия поставки материальных ресурсов и энергетических ресурсов, для их описания используются следующие переменные:

$T_{R_i}, i = \overline{1, n_R}$  – минимальный срок поставки  $i$ -го ресурса;

$C_{R_i}, i = \overline{1, n_R}$  – цена  $i$ -го ресурса;

$n_R$  – общее количество используемых видов ресурсов.

В элементе «государство» выделены две основные подсистемы, с которыми взаимодействует предприятие: денежно-кредитная система; налоговая система.

*Денежно-кредитная система* представлена набором переменных:

$k_{\%}$  – процентная ставка за банковский кредит для исследуемого предприятия;

$\tau_{\%}$  – период уплаты процентов за банковский кредит;

$k_{\alpha}$  – уровень инфляции за месяц.

Уровень процентной ставки представлен в следующем виде [21]:

$$k_{\%} = k_{\%}^* + ПИ + ПКР + ПЛ + ПРСП ,$$

где  $k_{\%}^*$  – процентная ставка, свободная от рисков; ПИ, ПКР, ПЛ, ПРСП – премии за инфляцию, кредитный риск, ликвидность и за риск, связанный со сроком погашения, соответственно.

Значения величины процентной ставки и уровня инфляции, закон их изменения в течение периода исследования устанавливаются перед началом моделирования на основе результатов обработки экспертных оценок развития финансового рынка.

*Налоговая система* представлена множеством налогов:

$$N = \{(k_{Ni}; o_{Ni}; s_{Ni}; t_{Ni})\}, i = \overline{1, n_N},$$

где  $k_{Ni}$  – ставка  $i$ -го налога;  $o_{Ni} \in O_N$  – объект  $i$ -го налога из множества возможных объектов налогообложения;  $s_{Ni} \in S_N$  – источник, из которого предприятие может оплачивать  $i$ -й налог, из множества возможных источников уплаты налогов;  $t_{Ni}$  – срок оплаты  $i$ -го налога;  $n_N$  – общее количество налогов.

Виды налогов, их ставки, объекты налогообложения, источники оплаты определяются действующим законодательством.

### 3.5 Формализация основных компонентов в базовой имитационной модели

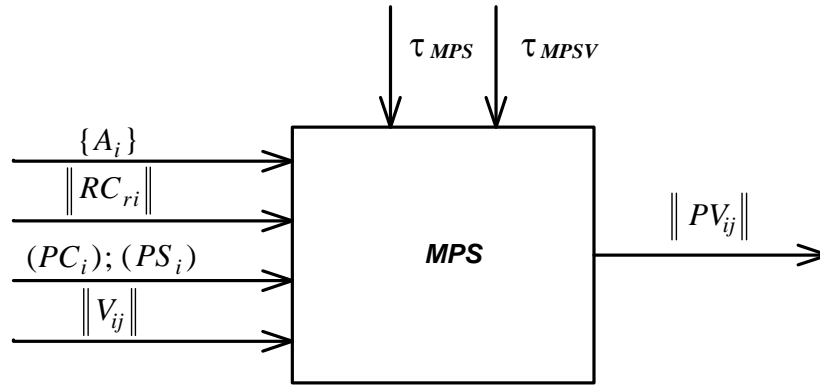
**План производства (MPS) в базовой имитационной модели.** Формализация основных компонентов БИМ в соответствии с концепцией MRP II представлена на примере составления плана производства (MPS) [120].

Переменные для моделирования MPS (рисунок 3.14):  $\{A_i\}$  – множество основных видов выпускаемой продукции;  $PC_i$  – производственная мощность оборудования предприятия по  $i$ -му виду продукции (в количестве продукции за один рабочий день),  $(PS_i)$  – складские запасы  $i$ -го вида готовой продукции;  $RC_{ri}$  – нормативное значение расхода  $r$ -го вида материального ресурса на производство единицы  $i$ -го вида продукции,  $\|V_{ij}\|$  – объемы отгрузки  $i$ -го вида готовой продукции в  $j$ -м интервале планирования;  $\tau_{MPS}$  – временной период планирования;  $\tau_{MPSV}$  – временной интервал планирования;  $\|PV_{ij}\|$  – объемы производства  $i$ -го вида продукции в  $j$ -м интервале планирования.

План производства  $\|PV_{ij}\|$  является результатом выполнения операторов  $P_1, \dots, P_6$ , представленных следующей логической схемой:

$$\|PV_{ij}\| = P_1 P_2 P_3 P_4 P_5 P_6, \quad (3.27)$$

каждый из которых детально рассмотрен далее.



**Рисунок 3.14 – Формализация составления плана производства**

$P_1$  – формирование предварительного варианта плана  $\|PV_{ij}\|$  с учетом интенсивности реализации готовой продукции:

$$\|PV_{ij}\| = \|V_{ij}\|, i = 1, \dots, n_A; j = 1, \dots, \tau_{MPS} / \tau_{MPSV}, \quad (3.28)$$

$$V_{ij} = SVD(a_1, \dots, a_{|A|})_i \cdot \tau_{MPSV} / SID(a_1, \dots, a_{|A|})_i, \quad (3.29)$$

где  $SVD(a_1, \dots, a_{|A|})_i$  – функция, задающая объемы  $i$ -го вида готовой продукции, отгружаемой потребителю, в общем случае по случайному закону с параметрами  $a_1, \dots, a_{|A|}$ , например,  $SVD(a_1, \dots, a_{|A|}) = Norm(a_1, a_2)$  для нормального закона распределения случайной величины ( $a_1$  – математическое ожидание;  $a_2$  – среднеквадратическое отклонение);  $SID(a_1, \dots, a_{|A|})$  – функция, задающая длительность временного интервала между отгрузками  $i$ -го вида продукции в общем случае по случайному закону с параметрами  $a_1, \dots, a_{|A|}$ , например,  $SID(a_1, \dots, a_{|A|}) = Exp(a_1)$  для экспоненциального закона распределения случайной величины ( $a_1 = \lambda$ , где  $\lambda$  – интенсивность случайной величины).

$P_2$  – оператор удаления произведенных отгрузок продукции из предварительного списка реализаций  $\|P_{ijk}\|$  и создание нового списка  $\|P'_{ijk}\|$ :

$$\|P'_{ijk}\| = \begin{cases} 0 & | V_{ijk} = P_{ijk}; \\ P_{ijk} & | V_{ijk} = 0; \\ P_{ijk} - V_{ijk} & | 0 < V_{ijk} < P_{ijk}, \end{cases} \quad (3.30)$$

$$i = 1, \dots, |V|; j = 1, \dots, |MPSH|; k = 1, \dots, |HS|,$$

где  $|MPSH|$  – количество временных интервалов планирования;  $|HS|$  – количество отгрузок готовой продукции в одном интервале планирования.

Следует заметить, что неотгруженная продукция в  $k$ -ой отгрузке должна быть отгружена в  $(k+1)$ -ой отгрузке в первую очередь.

$P_3$  – сортировка списка отгрузок для определения продукции, отгружаемой, в первую очередь, в следующей отгрузке:

$$P_3 : \|P'_{ijk}\| \xrightarrow{<_k} \|P'_{ijk}\|_{<_k}, \quad (3.31)$$

где  $<_k$  – отношение порядка.

$P_4$  – составление предварительного плана следующей отгрузки, включая просроченные отгрузки, с учетом известных отгрузок будущих периодов:

$$\|P'_{ij(k+1)}\| = \begin{cases} P_{ij(k+1)} & | V_{ijk} = P_{ijk}; \\ P_{ij(k+1)} + (P_{ijk} - V_{ijk}) & | 0 < V_{ijk} < P_{ijk}; \\ P_{ij(k+1)} + P_{ijk} & | V_{ijk} = 0, \end{cases} \quad (3.32)$$

$$i = 1, \dots, |A|; j = 1, \dots, |MPSH|; k = 1, \dots, |HS|.$$

где  $\|P'_{ij(k+1)}\|$  – предварительный план для  $(k+1)$ -ой отгрузки готовой продукции,  $P_{ijk}$  – план выполнения  $k$ -ой отгрузки продукции,  $V_{ijk}$  – продукция  $i$ -го вида в  $j$ -м временном интервале планирования в  $k$ -ой реализации;  $|A|$  – общее количество видов продукции.

$P_5$  – корректировка плана производства с учетом фактических запасов готовой продукции на складе:

$$\|PV_{ij}''\| = \begin{cases} 0 & \left| PS_{ij} \geq \sum_{k=1}^{|HS|} PV_{ijk}; \right. \\ P_{ij} & \left| PS_{ij} = 0; \right. \\ P_{ij} - PS_{ij} & \left| PS_{ij} < \sum_{k=1}^{|HS|} PV_{ijk}, \right. \end{cases} \quad (3.33)$$

$$i = 1, \dots, |A|; j = 1, \dots, |MPSH|,$$

где  $\|PV_{ij}''\|$  – скорректированный план производства с учетом фактических запасов готовой продукции на складе;  $PS_{ij}$  – складские запасы  $i$ -ой готовой продукции в  $j$ -м интервале планирования;  $PV_{ijk}$  – объем  $k$ -ой отгрузки  $i$ -ой готовой продукции в  $j$ -м интервале планирования.

$P_6$  – составление плана производства с учетом длительности производственного цикла, удаление предыдущего плана:

$$\|PV_{ij}'''\| = \|PV_{i(j-|C_i|)}''\|, \quad i = 1, \dots, |A|; j = 1, \dots, |MPSH|, \quad (3.34)$$

$$|C_i| = \left\lceil \frac{PV_i/PC_i}{\tau_{MPSTV}} \right\rceil, \quad (3.35)$$

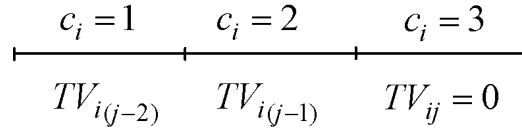
где  $\|PV_{ij}'''\|$  – план производства,  $PC_i$  – производственная мощность оборудования предприятия по  $i$ -му виду продукции, ед./дн.;  $\tau_{MPSTV}$  – длительность каждого из интервалов планирования, дней;  $\lceil \cdot \rceil$  – округление до целого числа в сторону больших значений,  $|C_i|$  – количество интервалов планирования для производства заданного объема  $i$ -го вида продукции,  $C_i = \{c_i\} = \{1, 2, \dots, |C_i|\}$ .

На интервалах планирования определяется также незавершенное производство  $TV_i$ :

$$TV_i(j - (|C_i| + i)) = \begin{cases} PV_{ij} - (|C_i| - C_i) \cdot (PC_i \cdot \tau_{MPSTV}), & |c_i| < |C_i|; \\ 0 & |c_i| = |C_i|, \end{cases} \quad (3.36)$$

$$i = 1, \dots, |A|; j = 1, \dots, |MPSH|; c_i = 1, \dots, |C_i|.$$

Например,  $c_i = 1, 2, 3$  (рисунок 3.15).



**Рисунок 3.15 – Формализация незавершенного производства**

Тогда (см. рисунок 3.15) имеем:

$$TV_{i(j-2)} = PV_{ij} - 2 \cdot PC_i \cdot |\tau_{MPSTV}|, \quad c_i < |c_i|, \quad (3.37)$$

$$TV_{i(j-1)} = PV_{ij} - 1 \cdot PC_i \cdot |\tau_{MPSTV}|, \quad c_j < |c_j|, \quad (3.38)$$

$$TV_{ij} = 0, \quad c_i = |c_i|. \quad (3.39)$$

График поставки ресурсов для требуемого интервала планирования:

$$R_{rj} = PV_{ij} \cdot RC_{ri} \mid r = 1, \dots, |R|; \quad (3.40)$$

$$i = 1, \dots, |A|; j = 1, \dots, |MPSH|,$$

где  $RC_{ri}$  – нормированное значение расхода  $r$ -го вида материального ресурса на изготовление единицы  $i$ -го вида продукции.

**Планирование потребностей в сырье и материалах (MRP).** Переменные, используемые для моделирования  $MRP$  представлены на рисунке 3.16:  $RS_r$  – запасы  $r$ -го вида ресурса на складе;  $RP_r$  – цена  $r$ -го вида ресурса;  $RV_r$  – объемы заказанного, но не полученного ресурса  $r$ -го вида;  $\|R_{rj}\|$  – матрица потребности ресурсов, в которой элементы представляют объем необходимого  $r$ -го вида материального ресурса в  $j$ -м интервале планирования,  $r = 1, \dots, |R|$ ;  $j = 1, \dots, \tau_{MPS} / \tau_{MPSV}$ , где в свою очередь  $|R|$  – общее количество видов ресурсов;  $\tau_{MPS}$  – длительность моделирования;  $\tau_{MPSV}$  – длительность интервала планирования в модели;

$W_{VAT}$  – ставка налога на добавленную стоимость (НДС);

$t$  – текущее время моделирования;

$DR_r$  – заказ на поставку ресурса  $r$ -го вида

$$DR_r = \langle DR_r^d, DR_r^t, DR_r^v, DR_r^p \rangle, \quad (3.41)$$

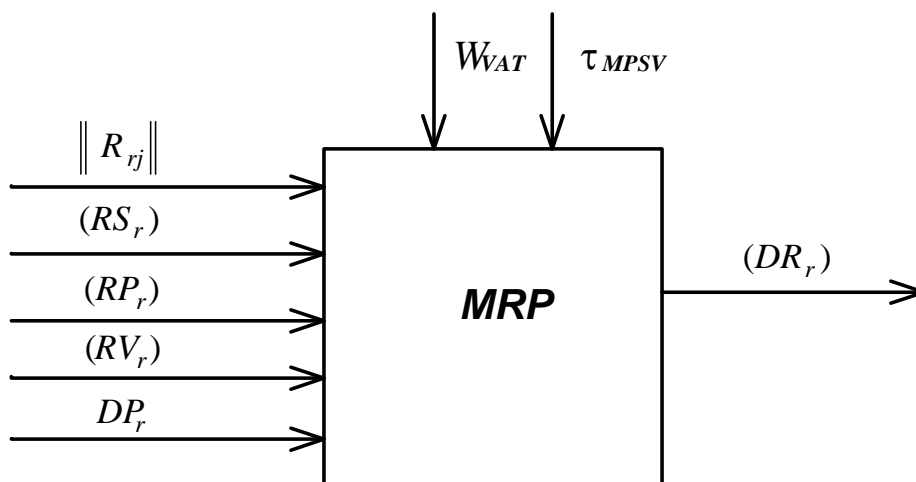
где в свою очередь:

$DR_r^d$  – срок оплаты заказа на поставку ресурса  $r$ -го вида,

$DR_r^t$  – время до оплаты заказа на поставку ресурса  $r$ -го вида,

$DR_r^v$  – объем  $r$ -го вида ресурса в заказе,

$DR_r^p$  – цена  $r$ -го вида ресурса в заказе (рисунок 3.16).



**Рисунок 3.16 – Управление поставками ресурсов**

Формализация заказов на поставку материальных ресурсов с учетом их потребности  $\|R_{rj}\|$  в  $j$ -м периоде планирования представлен ниже. Объем ресурса  $r$ -го вида в заказе на поставку в начале моделирования:

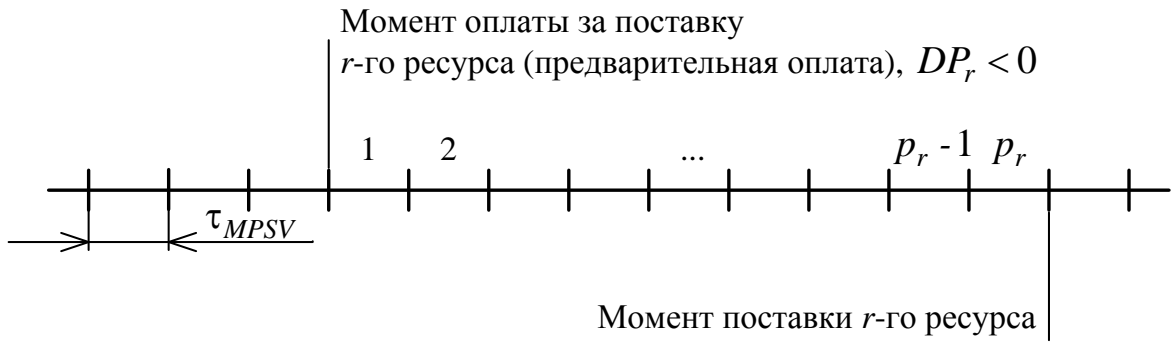
$$DR_r^v = \begin{cases} R_{r1} - RS_r | DP_r \geq 0; \\ R_{rp_r} + R_{HBO_r} - (RV_r + RS_r) | DP_r < 0, \end{cases} \quad (3.42)$$

где  $R_{r1}$  – потребность в  $r$ -м ресурсе для начального (первого) интервала планирования;

$RS_r$  – запас  $r$ -го вида ресурса на складе в текущий момент моделирования  $t$ ;

$DP_r$  – срок оплаты за ресурс  $r$ -го вида (определяется условиями контракта с поставщиком  $r$ -го ресурса) (рисунок 3.17).





**Рисунок 3.17 – Предварительная оплата поставки  $r$ -го ресурса**

Количество интервалов планирования, предшествующих поставке ресурса  $r$ -го вида, определяемых условиями предварительной оплаты ( $DP_r < 0$ )

$$|p_r| = \left\lceil \frac{-DP_r}{\tau_{MPSV}} \right\rceil, \quad (3.43)$$

где  $\lceil \cdot \rceil$  – операция округления до целого числа в большую сторону.

$$R_{HBO_r} = \sum_{j=1}^{|p_r|} R_{rj}, \quad j = 1, \dots, |p_r|, \quad (3.44)$$

$R_{HBO_r}$  – объем ресурса  $r$ -го вида на ближайшие интервалы планирования при предварительной оплате за  $r$ -й ресурс ( $DP_r < 0$ ).

$R_{rp_r}$  – объем ресурса  $r$ -го вида на момент поставки в  $p_r$ -м (последнем) интервале планирования, который рассчитывается подсистемой  $MPS$ ,  $r = 1, \dots, |R|$ .

Срок оплаты заказа на поставку ресурса  $r$ -го вида:

$$DP_r^d = DP_r, \quad r = 1, \dots, |R|, \quad (3.45)$$

где  $DP_r$  определяется некоторой функцией.

Время оплаты заказа на поставку  $r$ -го ресурса

$$DR_r^t = \begin{cases} t - DP_r & | DP_r < 0; \\ t & | DP_r \geq 0. \end{cases} \quad (3.46)$$

Цена  $r$ -го ресурса в заказе на поставку

$$DP_r^p = RP_r \cdot (1 + W_{VAT}), \quad r = 1, \dots, |R|. \quad (3.47)$$

Аналогично на основе логико-алгоритмической формализации представлены и остальные компоненты базовой модели (см. рисунок 3.9).

### 3.6 Построение имитационной модели промышленного предприятия

#### 3.6.1 Процессный способ имитации динамики взаимодействия компонентов промышленного предприятия

На промышленном предприятии, как правило, все функциональные действия  $w^{\ell\pm\tau}$  подсистем  $S^{\ell\pm\tau}$  на  $\ell\pm\tau$ -уровнях системы различны. Функционирование подсистем представляет собой последовательность  $w^{\ell\pm\tau}$ , которые выполняются на некотором временном интервале  $t^\ell$ . В результате выполнения  $w^{\ell\pm\tau}$  система переходит в состояние  $kC^\ell$ . Каждое из состояний  $kC^\ell$  связано с соответствующей компонентой  $K_i^{\ell\pm\tau} \in S^{\ell\pm\tau}$ . При построении ИМ  $w^{\ell\pm\tau}$  аппроксимируются упрощенными функциональными действиями  $w^{\ell\pm\tau*}$ . В ИМ каждое  $w^{\ell\pm\tau*}$  описывается в общем случае некоторым алгоритмом  $АЛ^{\ell\pm\tau}$  и выполняется за время  $\Delta t$ . Пару  $(АЛ^{\ell\pm\tau}, \Delta t)$  обычно называют активностью ИМ и обозначают  $АК^{\ell\pm\tau}$  [59].

Условия появления состояний, приводящих к выполнению  $w^{\ell\pm\tau}$ , также индивидуальны. У каждой компоненты существует определенная последовательность выполнения  $w^{\ell\pm\tau}$ . В определенный момент времени в заданной компоненте может выполняться только одно функциональное действие  $w^{\ell\pm\tau}$ .

При процессном способе организации квазипараллелизма декомпозиция системы определяется следующим. Во-первых, каждой компоненте  $K_i^{\ell\pm\tau} \in S^{\ell\pm\tau}$  соответствует свой процесс. Таким способом достигается соответствие структуры ИМ и реальной системы. Во-вторых, все  $w^{\ell\pm\tau}$  аппроксимируются соответствующими активностями, исходя из величины ошибки аппроксимации. Если ошибка аппроксимации некоторых  $w^{\ell\pm\tau}$  на интервале  $\Delta t$  велика, то этот интервал разбивается на несколько более мелких интервалов, на каждом из которых часть  $w^{\ell\pm\tau}$  аппроксимируется своей активностью. Таким образом, компоненты модели делят на последовательность активностей, для которых ошибки аппроксимации  $w^{\ell\pm\tau}$  находятся в допустимых пределах для данного исследования. Важным для процессного способа имитации является то, что условия свершения событий индивидуальны для каждой компоненты реальной системы  $K_i^{\ell\pm\tau} \in S^{\ell\pm\tau}$  и активности  $АК^{\ell\pm\tau}$  тесно взаимосвязаны между собой.

Процессным способом можно организовать имитацию любых достаточно сложных систем. Однако процессный способ наиболее эффективен, когда требуется высокий уровень детализации выполнения функциональных действий  $w^{\ell \pm \tau}$  при их аппроксимации с помощью АЛ $^{\ell \pm \tau}$  и сама ИМ используется для поиска так называемых узких мест в системе. При такой постановке задачи на моделирование очень важным является соблюдение структуры имитационной модели и объекта моделирования, что обеспечивается процессным способом моделирования. Процессный способ имитации обладает хорошими изобразительными возможностями при реализации многоуровневого и модульного подхода к моделированию систем. Это преимущество возрастает по мере роста сложности модели.

Вся имитационная модель может быть представлена в виде набора описаний процессов, каждый из которых соответствует одному определенному классу. Между компонентами и отдельными алгоритмами их функционирования могут быть установлены информационные и управляющие связи. Алгоритм функционирования ИМ представляется последовательным взаимодействием процессов и УПМ. Причем в процессы объединяются активности, определяющие функционирование одной и той же компоненты модели. Таким образом, достигается полное соответствие компонент реальной системы и ее ИМ, когда каждой компоненте объекта моделирования соответствует свой процесс [143].

### 3.6.2 Декомпозиция функционирования компонентов системы

Декомпозиция системы на компоненты проведена согласно функциональной схеме деятельности промышленного предприятия. Функциональные действия объединены по принадлежности к одной реальной компоненте системы. В результате получены компоненты, представленные в таблице 3.1.

Список компонентов следующий (см. таблицу 3.1):  $K_1$  – планирование производства;  $K_2$  – заключение и выполнение контрактов;  $K_3$  – финансы и бухгалтерский учет;  $K_4$  – производство продукции.

Таблица 3.1 – Перечень процессов ИМ промышленного предприятия

Компонента	Обозначение	Наименование
К <sub>1</sub>	ПК <sub>11</sub> : АК <sub>11</sub>	<i>Планирование производства:</i> - составление плана-графика производства
	ПК <sub>21</sub> : АК <sub>21</sub>	<i>Управление поставками материальных ресурсов:</i> - обработка заказов на поставку ресурсов
К <sub>2</sub>	ПК <sub>22</sub> : АК <sub>22.1</sub> АК <sub>22.2</sub> АК <sub>22.3</sub>	<i>Поставка ресурсов:</i> - предоплата поставки ресурсов; - выполнение поставки ресурсов; - оплата поставки ресурсов
	ПК <sub>23</sub> : АК <sub>23</sub>	<i>Поступление заказов на готовую продукцию:</i> - обработка заказов на отгрузку продукции
	ПК <sub>24</sub> : АК <sub>24.1</sub> АК <sub>24.2</sub> АК <sub>24.3</sub> АК <sub>24.4</sub>	<i>Реализация готовой продукции:</i> - генерация отгрузки готовой продукции; - отгрузка готовой продукции; - оплата готовой продукции; - предоплата готовой продукции
	ПК <sub>31</sub> : АК <sub>31</sub>	<i>Выдача заработной платы (ЗП):</i> - расчет и выдача ЗП производственным рабочим
К <sub>3</sub>	ПК <sub>32</sub> : АК <sub>32</sub>	<i>Уплата налогов:</i> - расчет и уплата налогов
	ПК <sub>33</sub> : АК <sub>33</sub>	<i>Оплата постоянных затрат и издержек:</i> - расчет и оплата постоянных затрат и издержек
	ПК <sub>34</sub> : АК <sub>34</sub>	<i>Получение кредитов:</i> - получение кредита
	ПК <sub>35</sub> : АК <sub>35.1</sub> АК <sub>35.2</sub>	<i>Обслуживание кредита:</i> - поступление денежных средств по кредиту - выплата процентов и погашение кредита
	ПК <sub>36</sub> : АК <sub>36</sub>	<i>Бухгалтерский учет:</i> - формирование статей бухгалтерского учета
	К <sub>4</sub>	ПК <sub>41</sub> : АК <sub>41</sub>
ПК <sub>42</sub> : АК <sub>42.1</sub> АК <sub>42.2</sub>		<i>Технологический процесс:</i> - запуск партии продукции в производство; - окончание производства продукции и поступление готовой продукции на склад
К <sub>5</sub>	ПК <sub>51</sub> : АК <sub>23</sub>	<i>Инфляция:</i> - расчет индекса инфляции и корректировка цен

Примечание. ПК – процесс компоненты; АК – активность процесса

Фактически отдельная компонента модели соответствует каждому блоку диаграммы верхнего уровня функциональной схемы (см. рисунок 3.7), за исключением функциональных действий по планированию объема реализации и формированию плана-графика производства, которые отнесены к одной компоненте, и ведению складского хозяйства, для которых отдельная компонента не выделяется.

Во внешней среде выделены следующие компоненты: потребители продукции (формируют поток заказов на продукцию предприятия); поставщики материальных и энергоресурсов; государство (денежно-кредитная и налоговая подсистемы).

Функциональные действия, соответствующие процессу выполнения контрактов внешними контрагентами, отнесены к компоненте  $K_2$ . Это позволяет моделировать все функциональные действия по выполнению контрактов с помощью одной компоненты ИМ. Полный перечень компонент ИМ приведен в таблице 3.1. Все компоненты взаимодействуют с другими посредством информационных и управляющих связей, которые соответствуют связям между блоками функций на функциональной схеме предприятия.

Функциональные действия отдельных компонент модели могут выполняться одновременно, поэтому имитацию таких функциональных действий необходимо выполнять отдельными процессами. При этом одна реальная компонента будет представлена несколькими процессами (приложение Б).

Конкретный состав процессов, соответствующих компоненте  $K_4$ , зависит от типа производственной структуры (предметный, технологический и предметно-технологический) и особенностей технологического процесса на конкретном предприятии. В случае предметной производственной структуры состав процессов компоненты  $K_4$  соответствует специализации подразделений предприятия на изготовление определенного изделия или группы однотипных изделий.

Если предприятие имеет технологическую структуру, то процессы выделяются в соответствии со специализацией цехов предприятия на выполнение определенной части технологического процесса или отдельной стадии произ-

водственного процесса. В первом случае отдельный процесс используется для моделирования производства изделия определенного вида, во втором – определенной части технологического процесса.

При предметной производственной структуре связи между процессами определяются спецификацией выпускаемых изделий, при технологической структуре производства – конкретным технологическим процессом. Если предприятие имеет смешанную производственную структуру, то часть процессов используется для моделирования производства отдельных видов продукции, а часть – для моделирования определенных технологических операций. Уровень детализации моделирования производства определяется выбранными показателями качества [109].

### **3.6.3 Технология модификации состава и структуры имитационной модели промышленного предприятия**

При эксплуатации имитационной модели для ее коррекции в реальных условиях деятельности промышленного предприятия требуется изменение структуры модели путем введения новых компонентов или процессов.

*Создание процессов* выполняет оператор  $ADD\_PROCESS(A_i, [t], [p])$ , где  $A_i$  – первая активность создаваемого процесса,  $t$  – время первого запуска активности (необязательный параметр),  $p$  – приоритет создаваемого процесса (необязательный параметр). Синхронизация процессов реализуется оператором  $WAIT(t, [A_i])$ , где  $t$  – время ожидания,  $A_i$  – здесь следующая активность создаваемого процесса (необязательный параметр). Завершение текущего процесса и удаление его из списка процессов УПМ реализуется оператором  $END$ .

Чтобы выполнить отладку ИМ, используются операторы  $WRITE$ , включающие программную реализацию соответствующей активности. При этом система моделирования (СМ) обеспечивает запись информации об активности (наименование, значение локального времени, глобальные данные модели) в  $XML$ -файл для дальнейшей обработки и поиска допущенных ошибок.

После завершения моделирования имеется возможность проведения обработки полученных результатов и анализа их с использованием средств автоматизации. Чтобы упростить процедуру верификации ИМ путем замены стохастические значений переменных детерминированными, при испытании модели можно зафиксировать последовательность значений генератора случайных чисел.

### **3.6.4 Формализация взаимодействия компонентов имитационной модели предприятия с внешней средой**

Исходные данные для ввода в ИМ представлены двумя составляющими: управляющими параметрами (регламентированы нормативными документами, например, нормированные значения расхода материальных ресурсов) и входными характеристиками. Управляющие параметры могут быть записаны в файл входных данных имитационной модели без предварительной статистической обработки и проверки на наличие грубых ошибок. Это вызвано тем, что актуальность и достоверность информации обеспечивается компанией, решающей задачу сопровождения КИС на предприятии, либо операторами, которые вносят данные из соответствующих нормативных документов в КИС. Поэтому получение управляющих параметров и ввод их в ИМ сводится к автоматизации импорта массивов требуемых данных из КИС в ИМ.

Запись входных характеристик в файл входных данных ИМ должна выполняться после их предварительной обработки, например, средствами статистического пакета *STATISTICA*. Такое требование обусловлено тем, что данные представлены в КИС предприятия в виде информации, записанной по показаниям контрольно-измерительных приборов, внесенной из товарно-транспортных накладных, из документов контрактов и иной документации, имеющейся на предприятии.

В КИС могут храниться недостоверные данные  $D_{нд}$ , которые определены ошибками измерения  $D_{ои}$ , ошибками снятия показаний контрольно-измерительных приборов  $D_{оп}$ , ошибками операторов при внесении информации в базу данных КИС  $D_{оо}$ :

$$D_{нд} = D_{ои} + D_{оп} + D_{оо} .$$

Для исключения недостоверной статистической информации при ее использовании, например, для получения аналитических зависимостей, которые могут быть использованы в процессе моделирования, выполняется следующая процедура:

1) импортируются необходимые данные из таблиц базы данных КИС в пакет *STATISTICA* через табличный процессор MS Excel или непосредственно, в зависимости от формата базы данных КИС (выгрузка данных в *MS Excel* поддерживается большинством современных КИС);

2) выполняется обработка импортированной статистической информации для исключения недостоверных данных с привлечением экспертов. Для автоматизации исключения недостоверных данных знания экспертов могут быть сохранены в системе *STATISTICA* посредством реализации специального макроса;

3) получают регрессионные зависимости, которые могут использоваться в процессах ИМ. Для каждой из входных характеристик, которые могут изменяться случайным образом, определяется тип закона распределения, среднее значение и стандартное отклонение.

Обработанные данные затем импортируются в файл данных модели [134, 143].

### 3.6.5 Исследование ИМ подсистем промышленного предприятия

Исследование свойств ИМ включает количественную оценку погрешности имитации; оценку времени так называемого разгона модели (переходного процесса) и оценку устойчивости полученных результатов; оценку чувствительности откликов модели при варьировании значений входных параметров. При этом следует отметить, что до настоящего времени, как отмечает Дж. С. Карсон (J. S. Carson II, 2003), отсутствуют общие правила для определения длины имитационного эксперимента и числа повторений опытов. В каждом случае все определяется моделью и решаемой задачей. Число повторений определяется точностью



исследуемой величины, задаваемой длительностью доверительного интервала [164].

Обработка статистики имитационных экспериментов для исследования свойств ИМ выполняется с использованием пакета *STATISTICA* и/или программных приложений, входящих в состав ПТКИ *BelSim*. Если некоторые процессы ИМ выполняют одинаковые функции, но при этом имеются разные алгоритмы их выполнения, то необходимо исследовать свойства ИМ для каждого варианта реализации процессов, что является *особенностью исследования ИМ* процессов функционирования промышленного предприятия. Выбор определенных алгоритмов приводит к различному составу процессов ИМ, вызывающих разные свойства ИМ.

*Погрешность моделирования* оценивают дисперсией  $D_{им}$ , которая вызвана принципиально неточным описанием исследуемых процессов (оценивается дисперсией  $D_{ап}$ ) и ошибками  $D_{сз}$ , обусловленными получением случайных значений параметров ИМ:

$$D_{им} = D_{ап} + D_{сз}.$$

В общем случае Б. В. Шмейсер (B. W. Schmeiser, 2001) определяет пять источников ошибок при исследовании модели: ошибки программирования (иначе ее называют ошибкой верификации; аналогично для моделирования ее называют ошибкой валидации); вычислительные ошибки, связанные с ограничениями вычислений на компьютере; ошибки случайных чисел (чем быстрее становятся компьютеры, чем больше размер памяти, тем более повышается чувствительность к ошибкам случайных чисел); ошибка формирования закона распределения случайной величины; ошибка статистической выборки (метод Монте-Карло является, по существу, статистическим методом, поэтому статистические ошибки неизбежны) [212].

*Стационарность* режима моделирования характеризуется некоторым установившимся равновесием процессов в модели, когда при дальнейшей имитации новой информации о системе не получают [59].

Под *устойчивостью* результатов моделирования, как правило, понимают сходимость контролируемого параметра к определенной величине при увеличении времени имитации варианта сложной системы [59]. Обычно устойчивость результатов моделирования оценивают дисперсией значений отклика. Если эта дисперсия при увеличении времени моделирования  $T_m$  не увеличивается, то результаты моделирования полагают устойчивыми [54].

Исследование переходного процесса (разгона модели) и устойчивости полученных результатов предполагает сбор статистики в ИМ для каждого отклика  $\{Y_c\}$  через определенные промежутки времени  $\Delta t_i$ . Соответственно, в *XML*-файл, сохраняющий результаты моделирования, значения откликов ИМ записываются через промежутки времени  $\Delta t_i$ . Затем выполняется анализ полученной статистики моделирования средствами пакета *STATISTICA*.

*Анализ длины переходного процесса* выполняется построением графиков зависимостей  $\{Y_c(t_i)\}$  и последующим определением времени  $\tau_i$  перехода в стационарный режим самого медленно изменяющегося отклика. Это время определяет длину исследуемого переходного процесса [130].

Чтобы *оценить устойчивость результатов имитации*, должны быть построены графические зависимости средних значений откликов ИМ от модельного времени  $\{\hat{Y}_c(t_i)\}$ . Результаты имитации устойчивы, если с ростом модельного времени не увеличиваются в несколько раз.

Исследование *чувствительности имитационной модели* выполняют по типовой методике, в соответствии с которой для каждого параметра модели  $\{X_c\}$  и внешней среды  $\{Z_c\}$  указывают средние значения и значения с отклонениями в обе стороны на определенную ширину интервала (уровни +1 и -1). Для каждого параметра проводятся эксперименты с уровнями +1 и -1 при значениях остальных параметров на уровне 0. После нахождения вектора откликов  $Y^{i(1)}$  и  $Y^{i(-1)}$  итоги экспериментов записываются в *XML*-файл для обработки в пакете *STATISTICA*. В результате исследований определяются отклики, наиболее чувствительные к изменению входных параметров [59].

Для ИМ промышленного предприятия зачастую *проверка адекватности* заменяется *верификацией* модели объекта, поскольку эта процедура является весьма затратным мероприятием. Оценка адекватности ИМ реальному объекту проводится в тех случаях, когда для моделируемой системы можно провести натурные эксперименты, в ходе которых можно определить отклики системы при известных значениях входных характеристик и управляющих переменных.

### **3.6.6 Задачи, решаемые на основе имитационной модели промышленного предприятия**

Разработанная модель промышленного предприятия, именуемая базовой имитационной моделью (БИМ), ориентирована в первую очередь для малых и средних предприятий нефтехимической и химической промышленности, легкого машиностроения, строительной индустрии, текстильной и легкой промышленности, деревообрабатывающих и целлюлозно-бумажных предприятий. В соответствии с ограничениями модели такие предприятия по характеру процесса переработки сырья могут быть с непрерывным и прерывным процессами производства, по типу производства – массового и серийного.

Предложенная имитационная модель производственно-экономической деятельности промышленного предприятия обеспечивает решение следующих задач:

а) *Оценка влияния алгоритмов организации бизнес-процессов в информационной системе (ИС) предприятия и их параметров на финансово-экономические показатели предприятия* для поиска путей реинжиниринга бизнес-процессов. В процессе комплексного анализа необходимо не просто найти рациональный вариант состава и структуры ИС на основе суперпозиции интегральных показателей качества функционирования отдельных подсистем ИС, но и учесть ограничения, накладываемые на каждую исследуемую подсистему другими подсистемами сложной системы. Для этого в ИМ подсистем включаются процессы, реализующие функции ИМ каждой подсистемы, которая передает входные характеристики и управляющие параметры исследуемой подсистеме.

Алгоритмы бизнес-процессов реализуют функции подсистемы на более высоком уровне детализации, чем ИМ других подсистем, используя аналитические зависимости и операторы ветвления. Выходные характеристики процесса влияют на отклики модели непосредственно или через промежуточные переменные. Для комплексного анализа систем в этом случае можно использовать метод многокритериальной оптимизации. Интегральный показатель качества функционирования сложной системы определяется на основе суперпозиции показателей качества отдельных компонентов [8, 143].

б) *Рациональный выбор состава, структуры и параметров системы управления производственными процессами для поиска путей снижения себестоимости продукции.* Для рационального выбора параметров производственного процесса принятие решений осуществляется на основе показателей его эффективности: производственная мощность, длительность производственного цикла, коэффициенты загрузки оборудования и себестоимость продукции. При этом может использоваться как метод главного критерия (которым является, как правило, себестоимость продукции), так и метод «свертки» компонентов вектора откликов для учета других показателей эффективности системы управления производственным процессом [143];

в) *Оценка выбора альтернативных вариантов – целесообразности запуска отдельного производства на краткосрочный период при повышенном спросе на продукцию или равномерный выпуск продукции с хранением ее на складе.* Модель планирования производственной нагрузки с жесткой координацией основана на том, что производственная нагрузка зависит от заказов, поступающих на уровне производственного объединения. Производственная нагрузка не может превышать некоторого максимального значения  $ВП_{maxi}$  и не может быть ниже  $ВП_{mini}$ , обусловленного особенностями технологического процесса. Продукция производится непрерывно, загружается в тару и перемещается на склад. Если заказы отсутствуют, производство останавливается и требуются дополнительные затраты на его запуск. Модель с интегральной координацией основана на постоянной производственной нагрузке, когда за определенный длительный

период времени требуется выпустить заданное количество продукции. Если спрос на продукцию отсутствует, она поступает на склад. С другой стороны, при отсутствии сырья, требуется брать кредиты для его закупки. Финансовые средства могут отсутствовать, когда продукция не реализуется. При этом следует определить максимально допустимые цены на закупку сырья и учитывать дополнительные складские затраты на хранение готовой продукции [108].

г) *Оценка целесообразности закупки сырья в период низких цен с последующим хранением на складе или ритмичная закупка сырья в течение заданного периода.* При определении максимально допустимых закупочных цен себестоимость продукции и время перехода на другой ассортимент используются в качестве ограничений в ходе оптимизации. Возможны следующие варианты постановки данной задачи:

1) Необходимо определить максимальное значение закупочной цены основного сырья. В этом случае задача сводится к максимизации значения закупочной цены для заданного вида основного сырья при однозначно заданном плане выпуска продукции, себестоимости продукции и времени перехода на другой ассортимент, не превышающем заданные значения. В качестве ограничения для каждого цеха предприятия задается также минимальная и максимальная производственная нагрузка на выпуск продукции;

2) Необходимо установить максимальное значение закупочной цены  $\Pi_i$  для  $N$  видов основного сырья ( $i=1, \dots, N$ ). В этом случае при поиске решения может быть использована процедура, аналогичная процедуре метода последовательных уступок: для выбранного одного вида сырья определяется максимальное значение закупочной цены  $\max \Pi_1$  при фиксированных регламентированных значениях цен на другие виды основного сырья. Для найденного максимального значения назначается некоторая уступка  $\Delta \Pi_1$ . Задавая в качестве ограничения  $\max \Pi_1 - \Delta \Pi_1 \leq \Pi_1 \leq \max \Pi_1$ , определяется максимальное значение для второго вида основного сырья  $\max \Pi_2$ . Далее назначается уступка для максимального значения цены второго вида основного сырья  $\Delta \Pi_2$  и т.д. [143].

д) *Оценка эффективности заключения контрактов на поставку необходимых ресурсов, реализацию готовой продукции с учетом экономических условий контракта.* Одна из главных составляющих маркетинговой деятельности предприятий – ценовая политика. При этом подразумевается использование ориентированной на потребителя системы обоснования цен. Различные условия контракта оказывают влияние на финансовое состояние предприятия. Контракт с отсрочкой платежа при реализации продукции приводит к потерям предприятия из-за невовлечения денежных средств в производственный оборот. Один из недостатков отсрочки платежа для продавца – сложность определения потерь от отсутствия оборотных средств и темпов инфляции и, соответственно, связанного с этим риска. Возможны следующие варианты решения задач оценки эффективности контрактов:

1) Анализ различных показателей финансовой деятельности предприятия (чистой прибыли, рентабельности и т.д.) при заключении контрактов купли, продажи и дилерского обслуживания и сравнение их с теми же показателями без заключения контрактов;

2) Оценка влияния процента и срока предоплаты, а также срока оплаты поставки в контракте купли с фиксированными ценами при одних и тех же исходных данных о состоянии предприятия на показатели финансовой деятельности предприятия; поиск наилучшего способа оплаты поставок по критерию, например, чистой прибыли;

3) Оценка влияния сроков поставок ресурсов в контракте купли с фиксированными ценами при одних и тех же исходных данных о состоянии предприятия на показатели финансовой деятельности предприятия;

4) Оценка влияния процента и срока предоплаты, а также срока оплаты отгрузки в контракте продажи при одних и тех же исходных данных о состоянии предприятия на показатели финансовой деятельности предприятия; поиск наилучшего способа оплаты отгрузок по критерию, например, чистой прибыли;

5) Оценка влияния сроков отгрузок продукции в контракте продажи при одних и тех же исходных данных о состоянии предприятия на показатели финансовой деятельности предприятия;

б) Оценка влияния интервалов между отгрузками, объемов продукции в отгрузках по контракту на дилерское обслуживание на показатели финансовой деятельности предприятия при фиксированных исходных данных о состоянии предприятия [143].

е) *Минимизация длительности переходного процесса при смене ассортимента выпускаемой продукции, сокращение потерь при переходных процессах, снижение себестоимости выпускаемой продукции.* Данные по основному сырью, вспомогательным материалам, энергетическим и постоянным затратам хранятся в КИС. На основе обработки статистических данных определяются зависимости расхода энергоресурсов от выработки. Полагают, что расход энергоресурсов не зависит от вида выпускаемого продукта. В качестве ограничения используется суточный план выпуска продукции, для которого по результатам моделирования задаются оптимальные значения интенсивности и продолжительности подачи технологической добавки [143].

ж) *Оптимальное распределение производственной нагрузки по цехам промышленного предприятия.* На основе обработки статистических данных, хранящихся в базах данных КИС выявляются зависимости расхода энергоресурсов, основного сырья, вспомогательных материалов и постоянных затрат от выработки продукции. Выявленные зависимости с течением времени могут изменяться в результате проведения организационно-технологических мероприятий. При известных значениях минимальной и максимальной суточной производительности каждого цеха, планируемом суточном плане выпуска требуется распределить производственную нагрузку между цехами для минимизации себестоимости продукции за счет уменьшения постоянных и энергетических затрат на единицу вида продукции [126].

### 3.7 Выводы по третьей главе

В третьей главе диссертационной работы получены следующие основные результаты.

1. Метод построения имитационной модели (ИМ) предприятия состоит в создании базовой ИМ с учетом его многоуровневой иерархической структуры. На основных этапах построения ИМ используются внутренние информационные ресурсы корпоративной информационной системы (КИС): построение вербальной и концептуальной модели с применением концепции SADT в нотации IDEF0; формализация процессов функционирования предприятия процессным способом моделирования; программная реализация ИМ с применением концепции ООП; верификация программы ИМ, автоматизация ввода исходной информации с применением XML-технологии; испытание ИМ системы с использованием коммерческих пакетов статистической обработки данных имитационного эксперимента.

2. Разработан итерационный метод организации распределенного проектирования ИМ сложных систем в КИС промышленных предприятий, отличающийся учетом человеческого фактора с использованием понятий ресурсов  $f$ -,  $p$ -типа и их информационных оценок, что позволяет оценить скорость, качество и риски выполнения проектных работ по разработке ИМ при используемых ресурсах.

3. Сформулирована и решена оптимизационная задача многокритериального управления технологическим процессом (ТП) организации проектирования ИМ производственно-экономической деятельности промышленного предприятия с КИС, основанной на использовании принципа оптимальности Беллмана. Показана возможность представления ТП в виде взвешенного ориентированного ациклического графа, вершинами которого являются этапы проектирования, нагрузки на дуги графа – стоимости ресурсов  $f$ -,  $p$ -типа, при этом принято лексикографическое упорядочение критериев оптимизации по стоимости ресурсов и/или временным затратам.

4. Разработана методика решения задачи определения последовательности оптимальных управлений и состояний ТП организации проектирования ИМ способом пошаговой оптимизации, в соответствии с которой решение задачи дости-



гается согласно методу динамического программирования по принципу оптимальности Беллмана с введением лексикографического упорядочения критериев оптимизации. Применимость методики доказана для одно- и двумерных характеристик потока на дугах графа ТП таких, что нагрузки на исходящих дугах, инцидентных одной и той же вершине, эквивалентны из-за особенностей временных и/или стоимостных затрат при проектировании модели на одном этапе.

5. При построении концептуальной модели использована многоуровневая структура промышленного предприятия. На каждом из уровней иерархии объект имитации описывается функциональными действиями и структурой элементов системы, включая координатора взаимодействий. Координирующий элемент определяется описателями внутренней и окружающей среды, задаваемых структурой внутриуровневых и межуровневых отношений и соответствующими функциональными действиями. Такое представление позволяет формализовать стратегии координации на каждом из уровней, построить ИМ и исследовать стратегии для определения их эффективности.

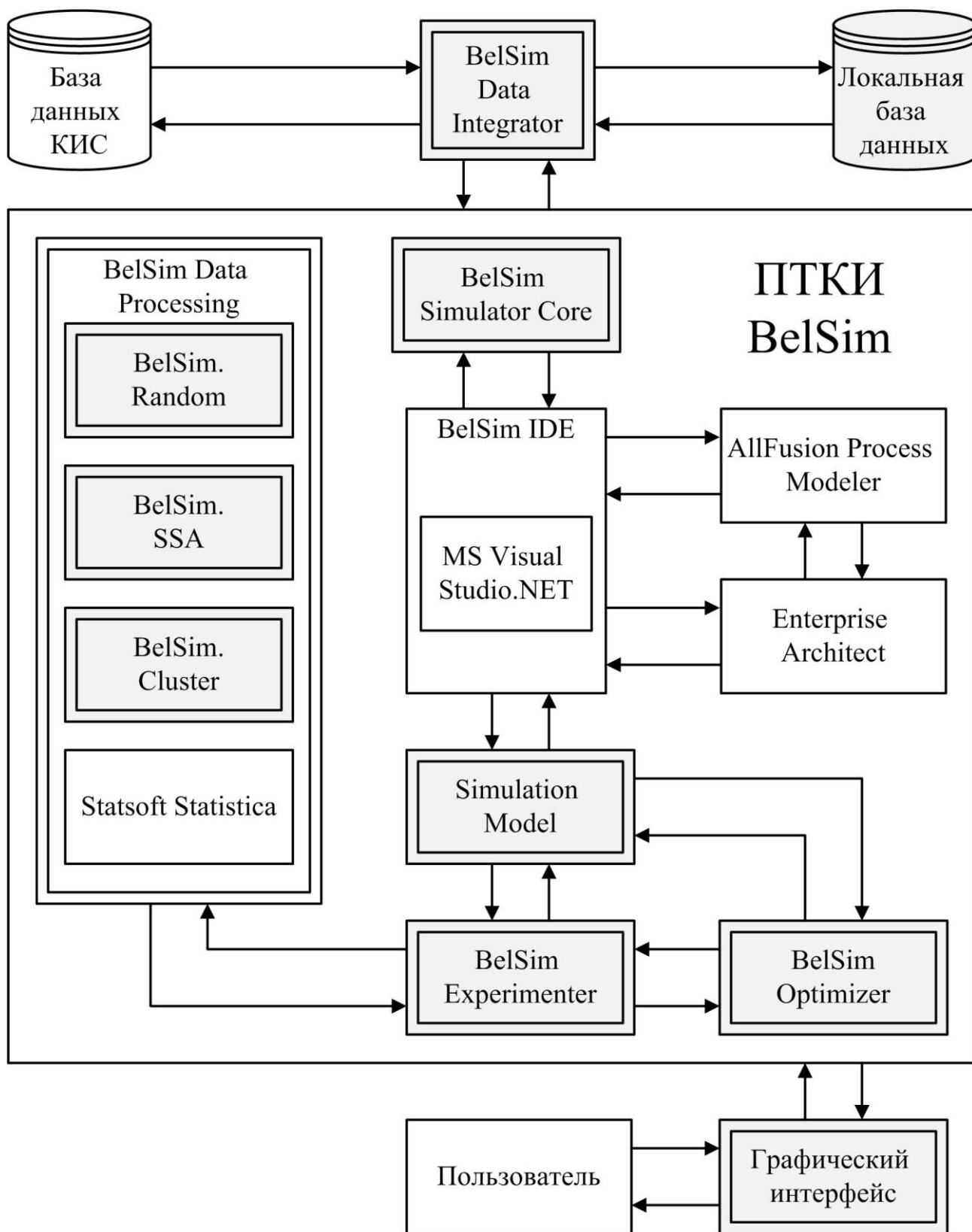
6. При разработке базовой имитационной модели функционирования предприятия использован системный подход к анализу сложной многоуровневой системы. Рассмотрены процессы планирования производства, снабжения ресурсами, управления финансами предприятия, организации производства и сбыта продукции, дано описание его окружающей среды.

## ГЛАВА 4. РАЗРАБОТКА ПРОГРАММНО-ТЕХНОЛОГИЧЕСКОГО КОМПЛЕКСА ИМИТАЦИИ СЛОЖНЫХ СИСТЕМ

### 4.1 Состав и структура комплекса

Предлагаемый программно-технологический комплекс имитации (ПТКИ) BelSim (рисунок 4.1) реализует метод построения имитационной модели процессов функционирования промышленного предприятия, представленный в разделе 3 [137, 184]. ПТКИ BelSim построен в соответствии с системными принципами, например, функциональной ортогональности (программные приложения выполняют функции, не пересекающиеся с функциями других программ) и рациональности (дублирование реализуемых функций исключается), многоцелевого назначения (программные средства решают широкий класс задач в заданной предметной области с разным назначением и достигаемыми целями), процедурной открытости (используемые методы и программные приложения не поддерживают структурную взаимосвязь и сохраняют функциональное взаимодействие при их замене и/или структурном и функциональном объединении) и рационального дополнения (допускается расширение путем использования дополнительных методов и средств, не вступающих в противоречие между собой и комплексом) [40]. Структура комплекса состоит из следующих основных компонентов с соответствующими функциями: BelSim IDE (*Integrated Development Environment*) – интегрированная среда разработки программного обеспечения модели; BelSim Optimizer (*Оптимизатор*) – поиск решений в оптимизационных задачах; BelSim Simulator Core – система моделирования; BelSim Experimenter (*Экспериментатор*) – автоматизация имитационных экспериментов; BelSim Data Processing – обработка данных; BelSim Data Integrator (*Интегратор*) – интеграция с базами данных КИС [5, 137, 221].

**Интегрированная среда разработки BelSim IDE** (рисунок 4.1) основана на коммерческой программной системе Microsoft Visual Studio .NET для программной реализации имитационных моделей на языке C++.



**Рисунок 4.1 – Структура программно-технологического комплекса имитации сложных систем BelSim**

Назначение компонентов программно-технологического комплекса BelSim представлено в таблице 4.1.

**Таблица 4.1 – Основные компоненты разработанного программно-технологического комплекса имитации BelSim**

Наименование	Назначение
BelSim IDE	Интегрированная среда разработки с использованием программы MS Visual Studio.NET
BelSim Simulator Core	Система имитационного моделирования на основе процессного способа имитации
AllFusion Process Modeler	Коммерческий программный продукт для разработки концептуальных IDEF моделей
Enterprise Architect	Коммерческий программный продукт для разработки моделей на основе унифицированного языка моделирования UML
BelSim Data Processing: 1 BelSim.Random; 2 BelSim.SSA; 3 BelSim Cluster; 4 Statsoft Statistica	Подсистема обработки входных данных модели и статистики имитационных экспериментов, включающая: - модуль обработки статистических данных; - модуль обработки числовых последовательностей данных; - модуль кластеризации многомерных данных; - коммерческий пакет программ для статистической обработки данных
Simulation Model	Базовая имитационная модель
BelSim Experimenter	Компонент для автоматизации имитационных экспериментов
BelSim Optimizer	Компонент для решения задач оптимизации на основе алгоритмов случайного поиска
BelSim Data Integrator	Компонент для интеграции с корпоративной информационной системой предприятия

ПО **AllFusion Process Modeler** (ранее известное как BPWin) для автоматизации проектирования функциональной модели исследуемой системы на основе

методологии SADT в нотации IDEF0 применяется на этапах содержательного описания и концептуального моделирования исследуемого объекта. С увеличением сложности моделируемой системы применение специализированного CASE-средства функционального моделирования AllFusion Process Modeler для разработки и графического построения IDEF0-диаграмм обеспечивает поддержку соответствия методологии SADT, документирование процесса проектирования модели, реализует автоматическую проверку логической корректности разрабатываемой функциональной модели объекта.

**Система моделирования BelSim Simulator Core.** Для имитационного моделирования исследуемой системы принят процессный способ имитации. Построение имитационных моделей (ИМ) реализовано в интегрированной среде Microsoft Visual Studio .NET разработки приложений на языке C++ и BelSim Simulator Core, которые являются основным ядром комплекса. Система моделирования (СМ) BelSim Simulator Core построена на основе стандартных средств объектно-ориентированного языка программирования ANSI C++, являясь его расширением, что позволяет использовать для реализации ИМ любую доступную среду программирования на языке C++.

Однако при создании сложных моделей следует использовать многофункциональную интегрированную среду разработки с развитыми средствами как написания, так и отладки программ (например, Microsoft Visual C++ .NET). Многие разработчики имитационных моделей отмечают достоинства Microsoft .NET Framework [80], например, развитие средств взаимодействия между имитационными моделями различных объектов, удобство сотрудничества разработчиков моделей и их пользователей через Интернет, использование объектно-ориентированного программирования.

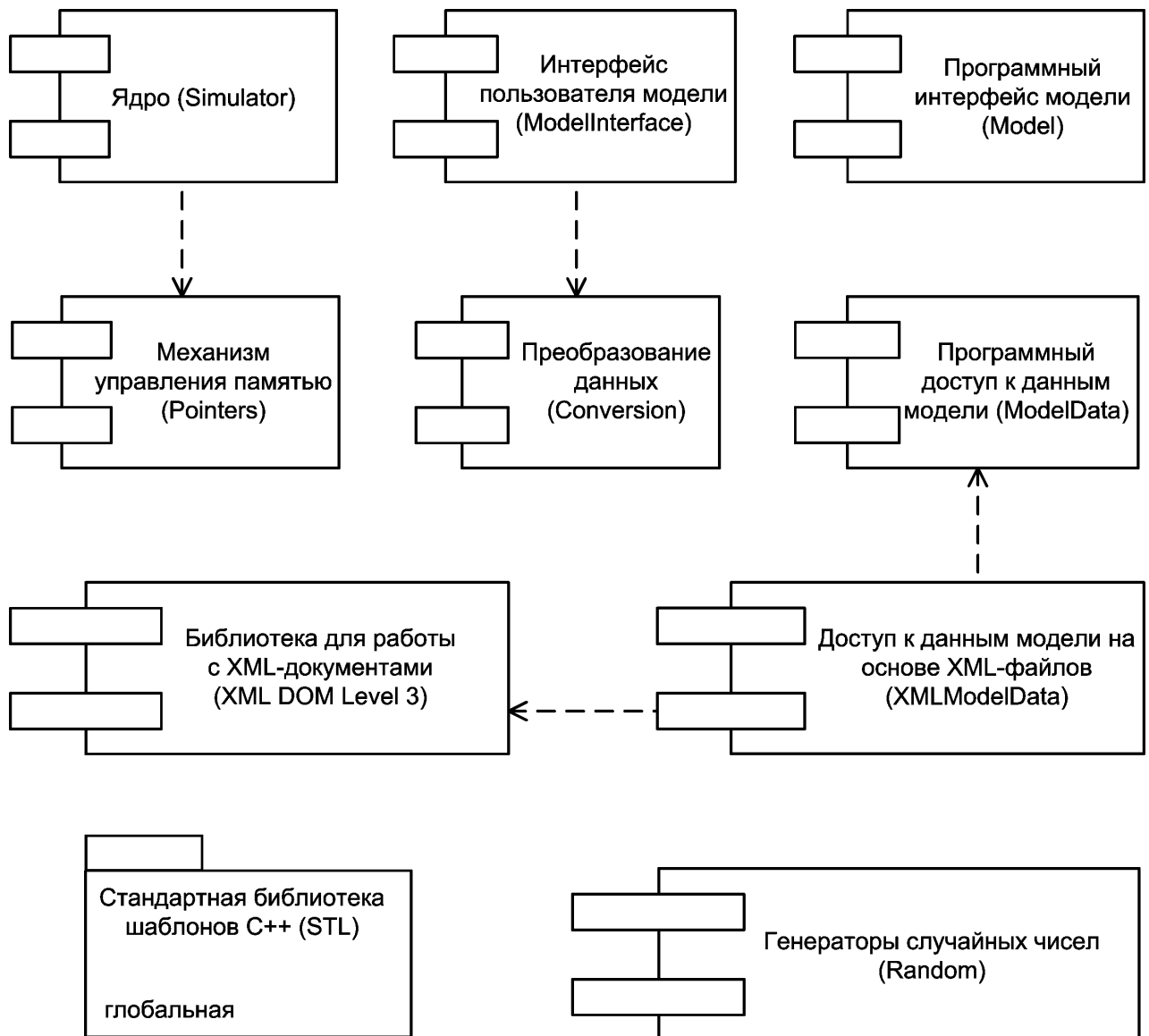
**Подсистема BelSim Experimenter** предназначена для планирования, автоматизации проведения и предварительной обработки статистики имитационных экспериментов (ИЭ). Планирование ИЭ в подсистеме BelSim Experimenter поддерживается разработанным программным приложением *ExperimentDesigner* на основе платформы Microsoft .NET Framework. Если для проведения исследо-

ваний требуется составление плана дробного факторного эксперимента, то используется внутреннее программное приложение *Experimental Design* математического пакета *STATISTICA* и программный модуль (ПМ) *DesignOfExperiment*, специально разработанный для сохранения итогов работы *Experimental Design*. Для обработки статистических результатов ИЭ разработан ПМ *ExperimentData* для пакета *STATISTICA*. Визуализация итогов имитационных экспериментов, обработка статистических результатов имитации и их анализ могут быть реализованы при использовании любых доступных статистических программ при условии автоматизации импорта данных из внешних источников информации по аналогии, например, с математическим пакетом *STATISTICA* [109].

#### **4.2 Система имитационного моделирования BelSim Simulator Core**

На этапе программирования и отладки создается программа модели. Для эффективной реализации ИМ необходим аппарат моделирования, который должен поддерживать способы организации данных для простого и эффективного моделирования, а также средства формализации и представления динамических свойств исследуемой системы. Использование существующих специализированных средств имитации процессным способом затруднено или невозможно, вследствие использования ими специфического аппаратного (тип ЭВМ) и программного (базовый язык программирования) обеспечения. Одна из немногих доступных современных СМ MICIC4 [54], разработанная в Гомельском государственном университете им. Ф. Скорины под руководством профессора И. В. Максимея, не позволяет обеспечить требуемую для рассматриваемой предметной области краткость и точность выражения понятий вследствие закрытости данной СМ и ограничений базового языка программирования. Возможности создания библиотеки дополнительных стандартных компонентов, которые могут использоваться при построении ИМ, также ограничены [109].

На этом этапе используется СМ BelSim Simulator Core (рисунок 4.2), которое позволяет учесть вышеуказанные требования и преодолеть недостатки существующих систем имитационного моделирования.



**Рисунок 4.2 – Диаграмма компонентов BelSim Simulator Core**

Использование концепций ООП и языка C++ позволяет устранить ограничения на описание модели и создать набор абстрактных типов данных (АТД) для описания модели на языке, максимально соответствующем предметной области. При проектировании СМ BelSim Simulator Core в основу положен принцип открытости, что в результате позволяет без каких-либо ограничений расширять ее функциональные возможности, обеспечивая при этом обратную совместимость вследствие инкапсуляции деталей реализации. Каждый компонент СМ BelSim Simulator Core (см. рисунок 4.2) содержит описание классов, отражающих соответствующий аспект СМ. Все компоненты используют стандартную библиотеку шаблонов C++ STL.

Программный интерфейс модели в CM BelSim Simulator Core используется для представления модели как единого целого и обеспечения доступа к ней посредством унифицированного набора методов, позволяющего использовать средства автоматизации моделирования. Доступ к данным модели (XMLMod-elData) обеспечивается на основе файлов в формате XML [97], работа с которыми осуществляется посредством библиотеки для работы с XML-документами (интерфейс XML DOM Level 3).

Генераторы случайных чисел (Random) (см. рисунок 4.2) построены с использованием стандартных функций языка C++ `int rand(void)`, `void srand(unsigned seed)` библиотеки `<stdlib.h>`.

Основным элементом CM является управляющая программа моделирования (УПМ). Семантика и отношения между классами УПМ представлены на диаграмме классов (рисунок 4.3).

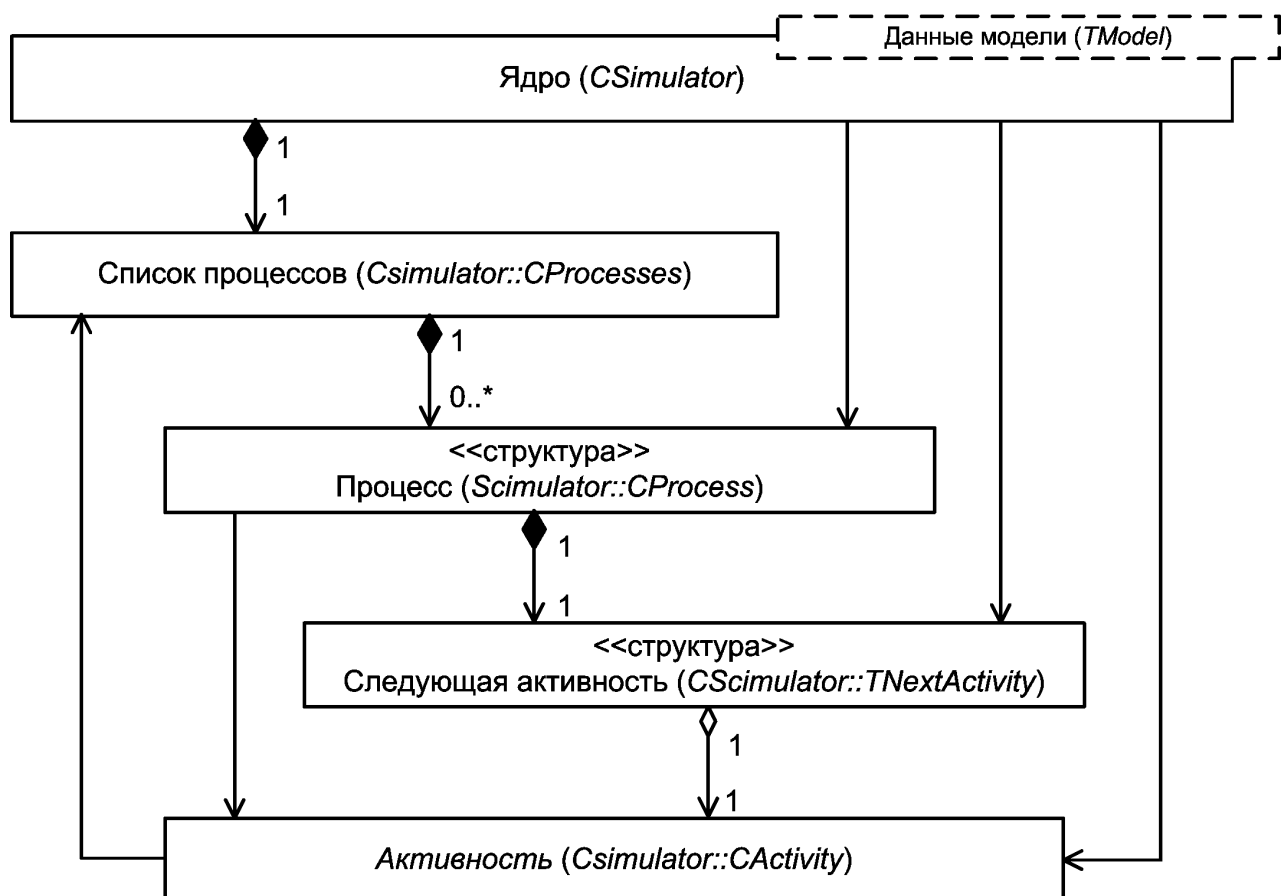


Рисунок 4.3 – Диаграмма классов УПМ в BelSim Simulator Core



Функции УПМ реализует параметризованный класс `CSimulator`, который включает в себя следующие вложенные классы [109]:

- `SActivity` – абстрактный базовый класс для представления абстракции *Активность*;
- `TProcess` – структура, содержащая данные о процессе: идентификатор, приоритет, сведения о моменте следующей активизации процесса (экземпляр структуры `TNextActivity`);
- `SProcesses` – класс-коллекция, содержащая список экземпляров структур `TProcess` для работающих процессов;
- `TNextActivity` – структура данных о моменте активизации процесса: время и выполняемая активность.

Функция `CSimulator::run` запускает основной цикл моделирования. Схема алгоритма этой функции приведена на рисунке 4.4, в соответствии с которым выполняются следующие действия: последовательная активизации процессов, смена модельного времени, проверка условий окончания моделирования (окончание заданного периода модельного времени, завершение всех процессов, вызов оператора останова активностью какого-либо процесса).

Важной частью языка моделирования является механизм *Управление памятью*. В его основу положен метод на основе подсчета ссылок. Основная идея заключается в отслеживании количества указателей, ссылающихся на объект – когда счетчик становится равным нулю, объект удаляется. Подсчет ссылок обладает рядом существенных ограничений, которые снижают его практическую ценность и объясняют малую распространенность. Во-первых, данный метод слишком полагается на соблюдение всех правил программистом (каждый раз при получении или копировании адреса объекта необходимо увеличивать, а при завершении использования адреса – уменьшать счетчик ссылок).

Во-вторых, возникает проблема зацикливания, когда какая-либо группа объектов содержит ссылки друг на друга, но внешние ссылки на объекты этой группы отсутствуют.

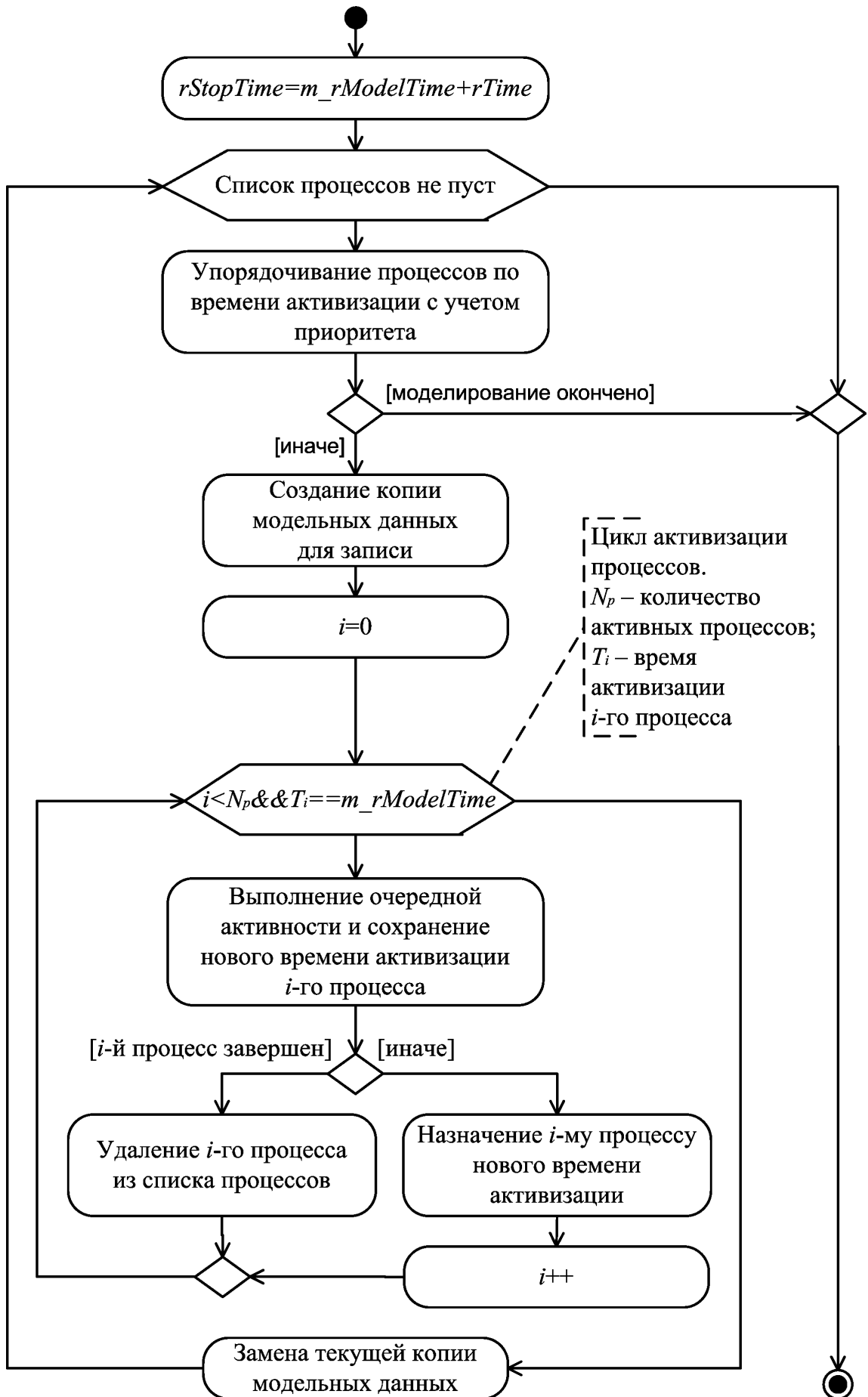


Рисунок 4.4 – Схема алгоритма цикла моделирования

Для устранения недостатков метода управления памятью на основе подсчета ссылок создан специальный класс-шаблон «умный указатель»  $SP<Type>$ , заменяющий стандартный тип указателя. Для этого перегружены операторы присваивания ( $operator =$ ), разыменования ( $operator *$ ), селектора членов ( $operator ->$ ), приведения типа к стандартному указателю на объект ( $operator Type*$ ).

Экземпляр класса  $SP<Type>$  обеспечивает автоматический подсчет ссылок, при необходимости увеличивая или уменьшая счетчик в конструкторах, деструкторе и операторе присваивания.

Если значение счетчика уменьшается до нуля, указываемый объект удаляется. Чтобы исключить возможность создания объектов в обход данного механизма, все объекты создаются с помощью производящих функций, которые являются статическими членами соответствующего класса и возвращают «умный указатель» на объект. При этом конструкторы объектов объявлены закрытыми. «Несанкционированное» (без увеличения счетчика ссылок) копирование запрещено. Для этого доступ к конструктору копии и оператору присваивания также закрыт.

Счетчики ссылок представлены ассоциативной коллекцией, в которой каждому значению указателя соответствует целое число – количество ссылок на данный адрес памяти. Проблема снижения быстродействия решена путем использования бинарного дерева в качестве внутренней структуры данных, что обеспечивает логарифмическую зависимость времени поиска требуемых элементов от их общего количества [109].

### **4.3 Технология алгоритмизации имитационной модели в BelSim Simulator Core**

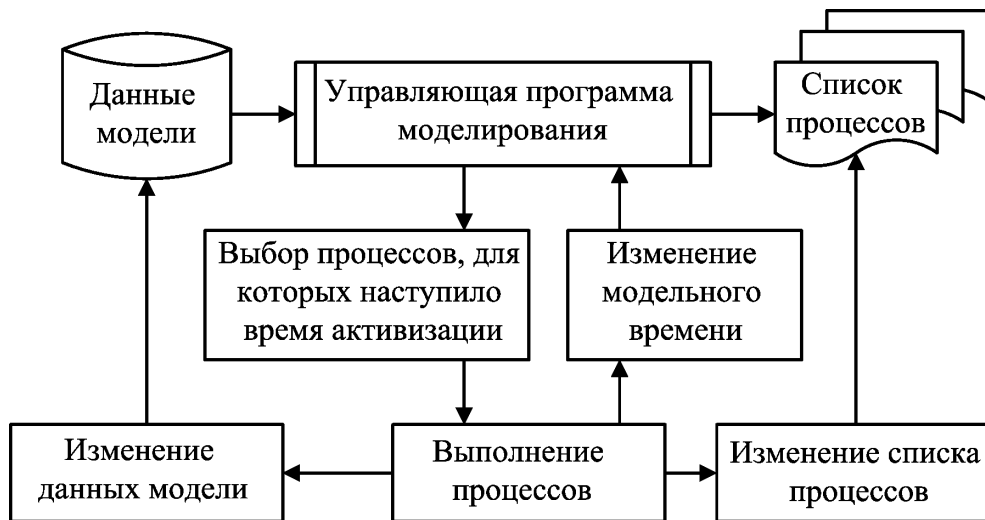
Рассмотрим содержание основных этапов технологии построения алгоритмического обеспечения ИМ.

*Создание процессов.* Оператор  $CREATE\_PROCESS(t, A_i, p)$ , где  $t$  – время активизации,  $A_i$  – начальная активность процесса,  $p$  – приоритет активности, со-

здает новый процесс с начальной активностью  $A_i$ , приоритетом  $p$  и назначает момент его первой активизации  $t$ .

Оператор  $RUN(T)$ , где  $T$  – модельное время имитации, запускает цикл моделирования длительностью  $T$  [109].

*Синхронизация процессов и их взаимодействия с УПМ.* На этапе построения алгоритмического обеспечения имитационной модели формальное описание системы, полученное на предыдущем этапе, преобразуется в описание алгоритмического обеспечения имитационной модели. При этом прорабатываются вопросы синхронизации процессов и их взаимодействия с управляющей программой моделирования (УПМ), задания начальных условий, организации сбора статистики и окончания имитации (рисунок 4.5).



**Рисунок 4.5 – Реализация имитационной модели на основе процессного способа имитации в BelSim Simulator Core**

Синхронизация процессов включает в себя реализацию двух функций:

- а) внутренняя синхронизация процессов по времени путем добавления в алгоритмы их функционирования операторов изменения временной координаты;
- б) синхронизация межпроцессных взаимодействий (информационного обмена) исходя из функциональной модели, построенной на этапе формализации.

Синхронизация и взаимодействие процессов осуществляются посредством обращения к УПМ с помощью набора специальных операторов синхронизации,

которые организуют возврат на УПМ и модификацию временной координаты процесса, разбивая алгоритм его функционирования на кванты – активности:

1) WAIT ( $t, A_i$ ), где  $t$  – время ожидания процесса в модельном времени,  $A_i$  – следующая активность процесса. Оператор изменяет временную координату и состояние процесса. В результате выполнения оператора процессу назначается момент следующей активизации по окончании ожидания длительностью  $t$  и при последующей активизации процесса управление будет передано на выполнение активности  $A_i$ . Возможен сокращенный вариант данного оператора – WAIT( $t$ ), когда указывается только время ожидания. В этом случае состояние процесса (текущая активность) не изменяется;

2) WAIT\_WHILE ( $E, A_i$ ), где  $E$  – логическое выражение,  $A_i$  – следующая активность процесса. Оператор изменяет состояние процесса, однако момент следующей активизации процесса не назначается до тех пор, пока выполняется условие  $E$ . УПМ проверяет условие  $E$  после каждого изменения модельного времени  $t_i$  и в случае его невыполнения ставит процесс в очередь для активизации в момент времени, равный текущему значению  $t_i$ . Если изменение состояния процесса не требуется, второй параметр оператора не указывается;

3) END. Оператор завершает текущий процесс и удаляет его номер из списка процессов в УПМ [109].

В начале очередного цикла алгоритма функционирования УПМ после процедуры изменения модельного времени и перед активизацией процессов запоминается текущее состояние модели путем создания копии структуры данных со значениями всех параметров. Далее каждому процессу при его активизации передается два набора (экземпляра структуры) параметров отдельно для чтения и для записи. Таким образом, информационная стыковка процессов, активизируемых в один и тот же момент модельного времени, обеспечивается тем, что все процессы используют идентичные значения входных параметров, зафиксированные в момент изменения модельного времени. Если процессу требуется доступ к выходным данным других процессов, то вызывается оператор WAIT(0), при этом изменения модельного времени не происходит, а требуемые параметры

передаются процессу при последующей его активизации как входные, доступные для чтения. Помимо этого, используется механизм приоритетов процессов: каждому процессу назначается приоритет, и активизация процессов в один и тот же момент модельного времени осуществляется в порядке убывания приоритетов. Это позволяет синхронизировать процессы-источники и процессы-потребители информации.

*Задание начальных условий.* Вторым шагом построения алгоритмического обеспечения имитационной модели является разработка алгоритма задания начальных условий, который выполняется один раз перед началом моделирования и содержит следующие блоки операторов: инициализация глобальных структур данных модели, инициализация локальных данных процессов; последовательность вызовов оператора `CREATE_PROCESS` для формирования списка процессов в УПМ; вызов оператора `RUN` для запуска процесса имитации.

*Сбор статистики.* Следующим шагом является организация сбора статистики. Для этой цели следует предусмотреть один или несколько процессов. Алгоритмы функционирования этих процессов должны обеспечивать их активизацию с определенным временным интервалом или при определенных условиях, вычисление и запись результатов моделирования. Для выполнения таких действий необходимо получить данные о состоянии модели при активизации процесса сбора статистики, а также типовые данные, представляемые УПМ: время формирования, время первой и последней активизации, время окончания процессов.

*Окончание моделирования.* Завершающим шагом этапа построения имитационной модели является задание условий окончания моделирования. Командой завершения имитации является вызов оператора `STOP`. Проверку условия окончания моделирования можно организовать с помощью отдельного процесса или добавить необходимые операторы в алгоритм сбора статистики. По умолчанию цикл имитации УПМ завершается по прошествии периода  $T$  модельного времени, указанного при вызове оператора `RUN` [109].

#### 4.4 Технология разработки программных модулей имитационной модели в BelSim Simulator Core

ИМ в СМ BelSim Simulator Core представляет собой один или несколько программных модулей на базовом языке программирования C++, содержащих описание структур данных модели и их реализацию, классов активностей процессов, УПМ, интерфейса модели, а также любых других необходимых классов, структур данных и функций. Типовой программный модуль ИМ включает блоки, содержащие описания: директив включения заголовочных файлов СМ; структуры (класса) глобальных данных модели; классов активностей процессов и их реализацию; реализацию классов программного интерфейса модели и ее внешнего интерфейса. Список заголовочных файлов и их назначение приведены в таблице 4.2.

**Таблица 4.2 – Заголовочные файлы СМ BelSim Simulator Core**

Файл	Тип данных	Назначение
<i>simulator.h</i>	<i>CSimulator, TProcess, CProcesses, CActivity, TNextActivity</i>	Описание базовых параметризованных классов ядра СМ
<i>model.h</i>	<i>CModel</i>	Программный интерфейс модели
<i>ModelData.h</i>	<i>CModelData</i>	Программный интерфейс для доступа к данным модели
<i>XMLModelData.h</i>	<i>CXMLModelData</i>	Описание класса, реализующего интерфейс для доступа к данным модели, представленным в виде XML-файла
<i>pointers.h</i>	<i>SP</i>	Реализация механизма управления памятью
<i>random.h</i>	<i>CRandom, CRandomUniform, CRandomNormal, CRandomExponential, CGammaFunction</i>	Описание генераторов случайных чисел
<i>ModelInterface.h</i>		Библиотека шаблонов функций и макросов для реализации интерфейса пользователя модели
<i>conversion.h</i>		Описание функций преобразования данных

Обязательным является включение файла *simulator.h*, содержащего описание классов, составляющих ядро СМ – УПМ для организации имитации процессным способом. Функции УПМ реализует параметризованный класс *CSimulator*. Параметром его выступает тип данных *TModel*, описывающий глобальные переменные модели. В состав глобальных данных модели включаются входные и выходные переменные модели, а также переменные, к которым должны иметь доступ одновременно несколько процессов. Глобальные данные оформляются в виде структуры (класса), которая может содержать вспомогательные элементы-функции, облегчающие доступ к элементам-данным. Эта структура указывается в качестве параметра класса, а ее экземпляр передается как параметр конструктора объекта УПМ. Никаких ограничений на типы данных модели не накладывается [109].

Активности процессов являются классами-функциями от базового класса *CSimulator <TModel >:: CActivity*, где *TModel* – наименование типа, представляющего глобальные данные модели. Класс *CActivity* является абстрактным базовым классом и содержит чистую виртуальную функцию *operator()*, вызываемую УПМ при очередной активизации процесса, которая должна быть определена для представления алгоритма активности. Параметрами функции *operator()* являются модельное время; указатель на константную копию экземпляра структуры данных модели на момент модельного времени, указанный в первом параметре до вызова активностей других одновременно активизируемых процессов; указатель на текущий экземпляр структуры данных модели; указатель на коллекцию процессов УПМ, обеспечивающий доступ к локальным данным других процессов и создание новых. Вызовы операторов синхронизации реализуются посредством оператора *return* языка С++ с указанием соответствующего возвращаемого значения типа *TNextActivity*, содержащего время ожидания до очередной активизации и указатель на следующую активность процесса. Инициализация данных возможна посредством передачи соответствующих параметров в конструктор активности. Порядок размещения описаний классов активностей и их реализаций произвольный, однако целесообразно выделить в отдельный блок перед блоком их реализаций [109].



Программный интерфейс проектируемой имитационной модели обеспечивает доступ к ней посредством унифицированного набора методов и позволяет использовать средства автоматизации моделирования. Для реализации интерфейса создается класс, наследующий интерфейс *CModel*. *CModel* содержит две функции: *getModelData* и *run*.

Функция *CModel::getModelData* вызывается для получения сведений о модели. Параметром функции является ссылка на интерфейс *CModelData* объекта-приемника информации о модели. Посредством вызова методов *setName*, *setVersion*, *setRevision*, *setCopyright*, *addParameter*, *addParameterArray*, *addResponse* и *addResponseArray* указываются сведения о наименовании, версии, авторских правах, параметрах и откликах модели, соответственно.

Функция *CModel::run* вызывается для проведения имитационного эксперимента. Параметрами функции являются: ссылка на интерфейс *CModelData* для чтения входных и записи выходных значений соответственно параметров и откликов; целое число для инициализации генераторов псевдослучайных чисел, что обеспечивает возможность точного повторного воспроизведения результатов экспериментов [109].

В текст функции включают следующие блоки операторов, которые в совокупности представляют задание начальных условий моделирования:

а) Блок инициализации глобальных данных модели. Вначале создается экземпляр структуры данных модели:

$$SP<TModel> spModel(new TModel),$$

где *TModel* – имя типа (структуры или класса), представляющего глобальные данные модели;

*spModel* – указатель на созданный экземпляр структуры данных. Затем производится инициализация его элементов.

Чтение исходных данных осуществляется посредством вызова метода *getParameterValue* интерфейса *CModelData*.

б) Объявление объекта УПМ:

*CSimulator*<*TModel*> *simulator(spModel)*,

где *simulator* – имя создаваемого объекта УПМ.

в) Блок инициализации процессов, включающий последовательность операторов для инициализации структур данных процессов и вызов метода *CSimulator::addProcess* для формирования списка процессов УПМ.

г) Блок инициализации генераторов псевдослучайных чисел на основе значения второго параметра функции.

д) Вызов метода *run* объекта УПМ для запуска процесса имитации.

Завершающим элементом является реализация внешнего интерфейса имитационной модели. Будучи стандартным для всех имитационных моделей, построенных по настоящей методике, он позволяет задействовать средства автоматизации проведения экспериментов и анализа их результатов. Для реализации интерфейса используется макрос

*MODEL\_MAIN(Model, ModelData)*,

где *Model* – имя класса, реализующего программный интерфейс модели;

*ModelData* – имя класса для доступа к данным модели. Указанный макрос помещает в текст программного модуля имитационной модели функцию *main*, обрабатывающую командную строку для организации интерфейса пользователя [109].

Объектно-ориентированный подход и открытость СМ находят выражение в гибкости при создании программы ИМ. Для большинства задач существуют стандартные рекомендуемые средства, однако разработчик не обязан использовать их, заменяя их собственными, наилучшим образом подходящими для данной конкретной ситуации.

#### **4.5 Программная реализация базовой имитационной модели промышленного предприятия в BelSim IDE**

Анализ функциональной модели показывает, что необходимо обеспечить синхронизацию следующих процессов источников и потребителей информации

(см. таблицу 3.1): ПК<sub>36</sub> (бухгалтерский учет), ПК<sub>23</sub> (поступление заявок на отгрузку продукции), ПК<sub>11</sub> (планирование производства). С учетом выбранного способа аппроксимации функциональных действий компонент, когда выполнение функционального действия предшествует изменению временной координаты, процессам назначаются следующие приоритеты: ПК<sub>36</sub> – Highest, ПК<sub>23</sub> – Normal+2, ПК<sub>11</sub> – Normal+1, где Normal – обычный приоритет, Highest – наивысший приоритет.

В блоке задания начальных условий моделирования должна быть обеспечена инициализация значений входных переменных, однозначно определяющих состояние модели.

Для организации сбора статистики требуется запоминать состояние имитационной модели через определенные временные интервалы, что реализуется отдельным процессом. Так как основой для формирования статистики моделирования являются данные «Бухгалтерская отчетность», требуется синхронизация с процессом ПК<sub>36</sub> (бухгалтерский учет). Для этого процессу сбора статистики следует назначить приоритет Highest-1. Моделирование завершается через определенный период (задается пользователем) модельного времени [109].

Имитационная модель представлена одним модулем *Enterprise3.cpp*, содержащим описание структур данных модели и их реализацию, классы активностей процессов, класс программного интерфейса модели, внешний интерфейс модели. Статическая структура классов, представляющих данные модели, приведена на рисунке 4.6.

Глобальные данные представлены структурой *TEnterprise* (приложение В, таблица В.1), которая включает ряд вспомогательных классов: *TDistribution* (параметры случайной величины), *TAccount* (счет бухгалтерского учета), *TWarehouseItem* (учетная единица на складе), *TBatch* (партия продукции или сырья). Для представления абстракций контракта и кредита используются классы *TContract* и *TCredit* соответственно.

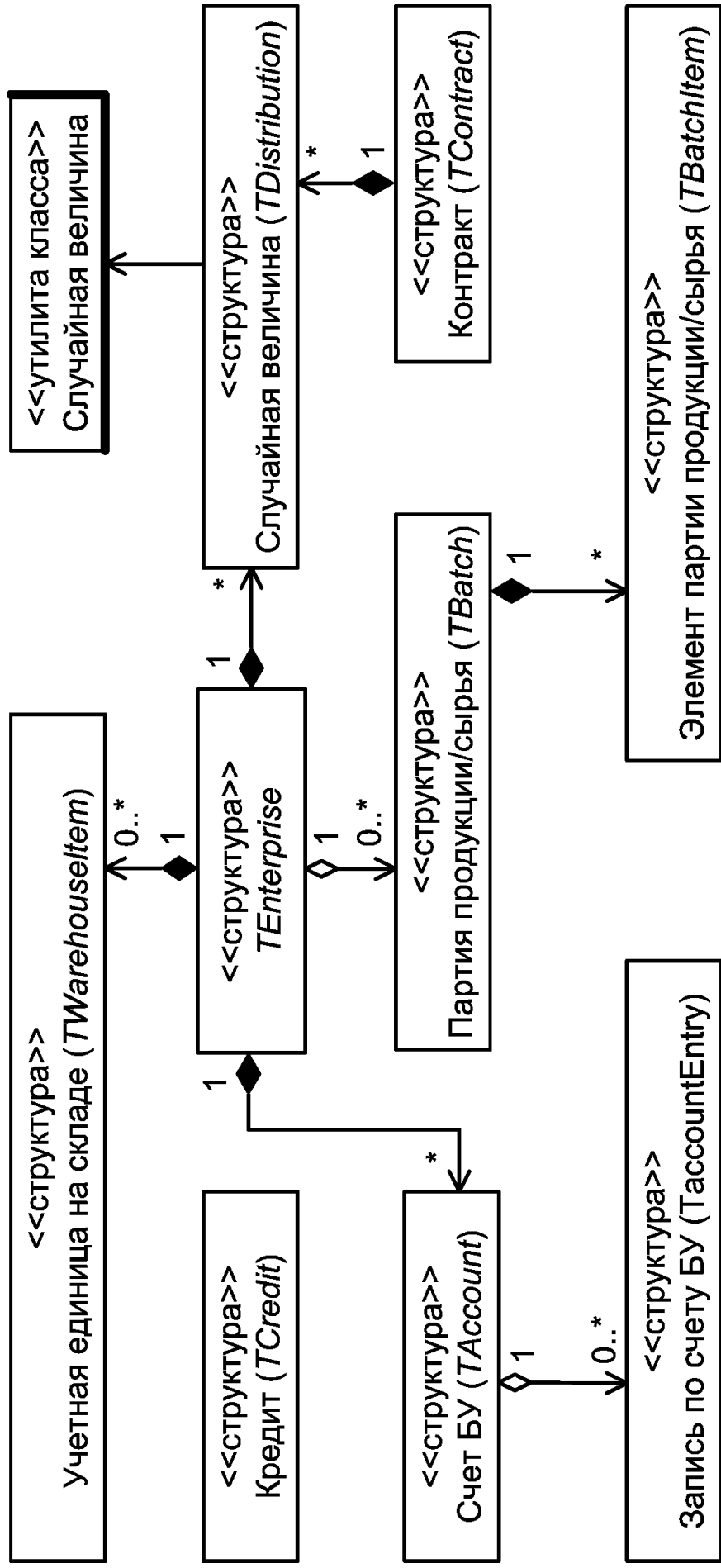


Рисунок 4.6 – Диаграмма структур данных модели

Классы-функции, описывающие активности процессов, представлены в таблице В.2 приложения В.

Класс *CEnterprise* предназначен для реализации программного интерфейса модели, получения входных и выходных значений переменных, инициализации и исполнения модели с помощью интерфейса *CModel* [109].

Для реализации внешнего интерфейса модели используются стандартные средства системы моделирования: в текст программного модуля модели добавлена реализация функции `main ()`:

*MODEL\_MAIN(CEnterprise, CXMLModelData),*

выполняющая обмен данными на основе *XML*-файлов. Для эксплуатации имитационная модель представлена программным модулем *Enterprise3.EXE*.

Представленная выше базовая ИМ, входящая в ПТКИ *BelSim*, позволяет с минимальными затратами и в кратчайшие сроки реализовать ИМ для решения практических задач в составе *ERP*-системы управления предприятием, благодаря высокому уровню автоматизации основных этапов разработки и использования ИМ [77].

#### **4.6 Автоматизация планирования, проведения и обработки результатов экспериментов в BelSim Experimenter**

Эксперименты с имитационными моделями сложных систем требуют значительных временных затрат. При этом важно исключить или значительно уменьшить ошибки при построении и реализации плана эксперимента, что становится не простой задачей при большом количестве исследуемых параметров и откликов модели. Автоматизация процедуры проведения эксперимента является необходимой задачей, для решения которой разработана программная подсистема *BelSim Experimenter* (см. рисунок 4.1). Программная система состоит из нескольких блоков: блок ввода/вывода информации, блок создания плана эксперимента, блок подготовки данных для проведения опытов, блок проведения опытов и блок предварительного статистического анализа (рисунок 4.7) [109].

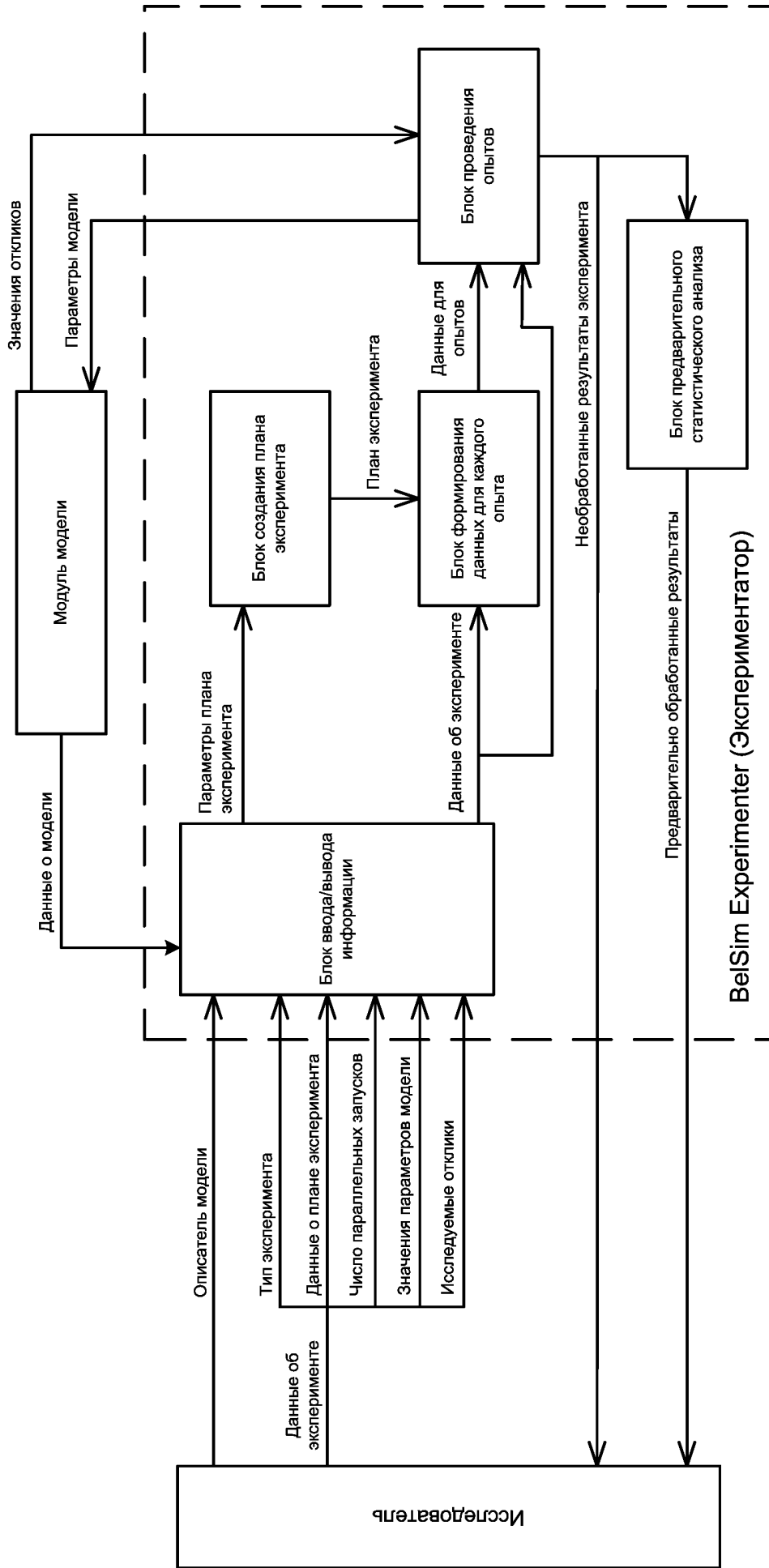


Рисунок 4.7 – Схема информационных потоков в подсистеме BelSim Experimenter

Блок ввода/вывода информации (см. рисунок 4.7) направлен на поддержку интерфейса, обеспечивающего удобный ввод необходимой информации. Реализован в виде приложения *ExperimentDesigner*, представляющего собой набор форм с использованием Microsoft Visual C++ .NET, Microsoft .NET Framework.

Блок создания плана эксперимента формирует его в зависимости от указанного типа, количества параметров и опытов на основе общепринятых методик [59]. Указанный блок использует внутренний модуль *Experimental Design (DOE)* приложения STATISTICA 6.0.

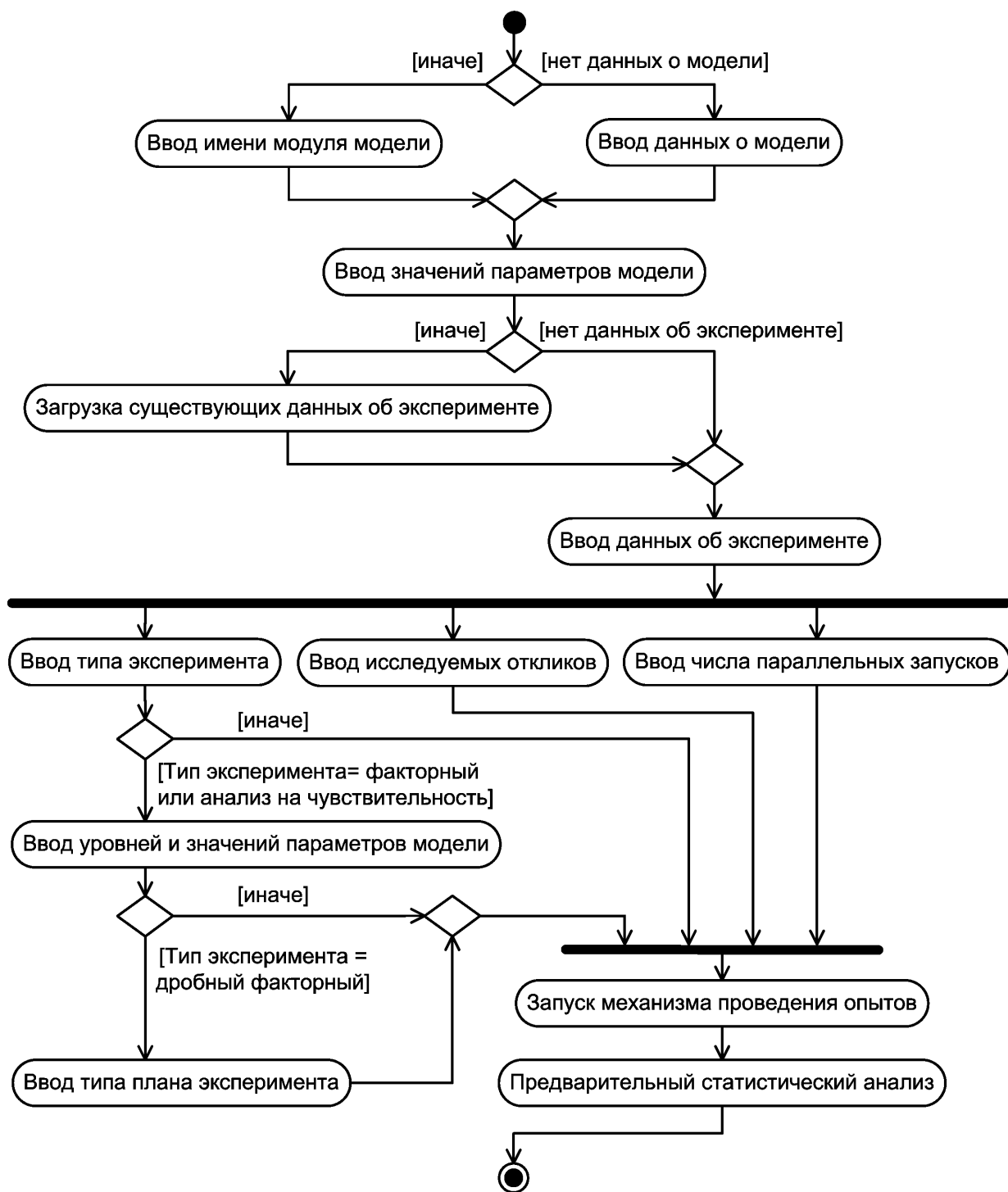
Блок формирования данных для каждого опыта производит сопоставление уровней из плана эксперимента и имен параметров модели. Реализован в виде макроса в приложении статистического анализа STATISTICA 6.0.

Блок проведения экспериментов по представленному плану эксперимента проводит опыты над моделью: осуществляет взаимодействие с моделью, сохраняет результаты опытов, для удобства пользователя обеспечивает вывод информации о ходе проведения эксперимента. Блок проведения экспериментов, механизмы взаимодействия с моделью и потоками данных реализованы в консольном приложении *Experimenter*.

Блок предварительного статистического анализа выводит результаты в удобной форме и проводит необходимые действия для последующего анализа результатов. Блок реализован в виде программного модуля *ExperimentData* в приложении STATISTICA 6.0.

Технология применения подсистемы *BelSim Experimenter* при проведении экспериментов различного типа представляет собой последовательность действий, изображенную на диаграмме (рисунок 4.8) [104], выполненной с использованием унифицированного языка моделирования UML [55].

*BelSim Experimenter* проводит следующие типы экспериментов (поток «*Тип эксперимента*»): оценка погрешности имитации; анализ длины разгона (переходного процесса) модели и оценка устойчивости результатов имитации; анализ чувствительности откликов к изменению значений входных параметров модели; полный и дробный факторные эксперименты [104].

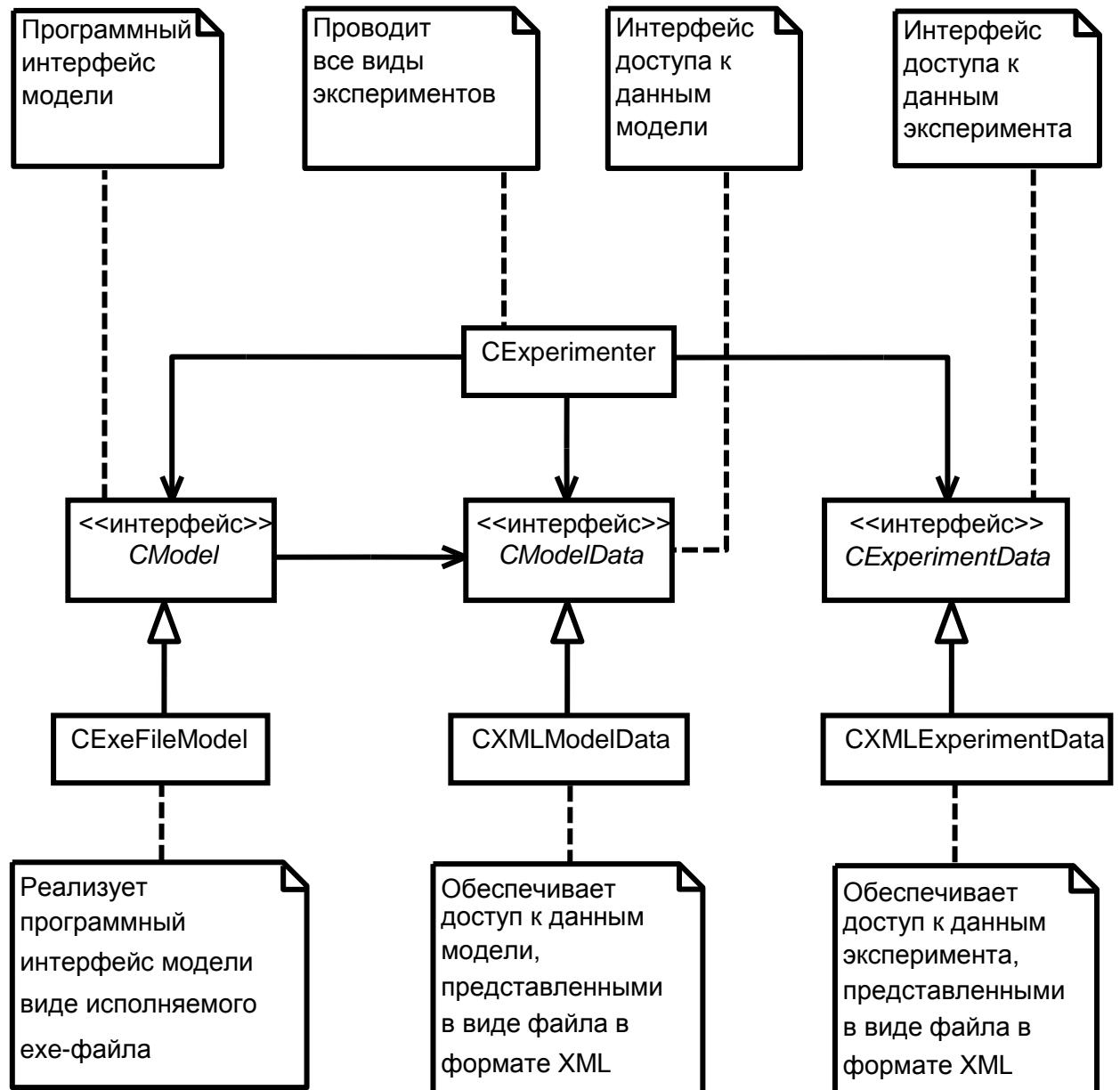


**Рисунок 4.8 – Технология применения подсистемы *BelSim Experimenter* для проведения экспериментов**

Планирование, проведение и обработка статистики имитационных экспериментов выполняются с потоками данных, организованными в виде XML-файлов данных модели [143] (приложение Г) и файлов данных эксперимента (см. приложение Г), доступ к которым реализован через стандартный интерфейс XML DOM Level 3 с использованием библиотеки MSXML 4.0 [215].



Проектирование программы *BelSim Experimenter* выполнено с использованием объектно-ориентированной технологии, в основе которой лежит объектная модель. Данная модель направлена на описание ключевых абстракций и механизмов, которые формируют предметную область и архитектуру блока (рисунок 4.9).



**Рисунок 4.9** – Диаграмма классов программы *Experimenter*

Для описания ролей и обязанностей объектов использована диаграмма классов (см. рисунок 4.10), для выражения решений о поведении блока – диаграммы состояний/активностей и сценарии. Все наименования в модуле удовле-

творяют правилам построения идентификаторов C++, так как модель экспортируется в код C++.

Основная действующая абстракция (механизм проведения опытов) – CExperimenter. Ответственность – организация всех видов экспериментов. Жизненный цикл объектов класса заключается в выполнении операции run():

```
void run(CModel &model, CModelData &modelData, CExperimentData
&experimentData),
```

которая осуществляет проведение эксперимента на основе данных ExperimentData над моделью с интерфейсом model, доступ к данным которой осуществляется через интерфейс modelData.

Посредством операций getModelData() и run() интерфейса CModel осуществляется получение данных и прогон модели. Один из вариантов взаимодействия модели в виде исполняемого модуля и блока проведения опытов реализован в классе CExeFileModel. Объекты данного класса имеют жизненный цикл, представленный на рисунке 4.10. При инициализации создается экземпляр класса, соответствующий модулю с именем sModelFileName. Из состояния ожидания он переводится сообщениями run и getModelData в состояния, из которых производится вызов модуля модели соответствующими параметрами, представленными ниже:

$$/I \mid /R \langle \text{файл} \rangle [/RS:\langle \text{число} \rangle] [/IO:\langle \text{файл} \rangle] [/F]$$

где /I – команда создания файла данных модели со списком входных и выходных переменных; /R – команда запуска модели; <файл> – имя файла данных модели; /RS:<число> – начальное число для инициализации генераторов псевдослучайных чисел; /IO:<файл> – файл для вывода промежуточных результатов моделирования; [/F] – команда отказа от выполнения дополнительных операций (форматирования выходных данных и т.д.) с целью ускорения работы модели [109].

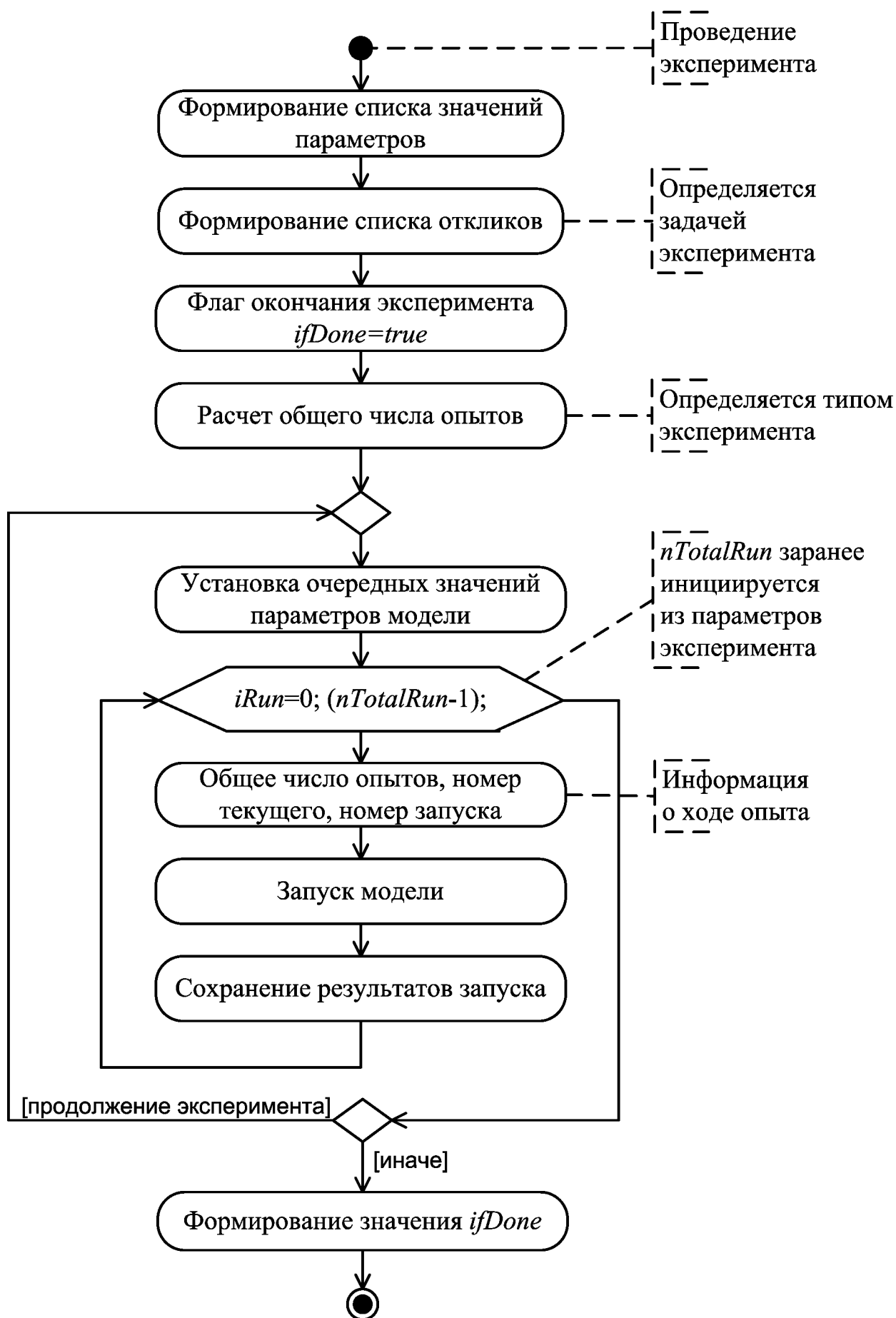
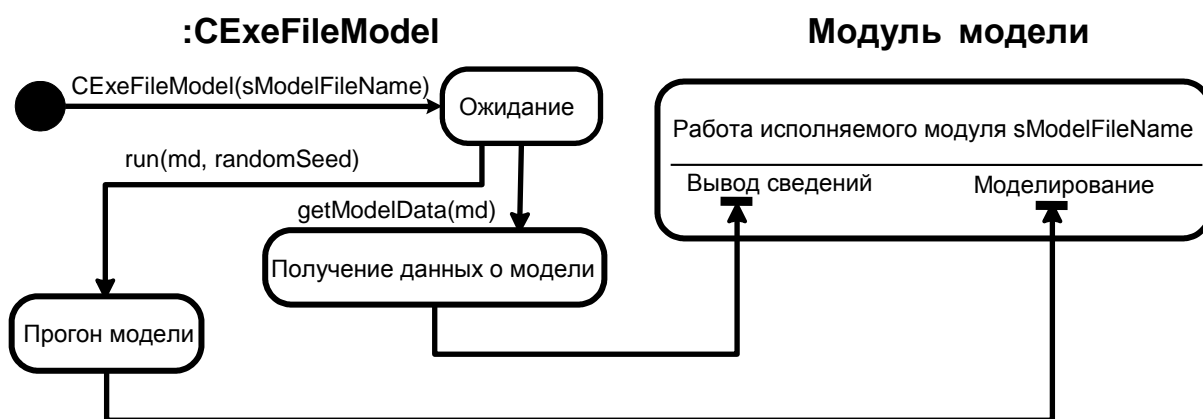


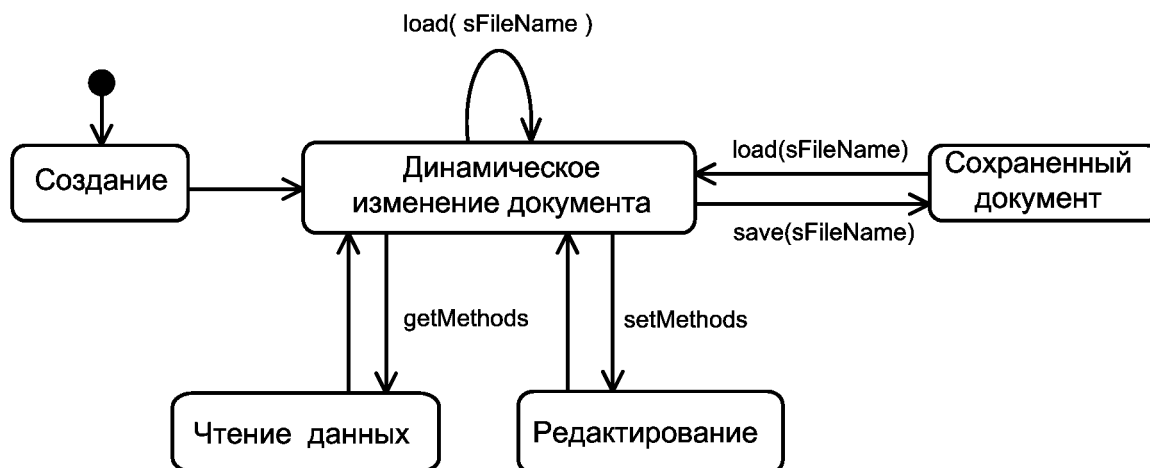
Рисунок 4.10 – Схема алгоритма блока проведения экспериментов

Интерфейсы *CModelData* и *CExperimentData* и их реализация в виде классов *CXMLModelData* и *CXMLExperimentData* используются для организации доступа к данным модели и эксперимента. Объекты этих классов имеют жизненный цикл, представленный на рисунке 4.11. После создания объект переходит в состояние «Создание», в котором формируются основные теги документа. Затем осуществляется переход в состояние «Динамическое изменение документа». Содержимое документа можно изменить группой операций *setMethods* в состоянии «Редактирование», прочитать – группой операций *getMethods* в состоянии «Чтение данных», загрузить новыми данными из существующего файла *sFileName* операцией *load*, сохранить в файле *sFileName* операцией *save*.



**Рисунок 4.11 – Диаграмма жизненного цикла экземпляров класса CExecFileModel**

Сценарий взаимодействия объектов в ходе проведения экспериментов следующий. Исследователем создается объект *model*, привязанный к модулю модели *sModelFileName*. Загружается файл данных модели сообщением *load* объекту *modelData:CXMLModelData*. Используя этот объект, выводится информация о модели. Загружается файл данных эксперимента сообщением *load* объекту *experimentData:CXMLExperimentData*. Используя этот объект, считывается информация об эксперименте и выводится на экран. Затем сообщением *run* экземпляру класса *CExperimenter* запускается механизм проведения опытов (рисунок 4.12). Результаты эксперимента сохраняются выдачей сообщения *save* объекту *experimentData*.



**Рисунок 4.12 – Диаграмма жизненного цикла экземпляров классов, реализующих интерфейсы CModelData и CExperimentData**

Взаимодействие с компонентами ПТКИ в процессе планирования, проведения и обработки статистики экспериментов показано на рисунке 4.13. Исследователь подает на вход системы необходимые данные о планируемом эксперименте. В соответствии с ними ПТКИ проводит эксперимент с моделью. Взаимодействие с моделью заключается в передаче модели исходных данных для ее работы. Модель после окончания работы выдает значения множества откликов, которые затем группируются системой и предоставляются пользователю в виде выходной информации. Моделью также выдаются сведения о параметрах и откликах. Используется структура информационных потоков, представленная на рисунке 4.7.

Поток «*Описатель модели*» может содержать следующие значения:

- имя модуля модели, тогда блок ввода/вывода информации сам запросит по данному имени поток «*Данные о модели*»;
- поток «*Данные о модели*».

Поток «*Данные об эксперименте*» обладает сложной структурой, в которой поток «*Тип эксперимента*» допускает следующие возможные значения: оценку погрешности имитации; длину разгона (переходного процесса) модели и оценку устойчивости статистики моделирования; оценку анализа чувствительности откликов при изменении значений входных параметров исследуемой модели; план полного и дробного факторного эксперимента [109].

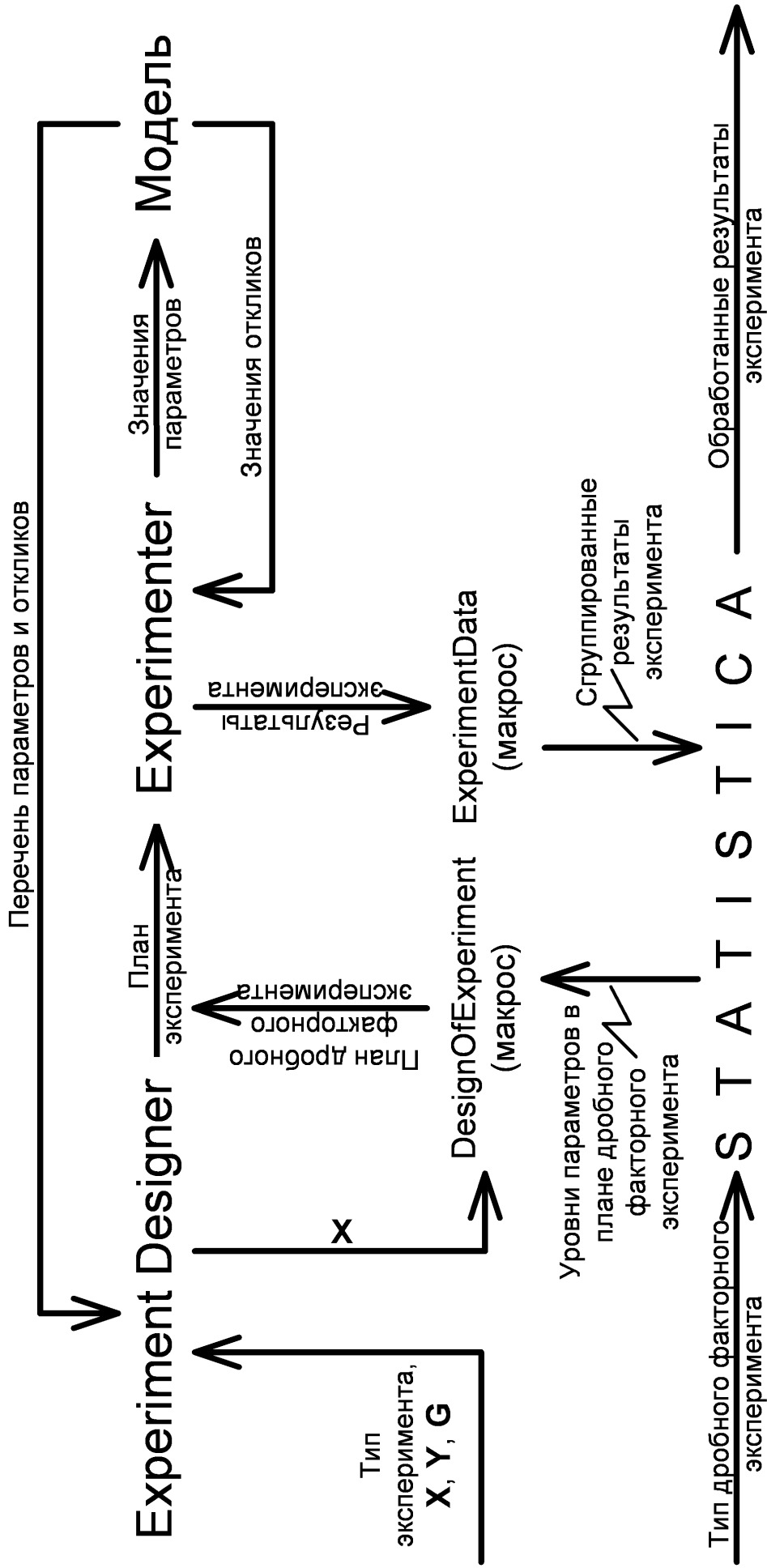


Рисунок 4.13 – Схема постановки и выполнения имитационных экспериментов

В зависимости от содержимого потока «*Тип эксперимента*» поток «*Данные о плане эксперимента*» представляет собой оценку погрешности моделирования или анализ длины разгона (переходного процесса) модели и устойчивости статистики моделирования – множество двоек <имя параметра, значение>; анализ чувствительности откликов модели к изменению значений входных параметров – множество структур (имя параметра, {<уровень, значение>}), причем уровень принимает значения из множества  $\{-1, 0, 1\}$ ; полный факторный эксперимент – множество структур (имя параметра, {<уровень, значение >}), причем уровень в пределах от  $-1$  до  $1$ ; дробный факторный эксперимент – множество структур (имя параметра, {<уровень, значение >}), причем уровень в пределах от  $-1$  до  $1$ , тип плана эксперимента.

Поток «*Значения параметров модели*» содержит информацию (множество пар <имя параметра, значение>) о параметрах модели, не участвующих в эксперименте.

Поток «*Исследуемые отклики*» – список имен исследуемых откликов.

Поток «*Данные о модели*» представляет собой множество имен параметров и откликов модели. Поток «*Значения откликов*» – это множество структур (имя отклика, список значений). Поток «*Параметры модели*» является множеством пар <имя параметра, значение>.

Поток «*Необработанные результаты эксперимента*» – это множество опытов. Опыт описывается списком исходных параметров (имя, значение, уровень) и списком запусков. Запуск описывается номером, списком откликов (имя, список значений).

Поток «*Параметры плана эксперимента*» формируется, когда поток «*Тип эксперимента*» содержит дробный факторный эксперимент. Он включает в себя информацию о количестве параметров, участвующих в эксперименте, о количестве опытов, типе плана эксперимента.

Поток «*План эксперимента*» содержит план эксперимента. Поток «*Данные для опытов*» – множество комбинаций уровней параметров модели <имя,

уровень> для каждого опыта в соответствии с планом дробного факторного эксперимента.

Последовательность выполнения операций при проведении экспериментов представлена на рисунке 4.14. Пользователю предоставляются развитые средства формализации, построения моделей, постановки экспериментов, анализа их результатов [109].

#### **4.7 Программное приложение BelSim Data Integrator для интеграции ПТКИ BelSim с корпоративной информационной системой**

Интеграция ПТКИ *BelSim* с КИС [38, 42] выполняется последовательностью действий по аналогии с механизмом интеграции программных информационных систем [117] по следующим этапам (рисунок 4.15) [143].

*Этап 1.* Установка параметров подключения к выбранному источнику данных информационной системы.

*Шаг 1.1.* Выбор пути к источнику данных, из которого необходимо получить данные для исследуемой имитационной модели.

*Шаг 1.2.* Выбор XML-файла со структурой входных данных имитационной модели.

*Шаг 1.3.* Формирование запроса к источнику данных с указанием названий таблиц и полей, в которых находятся требуемые данные для имитационной модели.

*Этап 2.* Подключение к источнику данных информационной системы промышленного предприятия (например, «1С: Предприятие») и чтение требуемой информации с применением технологии *ADO*, используя библиотеку *msado15.dll*.

*Этап 3.* Запись считанных данных в XML-файл входных данных имитационной модели (после выбора данных их следует представить в виде XML-файла, который *BelSim* может использовать для проведения имитационных экспериментов). Доступ к содержимому XML-документов реализуется через интерфейс, которым является объектная модель *DOM (Document Object Model)* [134].



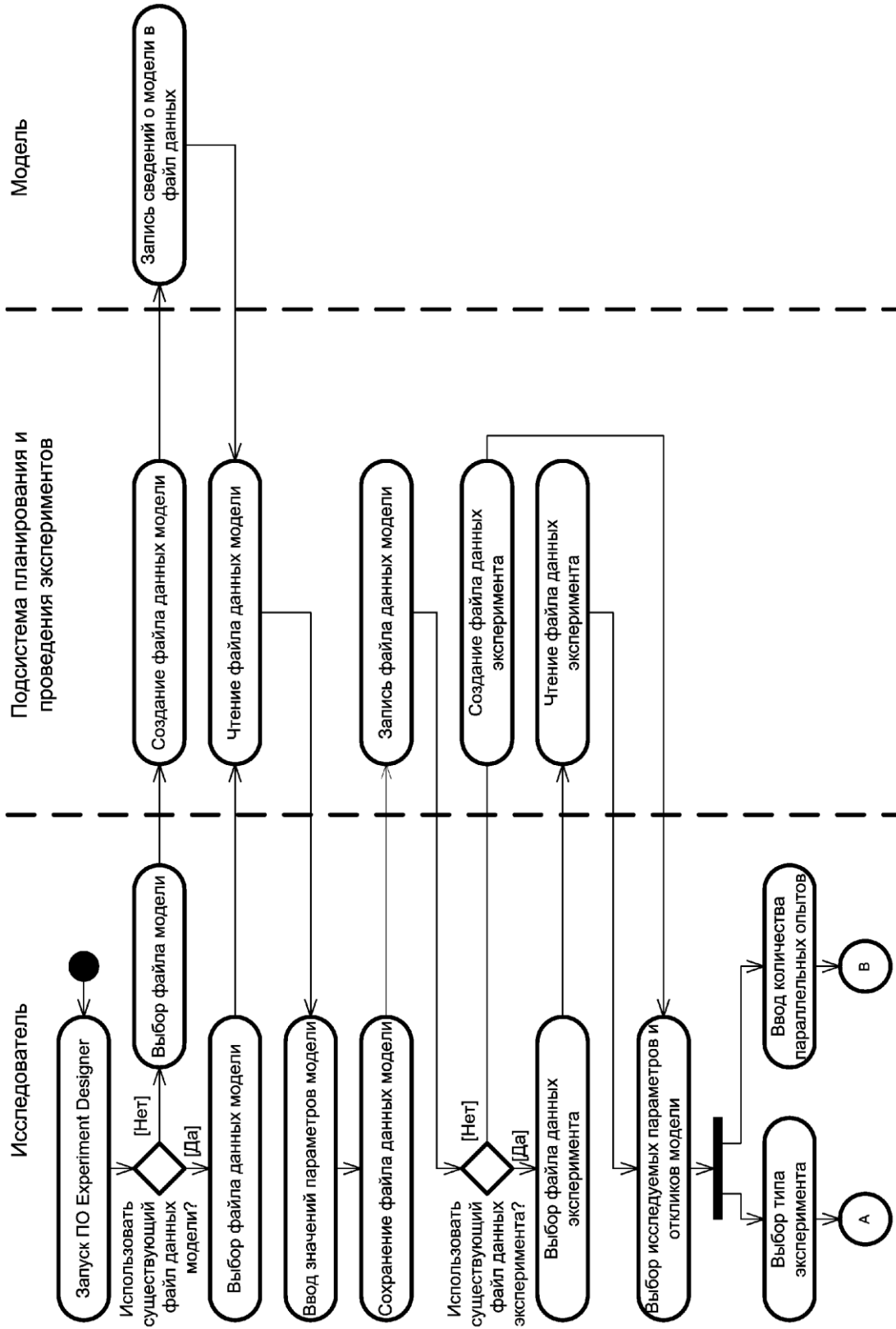
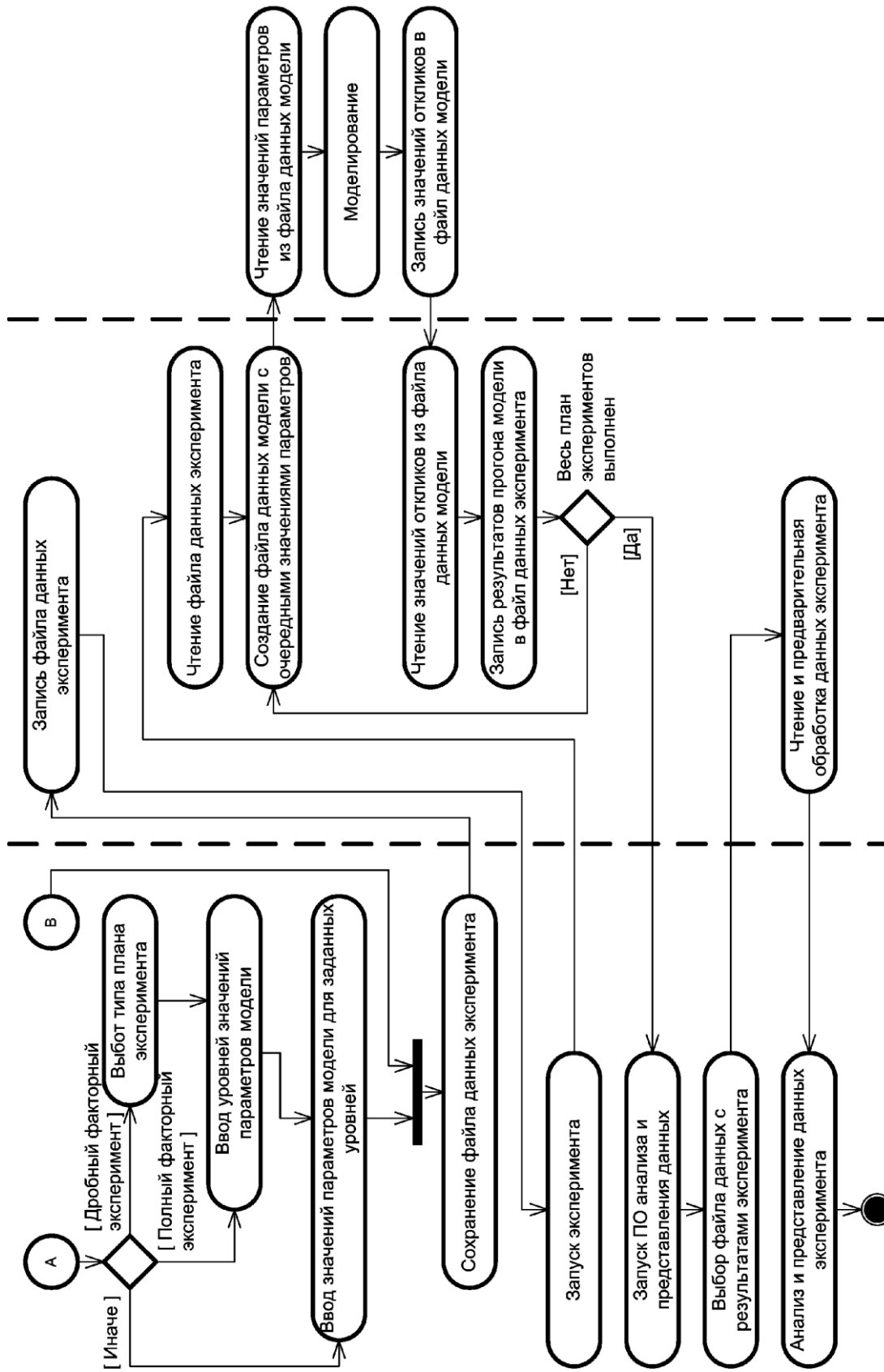
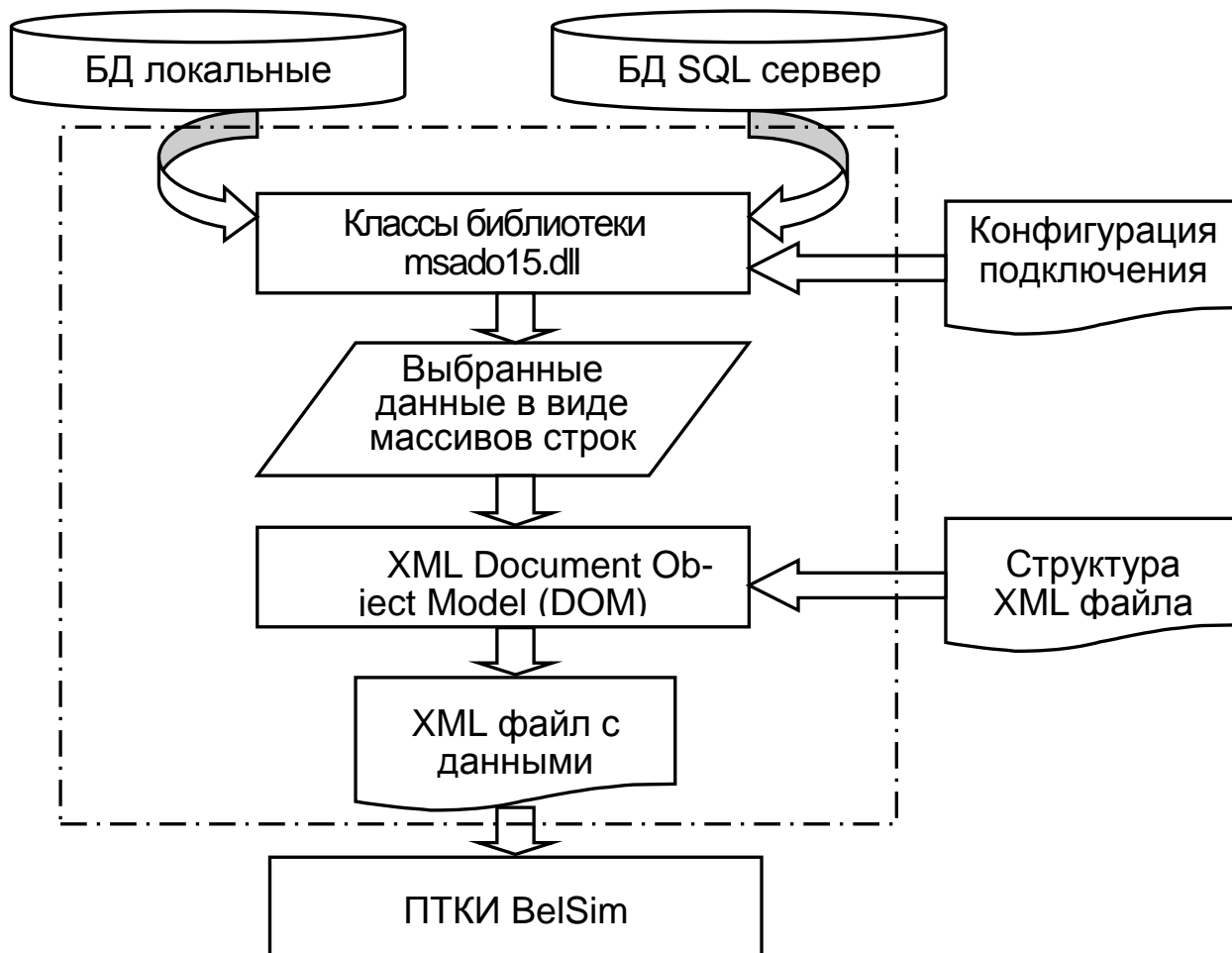


Рисунок 4.14 (Лист 1) – Диаграмма выполнения операций при планировании, проведении и обработке результатов имитационных экспериментов



**Рисунок 4.14 (Лист 2) – Диаграмма выполнения операций при планировании, проведении и обработке результатов имитационных экспериментов**



**Рисунок 4.15 – Логическая структура интеграции ПТКИ BelSim в КИС**

Программное приложение *BelSim Data Integrator* (рисунок 4.16) считывает параметры для подключения к источнику данных из файла *Settings.cfg*, подключается к источнику и выбирает требуемые данные, используя библиотеку *msado15.dll*. Используя *DOM* и *XML*-файл, содержащий структуру документа, формируется *XML*-файл для передачи в *Experiment Designer* ПТКИ *BelSim* (на рисунке 4.13 *XML*-файл условно не показан).

За сбор и конвертирование данных в приложении *BelSim Data Integrator* отвечает класс *DBF2XML* с атрибутами: *NumAllParameters* – задает общее количество параметров файла *Settings.cfg*; *BelsimPath* – указывает путь к директории с установленным ПТКИ *BelSim*; *OnecPath* – указывает путь к базам данных; *ModelStructureXML* – имя *XML*-файла с шаблоном выходного документа; *TablesAndColumns* – указывает на динамический массив с названиями таблиц и

полей, для импорта данных; *SQLServer* – имя сервера с хранящимися базами данных (если используется СУБД *SQLServer*, см. рисунок 4.15).

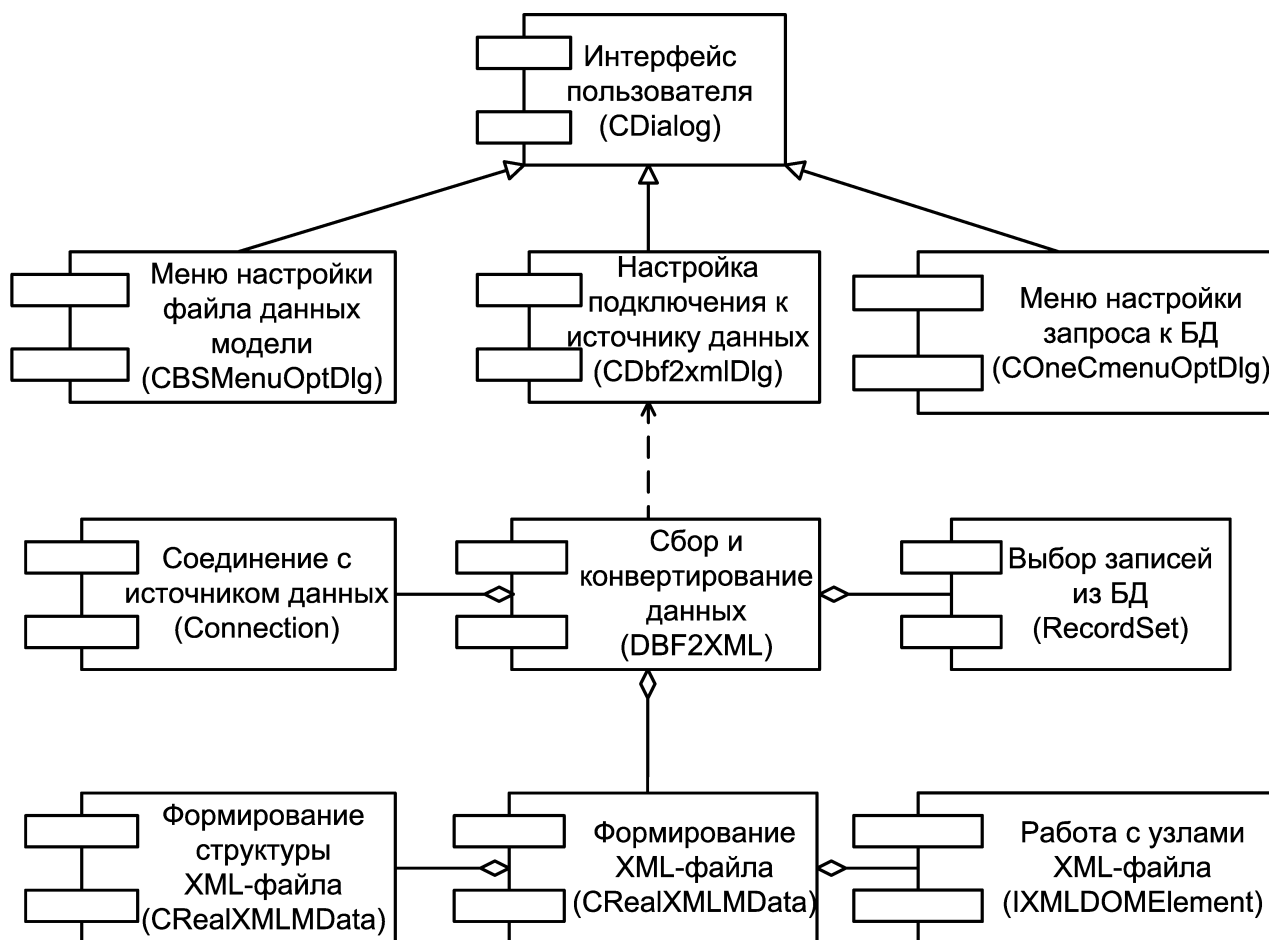


Рисунок 4.16 – Диаграмма классов приложения *BelSim Data Integrator*

Ядром приложения *BelSim Data Integrator* является метод *convert()* класса *DBF2XML* для сбора и конвертирования данных по алгоритму: сформировать объект класса *CRealXMLMData*; записать структуру XML-файла; сформировать объекты *Connection* и *Recordset*, используя *COM*; установить соединение с источником данных; сформировать и выполнить запрос к источнику данных; организовать динамический массив строк; перейти к типу *CString* и копировать записи из запроса в массив строк; затем – в XML-файл входных данных модели; установить окончание связи с источником данных. Используемый при этом класс *Connection* хранит информацию о соединении с источником данных, например, имя, расположение данных, имя провайдера *OLE DB* и т. д. Функции класса *Connection* открывают

источник данных, выполняют запрос к нему и возвращают результат запроса, закрывают подключение к источнику информации [143].

Класс *Recordset* содержит методы для обновления данных в базе данных и перемещения указателя текущей записи по результирующему набору, содержит поля с числом записей в нем, тип блокировки и т. д.

Класс *CRealXMLModelData* (см. рисунок 4.16) реализует доступ к классам библиотеки *MSXML* с помощью *COM*-технологии директивой *#import <msxml4.dll>*. При этом используются функции класса *IXMLDOMDoc* и функции класса *IXMLDOMElement*.

Функции классов *CDialog*, *CBSMenuOptDlg*, *Cdbf2xmlDlg*, *COneCmenuOptDlg* (см. рисунок 4.16) обеспечивают реализацию пользовательского интерфейса [143].

#### 4.8 Организация распределенных вычислений в *BelSim Experimenter*

Используемая концепция имитационного моделирования на основе метода Монте-Карло хорошо согласуется с моделью параллельных вычислений *MPMD* (*Multiple Program – Multiple Data*) [25]. Во-первых, иерархическая структура промышленного предприятия состоит из параллельно функционирующих компонент на разных уровнях иерархии, что приводит к использованию в ИМ модулей, одновременно реализующих поведение отдельных компонент и подсистем объекта; во-вторых, разнородность компонент и подсистем приводит к разнородности применяемых математических схем и, соответственно, порождает разнородность программной реализации модулей ИМ [71].

Эксперименты с имитационными моделями процессов функционирования промышленного предприятия требуют значительных временных затрат, для сокращения которых в ПТКИ имеется возможность распределения вычислений с использованием библиотеки *MPI* функций. Реализация функций обмена данными между распределенными вычислительными процессами выполнена для языка *C++* в программном приложении *MPICH* [100]. Для реализации *MPI*-версии в программе *Experimenter* вносятся следующие изменения [127].

*Шаг 1.* Вводятся дополнительные переменные: *MyId* для определения номера активного процесса, *NumProcs* для задания общего числа процессов в исполнительной среде, *Start* для оценки длительности функционирования *BelSim Experimenter* в режиме распределения вычислений (рисунок 4.17).

*Шаг 2.* Для слияния временных XML файлов с итогами имитационного эксперимента используется функция `mergeXML`:

```
void mergeXML(wstring fileNameDist, wstring fileNameSrc, int madeMerge)
```

где `fileNameDist` – путь к файлу для добавления файла-источника;

`fileNameSrc` – путь к файлу-источнику;

`madeMerge` – количество вызовов функции `mergeXML`.

В MPI-программе после определения переменных вводятся три обязательных строки:

```
MPI_Init(&argc,&argv);
```

```
MPI_Comm_size(MPI_COMM_WORLD,&NumProcs);
```

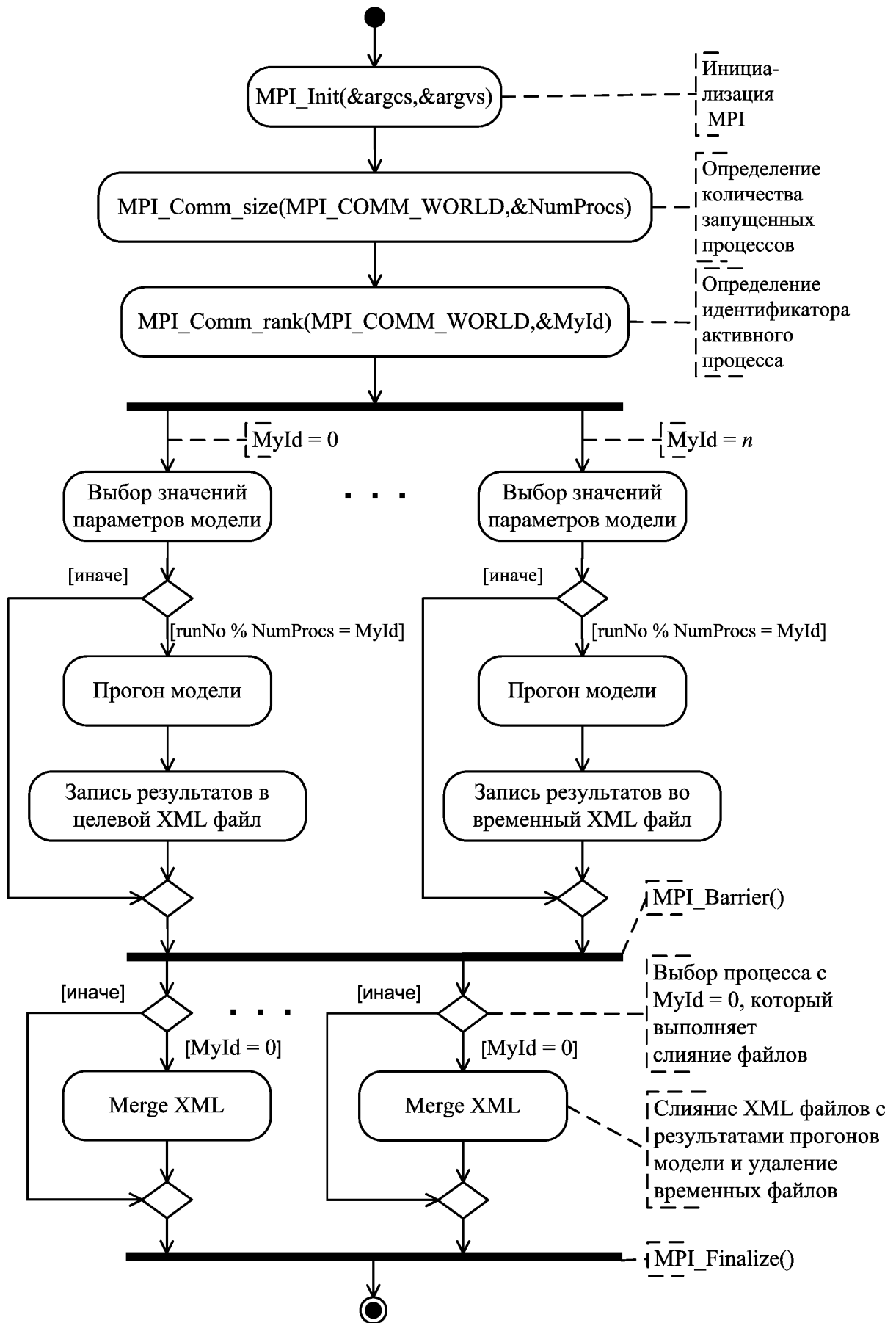
```
MPI_Comm_rank(MPI_COMM_WORLD,&MyId);
```

Обращение к `MPI_Init` должно быть первым обращением, т. к. оно инициализирует MPI.

Коммуникатор `MPI_COMM_WORLD` описывает состав процессов и связи между ними. Вызов `MPI_Comm_size` возвращает в `NumProcs` количество процессов (прогонов модели), задаваемых пользователем в программе. Вызывая `MPI_Comm_rank`, каждый процесс выясняет свой номер (`rank`) в группе, связанной с коммуникатором [100].

Синхронизация вычислительных процессов обеспечивается функцией `MPI_Barrier`. Эта функция вызывается каждым из активных процессов и приостанавливает соответствующий процесс, ожидая вызова функции всеми остальными процессами группы. Функция `MPI_Barrier` может быть завершена только одновременно всеми активными процессами [22].

В приложении Д представлен фрагмент программы *Experimenter* с дополнениями для организации распределенных вычислений.



**Рисунок 4.17 – UML-диаграмма работы программы Experimentier с распределением вычислений**

Результаты параллельных прогонов модели записываются в свой отдельный временный XML-файл. Объединение данных эксперимента из временных XML-файлов в основном файле обеспечивается функцией mergeXML (см. рисунок 4.17), создающей структуру для временных XML-файлов docSrc и основного XML-файла docDist. При этом одному из активных процессов присваивается нулевой идентификатор, и он создает основной (целевой) файл. Остальные процессы создают временные файлы. Данные из docSrc записываются в docDist, который используется как основной файл с результатами эксперимента [127].

#### **4.9 Применение подсистемы обработки данных BelSim Data Processing при эксплуатации имитационной модели в системе управления**

Для автоматизации отдельных этапов процедуры принятия решений при эксплуатации имитационной модели в системе управления разработана подсистема BelSim Data Processing. В состав BelSim Data Processing включены модули обработки статистических данных BelSim2#.random [142]; обработки числовых последовательностей данных BelSim2#.SSA [145, 147, 148]; кластеризации многомерных данных BelSim Cluster [20, 115]; коммерческий пакет программ для статистической обработки данных Statsoft Statistica.

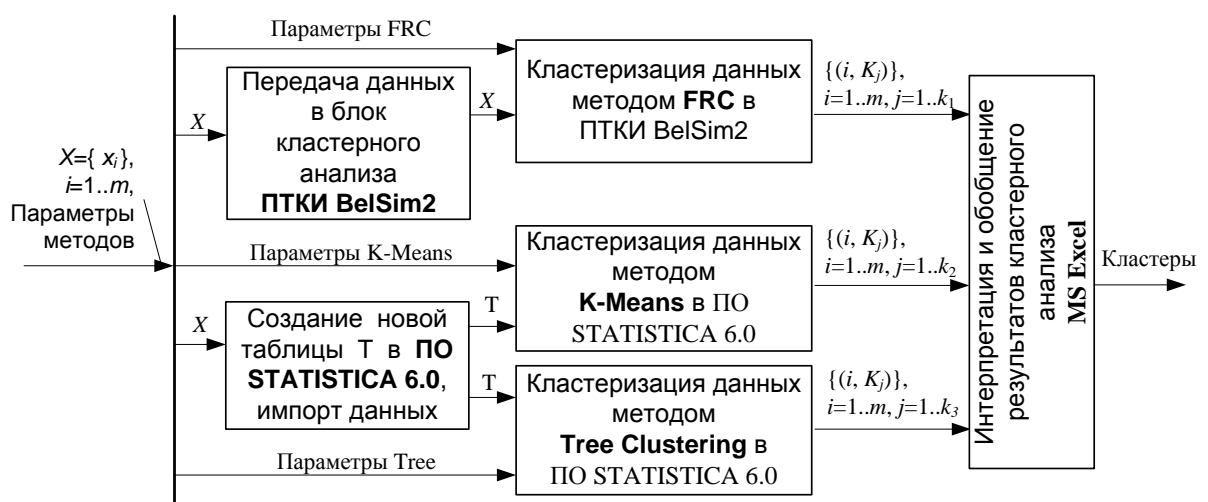
В блоке статистического анализа **BelSim2#.random** реализованы процедуры определения точечных оценок и построения статистических моделей распределения данных многомодальных выборок. Расчет кривой плотности распределения случайной составляющей в последовательности данных BelSim2#.random предназначен для построения кривой плотности обобщенного распределения, описывающей представленную выборку. Блок статистического анализа предназначен для автономного режима работы и в составе программно-технологического комплекса имитации сложных систем BelSim2# (приложение Е). Соответствие теоретического распределения закону распределения с исследуемой плотностью проверяется по критериям Колмогорова – Смирнова,  $\chi^2$ ,  $w^2$  Мизеса. Семейство кривых Пирсона включает семь основных и три частных типа распределений: равномерное, нормальное, экспоненциальное (рисунок 4.18).





**Рисунок 4.18 – Определение в BelSim2#random статистической модели распределения данных исследуемой выборки**

В блоке кластерного анализа BelSim Cluster реализуется процедура кластеризации многомерных данных. При этом для разбиения множества на кластеры предлагается использование нескольких разных методов кластеризации для контроля и уточнения результатов (рисунок 4.19).



**Рисунок 4.19 – Кластеризация многомерных данных с применением BelSim Cluster**

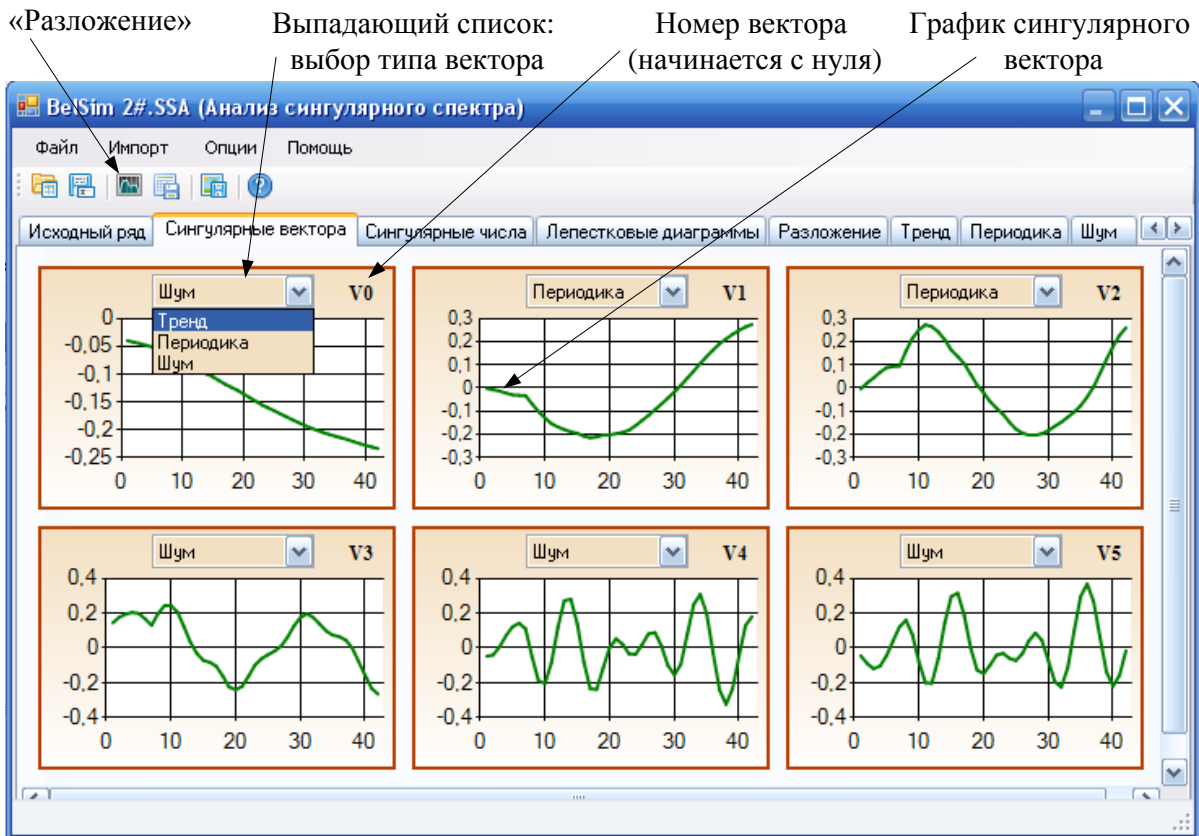
В результате разбиения исследуемого множества  $X$  на кластеры каждый из методов кластеризации ставит в соответствие порядковым номерам  $i = 1, \dots, m$  элементов  $x_i$  определенные номера кластеров  $K_j$ ,  $j = 1, \dots, k_l$ , где  $k$  – количество кластеров,  $l$  – номер метода кластеризации  $M_l$ . Интерпретация и обобщение результатов разбиения разными методами кластерного анализа выполняется в соответствии с предложенной методикой (см. рисунок 4.19).

**Блок анализа сингулярного спектра Belsim2#.SSA** предназначен для извлечения дополнительной информации из числовых последовательностей данных [149]. Сингулярный спектральный анализ (ССА) позволяет превратить путем декомпозиции исходную числовую последовательность данных в сумму небольшого количества независимых и поддающихся толкованию компонентов, таких как слабо выраженный тренд, периодическая и шумовая составляющая. Математической основой ССА является сингулярное разложение. Для успешного применения ССА следует последовательно пройти несколько шагов.

**Вложение.** На этом шаге выбирается ширина окна, от выбора которой зависят результаты применения ССА. После выбора ширины окна строится траекторная матрица  $A$  ряда, которая будет являться по условию ее построения ганкелевой.

**Сингулярное разложение.** Для матрицы  $S = A \cdot A^T$  находятся собственные числа  $\mu$  и ортонормированные собственные векторы  $U$  (рисунок 4.20). Упорядоченные по убыванию собственные числа, которые больше нуля, часто называются сингулярными числами, а соответствующие им собственные векторы – левыми сингулярными векторами  $U$ . После этого вычисляются векторы  $V$ , которые называются правыми сингулярными векторами, и находятся элементарные матрицы, на сумму которых раскладывается первоначальная траекторная матрица.

**Группировка.** На данном этапе элементарные матрицы группируются по принципу принадлежности к тренду, гармоническим колебаниям или к шуму.



**Рисунок 4.20 – Вкладка «Сингулярные вектора» в BelSim2#.SSA**

Диагональное усреднение. В соответствии с этим этапом каждый член числовой последовательности данных будет являться средним арифметическим соответствующей ему побочной диагонали траекторной матрицы.

В результате проделанных шагов получается несколько числовых последовательностей: тренд, гармонические колебания, шумовая составляющая [150].

#### 4.10 Выводы по четвертой главе

В четвертой главе диссертационной работы получены следующие основные результаты.

1. Средством реализации метода построения имитационной модели является разработанный программно-технологический комплекс имитации (ПТКИ) сложных систем BelSim. Структура ПТКИ включает BelSim IDE – интегрированную среду разработки приложений на языке C++, BelSim Simulator Core – систему имитационного моделирования на основе процессного способа имитации,

BelSim Optimizer (*Оптимизатор*) – подсистему для решения оптимизационных задач, BelSim Experimenter (*Экспериментатор*) – программу для проведения имитационных экспериментов, BelSim Data Integrator – для интеграции с КИС и ввода исходных данных. Разработана структура XML-файла, позволяющая организовать обработку статистики имитации. Предложена универсальная схема взаимодействия модулей ПТКИ.

2. Предложены средства интеграции ПТКИ *BelSim* с КИС предприятия. Для интеграции ПТКИ *BelSim* с КИС выполняются действия исследователя по установке параметров подключения к выбранному источнику данных информационной системы; подключение и чтение требуемой информации с применением технологии *ADO*, используя библиотеку *msado15.dll*; запись считанных данных в XML-файл входных данных имитационной модели; пересылка сформированного файла в ПТКИ *BelSim*.

3. С целью сокращения времени проведения имитационных экспериментов предложена технология модернизации программы *BelSim Experimenter* в ПТКИ *BelSim* для организации распределенных вычислений. Для этого используется библиотека на языке C++ *MPI*-функций обмена данными между исполняемыми процессами, которые реализуются системой *MPICH*, обеспечивающей выполнение всех функций *MPI*. Для синхронизации процессов при распределенных вычислениях используется функция *MPI\_Barrier*, которая вызывается каждым из активных процессов и приостанавливает соответствующий процесс, ожидая вызова функции всеми остальными процессами группы.

4. Для автоматизации отдельных этапов процедуры принятия решений при эксплуатации имитационной модели в системе управления разработана подсистема *BelSim Data Processing*. В состав *BelSim Data Processing* включены модули обработки статистических данных *BelSim2#.random*; обработки числовых последовательностей данных *Belsim2#.SSA*; кластеризации многомерных данных *BelSim Cluster*; коммерческий пакет статистических программ *STATISTICA*.

## ГЛАВА 5. ИССЛЕДОВАНИЕ ПРОГРАММНЫХ СРЕДСТВ ДЛЯ РЕАЛИЗАЦИИ МЕТОДА ПОИСКА РАЦИОНАЛЬНЫХ РЕШЕНИЙ НА ЭТАПЕ ЭКСПЛУАТАЦИИ ИМИТАЦИОННОЙ МОДЕЛИ ПРОМЫШЛЕННОГО ПРЕДПРИЯТИЯ

### 5.1 Система поиска рациональных решений *BelSim Optimizer*

В решении задачи поиска рациональных решений задействованы модули модели, алгоритма оптимизации, целевой функции. Их взаимодействие обеспечивает служебный модуль *BelSim Optimizer*. Схема потоков данных между указанными модулями дана на рисунке 5.1 [141].

Соответствие модулей, содержимого потоков математической модели следующее. Модули: Модель –  $A_M \in M$ ; Алгоритм оптимизации –  $A_{AO}$ ; Целевая функция –  $F_{ZO}$ ; Optimizer –  $\Phi_{ZO}$ . Потоки: Оптимизируемые параметры –  $X_{AO}$ ,  $Def_{AO}$ ; Значения постоянных параметров –  $\overline{x_G} \in X_G$ ; Погрешность решения –  $\epsilon_{AO}$ ; Значения параметров решения –  $\overline{sx_M^*} \in SX_M^*$ ; Значения параметров модели –  $\overline{sx_M} \in SX_M$ ; Значения откликов модели –  $\overline{sy_M} \in SY_M$ ; Данные о модели –  $\langle X_M, Y_M, Def_M \rangle \in M$ ; Значения параметров –  $\overline{sx_{AO}} \in SX_{AO}$ ; Значение целевой функции –  $F_{AO}(\overline{sx_{AO}}) = F_{ZO}(\overline{sx_{ZO}})$ ; Параметры алгоритма –  $P_{AO}$ ; Значения параметров целевой функции –  $\overline{sx_{ZO}} \in SX_{ZO}$ ; Вид целевой функции –  $F_{ZO} \in CF_{ZO}$ ; Параметры целевой функции –  $X_{ZO} \in CF_{ZO}$ .

Модулям (см. рисунок 5.1) соответствуют отдельные классы, формирующие структуру системы (рисунок 5.2). Класс *CCallBackFunction* представляет собой интерфейс вызова целевой функции задачи алгоритмом оптимизации.

Соответствие методов классов математической модели приведено в таблице 5.1 [141]. При рассмотрении модели ГА, используемой для решения задач оптимизации в ПТКИ *BelSim*, выделены следующие основные абстракции сущностей: популяция, индивидуум, генотип, которые можно выделить в отдельные классы *CPopulation*, *CIndividual* и *CGenotype* соответственно [109].

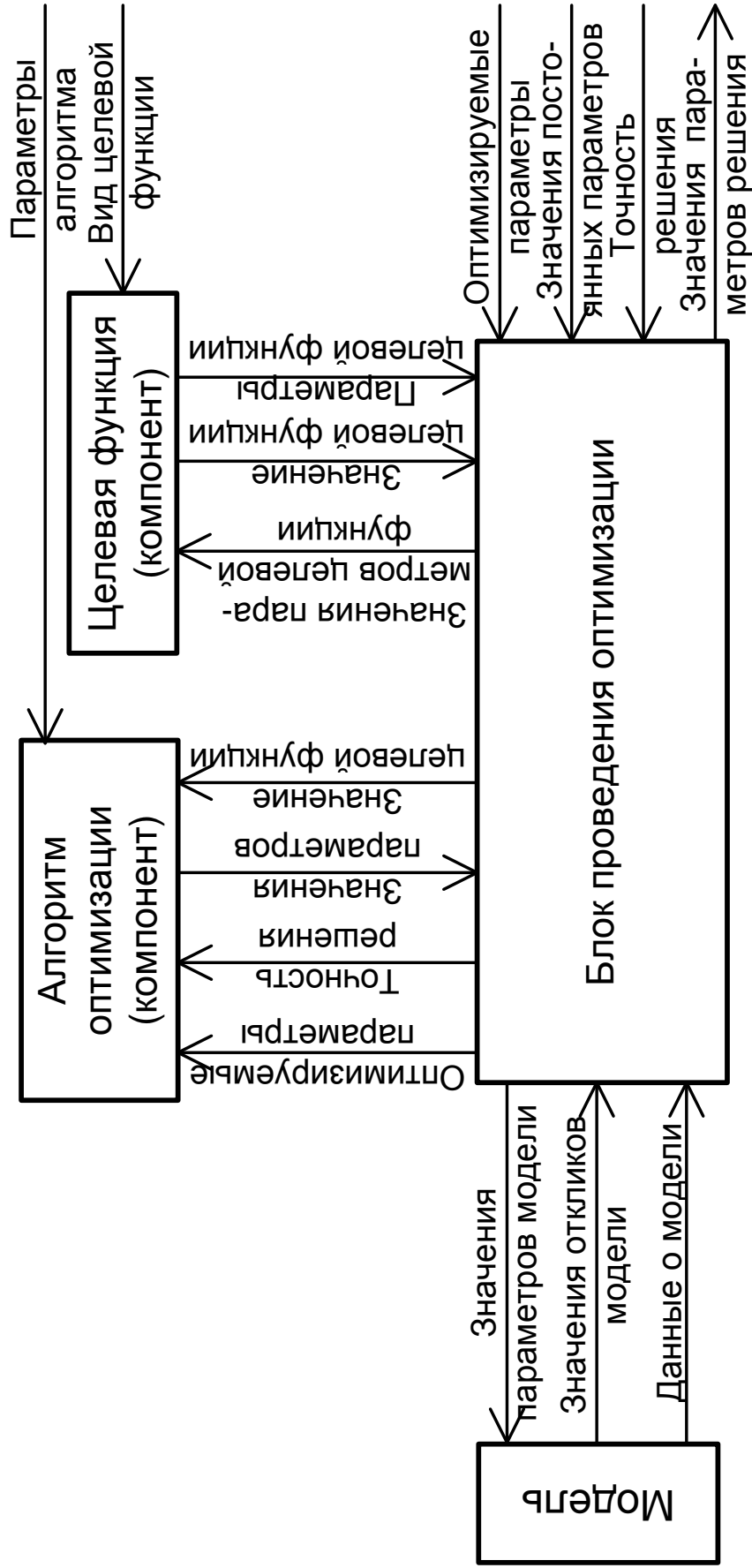


Рисунок 5.1 – Схема потоков данных в системе «Оптимизатор»

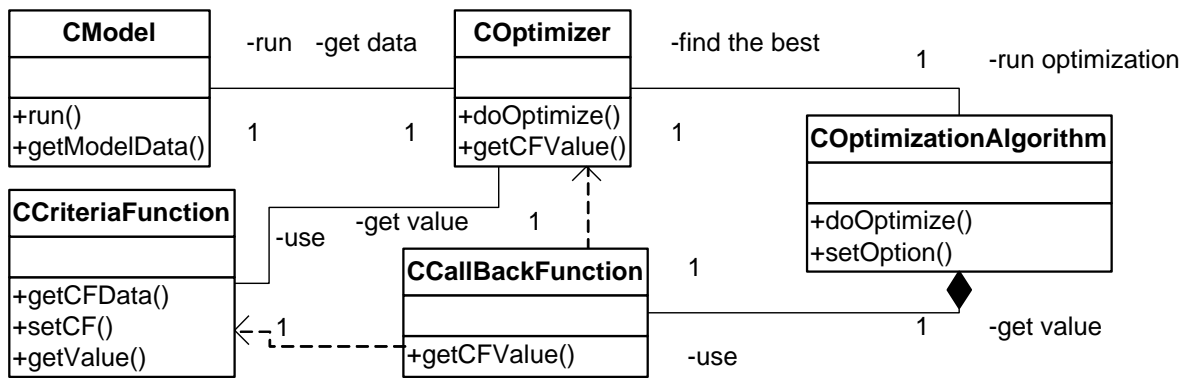


Рисунок 5.2 – Диаграмма классов системы оптимизации

Таблица 5.1 – Соответствие методов классов математической модели

Метод	Параметры	Возвращаемое значение	Назначение
<i>CModel</i> (Модель)			
<i>getModelData</i>	-	$\langle X_M, Y_M, Def_M \rangle \in M$	Вывод информации о модели
<i>run</i>	$\overline{sx}_M \in SX_M$	$\overline{sy}_M \in SY_M$	Запуск модели
<i>CCriteriaFunction</i> (Целевая функция)			
<i>getCFData</i>	-	$X_{ZO} \in CF_{ZO}$	Вывод информации о параметрах целевой функции
<i>setCF</i>	$F_{ZO} \in CF_{ZO}$	-	Формирование целевой функции
<i>getValue</i>	$\overline{sx}_{ZO} \in SX_{ZO}$	$F_{ZO}(\overline{sx}_{ZO})$	Вычисление значения целевой функции
<i>COptimizer</i> (Блок проведения оптимизации)			
<i>doOptimize</i>	$ZO$	$\overline{sx}_M^* \in SX_M^*$	Запуск задачи оптимизации
<i>getCFValue</i>	$\overline{sx}_{AO} \in SX_{AO}$	$F_{AO}(\overline{sx}_{AO}) = F_{ZO}(\overline{sx}_{ZO})$	Обращение к модели для получения значений параметров целевой функции, ее вызов и передача рассчитанного значения функции алгоритму оптимизации
<i>COptimizationAlgorithm</i> (Алгоритм оптимизации)			
<i>setOption</i>	$p_{AO} \in P_{AO}$	-	Установка параметра алгоритма
<i>doOptimize</i>	$X_{AO}, Def_{AO}, \varepsilon_{AO}, F_{AO} \in CF_{AO}$	$\overline{sx}_{AO}^* \in SX_{AO}^*$	Запуск алгоритма
<i>CCallbackFunction</i> (интерфейс вызова целевой функции задачи алгоритмом оптимизации)			
<i>getCFValue</i>	$\overline{sx}_{AO} \in SX_{AO}$	$F_{AO}(\overline{sx}_{AO})$	Вызывает метод <i>COptimizer::getCFValue</i> и возвращает его значение

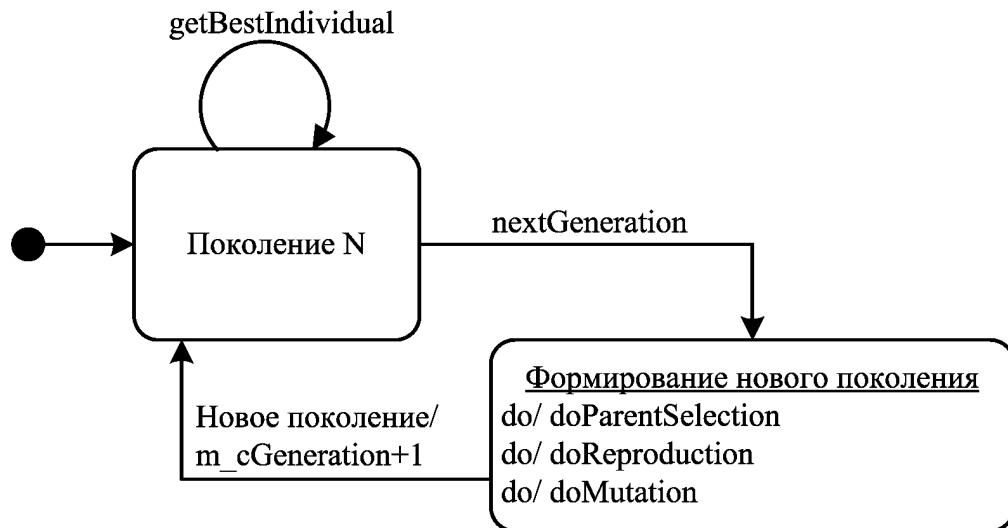
Описание класса *CPopulation*, представляющего абстракцию популяции, приведено в таблице 5.2.

**Таблица 5.2 – Представление абстракции Популяция**

Наименование	<i>CPopulation</i>	
Ответственность	Формирование очередного поколения	
Операции	<i>nextGeneration</i>	Формирование нового поколения
	<i>doParentSelection</i>	Отбор родителей
	<i>doReproduction</i>	Организация скрещивания родителей
	<i>doMutation</i>	Организация мутации полученной популяции
	<i>getBestIndividual</i>	Определение лучшего индивидуума
	<i>setCrossoverProbability</i>	Установка вероятности кроссовера пары
	<i>setMutationProbability</i>	Установка вероятности мутации индивидуума
	<i>operator[]</i>	Получение некоторого индивидуума
	<i>getGenerationNumber</i>	Возвращение номера поколения
Атрибуты	<i>m_size</i>	Размер популяции
	<i>m_vIndividuals</i>	Набор индивидуумов
	<i>m_cGeneration</i>	Номер поколения
	<i>m_crossoverProbability</i>	Вероятность кроссовера
	<i>m_mutaionProbability</i>	Вероятность мутации индивидуума

Экземпляр класса *CPopulation* имеет динамический цикл жизни, который отражен на диаграмме состояний и переходов на рисунке 5.3. При инициализации экземпляра класса *CPopulation* происходит создание начальной популяции и безусловный переход в состояние существования поколения. В данном состоянии можно получить информацию о лучшем индивидууме (операция *getBestIndividual*) и изменить параметры алгоритма, касающиеся воспроизведения (операции *setCrossoverProbability* и *setMutationProbability*). Передача сообщения *nextGeneration* о формировании нового поколения переводит его в соответствующее состояние, внутри которого происходит выполнение внутренних операций объекта (*doParentSelection*, *doReproduction*, *doMutation*). После окончания операций объект возвращается в состояние жизни, но уже нового поколения [109].





**Рисунок 5.3 – Диаграмма жизненного цикла класса CPopulation**

Описание класса *CIndividual*, представляющего абстракцию индивидуума, приведено в таблице 5.3.

**Таблица 5.3 – Представление абстракции Индивидуум**

Наименование	<i>CIndividual</i>	
Ответственность	Отражение механизма существования носителя генетического кода	
Операции	<i>make</i>	Создание индивидуума с определенной структурой генотипа
	<i>makeClone</i>	Создание себе подобного индивидуума
	<i>doMutation</i>	Мутация генотипа
	<i>doCrossover</i>	Скрещивание с другим индивидуумом
	<i>getQuality</i>	Получение значения приспособленности
Атрибуты	<i>m_genotype</i>	Генотип индивидуума
	<i>m_pGAManager</i>	Указатель на интерфейс взаимодействия с «внешней средой» алгоритма

Динамический цикл жизни экземпляра класса *CIndividual* отражен на диаграмме состояний и переходов (рисунок 5.4). Первым обязательным состоянием всякого объекта является *Рождение*, из которого после завершения одной из операций (*make* или *makeClone*) осуществляется безусловный переход в состояние существования. Сообщение *getQuality*, не изменяя состояния объекта, возвращает приспособляемость.

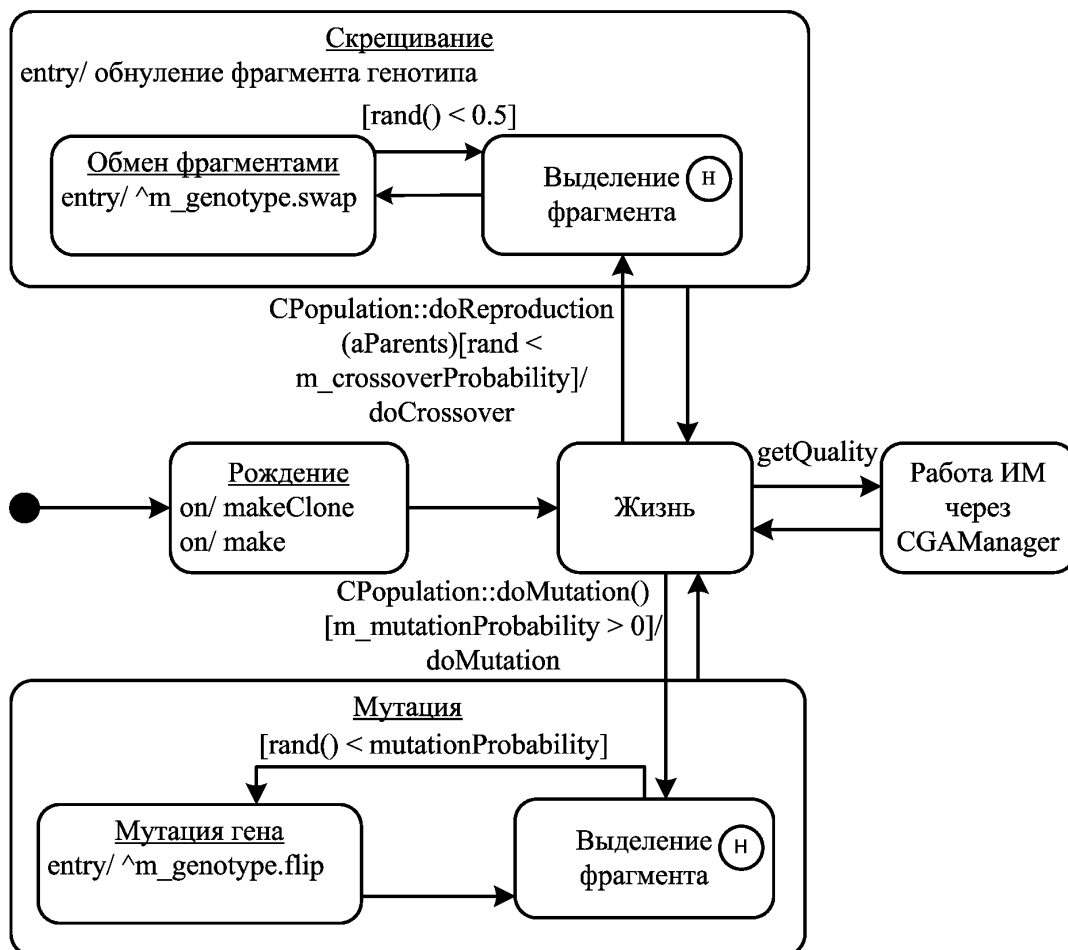


Рисунок 5.4 – Диаграмма жизненного цикла класса **CIndividual**

При истинности условия свершения скрещивания ( $CPopulation::doReproduction(aParents)[rand() < m\_crossoverProbability]$ ) операция  $doCrossover$  переводит объект в состояние скрещивания. В нем формируется потомок путем последовательного перехода для каждого выделенного фрагмента при выполнении условия  $(rand() < 0.5)$  в состояние рекомбинации. После этого идет возврат в прежнее состояние, но с новым содержанием. Схожим является состояние мутации (переход - операция  $doMutation$ ), где идет генная мутация (условие  $rand() < m\_mutationProbability$ ).

Представление абстракции генотипа приведено в таблице 5.4 [109]. При рассмотрении информационной модели системы оптимизации отдельно выделен интерфейс, обеспечивающий взаимодействие ГА с самим приложением. Данный интерфейс, представляющий собой абстракцию виртуальной машины *CGAManager*, является внешним объектом как для алгоритма (по сути, блок оп-

тимизации), так и для имитационной модели (экономическая часть задачи). Он занимает промежуточное положение и должен рассматриваться в обеих частях. Поэтому данная абстракция (таблица 5.5) включена в рассматриваемую объектную модель.

**Таблица 5.4 – Представление абстракции Генотип**

Наименование	<i>CGenotype</i>	
Ответственность	Хранение информации о генетическом коде	
Операции	<i>clear</i>	Очистка кода
	<i>assign</i>	Слияние с другим кодом в один
	<i>increase</i>	Увеличение кода на некоторое количество генов
	<i>swap</i>	Рекомбинация с другим кодом
	<i>operator[]</i>	Получение значения некоторого гена
Атрибуты	<i>m_genome</i>	Генетический код

**Таблица 5.5 – Представление абстракции виртуальной машины оптимизации**

Наименование	<i>CGAManager</i>	
Ответственность	Обеспечение взаимодействия с генетическим алгоритмом	
Операции	<i>addParameter</i>	Добавление управляемого параметра
	<i>delParameter</i>	Исключение параметра
	<i>setQualityFunction</i>	Установка связи с целевой функцией
	<i>doOptimize</i>	Запуск генетического алгоритма для поиска оптимума
	<i>getQuality</i>	Выдача значения целевой функции по генотипу
	<i>makeIndividual</i>	Создание индивидуума с определенной структурой генотипа и функцией качества
	<i>codeGenotype</i>	Создание генотипа по набору параметров
	<i>decodeGenotype</i>	Получение значений параметров по генотипу
	<i>codeInt</i>	Кодирование целочисленного параметра в генотип
	<i>decodeInt</i>	Декодирование целочисленного параметра из генотипа
	<i>codeReal</i>	Кодирование вещественного параметра в генотип
	<i>decodeReal</i>	Декодирование вещественного параметра из генотипа
Атрибуты	<i>m_vParameters</i>	Набор управляемых параметров – тип, диапазон изменения, точность
	<i>m_pQF</i>	Указатель на целевую функцию

Динамический цикл жизни экземпляра этого класса *CGAManager* отражен на диаграмме состояний и переходов (рисунок 5.5).

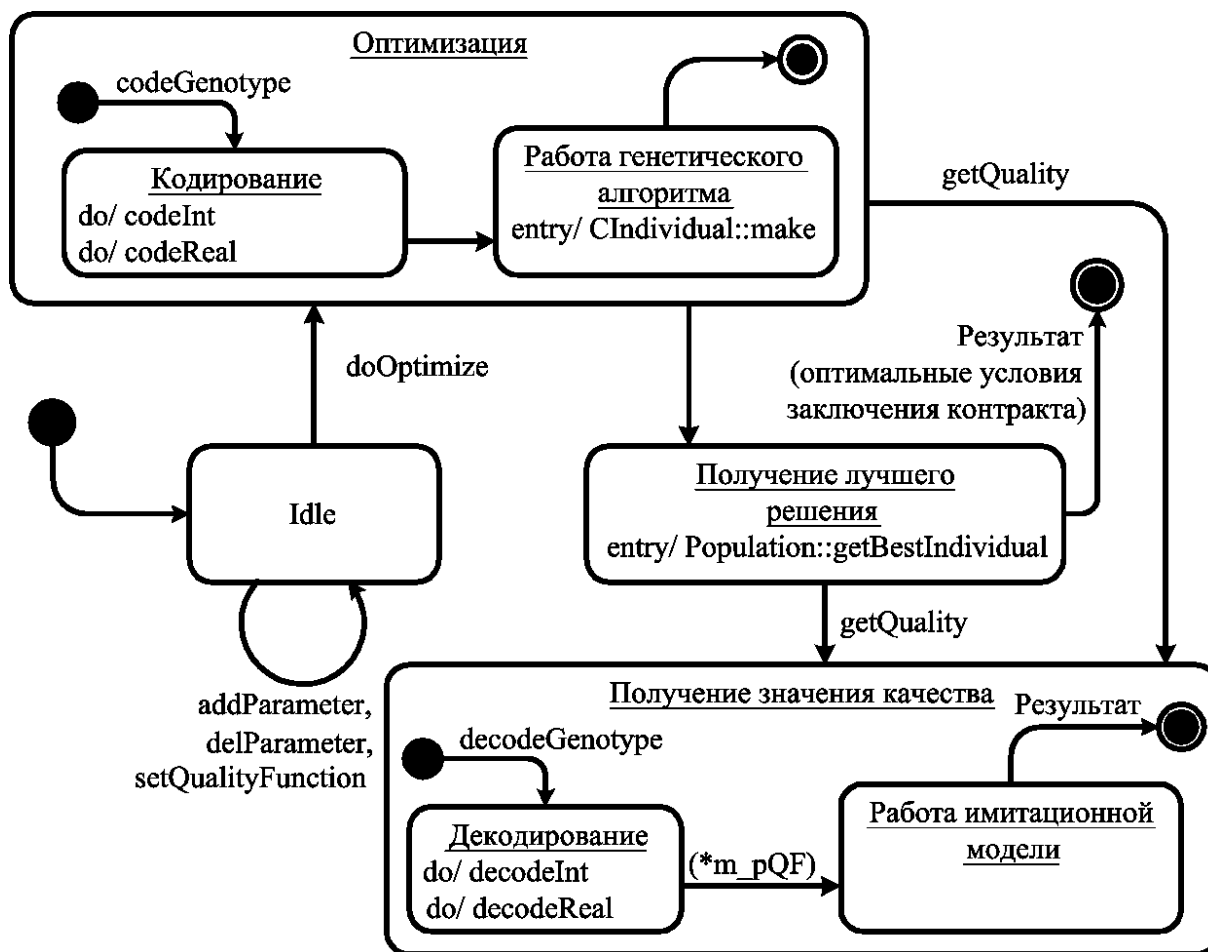
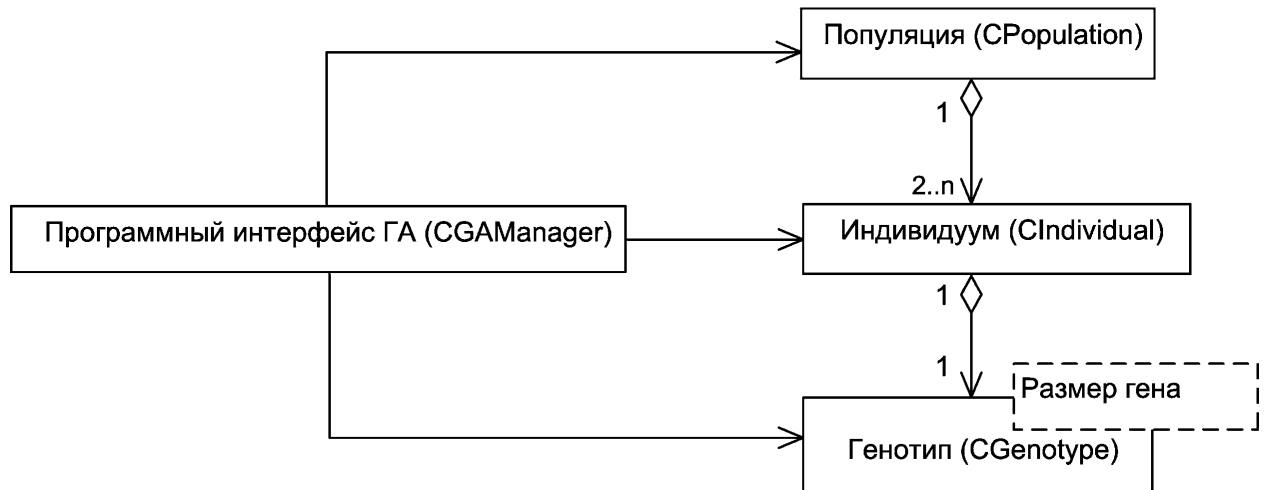


Рисунок 5.5 – Диаграмма жизненного цикла класса *CGAManager*

При появлении объект класса переходит в холостое состояние. Операции определения параметров целевой функции (*addParameter*, *delParameter*) и связи с ней (*setQualityFunction*) наполняют содержимое объекта [109]. Например, переход (операция *doOptimize*) в основное состояние оптимизации возможен лишь при наличии этих данных ( $m\_vParameters.size() > 0$ ,  $m\_pQF \neq NULL$ ). В этом состоянии первым внутренним сообщением *codeGenotype* объект переводится в субсостояние кодировки, где по мере необходимости используются операции *codeInt* и *codeReal*. Далее идет работа ГА – состояние ожидания для объекта, откуда периодически идет обратный вызов сообщением *getResultQuality*, переводящий объект в состояние декодирования. По завершении работы ГА объект переходит в состояние формирования оптимального решения в виде, понятном внешней системе. После получения лучшего индивидуума (*CPopulation::getBestIndividual*) декодируется

генотип (*getQuality*) и, вернув результат, возвращается в исходное состояние. Особенность состояния получения значения целевой функции в том, что после декодирования объект переходит в состояние ожидания, вызвав модель имитации указателем *m\_pQF* (см. таблицу 5.5).

Семантику абстракций, а также отношения между ними можно представить на диаграмме классов, ответственных за процесс оптимизации (рисунок 5.6) [109].



**Рисунок 5.6 – Диаграмма классов, ответственных за процесс оптимизации**

После определения основных абстракций строится сценарий работы системы. Работа системы заключается в получении оптимального решения, следовательно, основным сценарием является проведение оптимизации. Некий внешний объект (в общем случае экземпляр метакласса) в начале сценария путем посылки сообщений *addParameter*, *delParameter*, *setQualityFunction* объекту класса *CGAManager* создает прообраз интерфейса целевой функции, вызов которой осуществляет сам внешний объект.

Далее посылкой сообщения *doOptimize* тому же объекту инициируется процедура поиска оптимума. Объект класса *CGAManager* кодирует управляющие параметры в генотип (внутренняя операция *codeGenotype*), создает прототип объекта класса *CIndividual::make*, после чего запускает процесс эволюции сообщением *nextGeneration* объекту класса *CPopulation*, на что данный объект,

отобрав родителей (внутренняя операция *doParentSelection*), инициирует работу объекта класса *CIndividual* сообщениями *doCrossover*, *doMutation*.

*CIndividual* по мере необходимости обращается к объекту *CGenotype*, используя различные сообщения *swap*, *clear*, *assign* и др. В ходе всего этого процесса периодически объект *CPopulation* запрашивает приспособляемость объекта *CIndividual* сообщением *getQuality*.

В ответ идет одноименное сообщение интерфейсу взаимодействия *CGAManager*, который, декодировав генотип (внутренняя операция *decodeGenotype*), побуждает (функция обратного вызова (*\*m\_pQF*)) внешний объект запустить имитационную модель и дать результат при данном наборе значений параметров. По окончании моделирования эволюции объект *CGAManager* запрашивает лучшего индивидуума (*getBestIndividual*) и, декодировав его значение, возвращает внешнему объекту оптимальное решение.

Внешний объект взаимодействует с ГА только через специальный интерфейс, т. е. он не имеет никаких сведений о структуре алгоритма и наоборот (алгоритм о внешнем объекте). Имеет место контрактная модель взаимодействия клиент-сервер. Клиент, соблюдая правила интерфейса, выдает запрос на оптимизацию. Через интерфейс данные преобразуются в формат сервера (генетический алгоритм) и формируется запрос в понятной серверу форме. Сервер, приняв запрос, обрабатывает его, получает решение и, соблюдая правила интерфейса, возвращает его. Через интерфейс данные в требуемой форме выдаются клиенту [109].

## 5.2 Кластеризация целевых функций

**Идея кластеризации целевых функций.** Применение ГА в *BelSim Optimizer* для поиска рациональных решений предполагает настройку алгоритма оптимизации (АО) в зависимости от типа используемой целевой функции. При этом, для некоторых целевых функций, параметры ГА, обеспечивающие минимальное время проведения экспериментов с моделью и высокую точность рации-

ональных результатов, остаются неизменными [132, 143]. Говоря иначе, АО для групп целевых функций ведут себя одинаково.

Пусть исследователь располагает множеством  $\{CF_i\}, i=1, \dots, n$  целевых функций и АО с  $m$  параметрами (например, вероятность кроссовера и мутации, доля родителей от размера популяции и др.) и  $r$  откликами (лучшее и худшее решение поколения, среднее значение приспособленности поколения и др.). Изменения откликов АО при варьировании значений параметров, т. е. чувствительность АО, для определенных ЦФ отличаются между собой. Требуется разбить совокупность ЦФ на непересекающиеся подмножества (классы), в пределах которых АО с характеристиками по чувствительности эквивалентны.

Для разбиения совокупности объектов на классы эквивалентности с определенным свойством используют различные методы кластерного анализа, в основу которых могут быть положены эвристические методы перебора, методы математического программирования, использование матриц сходства, статистические распределения и др.

Предложена методика определения классов целевых функций с инвариантным поведением алгоритмов оптимизации. В соответствии с методикой применяют несколько разных методов кластерного анализа. Используемые методы кластерного анализа отличаются методологическими подходами к разбиению множества объектов на кластеры (классы) для проверки и уточнения результатов кластеризации [129]. Вначале разбиение выполняют двумя методами, например, K-Means и Tree Clustering. При несовпадении результатов разбиения дополнительно применяют третий метод, например, Fuzzy Relation Clustering (FRC) для уточнения кластеризации.

В итоге разбиения кластеризуемого множества каждым из методов образуется, в общем случае, разное количество классов объектов и, следовательно, разное соответствие индексации объектов множества и полученных классов. Поэтому требуется обобщение итогов кластеризации для определения классов ЦФ, в пределах которых АО эквивалентны.

**Методика определения классов целевых функций в ПТКИ BelSim.** Последовательность шагов определения классов ЦФ следующая.

**Шаг 1.** *Предварительный анализ АО.*

Для каждой тестовой ЦФ выполняется исследование чувствительности откликов АО при варьировании значений параметров АО. В результате анализа чувствительности формируются матрицы

$$S = \|s_{ij}\|, s_{ij} \in R, i = 1, \dots, r, j = 1, \dots, m, \quad (5.1)$$

где  $s_{ij}$  – чувствительность  $i$ -го отклика АО при варьировании значений  $j$ -го параметра.

Для последующего кластерного анализа матрицы  $S$  вида (5.1) преобразуются к вектору:

$$S = (s_{11}, s_{12}, \dots, s_{1m}, \dots, s_{r1}, \dots, s_{rm}). \quad (5.2)$$

Анализ чувствительности проводится в общем случае  $t$  раз для каждой ЦФ. Поэтому в итоге получают  $t \cdot n$  векторов вида (5.2), представленным множеством

$$\{S_i\}, i = 1, \dots, t \cdot n. \quad (5.3)$$

**Шаг 2.** *Кластерный анализ.*

При разбиении множества векторов (5.3) на классы используют ряд методов кластерного анализа для контроля и уточнения результатов.

В итоге множество (5.3) разбивается на  $k$  классов каждым из методов кластеризации. При этом индексам  $i = 1, \dots, t \cdot n$  элементов множества (5.3) поставлены в соответствие номера классов  $K_j, j = 1, \dots, k$ .

**Шаг 3.** *Интерпретация результатов кластерного анализа.*

Векторы  $S_i \in R^{m \cdot r}, i = 1, \dots, t \cdot n$  кластеризуемого множества (5.3) – результаты обработки статистических данных после экспериментов с  $n$  ЦФ. Поэтому результат кластеризации множества (5.3) может быть представлен матрицей относительных частот принадлежности ЦФ определенным классам:

$$P = \|p_{ij}\|, i = 1, \dots, n, j = 1, \dots, k, \quad (5.4)$$



где  $i$  – номер ЦФ,  $j$  – номер класса  $K_j$ ,  $k$  – количество классов,  $p_{ij} \in [0, 1]$  – относительная частота принадлежности  $i$ -й ЦФ классу  $K_j$ .

Величины  $p_{ij}$  в (5.4) рассчитываются как отношение количества попаданий ЦФ  $CF_i$  в класс  $K_j$  к общему количеству экспериментов с исследуемой ЦФ.

**Шаг 4. Обобщение итогов кластеризации разными методами.**

**Определение 1.** ЦФ  $CF_i$  принадлежит классу  $K_j$  ( $CF_i \in K_j$ ) тогда и только тогда, когда  $CF_i$  отнесена к данному классу  $K_j$ , по меньшей мере, двумя из трех используемых методов кластеризации.

Относительные частоты отнесения ЦФ к определенным классам одновременно тремя используемыми методами кластеризации ниже, чем относительные частоты отнесения ЦФ к определенным классам одним из используемых методов. Поэтому принята аналогия голосования большинством голосов – ЦФ относится к данному классу по меньшей мере двумя из трех используемых методов кластеризации.

**Утверждение.** Пусть  $P_1, P_2, P_3$  – матрицы вида (5.4) относительных частот принадлежности ЦФ  $CF_i$ ,  $i=1, \dots, n$  определенным классам в соответствии с тремя разными методами кластерного анализа. Тогда величины  $p_{ij} \in [0, 1]$  обобщенной (по Определению 1) матрицы  $P$  находятся суммированием коэффициентов  $P_{2,3}, P_{3,3}$  производящей функции

$$\varphi(z) = \prod_{l=1}^3 (q_{lij} + p_{lij} \cdot z) = \sum_{v=0}^3 P_{v,3} \cdot z^v,$$

где  $q_{lij} = 1 - p_{lij}$ ,  $p_{lij} \in [0, 1]$ ,  $l=1, \dots, 3$ ,  $i=1, \dots, n$ ,  $j=1, \dots, k$  – элементы матриц  $P_1, P_2, P_3$ , соответственно,

$$p_{ij} = (1 - p_{1ij})p_{2ij}p_{3ij} + p_{1ij}(1 - p_{2ij})p_{3ij} + p_{1ij}p_{2ij}(1 - p_{3ij}) + p_{1ij}p_{2ij}p_{3ij}. \quad (5.5)$$

**Доказательство.**

Пусть выполняются три независимых испытания, под которыми будем понимать кластерный анализ одним из трех разных методов.

Кластерный анализ проводится в одинаковых условиях с одними исходными данными и результаты разных методов кластеризации не зависят между собой.

Пусть событие  $A$  – попадание ЦФ в определенный класс в результате проведенного испытания. В каждом испытании событие  $A$  появляется со своей относительной частотой.

Введем обозначения:

–  $p_1, p_2, p_3$  – относительные частоты попадания ЦФ в некоторый класс в 1-м, 2-м, 3-м испытании, соответственно;

–  $q_1 = 1 - p_1, q_2 = 1 - p_2, q_3 = 1 - p_3$  – относительные частоты не попадания ЦФ в некоторый класс в 1-м, 2-м, 3-м испытании, соответственно.

Относительная частота появления события  $A$  ровно 0, 1, 2, 3 раза в указанных испытаниях равна коэффициенту при соответствующей степени  $z$  производящей функции:

$$\varphi(z) = \prod_{l=1}^3 (q_l + p_l \cdot z),$$

$$\begin{aligned} \varphi(z) &= (q_1 + p_1 \cdot z)(q_2 + p_2 \cdot z)(q_3 + p_3 \cdot z) = q_1 q_2 q_3 + (p_1 q_2 q_3 + q_1 p_2 q_3 + q_1 q_2 p_3)z + \\ &+ (q_1 p_2 p_3 + p_1 q_2 p_3 + p_1 p_2 q_3)z^2 + p_1 p_2 p_3 z^3 = \sum_{v=0}^3 P_{v,3} \cdot z^v. \end{aligned}$$

Таким образом, относительная частота того, что в этих трех испытаниях событие  $A$  не появится ни разу:  $P_{0,3} = q_1 q_2 q_3$ ; появится ровно один раз:

$P_{1,3} = p_1 q_2 q_3 + q_1 p_2 q_3 + q_1 q_2 p_3$ ; появится ровно два раза из трех:

$P_{2,3} = q_1 p_2 p_3 + p_1 q_2 p_3 + p_1 p_2 q_3$ ; появится ровно три раза из трех:  $P_{3,3} = p_1 p_2 p_3$ .

Тогда относительная частота события  $Y$  – попадания ЦФ в некоторый класс, по меньшей мере, в двух испытаниях из трех  $p(Y) = P_{2,3} + P_{3,3}$ ,

или

$$p(Y) = q_1 \cdot p_2 \cdot p_3 + p_1 \cdot q_2 \cdot p_3 + p_1 \cdot p_2 \cdot q_3 + p_1 \cdot p_2 \cdot p_3. \quad (5.6)$$

В итоге кластеризации множества (5.3) тремя методами получены матрицы относительных частот принадлежности ЦФ определенным классам  $P_1, P_2, P_3$  вида (5.4) с элементами  $p_{lij}, l = 1, \dots, 3$ :

$$P_l = \|p_{lij}\|, p_{lij} \in [0, 1], i = 1, \dots, n, j = 1, \dots, k, l = 1, \dots, 3.$$

Для обобщения результатов кластеризации тремя разными методами, применяя к элементам  $p_{lij}, l = 1, \dots, 3$  матриц  $P_1, P_2, P_3$  формулу (5.6), получим следующую обобщенную матрицу:

$$P = \|p_{ij}\|, p_{ij} \in [0, 1], i = 1, \dots, n, j = 1, \dots, k,$$

элементы которой вычисляются по формуле

$$p_{ij} = (1 - p_{1ij})p_{2ij}p_{3ij} + p_{1ij}(1 - p_{2ij})p_{3ij} + p_{1ij}p_{2ij}(1 - p_{3ij}) + p_{1ij}p_{2ij}p_{3ij},$$

$$i = 1, \dots, n, j = 1, \dots, k. \quad (5.7)$$

Доказательство закончено.

**Замечание 1.** ЦФ должна попадать хотя бы в один класс, поэтому сумма элементов строк обобщенной матрицы  $P$  должна быть равна единице. Тогда в общем случае элементы обобщенной матрицы  $P$  относительных частот принадлежности ЦФ  $CF_i, i = 1, \dots, n$  определенным классам  $K_j, j = 1, \dots, k$  вычисляются по формуле

$$p_{ij} = \begin{cases} (1 - p_{1ij})p_{2ij}p_{3ij} + p_{1ij}(1 - p_{2ij})p_{3ij} + p_{1ij}p_{2ij}(1 - p_{3ij}) + p_{1ij}p_{2ij}p_{3ij}, & j < k; \\ 1 - \sum_{j=1}^{k-1} p_{ij}, & j = k. \end{cases} \quad (5.8)$$

**Доказательство.**

Пусть события  $C_1, C_2, \dots, C_k$  – попадание ЦФ  $CF_i, i = 1, \dots, n$  в класс  $K_1, K_2, \dots, K_k$ , соответственно.

При отнесении методами кластерного анализа элементов множества (5.3) к определенному классу, ЦФ попадет хотя бы в один из  $k$  классов (произойдет одно из событий  $C_j, j = 1, \dots, k$ ). В таком случае события  $C_j, j = 1, \dots, k$  образуют полную группу событий с выполнением условия

$$p(C_1) + p(C_2) + \dots + p(C_k) = 1. \quad (5.9)$$

Тогда элементы  $p_{ij}$  матрицы  $P$  при  $j=1, \dots, k-1, i=1, \dots, n$  могут быть вычислены по формуле (5.7) при  $j=k, i=1, \dots, n$ , с учетом (5.9) – по формуле (5.8).

**Доказательство закончено.**

**Замечание 2.** Матрицы  $P_1, P_2, P_3$  для обобщения должны иметь одинаковую размерность. При разбиении множества (5.3) на  $k_1 \leq k, k_2 \leq k, k_3 \leq k$  классов, необходимо матрицы  $P_1, P_2, P_3$  привести к размерности  $k = \max\{k_1, k_2, k_3\}$ . Матрицы приводятся к требуемой размерности  $k$  дополнением столбцами с нулевыми относительными частотами попадания объекта в дополнительные классы.

**Шаг 5. Определение классов ЦФ.**

В итоге кластеризации множества (5.3) тремя разными методами построена обобщенная матрица  $P = \|p_{ij}\|, i=1, \dots, n, j=1, \dots, k$ , анализ которой позволяет определить принадлежность ЦФ определенному классу. ЦФ  $CF_i$  принадлежит классу  $K_{j_0}$  тогда, когда относительная частота принадлежности ЦФ классу  $K_{j_0}$  в построенной обобщенной матрице  $P$  максимальна.

**Пример построения классов целевых функций.** Пусть приняты тестовые ЦФ  $CF_i, i=1, \dots, n$  ( $n=5$ ):  $CF_1$  – Easom,  $CF_2$  – Six-hump,  $CF_3$  – Branin RCos,  $CF_4$  – Modified Branin RCos,  $CF_5$  – Goldstein&Price [166].

Алгоритм оптимизации – генетический алгоритм (ГА).

Параметры ГА:  $X_1$  – вероятность кроссовера,  $X_2$  – вероятность мутации,  $X_3$  – доля родителей от размера популяции,  $X_4$  – математическое ожидание для распределения количества точек разрыва. Количество параметров АО  $m=4$ .

Отклики ГА:  $Y_1$  – лучшее решение поколения,  $Y_2$  – худшее решение поколения,  $Y_3$  – среднее значение приспособленности поколения,  $Y_4$  – дисперсия приспособленности поколения,  $Y_5$  – лучшее решение,  $Y_6$  – количество запусков функции качества. Таким образом, количество откликов АО с использованием ГА  $r=6$ .

Значения параметров ГА заданы на трех уровнях (таблица 5.6), где <"-1"> означает "-1"-й уровень эксперимента, <"0"> – соответственно "0"-й, <"1"> – "1"-й уровень эксперимента.

**Таблица 5.6 – Значения параметров АО на уровнях эксперимента**

Уровни\параметры	$X_1$	$X_2$	$X_3$	$X_4$
<"-1">	0,01	0,01	0,5	2
<"0">	0,5	0,5	0,75	5
<"1">	0,99	0,99	1	8

Необходимо определить классы, к которым принадлежат целевые функции  $CF_1$  – Easom,  $CF_2$  – Six-hump,  $CF_3$  – Branin RCos,  $CF_4$  – Modified Branin RCos,  $CF_5$  – Goldstein&Price [166].

Для решения поставленной задачи в ПТКИ BelSim выполняются шаги методики определения классов целевых функций.

Элементы  $S_i$  множества (5.3), полученные в результате предварительного анализа АО на **шаге 1** методики, представлены  $4 \cdot 6 = 24$ -мерными векторами действительных чисел. Для каждой из пяти ЦФ проведен  $t = 31$  эксперимент по анализу чувствительности, общее количество элементов  $S_i$  множества (5.3) составляет  $|\{S_i\}| = 5 \cdot 31 = 155$ .

Для разбиения на классы множества векторов (5.3) (**шаг 2** методики) приняты методы кластеризации K-Means, Tree Clustering и FRC. Кластерный анализ методами K-Means и Tree Clustering выполняется с использованием математического пакета STATISTICA. Метод кластеризации FRC реализован в составе ПТКИ BelSim в программном модуле BelSim Cluster. Гипотеза относительно числа классов:  $k = 3$ .

Метод K-Means делит кластеризуемые данные ровно на  $k = 3$  классов. По запросу пакет STATISTICA предлагает построить график средних значений для каждого класса (Graph of means); вывести отдельно члены каждого класса (members of each cluster& distances) и др.

Полная классификация (Save classifications and distances), или – соответствие номерам  $i = 1, \dots, m$  элементов множества (5.3) номеров классов  $K_j$ ,  $j \in 1, \dots, k$ , сохраняется для последующего анализа.

При использовании метода Tree Clustering в пакете STATISTICA построены дендрограммы (Horizontal hierarchical tree plot), матрицы расстояний  $D$  (Distance matrix), матрицы Amalgation Schedule. Полную классификацию получают в результате анализа данных матрицы расстояний  $D$  и матрицы Amalgation Schedule.

Для использования метода кластеризации FRC данные передаются в модуль кластеризации многомерных данных BelSim Cluster в составе ПТКИ BelSim. Задается значение параметра  $\alpha \in [0, 1]$  метода FRC. С учетом варьирования значений параметра  $\alpha \in [0, 1]$  метод FRC позволяет построить ожидаемое количество классов. Полная классификация, построенная методом FRC, сохраняется для последующего анализа.

Используемые методы кластеризации разбили множество (5.3) на 3 класса.

В соответствии с методикой определения классов ЦФ разбиение выполняется двумя методами: K-Means и Tree Clustering. Интерпретированные результаты кластеризации элементов множества (5.3) (**шаг 3** методики) – матрицы  $P_1, P_2$  относительных частот принадлежности ЦФ  $CF_1, \dots, CF_5$  классам  $K_j$ ,  $j = 1, \dots, 3$  в соответствии с методами K-Means и Tree Clustering приведены ниже. Элементы матриц  $P_1, P_2$  рассчитаны как отношение количества попаданий элементов множества (5.3) в определенный класс к общему количеству экспериментов.

При получении разных результатов разбиений, как в данном случае (см. элементы  $p_{22}, p_{23}$  матриц  $P_1, P_2$ ), применяется метод FRC. Интерпретированный результат кластеризации методом FRC множества (5.3) (**шаг 3** методики) – матрица  $P_3$  – приведен ниже.

$$P_1 = \begin{pmatrix} 0,97 & 0,00 & 0,03 \\ 0,06 & 0,03 & 0,90 \\ 0,00 & 1,00 & 0,00 \\ 0,19 & 0,81 & 0,00 \\ 1,00 & 0,00 & 0,00 \end{pmatrix}, P_2 = \begin{pmatrix} 0,97 & 0,00 & 0,03 \\ 0,06 & 0,94 & 0,00 \\ 0,00 & 1,00 & 0,00 \\ 0,19 & 0,81 & 0,00 \\ 1,00 & 0,00 & 0,00 \end{pmatrix}, P_3 = \begin{pmatrix} 0,97 & 0,00 & 0,03 \\ 0,06 & 0,94 & 0,00 \\ 0,00 & 1,00 & 0,00 \\ 0,19 & 0,81 & 0,00 \\ 1,00 & 0,00 & 0,00 \end{pmatrix},$$

$$P = \begin{pmatrix} 1,00 & 0,00 & 0,00 \\ 0,01 & 0,88 & 0,11 \\ 0,00 & 1,00 & 0,00 \\ 0,10 & 0,90 & 0,00 \\ 1,00 & 0,00 & 0,00 \end{pmatrix}.$$

Матрица  $P$  – обобщенная матрица относительных частот принадлежности ЦФ определенным классам. Элементы матрицы  $P$  рассчитаны на основании (5.8) (**шаг 4** методики). Например, элементы 2-й строки  $p_{21}$ ,  $p_{22}$ ,  $p_{23}$  матрицы  $P = \|p_{ij}\|, i = 1, \dots, 5, j = 1, \dots, 3$  рассчитываются следующим образом:

$$p_{21} = (1 - 0,06) \cdot 0,06 \cdot 0,06 + 0,06 \cdot (1 - 0,06) \cdot 0,06 + \\ + 0,06 \cdot 0,06 \cdot (1 - 0,06) + 0,06 \cdot 0,06 \cdot 0,06 = 0,01,$$

$$p_{22} = (1 - 0,03) \cdot 0,94 \cdot 0,94 + 0,03 \cdot (1 - 0,94) \cdot 0,94 + \\ + 0,03 \cdot 0,94 \cdot (1 - 0,94) + 0,03 \cdot 0,94 \cdot 0,94 = 0,88,$$

$$p_{23} = 1 - 0,01 - 0,88 = 0,11.$$

В начале исследования ожидалось выделить три класса ЦФ. Методика определения классов ЦФ выявила наличие двух классов (**шаг 5** методики):

– классу  $K_1$  (с наименьшим значением чувствительности) со 100% относительной частотой принадлежат функции Easom и Goldstein&Price;

– классу  $K_2$  принадлежат функции Six-hump, Branin RCos, Modified Branin RCos с относительными частотами принадлежности классу 88 %, 100 % и 90 % соответственно [132].

Поведение ЦФ в каждом из выделенных классов отличается следующими особенностями: в класс  $K_1$  попадают одномодальные ЦФ с пологими участками,

в класс  $K_2$  – многомодальные ЦФ с множеством локальных экстремумов [4, 140].

### **5.3 Сингулярный спектральный анализ последовательностей данных на этапе эксплуатации имитационной модели**

**Идея применения накопленных данных для извлечения информации и принятия решений.** Эффективное использование имитационных моделей в системе управления производственно-экономической деятельностью предприятия сдерживается отсутствием методов и средств обработки данных, хранящихся в информационных системах, для получения дополнительной информации об особенностях исследуемых процессов. Для решения такой проблемы, например, разработан программный инструментарий, который позволяет подключаться к базе данных и извлекать их, представляя в CMSD – формате (CMSD – Core Manufacturing Simulation Data), пригодном для дискретно-событийных моделей [160].

В имитационном моделировании наиболее широко используются статистические методы обработки данных. Тип закона распределения входных данных для имитационных моделей вносит дополнительную погрешность при имитации, что вызывает необходимость применения обобщенных законов распределения Пирсона, Джонсона и др. Важным аспектом при сокращении погрешностей имитации является моделирование входных потоков данных, в частности, на основе метода сингулярного спектрального анализа (ССА) последовательностей числовых данных. ССА позволяет получить дополнительную информацию разложением числовой последовательности на аддитивные составляющие [150].

**Содержательное описание сингулярного спектрального анализа.** Числовые последовательности данных, хранящиеся в корпоративной информационной системе, могут иметь характер медленно меняющихся числовых значений с определенной тенденцией, которые в теории ССА именуют трендом или трендовой составляющей. Кроме тренда ССА позволяет выделить в числовой последо-



вательности реализацию случайной последовательности чисел (именуют шумовой составляющей или шумом) и периодическую составляющую (иначе именуемую периодикой). Сингулярный спектральный анализ числовых последовательностей состоит из этапов вложения, сингулярного разложения, группировки и диагонального усреднения.

Пусть числовая последовательность  $G$  длины  $n$  имеет аддитивные составляющие:  $G = G_T + G_H + G_N$  ( $G_T$ ,  $G_H$ ,  $G_N$  – трендовая, периодическая и шумовая аддитивная составляющая, соответственно). В соответствии с ССА-методом по числовой последовательности  $G$  строится матрица  $A$  определенного размера  $L \times K$ ,  $1 < L < n$ ,  $K = n - L + 1$  ( $L$  именуется длиной окна), которая по правилам построения является ганкелевой матрицей [52].

Определяется матрица  $S = A \cdot A^T$ , для которой составляется характеристическое уравнение  $S - \mu \cdot E = 0$  и вычисляются собственные числа  $\{\mu_k\}_{k=1}^L$ , а также левые  $\{U_k\}_{k=1}^L$  и правые ортонормированные собственные вектора  $\{V_k\}_{k=1}^L$ . Матрица  $A$  сингулярным разложением представляется в виде  $A = \sum_k A_k$ ,  $A_k = \sqrt{\mu_k} \cdot U_k \cdot V_k^T$ . Для последующего описания корень собственного числа матрицы  $S$  обозначим через  $\lambda_k = \sqrt{\mu_k}$ .

Наборы  $\langle \lambda_k, U_k, V_k \rangle$  группируются для получения трендовой, периодической и шумовой составляющей. Соответственно определяются матрицы  $A_T = \sum_{k \in T} A_k$ ,  $A_H = \sum_{k \in H} A_k$ ,  $A_N = \sum_{k \in N} A_k$ .

После выполнения этапа диагонального усреднения матриц  $A_T$ ,  $A_H$ ,  $A_N$  получают аддитивные составляющие  $G_{T\theta}$ ,  $G_{H\theta}$ ,  $G_{N\theta}$  (индекс обозначает восстановленную аддитивную составляющую) [148].

**Методика построения функции плотности распределения семейства Пирсона.** Пусть в ходе имитационных экспериментов получена числовая последовательность данных

$$X = \{x_i \mid x_i \in R, i = 1, \dots, n\}. \quad (5.10)$$

Требуется построить функцию плотности распределения выборки  $X$  на интервале  $[a, b]$  с использованием обобщенного закона распределения Пирсона.

Методика построения функции плотности распределения семейства Пирсона для выборки  $X$  реализована в программном модуле BelSim2#.random и представлена следующими шагами:

Шаг 1. Определение точечных оценок выборки  $X$ .

Для выборки  $X$  определяются следующие оценки: минимальное и максимальное значения выборки  $\text{Min}$ ,  $\text{Max}$ ; диапазон  $\text{Range}$ ; начальный момент первого порядка  $\nu_1$ ; центральные моменты  $\mu_2, \dots, \mu_4$ ; дисперсия  $D = \mu_2$ ; среднеквадратическое отклонение  $\sigma = \sqrt{D}$ ; коэффициенты асимметрии и эксцесса, введенные Пирсоном,  $\beta_1 = \mu_3^2/\mu_2^3$ ,  $\beta_2 = \mu_4/\mu_2^2$ ; а также в соответствии со статистическим пакетом STATISTICA:  $\gamma_3 = \sqrt{\beta_1}$ ,  $\gamma_4 = \beta_2 + 3$ .

Шаг 2. Построение функции  $f(x)$  плотности распределения Пирсона в соответствии с классификацией.

Для оценивания типа кривой плотности распределения Пирсоном введен показатель

$$\aleph = (\beta_1(\beta_2 + 3)^2) / (4(2\beta_2 - 3\beta_1 - 6)(4\beta_2 - 3\beta_1)). \quad (5.11)$$

При определенном значении показатель  $\aleph$  (5.11) определяет тип кривой плотности распределения Пирсона:  $\aleph < 0$  – тип I;  $\aleph = 0$  – типы II, VII;  $0 < \aleph < 1$  – тип IV;  $\aleph = 1$  – тип V;  $\aleph > 1$  – тип VI;  $\aleph = \pm\infty$  – тип III. Если  $\aleph = 0$  и  $\beta_1 = 0$ ,  $\beta_2 \neq 3$  – тип II. Если  $\aleph = 0$  и  $\beta_1 = 0$ ,  $\beta_2 = 3$  – тип VII. При  $(\beta_1, \beta_2) = (0; 1,8)$  – равномерное распределение  $R(x)$ ; при  $(\beta_1, \beta_2) = (0, 3)$  – нормальное распределение  $N(x)$ ; при  $(\beta_1, \beta_2) = (4, 9)$  – экспоненциальное распределение  $E(x)$ .

Значения параметров, описывающих функции  $f(x)$ , определяются через центральные моменты  $\mu_2, \dots, \mu_4$  и начальный момент первого порядка  $\nu_1$ .

Далее проверяется гипотеза принадлежности выборки  $X$  (5.10) закону распределения с плотностью  $f(x)$  по двум критериям согласия:  $\chi^2$  Пирсона,  $\lambda$

Колмогорова-Смирнова. При получении эквивалентных результатов по критериям  $\chi^2$  и  $\lambda$  статистическая проверка гипотезы останавливается. В противном случае дополнительно используют критерий  $\omega^2$  Мизеса, итог работы которого является заключением о проверке гипотезы.

Сохраняются следующие результаты работы критериев  $\chi^2$ ,  $\lambda$ ,  $\omega^2$ :

1) наблюдаемые значения критериев:

$$\Delta_{\chi^2} \in R_+ \cup \{0\}, \Delta_{\lambda} \in R_+ \cup \{0\}, \Delta_{\omega^2} \in R_+ \cup \{0\};$$

2) критические значения критериев:

$$\Delta_{\chi^2}^{\text{кр}} \in R_+, \Delta_{\lambda}^{\text{кр}} \in R_+, \Delta_{\omega^2}^{\text{кр}} \in R_+;$$

3) отношение наблюдаемых значений критериев к критическим:

$$dL_1 = \Delta_{\chi^2} / \Delta_{\chi^2}^{\text{кр}}; dL_2 = \Delta_{\lambda} / \Delta_{\lambda}^{\text{кр}}, dL_3 = \Delta_{\omega^2} / \Delta_{\omega^2}^{\text{кр}}; \quad (5.12)$$

4) логические результаты работы критериев:

$$bL_1 = \begin{cases} true, & dL_1 < 1; \\ false, & dL_1 \geq 1, \end{cases} bL_2 = \begin{cases} true, & dL_2 < 1; \\ false, & dL_2 \geq 1, \end{cases} bL_3 = \begin{cases} true, & dL_3 < 1; \\ false, & dL_3 \geq 1. \end{cases} \quad (5.13)$$

Шаг 3. Если на шаге 2 гипотеза о принадлежности выборки  $X$  закону распределения с плотностью  $f(x)$  отклоняется, выполняется проверка статистическими критериями  $\chi^2$ ,  $\lambda$ ,  $\omega^2$  остальных типов кривых семейства Пирсона.

По запросу пользователя выполняется построение функций равномерного, нормального и экспоненциального распределений с проверкой статистическими критериями  $\chi^2$ ,  $\lambda$ ,  $\omega^2$ .

Шаг 4. По итогам работы статистических критериев  $\chi^2$ ,  $\lambda$ ,  $\omega^2$  выбирается кривая плотности распределения. Выбор осуществляется на множестве кривых  $\{f_i(x)\}$ ,  $i = 1, \dots, |f_i(x)|$ , не отклоненных статистическими критериями, т. е. для (5.13) выполняется одно из следующих условий:

$$bL_1[i] \wedge bL_2[i] = true, \quad (5.14)$$

$$(bL_1[i] \vee bL_2[i]) \wedge bL_3[i] = true, \quad (5.15)$$

где  $[i]$  – номер функции  $f_i(x)$ ;  $bL_1, \dots, bL_3$  – результаты проверки  $f_i(x)$  по (5.13).

Функция плотности распределения  $f^*(x)$  описывает на интервале  $[a, b]$  выборку  $X$  наилучшим образом, если выполняется условие:

$$\begin{aligned} (f^*(x) = f_{i_0}(x) \in \{f_i(x)\})[\rho(X, f_{i_0}(x)) = \\ = \min\{\rho(X, f_i(x)) \mid i = 1, \dots, |f_i(x)|\}], \end{aligned} \quad (5.16)$$

где  $\rho(X, f_i(x))$  – евклидова метрика вида (5.17), если для всех  $i = 1, \dots, |f_i(x)|$  выполняется (5.14), т. е. использовались два критерия:  $\chi^2$  и  $\lambda$ :

$$dL = \rho(X, f_i(x)) = \sqrt{(dL_1[i])^2 + (dL_2[i])^2}, \quad i = 1, \dots, |f_i(x)|, \quad (5.17)$$

либо  $\rho(X, f_i(x))$  – евклидова метрика вида (5.18), если существуют такие функции  $f_i(x)$ , для которых использовались три критерия:  $\chi^2$ ,  $\lambda$ ,  $\omega^2$ :

$$dL = \rho(X, f_i(x)) = \sqrt{(dL_1[i])^2 + (dL_2[i])^2 + (dL_3[i])^2}, \quad i = 1, \dots, |f_i(x)|. \quad (5.18)$$

В случае, если все построенные кривые отклонены, рекомендуется исключение выбросов и/или разделить выборку на несколько однородных выборок с применением для каждой из них методики построения функции плотности распределения.

Методика построения функции плотности распределения реализована в программном модуле «BelSim2#.random», на вход которого подается файл model.xml с выборочными данными, значениями параметров и списком откликов.

**Технология применения ССА-метода и программного средства для определения типа шумовой составляющей.** Пусть числовая последовательность данных получена в результате имитационных экспериментов:

$G = \langle 95,0; 95,6; 97,1; 97,5; 98,0; 98,5; 99,0; 98,2; 99,2; 98,7; 99,3; 99,5; 100,0; 100,5; 101,0; 100,2; 99,4; 99,8; 101,2; 99,4; 102,0; 103,4; 104,4; 105,5; 104,0;$

105,0; 104,2; 105,2; 104,3; 105,4; 105,6; 106,6; 107,6; 105,8; 106,8; 107,2; 107,1; 106,1; 105,6; 106,0; 107,6; 107,7; 109,0>.

Задача построения модели последовательности числовых данных решается с применением программного модуля BelSim2#.random. Рассчитаны точечные оценки выборки  $G$ , в частности:  $\text{Min} = 95,00$ ,  $\text{Max} = 109,00$ ,  $\nu_1 = 102,54$ ,  $\mu_2 = 14,295$ ,  $\mu_3 = -7,698$ ,  $\mu_4 = 355,460$ ,  $\beta_1 = 0,020$ ,  $\beta_2 = 1,74$ ,  $\aleph = -0,0064$  (с заданной точностью  $\varepsilon = 0,1$  показатель  $\aleph = 0$ ). Построены функции плотностей распределений  $f_5(x)$ ,  $f_7(x)$ ,  $R(x)$ ,  $E(x)$ ,  $N(x)$ .

Результат работы программы BelSim2#.random:

$$f_5(x) = 6,244E30 * (x - 88,500)^{-20,020} * \exp(-281,084 / (x - 88,500));$$

$$f_7(x) = 0,114 * \exp(-(x - 102,540)^2 / 28,590);$$

$$R(x) = 0,071;$$

$$E(x) = 9,752E-3 * \exp(-1 * 9,752E-3 * x);$$

$$N(x) = 1 / (3,781 * (2 * \text{Pi})^{1/2}) * \exp(-1 * (x - 102,54)^2 / 28,590).$$

Функции плотностей распределений  $f_5(x)$ ,  $N(x)$ ,  $E(x)$  отклонены после проверки критериями  $\chi^2$ ,  $\lambda$ ,  $\omega^2$ , для отклонения гипотез о функциях  $f_7(x)$ ,  $R(x)$  нет оснований. Для функции  $R(x)$  значение  $dL$  (5.18) минимально, поэтому принимается соотношение  $f^*(x) = R(x)$ .

Исследование внутренней структуры числовой последовательности  $G$  выполнено с помощью ССА-метода. При этом числовая последовательность  $G$  приводится к нормированному виду следующими шагами [148] (методика реализована в программном модуле BelSim2#.SSA):

Шаг 1. Определяется среднее значение числовой последовательности  $G$ :  $\text{Mean} = 102,54$ .

Шаг 2. Определяется разность между элементами числовой последовательности и ее средним значением:  $G - \text{Mean}$ .

Шаг 3. Определяется максимальное по абсолютной величине значение из числовой последовательности  $G - \text{Mean}$ :  $\text{Max} = 7,54$ .

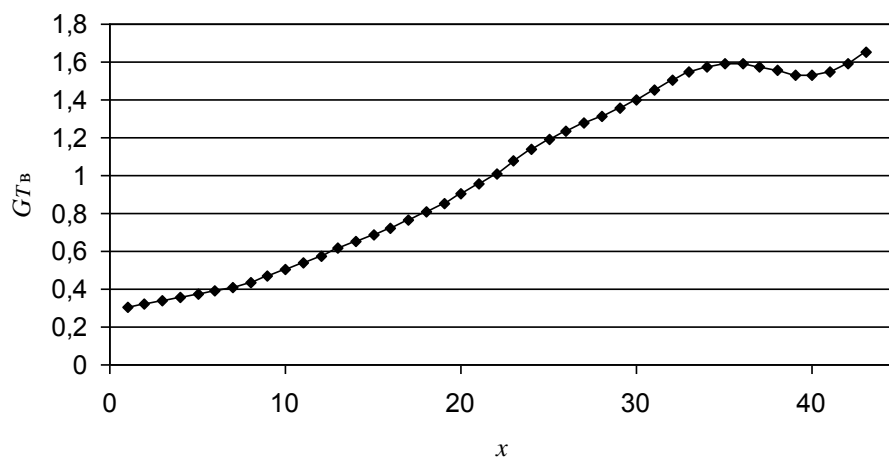
Шаг 4. Полученные на шаге 2 элементы числовой последовательности  $G$  – Mean делят на величину Max (см. шаг 3).

Шаг 5. Элементы числовой последовательности  $(G - \text{Mean})/\text{Max}$  (см. шаг 4) умножают на нормирующий множитель  $\beta$ , равный единице или десяти (принимаем  $\beta=1$ ).

Шаг 6. Элементы последовательности  $\beta \cdot (G - \text{Mean})/\text{Max}$  (см. шаг 5) сдвигают на значение параметра сдвига  $\gamma$  (в данном примере  $\gamma=1$ ).

Шаг 7. Применяя ССА-метод к нормированной числовой последовательности  $G_n$  (см. шаг 6) построена ее модель разложением на аддитивные составляющие: трендовую, периодическую и шумовую.

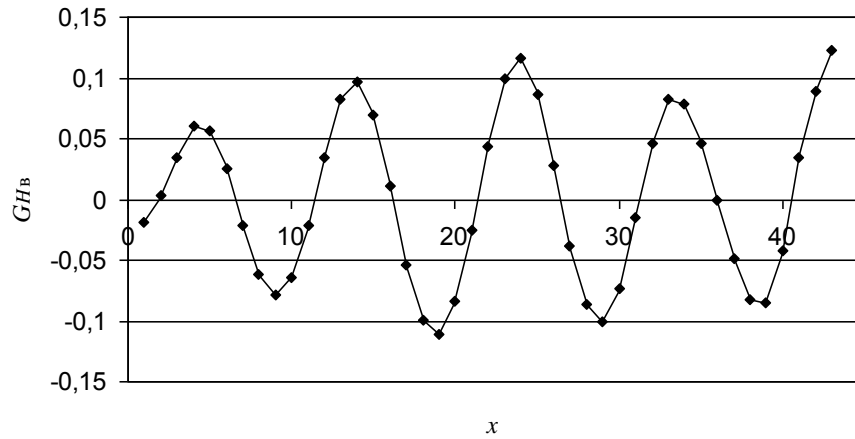
Восстановленный тренд  $G_{Tb}$  для нормированной числовой последовательности  $G_n$  представлен на рисунке 5.7.



**Рисунок 5.7 – Восстановленный тренд нормированной последовательности**

Для трендовой составляющей  $G_{Tb}$  может быть получено аналитическое выражение в виде регрессионной модели с применением математического пакета STATISTICA.

Восстановленная периодическая составляющая  $G_{Hb}$  для нормированной числовой последовательности  $G_n$  представлена на рисунке 5.8.



**Рисунок 5.8 – Восстановленная периодическая составляющая**

Аналитическая модель восстановленной периодической составляющей  $G_{НВ}$  может быть построена в математическом пакете MathCad с использованием дискретного преобразования Фурье.

Закон распределения восстановленной шумовой составляющей  $G_{НВ}$  определяется программным модулем BelSim2#.random. При этом рассчитаны точечные оценки выборки  $(\beta \cdot (G - \text{Mean})/\text{Max}) + \gamma$  (см. шаг 6):  $\text{Min} = -0,194$ ,  $\text{Max} = 0,136$ ,  $\nu_1 = -0,0104$ ,  $\mu_2 = 0,0056$ ,  $\mu_3 = -1,123$ ,  $\mu_4 = 8,743$ ,  $\beta_1 = 0,072$ ,  $\beta_2 = 2,796$ ,  $\aleph = -0,0884$  (с заданной точностью  $\varepsilon = 0,1$  показатель  $\aleph = 0$ ). Построены функции плотностей распределений  $f_1(x)$ ,  $f_2(x)$ ,  $f_3(x)$ ,  $f_5(x)$ ,  $f_7(x)$ ,  $R(x)$ ,  $E(x)$ ,  $N(x)$ .

Итоги работы программы BelSim2#.random:

$$f1(x)=5,077*(1+(x-0,0104)/0,321)^7,141*(1-(x-0,0104)/0,333)^7,426;$$

$$f2(x)=5,346*(1-(x - 0,0104)^2/0,153)^12,206;$$

$$f3(x)=5,524*\exp(99,60*(x-0,0104))*(1+(x-0,0104)/(-0,547))^54,47;$$

$$f5(x)=0,024*(x-0,289)^{-20,072}*\exp(-5,584/(x-0,289));$$

$$f7(x)=5,512*\exp(-(x-0,0104)^2/0,0112);$$

$$R(x)=3,032;$$

$$E(x)=-96,19*\exp(96,19 * x);$$

$$N(x)= 1 / (0,075 * (2 * \text{Pi})^{(1/2)})*\exp(-1*(x-0,0104)^2 / 0,0112).$$

Функции  $f_1(x)$ ,  $E(x)$  отклонены критериями  $\chi^2$ ,  $\lambda$ ,  $\omega^2$ , для отклонения

функций плотностей распределений  $f_2(x)$ ,  $f_3(x)$ ,  $f_5(x)$ ,  $f_7(x)$ ,  $R(x)$ ,  $N(x)$  нет оснований. Для функции  $f_7(x)$  значение  $dL$  (5.18) минимально, поэтому является справедливым соотношение  $f^*(x) = f_7(x)$ .

Таким образом, определена внутренняя структура исследуемой последовательности числовых данных, включающая трендовую составляющую (см. рисунок 5.7), периодическую составляющую (см. рисунок 5.8) и шумовую составляющую. Полученный результат представляет исследователю имитационной модели (в сравнении со статистическими методами) значительно больше дополнительной информации, используемой для принятия решения. Тем самым достигается повышение эффективности использования имитационных моделей в реальных производственных условиях [146, 148].

#### 5.4 Выводы по пятой главе

В пятой главе диссертационной работы получены следующие основные результаты.

1. Для реализации метода поиска рациональных решений предложена структура программной системы оптимизации и разработана схема потоков данных. Алгоритм оптимизации целевой функции взаимодействует с одной стороны с исследователем, указывающим список параметров и вид целевой функции, а с другой стороны – с базовой имитационной моделью предприятия. На входе подсистемы исследователь задает список параметров, которые подлежат оптимизации; список постоянных параметров, и необходимую точность нахождения оптимума. На выходе система возвращает исследователю оптимальные значения параметров решения задачи оптимизации. Приведен анализ жизненного цикла класса `CGAManager`, реализующего алгоритм оптимизации.

2. Предложена методика определения классов целевых функций в ПТКИ `BelSim` для алгоритмов оптимизации с эквивалентным поведением, основанных на случайном поиске и имеющих параметры, требующие дополнительной точной настройки для повышения эффективности работы самого алгоритма опти-



мизации. Методика использует несколько методов кластерного анализа для проверки и уточнения результатов. Вначале разбиение производится методами K-Means и Tree Clustering. Если получены разные результаты разбиений, применяется метод FRC. При этом решена задача построения классов исследуемых целевых функций по итогам их кластерного анализа разными методами.

3. Для повышения эффективности принятия решений при эксплуатации имитационной модели предложена методика использования в имитационном моделировании сингулярного спектрального анализа (ССА) последовательностей числовых данных для получения дополнительной информации о структуре данных в результате разложения исходной последовательности на трендовую, периодическую и шумовую составляющие.

4. Разработана технология применения ССА и программного средства для определения типа шумовой составляющей. Для проверки гипотезы о типе кривой плотности распределения семейства Пирсона выбраны следующие критерии согласия:  $\chi^2$  Пирсона,  $\lambda$  Колмогорова-Смирнова,  $\omega^2$  Мизеса. Если логические значения результатов работы критериев  $\chi^2$  и  $\lambda$  эквивалентны, то на этом этап статистической проверки гипотезы останавливается. В противном случае дополнительно применяется критерий  $\omega^2$ , итог работы которого принимается в виде заключения о проверке гипотезы.

## ГЛАВА 6. АПРОБАЦИЯ МЕТОДА ПОИСКА РАЦИОНАЛЬНЫХ РЕШЕНИЙ В ИНФОРМАЦИОННОЙ СИСТЕМЕ ПРОМЫШЛЕННОГО ПРЕДПРИЯТИЯ С ПРИМЕНЕНИЕМ ПРОГРАММНО-ТЕХНОЛОГИЧЕСКОГО КОМПЛЕКСА BELSIM

### 6.1 Состав задач, выбранных на апробацию

**Задача 1.** Одной из задач управления предприятием является задача *оптимизации распределения производственной нагрузки*. Особенно актуальной эта задача является для предприятий химической промышленности, что обусловлено следующими факторами. Себестоимость продукции химического производства определяется не только совершенством технологической схемы и степенью приближения режима процессов к оптимальному, но и в известной степени зависит от мощности производства. Эта зависимость носит в основном обратно пропорциональный характер. Относительные потери сырья и готового продукта на разных стадиях производства обратно пропорциональны величине перерабатываемых потоков и уменьшаются с увеличением производственной мощности практически линейно, приводя соответственно к снижению удельного расхода сырья на единицу готового продукта. На крупных предприятиях из-за больших объемов производства снижение себестоимости продукции даже на малую долю приводит к большой экономии в стоимостном выражении. Задача оптимизации распределения производственной нагрузки заключается в таком распределении ее между несколькими цехами, при котором себестоимость выпускаемой продукции для заданного ассортимента и объема выпуска будет минимальная.

Традиционно для решения подобных задач применяется корреляционно-регрессионный анализ. При этом требуется анализ большого количества данных, которые не всегда доступны. Кроме того, подобные модели не учитывают переходные процессы, неизбежно возникающие при изменении производственной нагрузки на химическом предприятии, что значительно снижает их практическую ценность в условиях современной экономики, требующей быстрого реагирования на изменения на рынках продукции и сырья.

Возможным решением указанной проблемы может быть применение имитационных моделей, адекватно отражающих динамику технологических процессов. Однако в этом случае сложно, а порой и невозможно, представить целевую функцию в аналитическом виде, либо последняя имеет сложный вид (наличие множества локальных экстремумов, точек разрыва), делающий невозможным применение традиционных методов оптимизации [126].

**Задача 2.** *Выявление свойств имитационных моделей контрактов* продажи, купли, на дилерское обслуживание и по товарному обмену для обоснования области применения аналитических моделей оценки эффективности указанных контрактов, поиск оптимальных решений при заключении контрактов в маркетинговой деятельности промышленного предприятия, оценка влияния параметров контрактов на финансово-экономическое состояние предприятия. Для оценки финансово-экономического состояния приняты следующие показатели:  $K_1$  – коэффициент текущей ликвидности;  $K_2$  – доля собственных оборотных средств в общей их сумме;  $K_3$  – коэффициент обеспеченности собственными средствами.

**Задача 3.** Промышленное объединение представляется многоуровневой системой, в которой решение нижестоящего уровня зависит от задания со стороны вышестоящего уровня, а решение вышестоящего уровня, в свою очередь, зависит от откликов нижестоящего уровня. Для такой системы важным является описание нижестоящего уровня как взаимодействие взаимосвязанных подсистем, каждая из которых преследует собственные цели. При этом основное внимание обращено на внутриуровневые и межуровневые взаимодействия. Такой подход обеспечивает при моделировании целостность рассмотрения сложной системы, удерживая в поле зрения всю систему в целом для решения задачи управления с позиции системной цели.

На самом высоком уровне рассматриваются вопросы распределения ресурсов и взаимовлияние между подсистемами с точки зрения всей системы в целом и вопрос об эффективности и допустимости ее режимов. Наиболее детально при моделировании рассматриваются только те подсистемы, в которых имеется

существенное снижение степени эффективности и допустимости режимов. *Выбор типа координации процесса планирования производственной программы* позволяет выбрать оптимальную стратегию управления промышленным производством [108].

**Задача 4.** Чтобы оценить возможность развития ПТКИ Belsim для решения проблем в смежных предметных областях ставится задача *разработки библиотеки моделей для исследования информационной системы промышленного предприятия*. Библиотека должна включать ИМ бизнес-процессов планирования производства и продаж продукции, ИМ информационной системы производственного процесса, ИМ бизнес-процесса управления складскими запасами материальных ресурсов, ИМ функционирования информационной системы в локальной вычислительной сети [143].

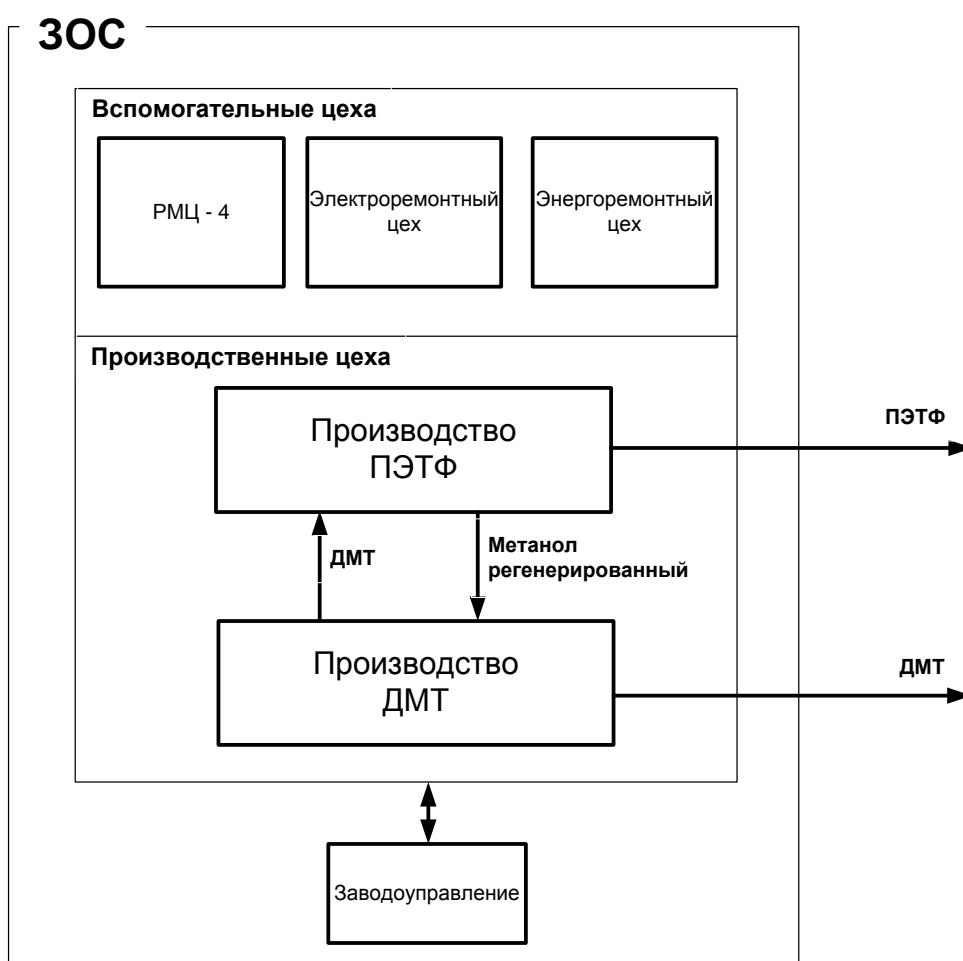
Решение задачи 4 представлено в монографии [143]. Библиотека имитационных моделей, разработанная в ПТКИ BelSim, существенно расширяет область его применения.

## **6.2 Решение задачи оптимизации производственного процесса**

### **6.2.1 Структура исследуемого производственного объединения**

Структура химических производственных объединений в своей структуре содержит несколько специализированных заводов [125]. На таких предприятиях при увеличении объемов производственной нагрузки, в условиях постоянной технологической схемы и ее параметров изготовления химической продукции, себестоимость конечного продукта уменьшается. Связано это с тем, что хотя расход основных и дополнительных материальных ресурсов на стадии химического преобразования почти не зависит от технических характеристик химического оборудования, относительные потери как основного сырья и вспомогательных материалов, так и готовой продукции уменьшаются с увеличением объемов их переработки. Такая тенденция имеет место, например, при производстве

диметилтерефтолата (ДМТ) на заводе органического синтеза (ЗОС) ОАО «Могилевхимволокно» (рисунок 6.1), когда почти прямолинейное увеличение мощности производства приводит к сокращению относительного расхода материальных и энергетических ресурсов на единицу готовой продукции. Если производство состоит из нескольких технологических потоков с разным оборудованием, то для составления производственной программы формулируется задача оптимального распределения нагрузки по цехам предприятия.



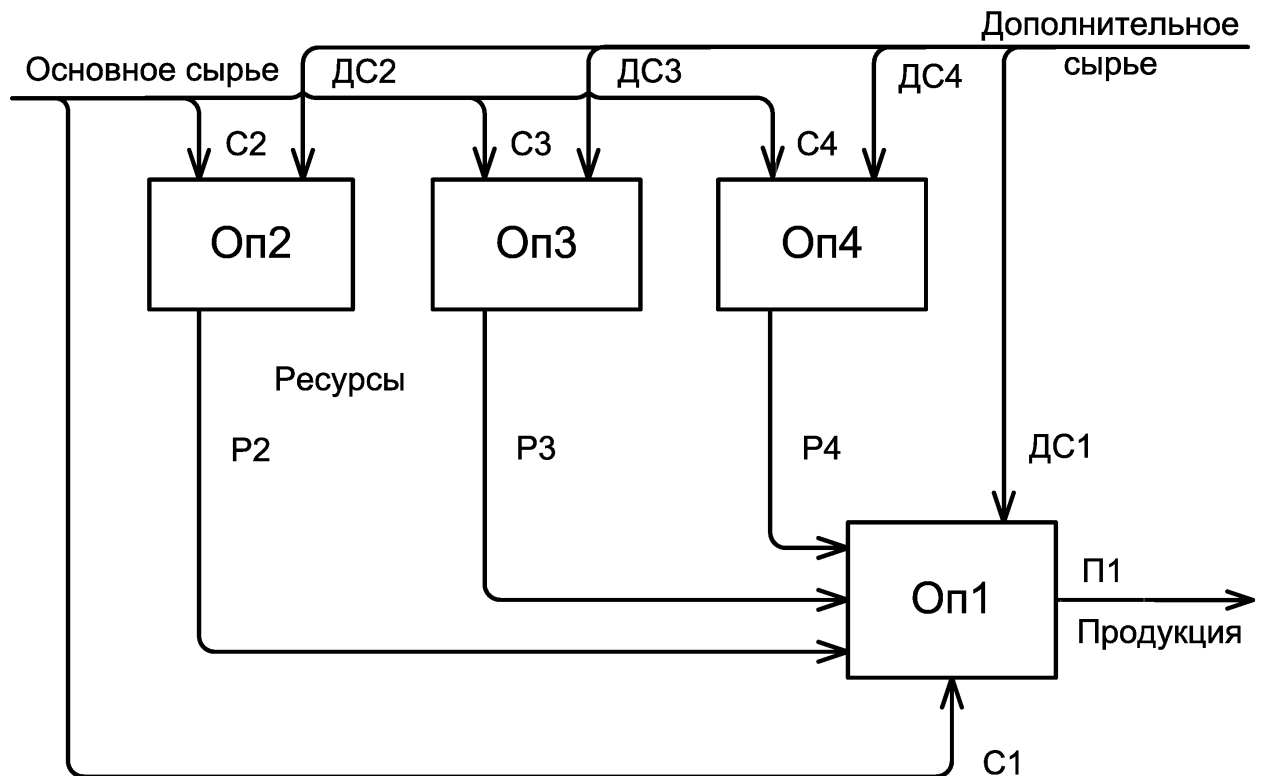
**Рисунок 6.1 – Структура завода органического синтеза**

В соответствии со стандартом MRPII, система управления предприятием должна включать такие функции, как планирование производственных мощностей, планирование производственных операций, управление на уровне производственных цехов. С ростом сложности производственных процессов возника-

ет проблема определения оптимального управления производством и планирования выпуска продукции. Следует подчеркнуть, что получившие широкое распространение и признание, как в отечественной, так и в зарубежной литературе методики расчета производственной мощности предприятия, цеха, участка основаны либо на принципе прямого счета, либо на принципе оптимизации загрузки оборудования с помощью линейно-программной модели. Расчеты на их основе выполняются для целей текущего (годового, квартального) планирования выпуска продукции. Используемая для этого информация является не только существенно агрегированной, но и не учитывает непрерывность технологического процесса, его цикличность и партионный характер движения предметов труда от одной производственной операции к другой.

Таким образом, для решения рассматриваемой проблемы следует учитывать значимые параметры непрерывного технологического процесса и характеристики, определяющие динамику его поведения. Одним из подходов к решению проблемы может быть имитационное моделирование производственного процесса, а эффективным средством реализации модели на основе имитации процессами может быть ПТКИ BelSim.

На рисунке 6.2 производственный процесс определен производственными операциями Оп1, ..., Оп4, из которых Оп2, ..., Оп4 перерабатывают ресурсы (основное сырье С2, ..., С4 и дополнительное сырье ДС2, ..., ДС4) в промежуточную химическую продукцию (ресурсы Р2, ..., Р4 соответственно). Операция Оп1 предназначена для переработки ресурсов Р2, Р3, Р4, сырья С1 и ДС1 в готовую химическую продукцию П1. Каждая из отмеченных операций Оп1, ... Оп4 отличаются нормативными показателями расхода сырья и ресурсов, производительностью выпуска продукции.



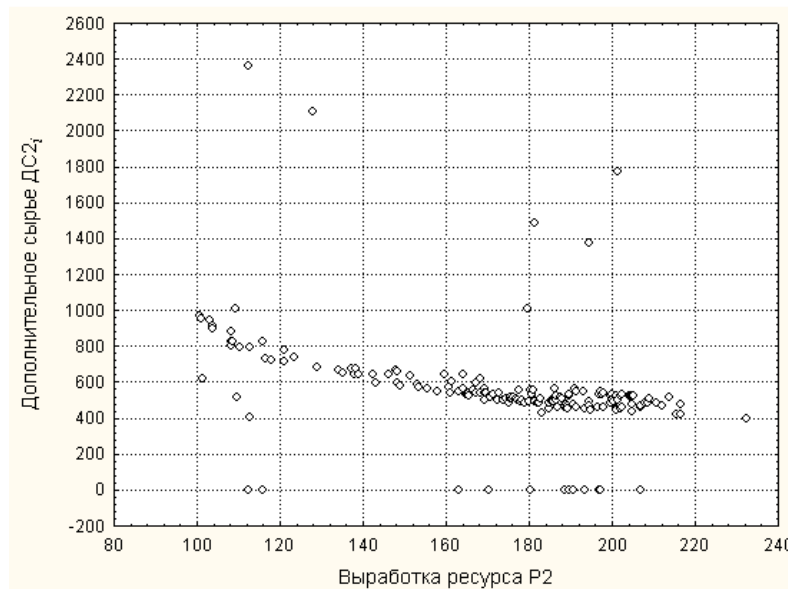
**Рисунок 6.2 – Схема производственного процесса**

Таким образом, в соответствии со схемой производственного процесса (см. рисунок 6.2) себестоимость выпускаемой продукции П1 зависит от загрузки операций Оп2, ... Оп4 и себестоимости выпускаемых ресурсов (Р2, ... Р4). В этом случае оптимальные параметры производственной программы можно определить, варьируя распределением производственной нагрузки между операциями для минимизации себестоимости готовой продукции.

### **6.2.2 Технология обработки исходных данных для моделирования**

Удельные нормы расхода основного сырья для всех операций определяются технологией производства и являются постоянными. Иначе обстоит дело с дополнительным сырьем. Удельный расход дополнительного сырья, исходя из опытных исследований, изменяется по обратно пропорциональному закону в зависимости от объема выпуска (выработки) продукции. Зависимость удельных норм от выработки может быть найдена в результате анализа и обработки статистической информации из баз данных информационной системы предприятия.

Такой анализ и обработка проведены в программе STATISTICA. На рисунке 6.3 представлены данные по удельному расходу дополнительного сырья  $ДС2_i \in ДС2$  для операции Оп2 в зависимости от суточной выработки в течение 5 лет.



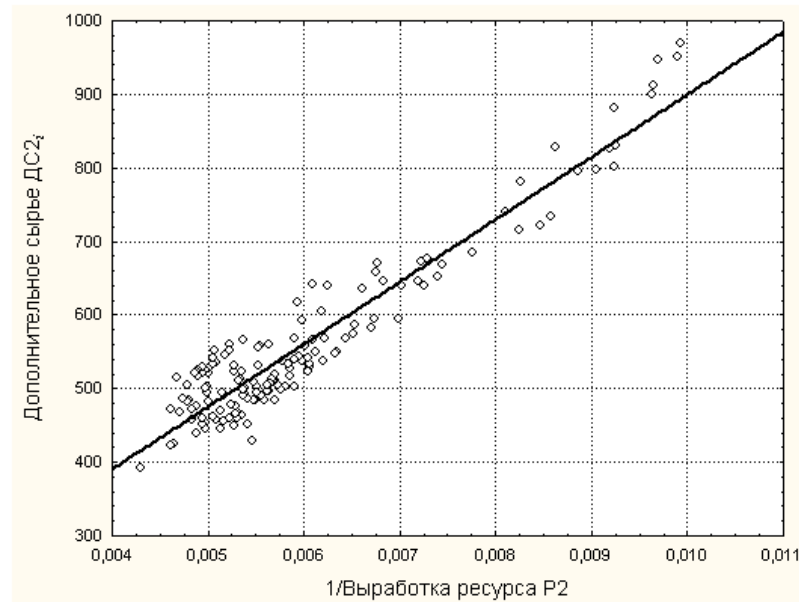
**Рисунок 6.3 – Удельный расход дополнительного сырья  $ДС2_i$  для операции Оп2 в зависимости от суточной выработки**

На графике (см. рисунок 6.3) явно просматривается некоторая зависимость, а также присутствуют некоторые, так называемые, «выбросы», которые обусловлены различными причинами, а именно: ошибки при измерении, ошибки при снятии показаний со счетчиков, ошибки при внесении информации в базу данных и т. д. Для того чтобы «очистить» данные от выбросов необходимо задать требуемый коэффициент корреляции. Коэффициент корреляции Пирсона находится для линейных зависимостей, следовательно, необходимо перейти от обратно пропорциональной зависимости к линейной. Это можно сделать, построив зависимость не от выработки, а от величины, обратной выработке.

На основании экспертных оценок сделан вывод о достоверности или ошибочности этих данных. В большинстве случаев разброс обусловлен ошибками измерений или ошибками внесения данных в базу данных. В пакете *STATISTICA* существует инструмент «щетка», при помощи которого можно удалить очевид-



ные выбросы. На рисунке 6.4 представлена зависимость удельного расхода дополнительного сырья ДС<sub>2i</sub> от величины, обратной суточной выработке, но уже без выбросов [221].

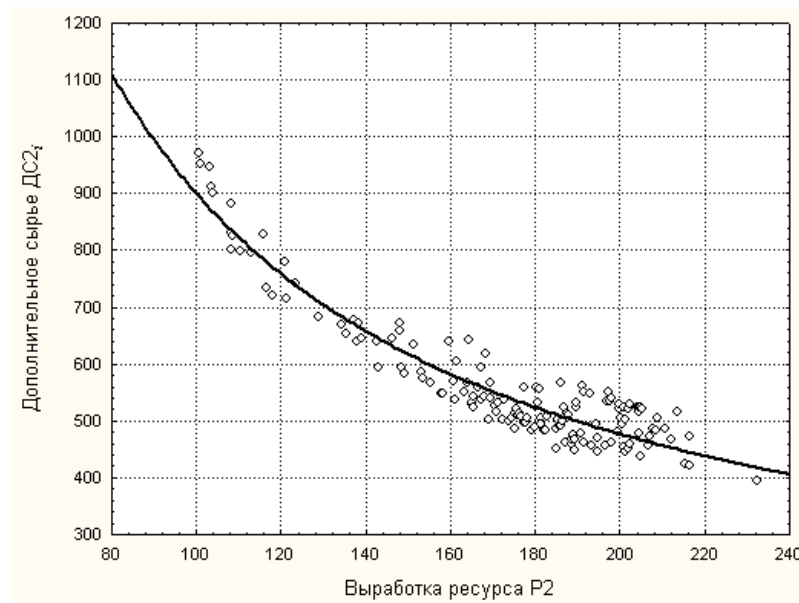


**Рисунок 6.4 – Удельный расход дополнительного сырья ДС<sub>2i</sub> от величины, обратной суточной выработке ресурса P2 без выбросов**

Для данных на рисунке 6.4 коэффициент корреляции составляет 0,95, что говорит о наличии достаточно сильной зависимости между двумя параметрами. Также *STATISTICA* автоматически выводит уравнение регрессии для данной зависимости. Можно построить график зависимости расхода дополнительного ресурса ДС<sub>2i</sub> от выработки уже без учета выбросов. Такой график представлен на рисунке 6.5 и аппроксимируется функцией вида

$$y = \frac{K}{x} + B, \quad (6.1)$$

где коэффициенты  $K$  и  $B$  принимаются из уравнения регрессии, построенного в пакете *STATISTICA*.



**Рисунок 6.5 – Зависимость удельного расхода дополнительного сырья ДС<sub>2i</sub> от суточной выработки ресурса P2 без учета выбросов**

Сплошной линией на рисунке 6.5 показана линия регрессии. Таким образом, получена зависимость удельного расхода дополнительного сырья ДС<sub>2i</sub> от суточной выработки ресурса P2 для операции Оп2. Аналогичным образом находятся остальные зависимости удельного расхода каждого дополнительного сырья от суточной выработки для операций Оп2, ..., Оп4 (см. рисунок 6.2).

На исследуемом предприятии основная информация о себестоимости продукции и нормах расхода сырья хранится в информационной системе 1С: Предприятие, а статистическая информация о нормах расхода дополнительного сырья хранится в программе MS Excel.

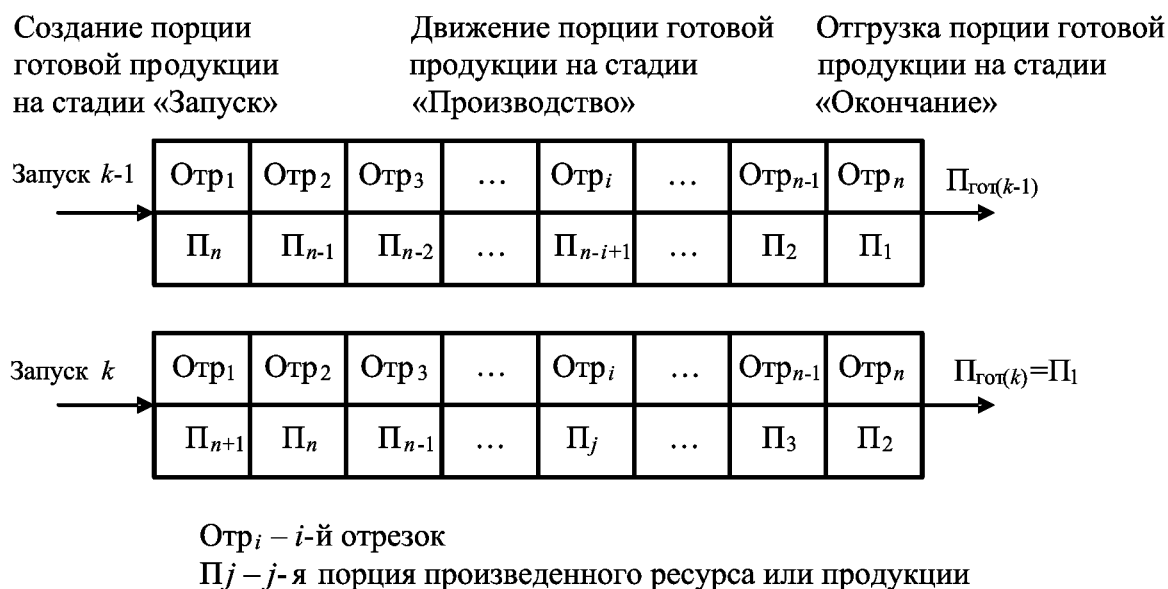
### **6.2.3 Имитационная модель непрерывного производственного процесса**

Рассматриваемый непрерывный производственный процесс моделируется процессным способом имитации [143]. Имитационная модель состоит из процессов, реализующих следующие функции:

- а) имитация работы отдельной производственной операции;

б) оценка состояния производственного процесса и сбор статистики результатов имитации.

Алгоритм процесса имитации работы отдельной производственной операции состоит из функциональных действий на стадиях «ЗАПУСК», «ПРОИЗВОДСТВО», «ОКОНЧАНИЕ». Моделируемый непрерывный производственный процесс разделяется на  $n$  отрезков. При имитации отрезками производственный процесс будет запускаться  $n$  раз, соответственно расход ресурсов и выпуск готовой продукции на каждом отрезке в  $n$  раз меньше. Движение ресурсов по отдельным отрезкам при реализации производственной операции представлено на рисунке 6.6, в соответствии с которым порции готовой продукции при выполнении процесса сдвигаются по одному отрезку вперед и из последнего отрезка  $Отр_n$  продукция  $П_{\text{гот}}$  отгружается на склад.



**Рисунок 6.6 – Движение ресурсов по отдельным отрезкам при реализации производственной операции**

На стадии «ЗАПУСК» проверяется наличие ресурсов на складе и если они есть, то происходит их выделение и добавление порции готовой продукции в первый отрезок; если сырья на складе недостаточно, то первый отрезок помечается пустым.

На стадии «ПРОИЗВОДСТВО» (рисунок 6.7) имитируется перемещение продукции в последовательности отрезков, а также сбои и отказы в производственной операции. При имитации сбоев и отказов перед запуском модели задается момент времени отказа и сбоя для отдельной операции. При достижении такого момента времени производство останавливается с некоторой задержкой и соответствующей потерей в изготовлении продукции.

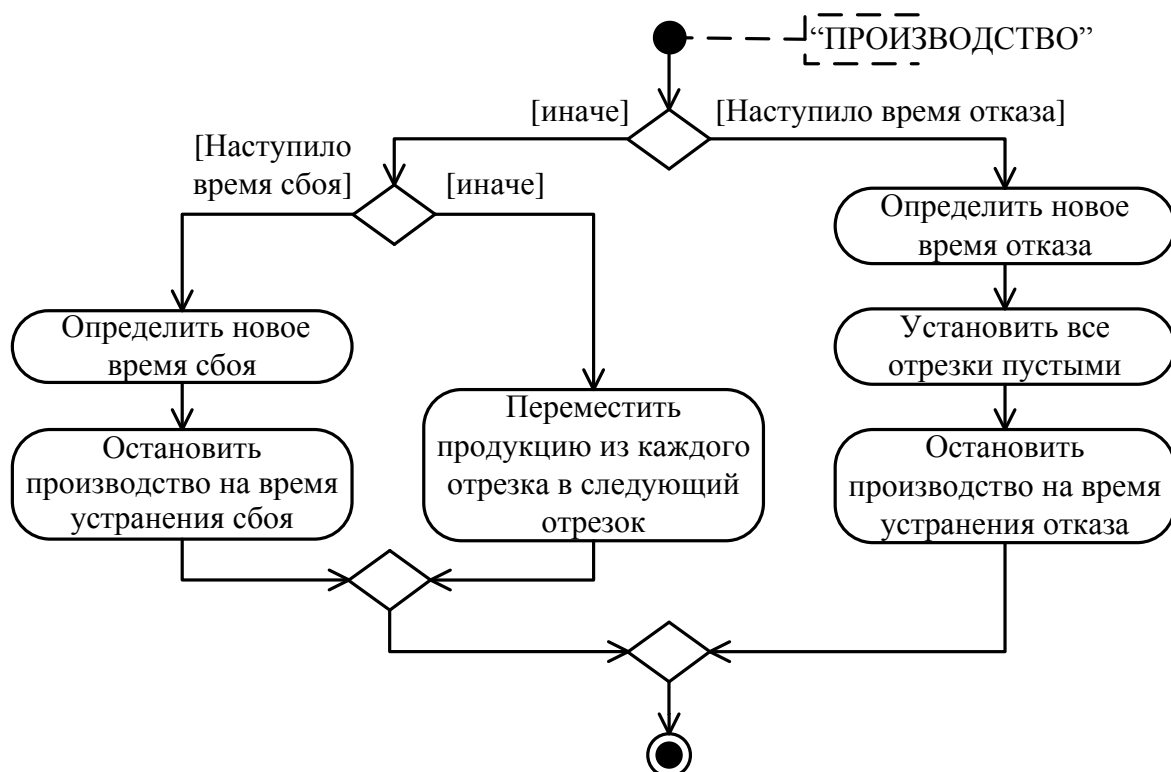


Рисунок 6.7 – Алгоритм стадии «ПРОИЗВОДСТВО»

Стадия «ОКОНЧАНИЕ» завершается перемещением произведенной продукции на склад.

Выполнение стадий в последовательности «ЗАПУСК», «ПРОИЗВОДСТВО», «ОКОНЧАНИЕ» определяет алгоритм, имитирующий непрерывную операцию с учетом производственных сбоев и отказов (при одном отрезке алгоритм имитирует дискретную производственную операцию) [143].

Алгоритм процесса для оценки состояния производственного процесса и сбора статистики результатов имитации состоит из двух стадий:

- 1) анализ текущих результатов имитации;

2) вывод статистики результатов имитации.

#### **6.2.4 Исследование имитационной модели непрерывного производственного процесса**

В имитационной модели производственного процесса (ПрП) (см. рисунок 6.2) длина разгона (переходного процесса) определяется длительностью запуска производства продукции, т.е. пока исследуемые производственные операции (ПрО) не войдут в стационарный режим функционирования. Исследование длины разгона модели проведено на основе параметра «Себестоимость готовой продукции», который стабилизируется после 135 ед. модельного времени. В течение этого периода обеспечено начало функционирования всех ПрП. Статистика, собираемая в течение разгона модели, подлежит исключению. Поэтому сбор статистики имитационной модели начинается после 135 ед. модельного времени [143].

Пусть для исследуемого непрерывного ПрП (см. рисунок 6.2) необходимо решить оптимизационную задачу нахождения минимальной себестоимости изготовленной продукции при известном нормативном показателе выпуска продукции, равным 600 т. Исходные данные реального ПрП приведены в таблице 6.1, где коэффициенты  $K$  и  $B$  (формула 6.1) определяются из уравнения регрессии, построенного по статистическим данным для расчета количества  $j$ -го ресурса, необходимого для реализации  $i$ -ой операции.

В таблице 6.2 указана себестоимость основного и дополнительного сырья. Ограничения на выработку ресурсов приведены в таблице 6.3. Операции Оп2, ..., Оп4 производят ресурсы Р2, ..., Р4 соответственно (см. рисунок 6.2). Для Оп3 и Оп4 устанавливается среднее значение загрузки операций. При этих условиях для Оп2 загрузка варьируется от минимального до максимального значения, что позволяет построить графическую зависимость (рисунок 6.8) себестоимости выпускаемой продукции от разных значений выработки Р2.

**Таблица 6.1 – Исходные данные реального производственного процесса**

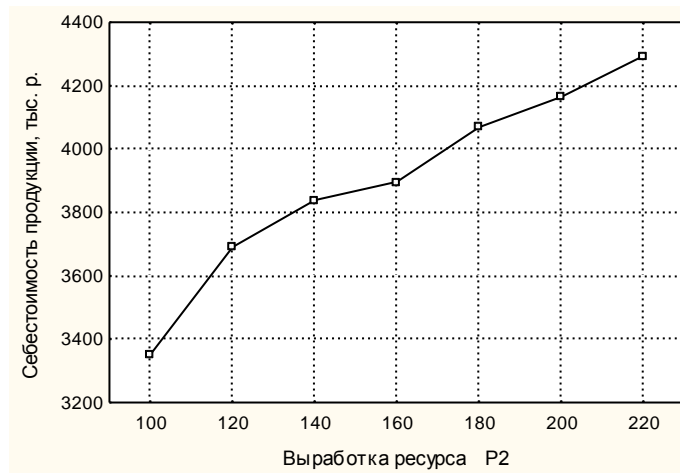
Операция-1			Операция-2		
Основное сырье	Кол-во, т/сут		Основное сырье	Кол-во, т/сут	
Сырье-4	10,00		Сырье-1	0,62	
			Сырье-2	0,38	
			Сырье-19	2,00	
			Сырье-4	14,80	
			Сырье-7	0,10	
			Сырье-8	0,20	
Дополнительное сырье	Коэф. В	Коэф. К	Дополнительное сырье	Коэф. В	Коэф. К
Дополнительное сырье-11	15,53	6156,96	Дополнительное сырье-11	531,65	27864,13
Дополнительное сырье-12	3,77	8,96	Дополнительное сырье-12	0,59	543,04
			Дополнительное сырье-13	53,10	84669,53
Дополнительное сырье-14	1,31	21,44	Дополнительное сырье-14	0,71	8212,35
			Дополнительное сырье-15	73,84	10718,96
Операция-3			Операция-4		
Основное сырье	Кол-во, т/сут		Основное сырье	Кол-во, т/сут	
Сырье-1	0,62		Сырье-1	0,62	
Сырье-2	0,38		Сырье-2	0,38	
Сырье-19	2,00		Сырье-19	2,00	
Сырье-4	14,80		Сырье-4	14,80	
Сырье-5	0,23		Сырье-5	0,23	
Дополнительное сырье	Коэф. В	Коэф. К	Дополнительное сырье	Коэф. В	Коэф. К
Дополнительное сырье-11	451,51	32576,49	Дополнительное сырье-11	346,19	51200,16
Дополнительное сырье-12	-1,54	593,65	Дополнительное сырье-12	-0,70	661,45
Дополнительное сырье-13	53,46	72708,80	Дополнительное сырье-13	105,08	103040,00
Дополнительное сырье-14	2,26	6852,12	Дополнительное сырье-14	0,4756	6007,26
Дополнительное сырье-15	75,38	6805,23	Дополнительное сырье-15	61,36	26490,78
Дополнительное сырье-16	0	12	Дополнительное сырье-16	0	19,6

**Таблица 6.2 – Себестоимость основного и дополнительного сырья**

Наименование ресурса	Цена, руб/т	Наименование ресурса	Цена, руб/т
Сырье-1	700000	Дополнительное сырье-11	50
Сырье-2	170000	Дополнительное сырье-12	10000
Сырье-4	1500	Дополнительное сырье-13	25
Сырье-5	200	Дополнительное сырье-14	50
Сырье-6	3500	Дополнительное сырье-15	60
Сырье-7	20	Дополнительное сырье-16	100000
Сырье-8	1500	Сырье-19	7042

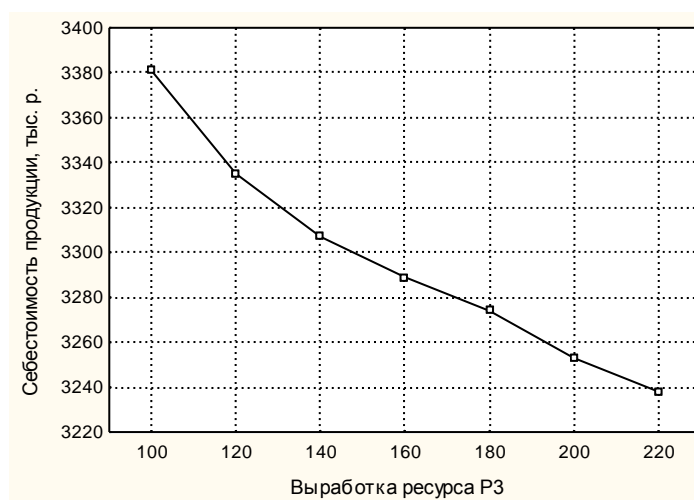
**Таблица 6.3 – Ограничения для операций по выработке ресурса**

Значение выработки	Выработка ресурса по операциям, т/сут		
	Операция-2 (Ресурс-2)	Операция-3 (Ресурс-3)	Операция-4 (Ресурс-4)
<b>Минимальное</b>	100	100	185
<b>Номинальное</b>	200	200	370
<b>Максимальное</b>	220	220	410



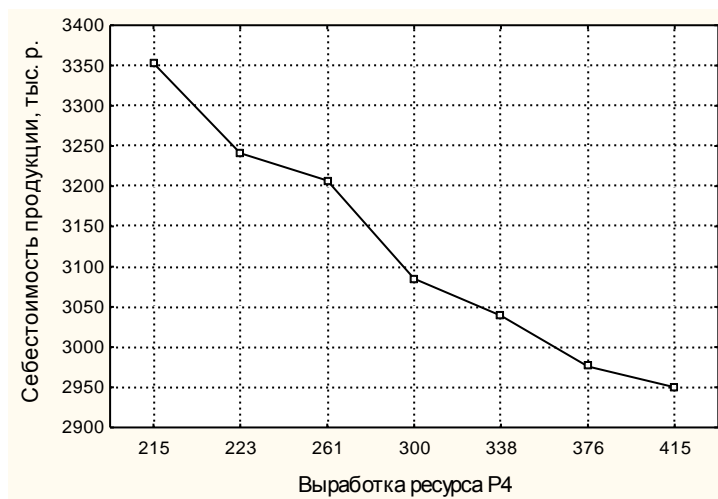
**Рисунок 6.8 – Зависимость себестоимости продукции от выработки ресурса P2**

По результатам построений (см. рисунок 6.8) определяется рациональная загрузка Оп2, обеспечивающая минимальное значение себестоимости продукции. Затем (при полученном значении выработки P2 операцией Оп2 и среднем значении загрузки Оп4) для Оп3 загрузка варьируется от минимального до максимального значения, чтобы построить графическую зависимость (рисунок 6.9) себестоимости выпускаемой продукции от разных значений выработки P3.



**Рисунок 6.9 – Зависимость себестоимости продукции от выработки ресурса P3**

По результатам построений (см. рисунок 6.9) определяется рациональная загрузка Оп3, обеспечивающая минимальное значение себестоимости продукции. При полученных рациональных значениях выработки ресурсов Р2 и Р3 для Оп4 загрузка варьируется от минимального до максимального значения (рисунок 6.10).



**Рисунок 6.10 – Зависимость себестоимости продукции от выработки ресурса Р4**

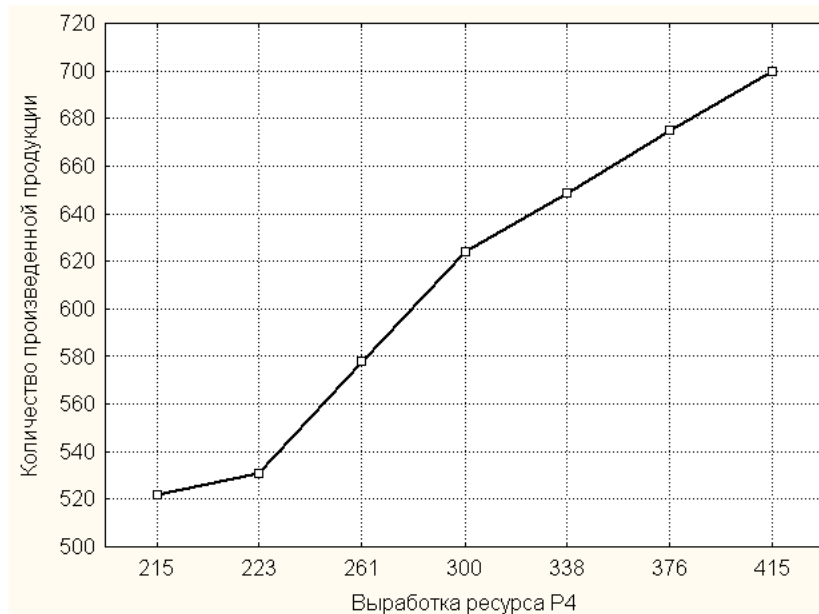
По результатам выполненных расчетов себестоимость продукции минимизируется при максимальных значениях выработки ресурсов Р3 (см. рисунок 6.9) и Р4 (см. рисунок 6.10). Поэтому для определения оптимальной загрузки операций строится графическая зависимость себестоимости продукции одновременно от выработки ресурсов Р3 и Р4.

Аналогично построена графическая зависимость объема выпущенной продукции от выработки ресурсов Р3 и Р4. На этом графике строится плоскость, соответствующая выпуску продукции 600 т. В результате пересечения плоскости с трехмерным графиком получается кривая, по которой определяют соотношение выработок ресурса Р3 и Р4 для заданного выпуска продукции.

Таким образом, для минимизации себестоимости выпускаемой продукции обеспечивают минимальную загрузку Оп2 и максимальную загрузку Оп3, а загрузка Оп4 зависит от требуемого выпуска продукции (рисунок 6.11). По выработке Р4 определяют в итоге себестоимость выпускаемой продукции. Опти-



мальные параметры загрузки операций следующие: Оп2 – 100 т/сут.; Оп3 – 220 т/сут.; Оп4 – 280 т/сут.; себестоимость выпускаемой продукции – 3151 тыс.р./т.



**Рисунок 6.11 – Зависимость количества выпущенной продукции от выработки ресурса P4 при максимальной выработке ресурса P3 и минимальной выработке ресурса P2**

Свойства исследуемой модели в соответствии с формулой (1.12) представлены следующим образом:

$$Q = \langle A(S, D, E) \rangle = \langle A(4, 4, 5) \rangle$$

т. к. в модели исследуется производственная деятельность  $A$  для нескольких операций предприятия ( $S = 4$ ). Модель в ходе производственной эксплуатации динамически изменяется в зависимости от параметров и структуры технологического оборудования ( $D = 4$ ), определяет оптимальное распределение выработки ресурсов для выпуска продукции между операциями производства ( $E = 5$ ).

### **6.2.5 Реализация динамической производственно-экономической модели ЗОС на уровне производства диметилтерефтолата**

Традиционная технология распределения производственной нагрузки между цехами состоит в следующем. Из-за отсутствия достаточного количества

сырья и заказов на производство диметилтерефтолата (ДМТ) предприятие не в состоянии использовать весь свой производственный потенциал и, таким образом, обеспечить работу всех трех цехов на проектных мощностях. Вследствие этого цех ДМТ-2 (Оп2) практически не работает, а производственные мощности распределяются между цехами ДМТ-3 (Оп3) и ДМТ-4 (Оп4) (см. рисунок 6.2). Причем руководствуются следующим соображением: так как проектная мощность цеха ДМТ-4 приблизительно в два раза больше чем у ДМТ-3, следовательно, исходя из обратно пропорционального закона снижения себестоимости продукции с увеличением загрузки, принимается решение обеспечить номинальную работу цеха ДМТ-4, а остальную часть производства сосредоточить в цехе ДМТ-3. В случае необходимости в строй вводится цех ДМТ-2. Как только план производства ДМТ снижается до определенного уровня, недостаточного для обеспечения работы трех цехов, цех ДМТ-2 останавливается.

При анализе эмпирических данных, хранящихся в корпоративной информационной системе (КИС), было установлено, что одним из путей снижения себестоимости продукции является распределение производства продукции между цехами таким образом, чтобы удельная стоимость сырья и энергии была минимальной. В качестве исходных данных для решения поставленной задачи используются данные о выпуске продукции при различной загрузке цехов, в качестве ограничений – общее количество продукции, которое необходимо произвести за определённый период (месяц, квартал), а также максимальный и минимальный выпуск продукции для каждого цеха. Необходимо отметить, что цеха характеризуются разным расходом сырья и энергии на единицу выпускаемой продукции, что обусловлено характеристиками оборудования и особенностями технологических процессов.

Методика решения такой задачи состоит из двух этапов [143]:

1. *Обработка эмпирических данных.* Данные по расходу ресурсов при разной загрузке цехов импортируются из корпоративной информационной системы в пакет STATISTICA. На основе полученных данных для каждого цеха

строятся зависимости удельных значений расхода сырья и энергии от выпуска продукции:

$$PP_{ik} = f(KП_k),$$

где  $PP_i$  – расход  $i$ -го вида ресурса (сырья или энергии) в  $k$ -ом цехе на единицу продукции,  $KП$  – количество продукции.

При построении зависимостей из рассмотрения исключаются данные, которые не попадают в диапазон значений  $(PP_{ik} - 2\sigma_{PPik}; PP_{ik} + 2\sigma_{PPik})$ .

2. *Оптимизация загрузки цехов.* Полученные на предыдущем этапе зависимости переносятся в табличный процессор Excel, где осуществляется переход от количественных зависимостей к стоимостным путем умножения расходов ресурсов на их цены ( $ЦP_i$ ). При поиске оптимального решения используется целевая функция

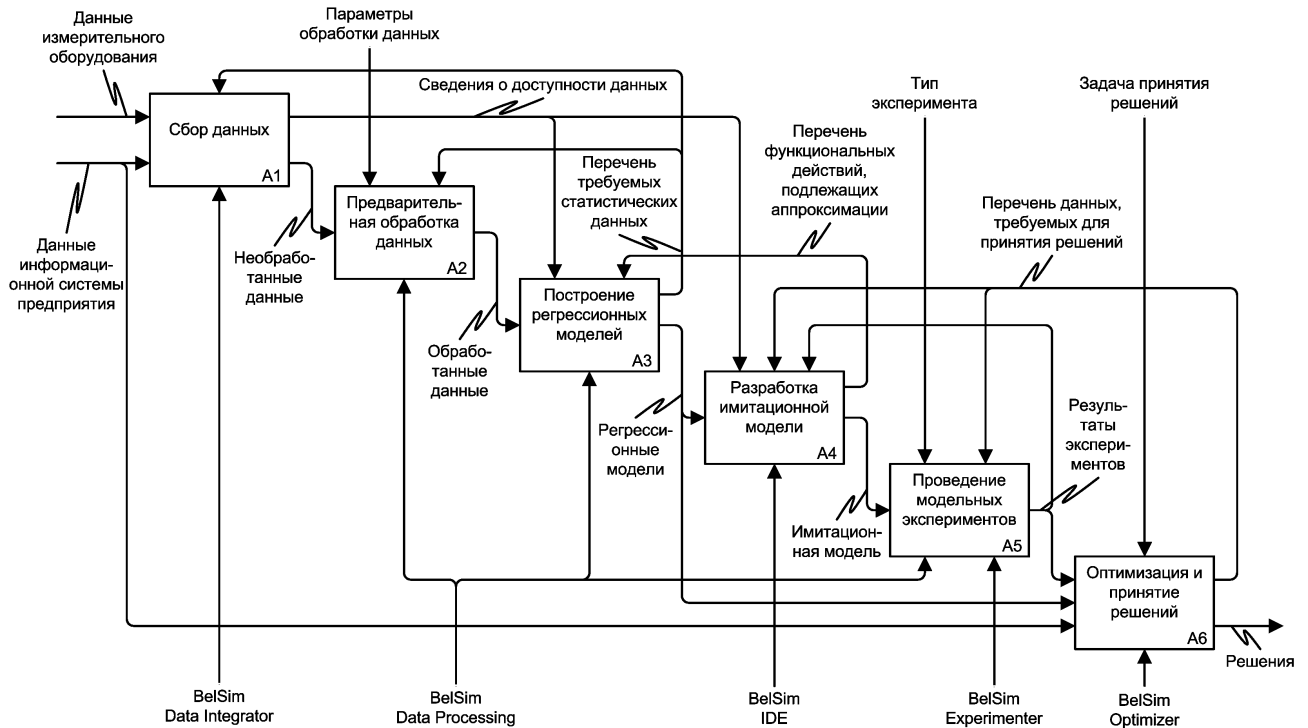
$$\sum_i ЦP_i \cdot \sum_k PP_{ik} \rightarrow \min.$$

В качестве ограничений при минимизации себестоимости продукции используется минимальная и максимальная загрузка цехов и общий выпуск продукции

$$\sum_i \sum_k PP_{ik}.$$

Представленная модель производственного процесса является сильно упрощенной и может применяться только при планировании производственных мощностей на некотором небольшом промежутке времени, поскольку не учитывает изменения цен на ресурсы. В реальных же условиях цены на ресурсы подвержены влиянию инфляции, а также сезонным колебаниям (это относится в первую очередь к ценам на сырье из-за его дефицита в определённые месяцы).

При построении и эксплуатации модели решают задачи обеспечения подготовки и ввода исходных данных; разработки автоматизированной технологии расчета требуемых показателей (рисунок 6.12). Для расчета, например, себестоимости выпускаемой продукции должны быть введены в модель сведения о ценах и нормативных показателях потребления основного сырья, расхода вспомогательных материалов и использования топливно-энергетических ресурсов.



**Рисунок 6.12 – Взаимодействие компонентов программно-технологического комплекса имитации сложных систем BelSim**

Для расчета себестоимости изготовленной продукции применяется формула

$$C = \frac{\sum_{i=1}^n c_{yди}(x_i) \cdot v_i}{V}, \quad (6.2)$$

где  $C$  – себестоимость ДМТ в целом по объединению,  $c_{yди}(x_i)$  – удельная себестоимость ДМТ, зависящая от производственной нагрузки  $x_i$   $i$ -го химического цеха,  $v_i$  – объем производства ДМТ  $i$ -м химическим цехом;  $V$  – объем производства ДМТ цехами объединения.

Функция удельной себестоимости представлена следующим выражением:

$$c_{yди}(x_i) = oc_i(x_i) + vc_i(x_i) + er_i(x_i) + pz_i(x_i), \quad (6.3)$$

где  $oc_i(x_i)$  – удельная себестоимость основного сырья;  $vc_i(x_i)$  – удельная себестоимость вспомогательных материалов;  $er_i(x_i)$  – удельная себестоимость энергоресурсов;  $pz_i(x_i)$  – удельные постоянные затраты, причем  $oc_i(x_i)$ ,  $vc_i(x_i)$ ,  $er_i(x_i)$ ,  $pz_i(x_i)$  для каждого  $i$ -го химического цеха зависят от их производственной нагрузки ( $x_i$ ).

В общем виде функции  $oc_i(x_i)$ ,  $vc_i(x_i)$ ,  $er_i(x_i)$ , которые можно обозначить  $f_r(x_i)$ , рассчитываются по формуле:

$$f_r(x_i) = \sum_{j=1}^n N_j(x_i) \cdot Ц_j, \quad (6.4)$$

где  $N_j$  – удельная норма  $j$ -го ресурса (основного сырья, вспомогательных материалов, энергоресурсов);  $Ц_j$  – цена или себестоимость  $j$ -го ресурса (основного сырья, вспомогательных материалов, энергоресурсов).

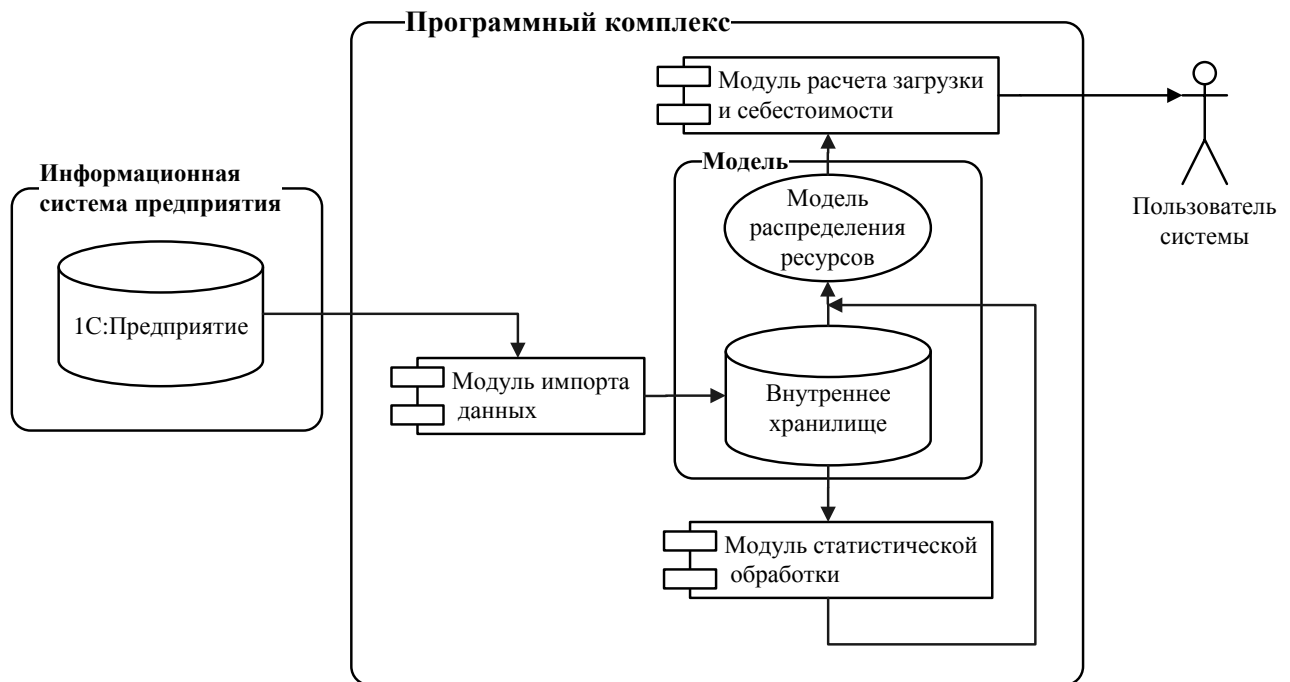
Удельные постоянные затраты рассчитываются по формуле:

$$pz_i(x_i) = \frac{ZP + OPr + NL}{x_i}, \quad (6.5)$$

где  $ZP$  – заработная плата основных и вспомогательных рабочих с учетом отчислений на социальное страхование;  $OPr$  – общепроизводственные расходы;  $NL$  – налоговые отчисления, определяемые действующими нормативными документами.

Данные в соответствии с формулами (6.2) – (6.5), требуемые для расчета себестоимости, представлены в информационной системе «1С: Предприятие». Поэтому в разрабатываемой модели должен быть реализован автоматизированный импорт необходимой информации. Данные о потреблении энергоресурсов хранятся в информационной системе на предприятии в виде данных, полученных по показаниям приборов учета. Эти данные обрабатываются в программе STATISTICA, полученные зависимости норм расхода энергоресурсов сохраняются и вводятся в модель. Далее вводят суточный план выпуска ДМТ, и программа выдаст расчетную загрузку цехов по производству ДМТ и себестоимость

ДМТ для каждого цеха и завода в целом. Полученные зависимости норм расхода энергоресурсов, планы выпуска, расчетные значения загрузки и себестоимости могут быть сохранены в корпоративной информационной системе на базе 1С: Предприятие 8.0, где также предусматривается чтение указанных данных при открытии модели (рисунок 6.13).



**Рисунок 6.13 – Схема технологии эксплуатации имитационной модели**

Программный комплекс состоит из следующих блоков [143]:

- блок импорта данных, состоящий из функций импорта данных по ценам, по нормам расхода материалов и энергоресурсов и сохранению данных в модели;
- блок статистической обработки данных, реализующий функции предварительной подготовки данных для анализа и непосредственного проведения регрессионного анализа;
- блок расчета загрузки цехов и себестоимости выпускаемой продукции, реализующий функции расчета загрузки цехов и себестоимости выпускаемой продукции;
- блок печати результата.

Пользователю доступны следующие настройки для каждого ресурса по каждому из цехов: фильтр на допустимые значения, анализируемый период, тип регрессионной модели, опция исключения ресурса из анализа, работа с выбросами, параметры вывода результата анализа [143].

### **6.2.6 Экспорт данных из системы «1С: Предприятие»**

В корпоративной информационной системе «1С: Предприятие» существует возможность создавать любые документы, справочники, регистры по своему усмотрению. Для разных предприятий существуют разнообразные информационные базы различной конфигурации. Основной задачей при моделировании производственного процесса является экспорт данных из справочников или документов в MS Excel.

Система «1С: Предприятие 8.0» может использоваться внешними приложениями в качестве OLE-сервера. Для запуска системы «1С: Предприятие» в качестве OLE-сервера из MS Excel выполняются следующие действия:

- создается OLE-объект с идентификатором “v8.Application”;
- выполняется инициализация системы «1С: Предприятие» методом Connect;
- вызываются свойства и методы системы «1С: Предприятие» как OLE-сервера.

Выбрать необходимые данные из справочника можно при помощи запроса. В системе «1С: Предприятие» использование запросов является основным способом получения сводных данных. Для выполнения запроса к данным OLE-базы достаточно с помощью метода NewObject создать объект Запрос и далее работать с ним стандартными методами.

В справочниках содержатся цены на основные и вспомогательные материалы. Для получения данных при помощи запроса из справочников, содержащих информацию о ценах на материалы и энергоресурсы, используется программа на языке Visual Basic.

После того, как данные при помощи запроса получены из системы «1С: Предприятие», они сохраняются на отдельном листе MS Excel в строго определенных ячейках. Этого можно добиться, задавая различные условия при запросе и используя сортировку. Например, при запросе данных о ценах на материалы, они сортируются по коду, и несмотря на то что запрос выбирает все поля, в ячейки можно вставлять значения любых атрибутов строки запроса, используя доступ к ним как к элементам объекта «Запрос». Далее на данные о ценах и нормах расхода материалов устанавливаются ссылки из модели. Таким образом, обеспечивается автоматизированный ввод исходных данных в модель.

Целью создания технологии моделирования и оптимизации системы управления производственным процессом является решение следующих задач:

*Задача 1.* Оптимальное распределение производственной нагрузки по цехам ДМТ ЗОСа, обеспечивающее минимизацию себестоимости выпускаемой продукции;

*Задача 2.* Минимизация длительности переходного процесса и сокращение потерь при переходе на другой ассортимент полиэтилентерефтолата (ПЭТФ);

*Задача 3.* Определение максимально допустимых закупочных цен на основное сырье.

Технология моделирования производства ДМТ включает следующие основные этапы (рисунок 6.14) [143].

*Этап 1. Импорт данных.*

*Шаг 1.1.* В модель записываются сведения о ценах на основное сырье, вспомогательные материалы, энергоресурсы и нормах их расхода из информационной системы «1С: Предприятие».

*Шаг 1.2.* Данные о расходе энергоресурсов в зависимости от выработки продукции сохраняются в информационной системе на базе MySQL в виде данных, полученных по показаниям приборов учета.





**Рисунок 6.14 – Общая схема технологии моделирования и оптимизации системы управления производственным процессом на ЗОС**

*Этап 2. Обработка статистических данных.*

*Шаг 2.1.* Информация о расходе энергоресурсов, хранящаяся в базе данных на основе MySQL, передается в табличный процессор MS Excel, а затем в пакет статистической обработки и анализа данных STATISTICA. Автоматизация данного шага обеспечивает постоянное обновление информации в модели, и, следовательно, возможность коррекции регрессионных зависимостей с учетом новых данных.

*Шаг 2.2.* Статистические данные обрабатываются средствами пакета STATISTICA и полученные зависимости норм расхода энергоресурсов от выра-

ботки импортируются в модель. Автоматизация этапа статистической обработки данных обеспечивает снижение трудоемкости данного шага и позволяет пользоваться системой лицам, не являющимся специалистами в области моделирования и статистической обработки данных.

*Этап 3. Решение оптимизационных задач.*

Для оптимизации используется программное обеспечение *SOLVER* табличного процессора *MS Excel*.

*Решение задачи 1 оптимального распределения производственной нагрузки по цехам ДМТ ЗОСа, обеспечивающее минимизацию себестоимости выпускаемой продукции*, реализуется следующей последовательностью шагов [143].

*Шаг 1.* Вводятся зависимости расхода энергоресурсов от выработки, полученные для каждого цеха ДМТ ЗОСа на основе обработки статистических данных. Для основного сырья, вспомогательных материалов и постоянных затрат данные импортируются из системы «1С: Предприятие».

*Шаг 2.* Вводится суточный план выпуска ДМТ, как ограничение при расчете оптимальной загрузки цехов. Ограничениями при оптимизации являются минимальная и максимальная суточная производительность каждого цеха.

*Шаг 3.* Вводится загрузка для каждого цеха, используемая в качестве начального приближения при оптимизации.

*Шаг 4.* Рассчитывается произведение расхода материалов и энергоресурсов на цену материала или энергоресурса. Рассчитывается себестоимость ДМТ для каждого цеха, как сумма затрат по статьям. В качестве целевой функции используется формула для расчета общей себестоимости по заводу.

*Шаг 5.* Минимизируется себестоимость продукции за счет изменения загрузки цехов, используя программу *SOLVER* табличного процессора *MS Excel* (рисунок 6.15).

*Решение задачи 2 минимизации длительности переходного процесса и сокращения потерь при переходе на другой ассортимент ПЭТФ.*

Рассчитать загрузку

Суточная норма выработки: 325

Расчет

Выход

Результирующие данные

	Выпуск, т/сут	Себестоимость, руб./т
ДМТ-2	100	2 234 234
ДМТ-3	50	1 530 000
ДМТ-4	175	2 444 680
Итого	325	2 069 638

Рисунок 6.15 – Расчет загрузки цехов и себестоимости продукции

Шаг 1. Вводятся зависимости расхода энергоресурсов от выработки на основе обработки статистических данных. Считается, что расход энергоресурсов не зависит от вида выпускаемого полимера. Для основного сырья, вспомогательных материалов и постоянных затрат данные импортируются из системы «1С: Предприятие» (рисунок 6.16).

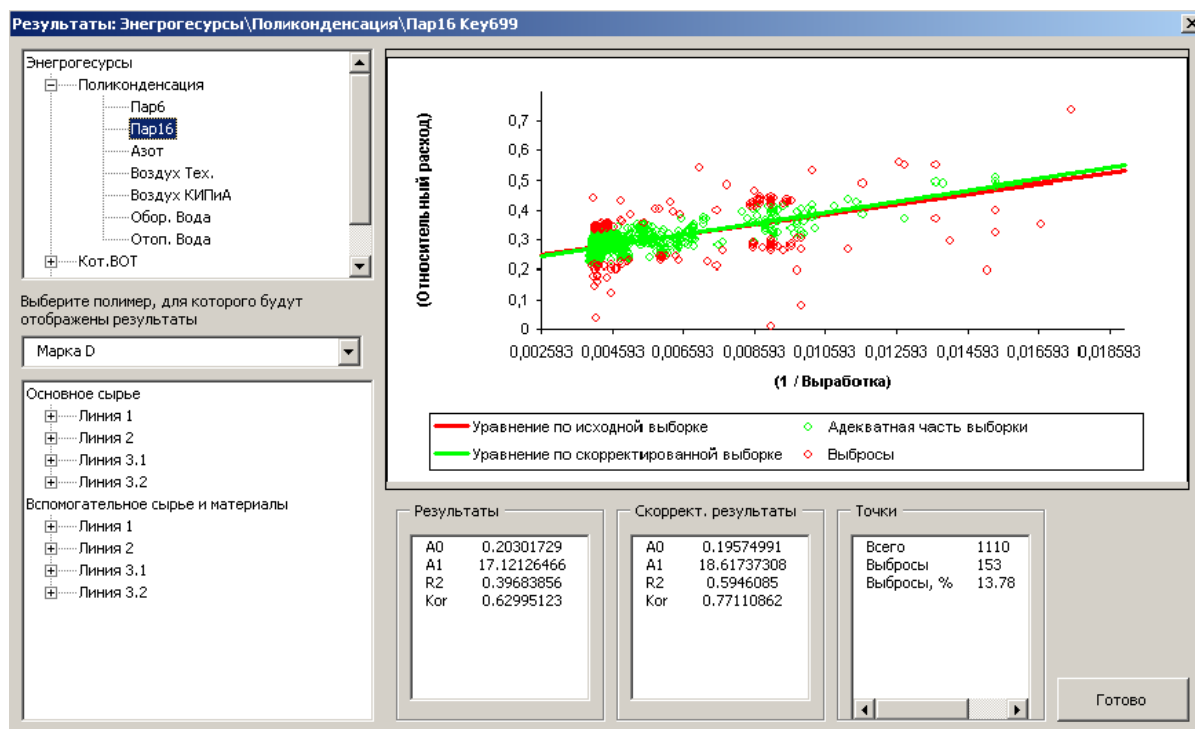


Рисунок 6.16 – Обработка статистических данных из 1С: Предприятие

*Шаг 2.* Вводится суточный план выпуска ПЭТФ, который используется в качестве ограничения при расчете оптимальной загрузки цехов.

*Шаг 3.* Задается интенсивность и продолжительность подачи дополнительного сырья.

*Шаг 4.* Рассчитывается себестоимость производства полимера при изменении ассортимента.

*Шаг 5.* Минимизируется время установления требуемой концентрации дополнительного сырья за счет изменения интенсивности его введения в технологический процесс [143].

*Решение задачи 3 определения максимально допустимых закупочных цен на основное сырье.*

При определении максимально допустимых закупочных цен себестоимость продукции для ДМТ и время перехода на другой ассортимент используются не в качестве целевых функций, а в качестве ограничений в процессе оптимизации. При этом возможны следующие варианты данной задачи.

*Вариант 1.* Необходимо установить максимальное значение закупочной цены для одного вида основного сырья. В этом случае задача сводится к максимизации значения закупочной цены для заданного вида основного сырья при однозначно заданном плане выпуска продукции (ДМТ и ПЭТФ), себестоимости продукции для ДМТ и времени перехода на другой ассортимент для ПЭТФ, не превышающем заданные значения. В качестве ограничения для каждого цеха ДМТ задается также минимальная и максимальная нагрузка.

*Вариант 2.* Необходимо установить максимальное значение закупочной цены  $C_i$  для  $N$  видов основного сырья ( $i=1, \dots, N$ ). В этом случае может быть использована процедура, аналогичная процедуре метода последовательных уступок: для выбранного одного вида сырья определяется максимальное значение закупочной цены  $\max C_1$  при фиксированных регламентированных значениях цен на другие виды основного сырья. Для найденного максимального значения назначается некоторая уступка  $\Delta C_1$ .

Задавая в качестве ограничения  $(\max \Pi_1 - \Delta \Pi_1) \leq \Pi_1 \leq \max \Pi_1$ , определяется максимальное значение для второго вида основного сырья  $\max \Pi_2$ . Далее назначается уступка для максимального значения цены второго вида основного сырья  $\Delta \Pi_2$  и т.д. [143].

### **6.3 Апробация средств оптимизации и распределенных вычислений для решения задачи о грузоперевозках на предприятии**

ПТКИ *BelSim* позволяет решать задачи моделирования как для процессов функционирования промышленного предприятия, так и для других предметных областей. Известной оптимизационной задачей является задача о планировании грузоперевозок на автотранспортном предприятии [143] с минимальными затратами. Постановка задачи: предприятие имеет  $N$  контрактов на доставку грузов. Каждый контракт ( $j=1, \dots, N$ ) определяет объем ( $O_j$ ) грузов, срок доставки ( $Sr_j$ ) и координату на карте местности. При этом известны:  $\|S_{fn}\|$  – расстояния от базы до остальных пунктов назначения;  $\|V_{fn}\|$  – средние скорости движения между базой и пунктами назначения ( $f, h = 0, \dots, N$ ).

Автопарк предприятия состоит из определенного количества грузовых машин ( $M$ ). Каждая  $i$ -ая машина ( $i=1, \dots, M$ ) имеет ограничения по грузоподъемности ( $T_i$ ), расходам на топливо ( $TO_i$ ) и повременной оплате труда водителя ( $Z_i$ ). Для расчета оплаты труда водителя учитывается время погрузки и разгрузки, определяемых нормативными показателями на скорость погрузки ( $Vp$ ) и разгрузки ( $Vr$ ). Контракт выполняется за одну поездку в установленные сроки. Если время доставки грузов увеличивается, превышая договорные обязательства на величину  $Tpr$ , то предприятие уплачивает пеню ( $Zpr$ ). Требуется построить план грузоперевозок ( $\|A_{ig}\|$  – матрица распределения контрактов по машинам,  $i=1, \dots, M$ ;  $g=1, \dots, 2*N+1$ ), при безусловном выполнении всех контрактов с минимальными общими затратами предприятия:

$$\sum_{i=1}^M \left( \sum_{j=1}^{2^*N} S_{A_{ij}A_{ij+1}} \cdot TO_i + \left( \sum_{j=1}^{2^*N} S_{A_{ij}A_{ij+1}} / V_{A_{ij}A_{ij+1}} + \sum_{j=1}^{2^*N+1} O_{A_{ij}} \cdot (Vp + Vr) \right) \cdot Z_i \right) + Zpr \cdot Tpr \rightarrow \min \quad (6.6)$$

Для решения задачи оптимизации грузоперевозок использована реализация генетического алгоритма (ГА) в подсистеме оптимизации ПТКИ BelSim.

Применение генетического алгоритма с параметрами по умолчанию с использованием матрицы  $\|A_{ig}\|$  для представления решения дает отрицательный результат. Требуется более эффективный способ кодирования решения задачи и рациональный выбор параметров генетического алгоритма. Проведены исследования следующего набора параметров генетического алгоритма:

$$GA = \langle \varepsilon_f, L_g, NI, NG, \delta_r(NI), p_m, p_k, M_{bp}, D_{bp}, O_t \rangle, \quad (6.7)$$

где  $\varepsilon_f$  – точность представления параметра функции;  $L_g$  – длина генотипа ( $L_g = M(\log_2 M + \log_2 N)$ );  $NI$  – размер популяции;  $NG$  – количество поколений;  $\delta_r(NI)$  – доля родителей от размера популяции;  $p_m$  – вероятность мутации;  $p_k$  – вероятность кроссовера;  $M_{bp}$  – математическое ожидание количества точек разрыва;  $D_{bp}$  – дисперсия количества точек разрыва;  $O_t = \langle \lambda_{om}, \{x_{om}\} \rangle$  отбор в следующее поколение,  $\lambda_{om}$  и  $\{x_{om}\}$  – тип и параметры оператора отбора родителей, соответственно.

В целях сужения пространства поиска решений, решение задачи представляется в виде  $N$  наборов вида  $\langle M_i, p_i \rangle$ , где  $i = 1, \dots, N$  – номер контракта,  $M_i \in Mz_i$  – номер машины, которая должна выполнять  $i$ -й контракт,  $Mz_i$  – множество машин с подходящей для  $i$ -го заказа грузоподъемностью,  $p_i$  – приоритет, определяющий порядок выполнения контрактов.

В результате пространство поиска решений значительно сокращается за счет исключения недопустимых вариантов, когда груз по одному контракту может доставляться более одного раза или не доставляться вовсе, а также возмож-

но превышение максимальной грузоподъемности машин. Для вычисления значения целевой функции выполняется преобразование указанного представления решения задачи в исходное. Для выбора лучших значений указанных выше параметров ГА в зависимости от длины генотипа, которая определяется размерностью задачи (количеством машин и контрактов) проведены исследования по указанной выше методике. В качестве функции качества использована целевая функция задачи, взятая со знаком минус [143].

Задача о грузоперевозках используется для исследования эффективности распределения вычислений. Экспериментальная база – локальная вычислительная сеть из трех компьютеров с основными характеристиками, представленными в таблице 6.4.

**Таблица 6.4 – Экспериментальная база для исследования распределенных вычислений в ПТКИ BelSim**

Компьютер	Процессор	ОЗУ, МБ	Сетевая карта	Операционная система
Compaq Evo N610c	Intel Pentium 4 / 2,0 ГГц	512	Ethernet / 100 Мбит/с; TCP/IP	Windows XP Professional
Compaq Evo N620c	Intel Pentium M / 1,4 ГГц			
HP Compaq nc 6000	Intel Pentium M / 1,6 ГГц			

Для оценки эффективности проведения эксперимента с распределенными вычислениями определена величина  $T_{ij}$  - длительность  $i$ -го эксперимента на  $j$ -м компьютере ( $i$  – порядковый номер эксперимента,  $i=1, \dots, 3$ ;  $j$  - идентификатор компьютера,  $j=PC1, \dots, PC3$ ) при заданном количестве прогонов модели в  $i$ -м эксперименте  $N_i$ . Результаты проведенного исследования представлены в таблице 6.5. Эффективность распределения вычислений оценена по выражению:

$$T_{i \max} = \max(T_i(PC_j)), \forall PC_j$$

где максимальное значение  $T_{i \max}$  продолжительности эксперимента определяется наибольшим значением  $T_i(PC_j)$  для наименее производительного компьютера.

**Таблица 6.5 – Результаты эксперимента с распределенными вычислениями в ПТКИ BelSim**

Количество прогонов модели, $N_i$	Продолжительность эксперимента, $T_{ij}$					
	на одном РС	на двух РС		на трех РС		
	T1(РС1), с	T2(РС1), с	T2(РС2), с	T3(РС1), с	T3(РС2), с	T3(РС3), с
60	21,99	12,22	12,81	8,17	8,24	8,91
100	35,73	19,11	19,90	13,26	13,34	14,38
200	71,69	35,94	37,28	26,18	26,10	28,07
400	146,64	73,14	76,29	54,30	54,44	58,37
600	227,89	108,81	114,38	79,46	79,61	86,12

Например, при сравнении экспериментов с  $N_i=400$  на одном и двух компьютерах продолжительность прогонов модели сокращается в 1,9, т.е. почти в два раза. На трех компьютерах продолжительность эксперимента в сравнении с одним компьютером сокращается в 2,5 раза. В соответствии с сетевым законом Амдала [100] при общей эффективности вычислений отсутствует прямо пропорциональное сокращение длительности имитационных экспериментов с расширением сети [127].

#### **6.4 Поиск оптимальных решений при заключении контрактов в маркетинговой деятельности промышленного предприятия**

##### **6.4.1 Исследование входных данных модели контракта продажи**

Задачей исследования является выявление свойств имитационных моделей контрактов продажи, купли, на дилерское обслуживание и по товарному обмену для обоснования области применения аналитических моделей оценки эффективности указанных контрактов. Эксперименты проведены с применением ПТКИ BelSim на основе базовой имитационной модели промышленного предприятия. В ходе проведения экспериментов все варианты файлов данных модели контракта продажи имеют одинаковые значения параметров, кроме одного, измене-



ние которого в ходе эксперимента осуществляется случайным образом по заданному закону распределения. Каждый вариант файла данных модели содержит только один из трех законов распределения случайных величин (равномерный, нормальный, экспоненциальный) [112, 143].

В качестве изменяемого в ходе эксперимента выбран параметр – "Реализация: Объем отгрузки:" со средним значением равным 500, стандартным отклонением равным 100 – для каждого варианта модели. Вид функции плотности распределения задан для первого варианта модели – 0 (равномерный); для второго варианта модели – 1 (нормальный); для третьего варианта модели – 2 (экспоненциальный).

В качестве откликов выбраны показатели деловой активности: Выручка от реализации (Y1), Чистая прибыль (Y2), а также Рентабельность продукции (Y3).

Тип эксперимента – полный факторный. При планировании эксперимента для каждого варианта задано по 100 прогонов модели.

Результаты проведенных исследований показывают, что закон изменения величин выходных откликов значительно зависит от закона изменения входных параметров [143]. Поэтому предлагается использование методики установления обобщенного типа выравнивающей кривой распределения и нахождения оценок параметров [64], чем достигается более высокая точность решения прикладных задач, в которых используется выравнивание статистических распределений.

#### **6.4.2 Пример моделирования контракта продажи**

Имитационная модель предприятия для оценки эффективности контракта используется дважды. Сначала следует имитация без контракта, затем – с контрактом продажи. Моделирование проведено при следующих основных условиях: НДС (отн.ед.) – 0,2; налог с выручки (отн.ед.) – 0,045; налог с заработной платы (отн.ед.) – 0,05; налоги с прибыли (отн.ед.) – 0,25; налоги с прибыли, остающиеся в распоряжении предприятия (отн.ед.) – 0,05; экологический налог (руб./ед.изд.) – 0,01; коэффициент износа основных средств (отн.ед.) – 0,5; коэффициент переоценки основных средств (отн.ед./год) – 1,35; норма амортиза-

ции основных средств (отн.ед./год) – 0,05; интервал планирования производства (дн.) – 5; вид функции распределения временных интервалов между изменениями цен – экспоненциальный; среднее значение временного интервала между изменениями цен (дн.) – 30; стандартное отклонение интервалов между изменениями цен (дн.) – 30; среднее значение коэффициента изменения цен (отн.ед./мес.) – 1,008; вид функции распределения коэффициента изменения цен – равномерный; стандартное отклонение коэффициента изменения цен – 0,01; сумма на расчетном счете (руб.) – 1 млн.

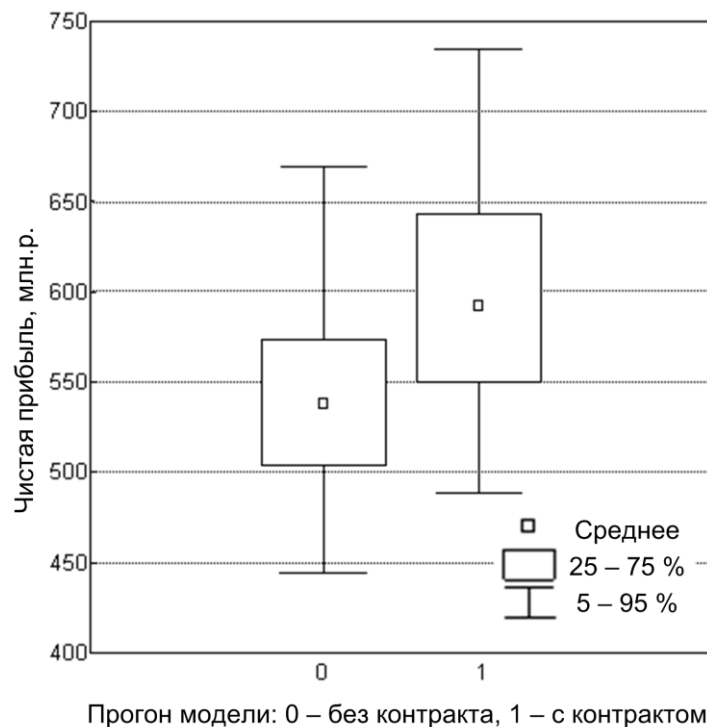
Отгрузка продукции осуществляется после заключения контракта продажи в течение 76 дней. Основные данные по отгрузкам, видам изделий, ценам на момент заключения контракта и рентабельности продукции представлены в таблице 6.6 по данным ОАО «Моготекс» (г. Могилев, Беларусь).

**Таблица 6.6 – Поставка продукции по контракту продажи**

Наименование изделия, (цена, руб./ед.изд.)	1 отгрузка (15 дн.): Объем, изд. (рентабельность, %)	2 отгрузка (34 дн.): Объем, изд. (рентабельность, %)	3 отгрузка (41 дн.): Объем, изд. (рентабельность, %)	4 отгрузка (68 дн.): Объем, изд. (рентабельность, %)	5 отгрузка (76 дн.): Объем, изд. (рентабельность, %)
1С119 (1420)		12866 (0)	5922 (0)	17548 (0)	
1С119К (2260)	750 (5,0)				
1С129К (2480)					3840 (16,0)
1С130К (760)					9360 (16,0)
2С2КВн (618)	40000 (5,0)	8753 (5,0)	13390 (5,0)	33670 (5,0)	98 (5,0)
4С31 (1240)			1249 (25,0)		
4С31КВ (1370)			2696 (18,0)		
4С3КВ (910)	815 (4,0)				
4С3КВг (2080)				3890 (15,0)	
5С34КВ (1130)	35000 (22,0)		7043 (22,0)	87 (22,0)	
5С35КВ (870)			4075 (4,0)	3436 (4,0)	22511 (4,0)
5С36КВ (1218)			570 (5,0)	53231 (16,0)	
8С55 (2480)				1129 (23,0)	
8С58КВн (812)	40000 (8,3)		4082 (8,3)		
8С111 (1220)			1446 (7,0)		
8С111К (1150)					1950 (5,0)
8С41К (2300)		14720 (13,0)	9007 (13,0)		489 (13,0)
8С41КВ (1870)			2462 (10,0)	4586 (10,0)	
8С4КВг (2390)				321 (22,0)	
9С88КВ (2050)		4940 (15,0)			
9С89КВ (1820)		4487 (6,0)			2192 (6,0)

Результаты имитации с контрактом и без контракта представлены на рисунке 6.17.

Таким образом, в соответствии с рисунком 6.17 исследователь может сделать вывод о целесообразности заключения исследуемого контракта при заданных исходных данных по критерию чистой прибыли предприятия.



**Рисунок 6.17 – Изменение чистой прибыли при заключении контракта продажи**

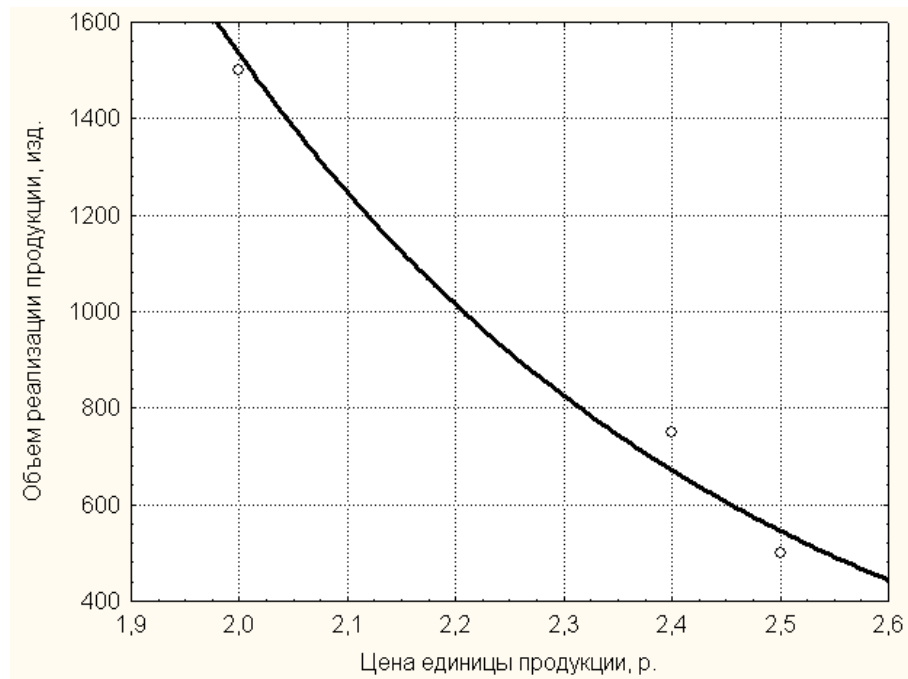
Аналогичные результаты получены и при использовании аналитической модели контракта продажи в программном комплексе *Contract Analyzer* [9, 107, 114], что подтверждает адекватность исследуемых моделей [105, 143]. Разработаны математические модели для прогнозирования эффективности наиболее часто используемых контрактов [138].

#### **6.4.3 Оценка влияния параметров контрактов на финансово-экономическое состояние предприятия**

Оценка финансово-экономического состояния предприятия проводится по трем основным показателям: K1 – коэффициент текущей ликвидности (Y74); K2

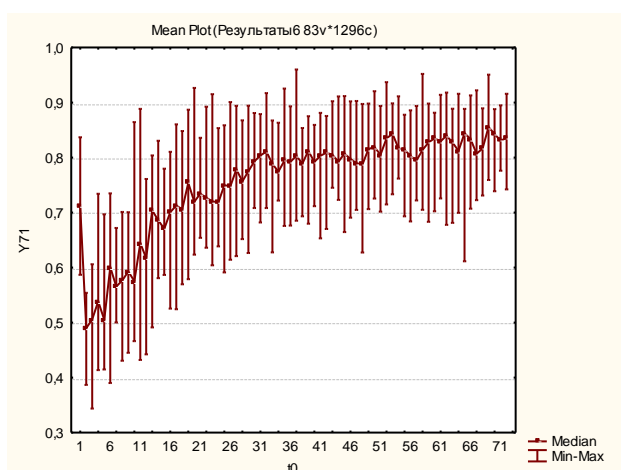
– коэффициент обеспеченности собственными оборотными средствами (Y71);  
 КЗ – коэффициент обеспеченности финансовых обязательств активами (Y73).  
 Дополнительно принят к анализу показатель рентабельности продукции (Y76).

Для исследований принята модель спроса [87], представленная на рисунке 6.18.

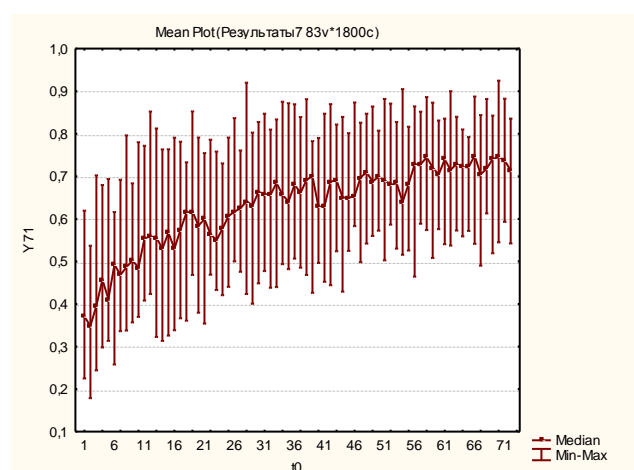


**Рисунок 6.18 – Модель спроса на выпускаемую продукцию**

По этим же показателям оценивается и влияние сроков платежа на финансово-экономическое состояние предприятия (рисунок 6.19, рисунок 6.20).



**Рисунок 6.19 – К2: срок платежа по контракту с предоплатой 30 дней**



**Рисунок 6.20 – К2: срок платежа по контракту с отсрочкой 10 дней**

Результаты экспериментов представлены для показателя  $K_2$  – доля собственных оборотных средств в общей их сумме ( $Y_{71}$ ). Эксперименты проведены при следующих значениях сроков платежа за отгруженную продукцию: предоплата 30 дней (см. рисунок 6.18); отсрочка платежа 10 дней (см. рисунок 6.19) [8, 111, 124]. Анализ результатов показывает ухудшение показателя  $K_2$  при отсрочке платежа на 10 дней по сравнению с предоплатой 30 дней.

## **6.5 Выбор типа координации процесса планирования производственной программы**

### **6.5.1 Целевая функция планирования производственной программы**

Составление производственной программы является итогом согласования следующих целей функционирования промышленного предприятия: получение максимальной прибыли; возможно полное удовлетворение потребностей рынка сбыта; максимальное снижение производственных издержек, в том числе и максимально полная загрузка технологического оборудования с учетом реальных финансовых и иных ресурсных возможностей предприятия.

Для их достижения используется целевая функция, которую можно представить в следующем виде:

$$\Phi = F(P, M, C, X, S), \quad (6.11)$$

где  $P \rightarrow \max$  – получение максимальной прибыли;

$M \rightarrow \max$  – максимальное использование производственных мощностей;

$C \rightarrow \min$  – минимальная себестоимость выпускаемой продукции;

$X \rightarrow \min$  – минимальные затраты на хранение выпускаемой продукции;

$S \rightarrow \min$  – минимальные затраты на уплату штрафов за не вовремя отгруженную продукцию.

Процесс разработки информационной системы для поддержки принятия решений при планировании производственной нагрузки включает в себя следующие основные блоки:

- создание базы данных для анализа рыночной конъюнктуры сырья и производимой продукции, построения имитационных моделей рынка сырья и сбыта продукции;
- создание базы данных о финансовом состоянии предприятия, производственных мощностей и уровней запасов сырья на складе;
- разработка интерфейса интеграции имитационных моделей с базой данных корпоративной информационной системы;
- разработка модуля для оценки вариантов и принятия решений при планировании производственной нагрузки на основе имитационного моделирования.

Укрупненная схема алгоритма решаемой задачи включает реализацию следующих основных функций: внесение данных; редактирование данных; выборка данных для моделирования; моделирование производственной нагрузки; анализ полученных результатов и выбор оптимального варианта планирования.

### **6.5.2 Модель планирования производственной нагрузки с жесткой координацией**

Модель с жесткой координацией основана на предположении, что производство продукции в определенном временном интервале (производственная нагрузка) зависит от объема заказов, поступающих из системы управления на уровне объединения. В качестве ограничения следует учитывать, что производственная нагрузка не может быть больше граничного значения  $ВПmax_i$  и не может быть ниже минимального значения  $ВПmin_i$ , обусловленных особенностями технологического процесса производства. При этом продукция производится непрерывно, загружается в тару и перемещается на склад.

При необходимости изменения плана выпуска продукции в соответствии с требованиями рынка на предприятии используется традиционная технология распределения производственной нагрузки между цехами. Поскольку проектная мощность цехов различная, то обеспечивается номинальная работа наиболее

мощного из них, а остальная часть выпуска продукции распределена в других цехах. В случае производственной необходимости в строй вводится резервный цех, который периодически останавливается, если снижается план реализации продукции. При этом ввод резервного цеха сопряжен с дополнительными значительными затратами. В условиях, когда требуется использование производственных мощностей всех цехов, такая технология не определяет их оптимальную загрузку и не обеспечивает минимальную себестоимость продукции.

Для алгоритма с жесткой координацией определены следующие параметры:  $Z_{Cn}$  – не заказанный объем складского запаса продукции;  $OZ_{kj}$  – объем заказа по  $j$ -му контракту;  $\sum_{i=2}^4 ПН_{i\max}$  – максимальная суточная производственная нагрузка по выпуску продукции;  $T \cdot \sum_{i=2}^4 ПН_i = OZ_{kj} - Z_{Cn}$  – требуемый выпуск продукции за период  $T$  с учетом складского запаса  $Z_{Cn}$  для выполнения  $j$ -го заказа объемом  $OZ_{kj}$ ;  $OZ_{ост}$  – текущее значение объема продукции, оставшееся для выполнения заказа;  $T_z$  – срок выполнения заказа по контракту;  $ВП_j$  – суточная производственная нагрузка по выпуску продукции. Алгоритм планирования производственной нагрузки с жесткой координацией с учетом введенных выше обозначений представлен на рисунке 6.21. Имитационная модель процесса планирования производства по указанному алгоритму реализована в среде ПТКИ BelSim.

### 6.5.3 Модель планирования производственной нагрузки с интегральной координацией

Модель с интегральной координацией основана на постоянной производственной нагрузке, когда за определенный длительный период времени требуется выпустить заданное количество продукции. Если спрос на продукцию отсутствует, она поступает на склад. При отсутствии сырья и материалов требуется брать кредиты для их закупки, т. к. финансовые средства могут отсутствовать. При этом следует определить максимально допустимые цены на закупку сырья и материалов с учетом складских затрат на хранение товарной продукции.

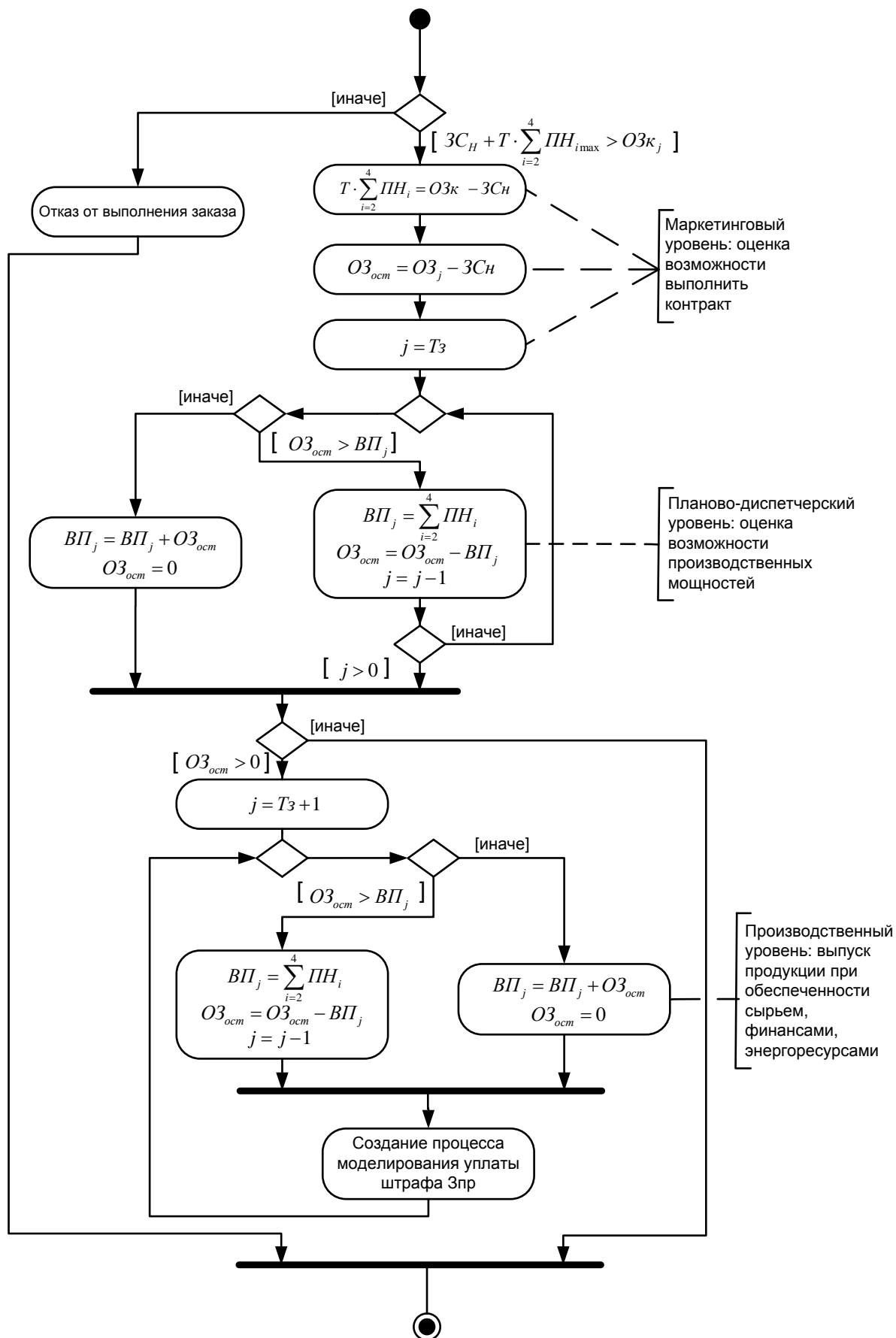


Рисунок 6.21 – Алгоритм процесса планирования с жесткой координацией



В алгоритме интегральной координации [143] на рисунке 6.22 выпуск продукции осуществляется с постоянной производственной нагрузкой. Такой вариант планирования обусловлен тем, что пуск цехов и остановов производства сопряжены со значительными затратами, которые увеличивают себестоимость выпускаемой продукции.

При отсутствии заказов предусмотрена поставка продукции на склад. Алгоритм (см. рисунок 6.22) построен при условии достаточного количества сырья для выпуска продукции.

Процесс планирования реализован в базовой имитационной модели ПТКИ BelSim. Внутренний модуль модели этого процесса задан следующими структурами данных.

Структура «*Order*»:

- *Date (D)* – дата получения заказа на поставку продукции;
- *NProduct (NP)* – тип изготавливаемой продукции;
- *Volume (V)* – объем продукции в соответствии с заказом;
- *DateShipment (DS)* – дата поставки изготовленной продукции;
- *MaxTimeExpect (MTE)* – максимальный допустимый временной интервал ожидания выполнения заказа после нарушения условий поставки по контракту;
- *ProdTypeDistribution (PTD)* – вероятностные параметры закона распределения типа продукции;
- *VolumeDistribution (VD)* – вероятностный параметр закона распределения объемов поступающих заказов;
- *ShipmentIntervalDistribution (SID)* – вероятностные параметры закона распределения даты выполнения заказа;
- *MaxWaitTDistribution (MWTD)* – вероятностные параметры закона распределения длительности ожидания выполнения заказа.

Структура «*Enterprise*»:

- *NumOrder (NO)* – число заказов в очереди на изготовление продукции;
- *ReleaseMaxCondProd (RMCP)* – максимально возможное производство условного типа продукции в течение рабочего дня;

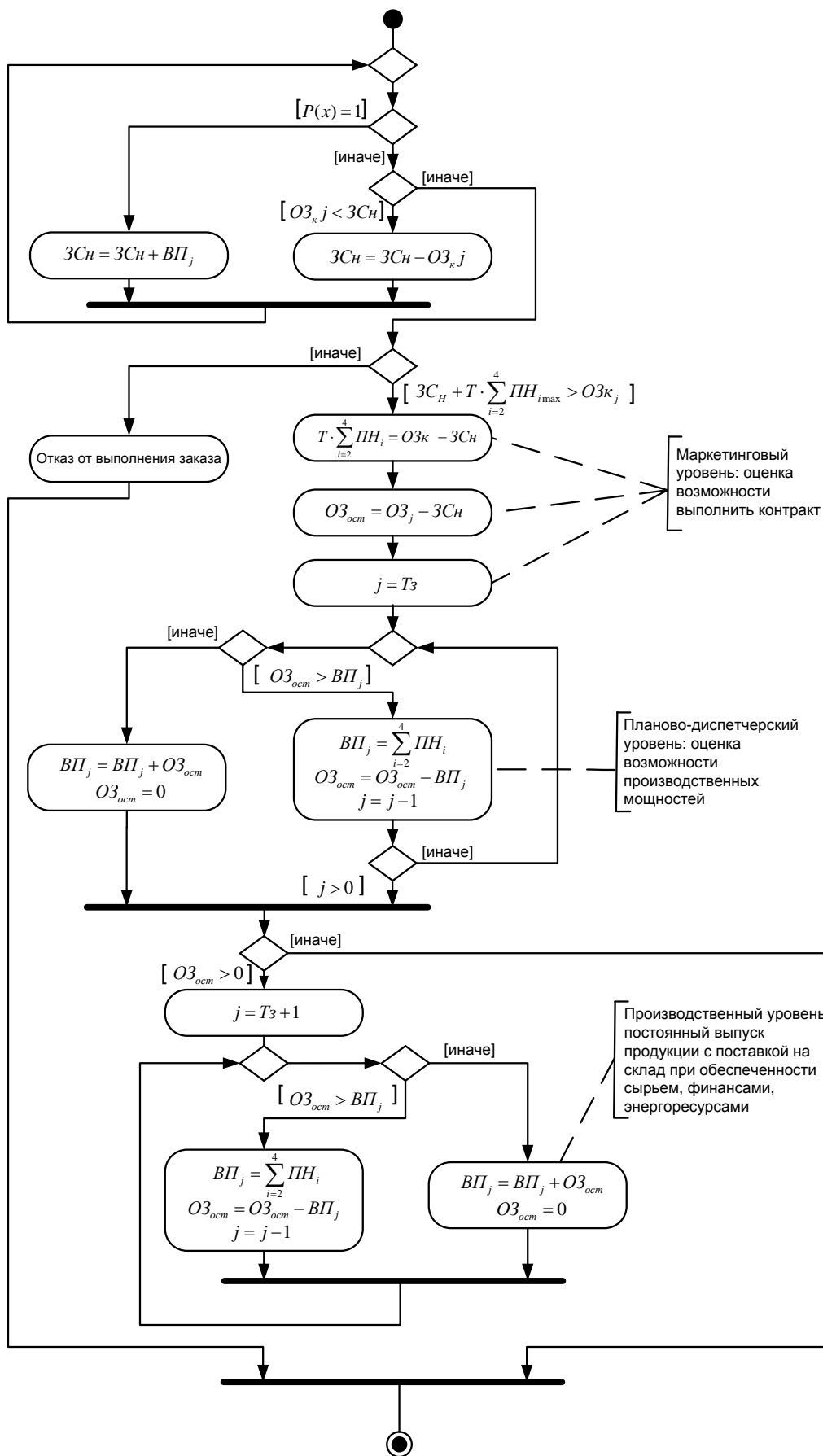


Рисунок 6.22 – Алгоритм процесса планирования с интегральной координацией

- *ReleaseMaxOverTimeCP (RMOTCP)* – наибольший объем производства условной продукции в течение одного рабочего дня;
- *AddExpenseCondProd (AECP)* – затраты на производство единицы условного типа продукции, изготовленной с превышением нормативных показателей;
- *ReducIndusPower (RIP)* – показатель снижения производительности оборудования из-за переналадок при смене типа выпускаемой продукции;
- *LoadingIndusPower (LIP)* – план-график загрузки производственного оборудования в  $i$ -ый временной интервал;
- *VariantPlan (VP)* – разработанный вариант плана производства продукции (производственной программы).

Структура «*Product*»:

- *NumProd (NP)* – тип производимой продукции;
- *MaxRelease (MR)* – максимальный объем выпуска  $i$ -го типа продукции в течение рабочего дня;
- $K_{of}$  – коэффициент для оценки максимального выпуска продукции относительно условного типа продукции ( $K_{of} = \text{MaxRelease} / \text{ReleaseMaxCondProd}$ );
- *CostsStorage (CS)* – складские затраты на единицу  $i$ -го типа продукции в течение одного рабочего дня;
- *Fine (F)* – размер штрафа за каждый день нарушения условий контракта по  $i$ -му типу продукции;
- *Stock (S)* – запасы готовой продукции  $i$ -го типа на складе;
- *MPS* – основной разработанный план-график производства готовой продукции;

*OrderGenerating* – процесс, в котором генерируются заказы и их параметров: тип продукции, объем заказа, дата поставки произведенной продукции, максимальный временной интервал ожидания после нарушения сроков поставки продукции в соответствии с заказом. Затем указанные данные перемещаются в структуру «*Order*», и применяются для реализации процессов планирования.

*HardCoord* – процесс, в котором происходит планирование производства с жесткой координацией по алгоритму, представленному на рисунке 6.21.

*IntegralCoord* – процесс, в котором происходит планирование производства с интегральной координацией по алгоритму [143], представленному на рисунке 6.22.

Структура *TEnterprise* – основная компонента в имитационной модели процессов функционирования деятельности предприятия. В системе имитационного моделирования *BelSim Simulator Core* процесс *CShipmentGeneraitin* запускается после текущего приращения модельного времени. Выполняется проверка времени поступления нового заказа. Если результат проверки – true, запускается процесс *PlanningOrderGenerating*, считывающий данные из структуры *Order*. Затем управление передается процессу *CMasterProductionScheduling*, передающему необходимые данные процессу *PlanningMPS1* или *PlanningMPS2*.

Вариант процесса планирования *PlanningMPS1* реализован по алгоритму, представленному на рисунке 6.21, процесса *PlanningMPS2* – по алгоритму, представленному на рисунке 6.22. При запуске каждого из процессов считываются входные данные из структур *Order*, *Interprise*, *Product*. Процессы формируют соответствующий план производства готовой продукции, который записывается в *TEnterprise*. Диаграмма соответствующих классов имитационной модели представлена на рисунке 6.23.

В рассматриваемом случае имеется несколько вариантов для выбора решений. Выбор при наличии нескольких вариантов и многих критериев представлен следующими примерами.

*Пример 1.* Пусть задано  $m$  – количество критериев,  $m = 2$ , множество оценок  $Y = \{y^1, y^2, y^3\}$  для соответствующих вариантов решений  $X = \{x_1, x_2, x_3\}$ , причем векторы оценок принимают следующие значения:  $y^1 = \{1,4; 1,1\}$ ;  $y^2 = \{1,3; 1,2\}$ ;  $y^3 = \{1,1; 1,3\}$ . В качестве критериев приняты следующие показатели промышленного предприятия (см. таблицу 2.1):  $K_1$  – коэффициент текущей ликвидности, отн. ед.,  $K_2$  – коэффициент обеспеченности собственными оборотными средствами, отн. ед. Представленные вектора оценок являются Парето-оптимальными. Это означает, что нет возможности сократить множество оценок  $Y$  и соответственно множество решений  $X$ .

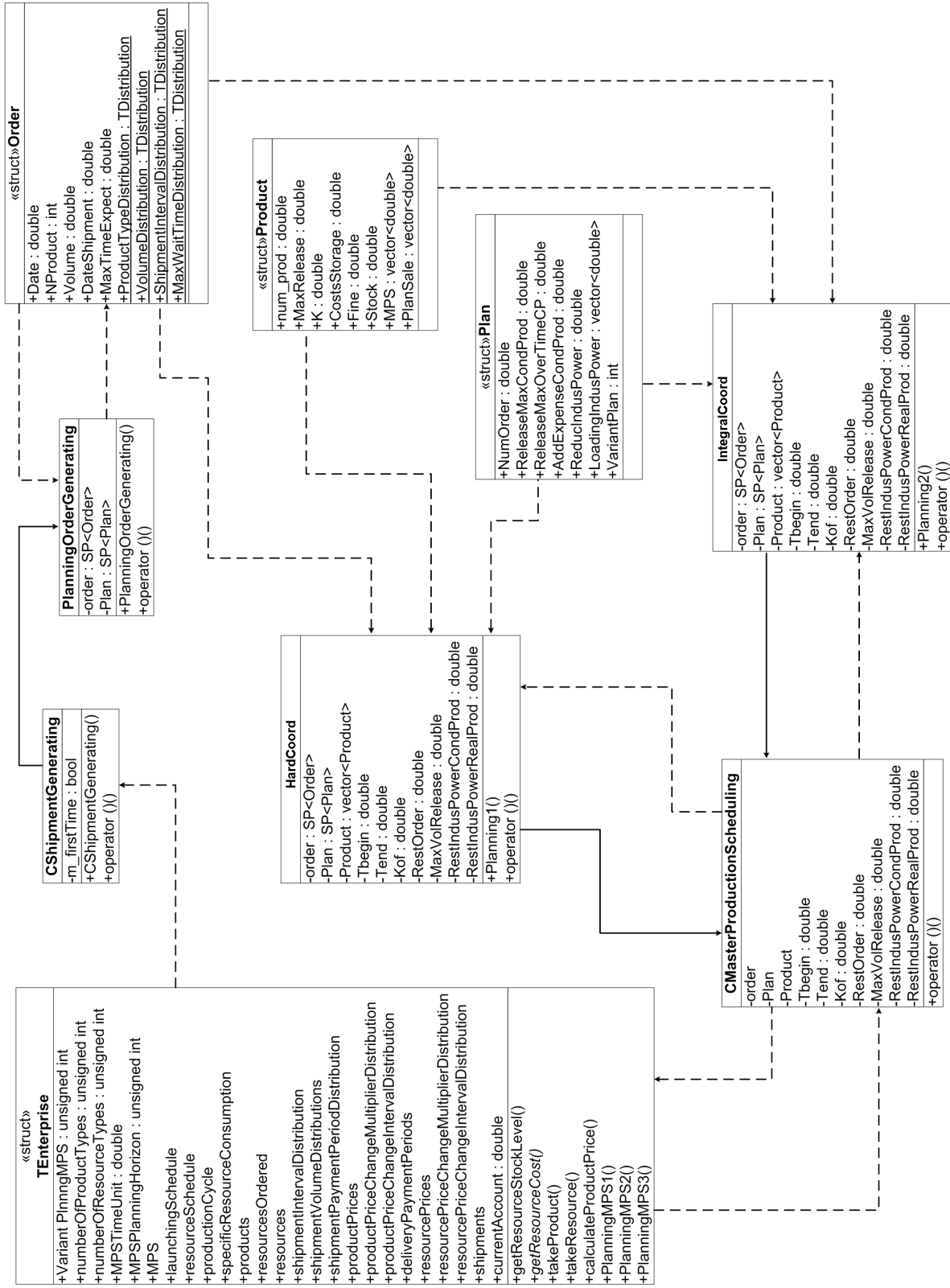


Рисунок 6.23 – Диаграмма классов имитационной модели планирования производственной программы

Пусть, с учетом предпочтений ЛПР, известно, что критерий  $K_1$  важнее критерия  $K_2$  с коэффициентом относительной важности  $\theta_{12} = 0,5$ . Тогда в соответствии с формулой (2.52) критерий  $K_2$  принимает вид:

$$\hat{y}_2 = 0,5y_1 + 0,5y_2 \quad (6.12)$$

и, следовательно,

$$\hat{Y} = \{(1,4; 1,25), (1,3; 1,25), (1,1; 1,20)\}. \quad (6.13)$$

В результате второй и третий вектора оказываются доминируемыми и в множестве  $\hat{Y}$  Парето-оптимальным является вектор (1,4; 1,25). Следовательно, этот вектор может быть выбранным.

Методика принятия решений при наличии нескольких вариантов и многих критериев реализована в программном модуле Decision making (рисунок 6.24).

№	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10
1	25.2808	0.96044	10	10	10	10	10	10	10	10
2	17.749	0.392064	10	10	10	10	10	10	10	10
3	8.46693	0.284553	0.2627	0.4046	0.1862	6.8609	0.595568	0.143055	0.905964	
4	4.72804	0.647447	0.5278	0.6988	0.2195	11.9621	0.663511	0.158732	0.949048	
5	5.48155	0.694073	1.2064	0.9059	0.5955	9.31169	0.783982	0.160342	0.962402	
6	5.14365	0.81196	1.9645	0.9490	0.6635	8.57332	0.799519	0.158951	0.970312	
7	3.18101	0.822065	4.0899	0.9624	0.7839	8.54115	0.775017	0.164198	0.977411	
8	2.70711	0.791436	3.7422	0.9703	0.7995	8.65023	0.869209	0.169976	0.982348	
9	1.34222	0.884234	7.3112	0.9774	0.7750	8.4091	0.931345	0.168275	0.984635	
10	1.40425	0.945713	14.779	0.9823	0.8692	8.15599	0.950102	0.168532	0.987487	

ИТОГ: 13520

Номер записи: 1 X Удалить

Состояние: Просмотр базы показателей производственно-финансовой деятельности промышленного предприятия Выход

Рисунок 6.24 – Показатели деятельности предприятия

*Пример 2.* Пусть задано  $m$  – количество критериев,  $m = 3$ , множество оценок  $Y = \{y^1, y^2, y^3, y^4\}$ , где  $y^1 = (1,1; 1,45; 1,2)$ ,  $y^2 = (1,2; 1,3; 1,1)$ ,  $y^3 = (1,3; 1,2; 1,1)$ ,  $y^4 = (1,5; 1,15; 1,2)$  для соответствующих вариантов решений  $X = \{x_1, x_2, x_3, x_4\}$ . В качестве критериев приняты следующие показатели промышленного предприятия (см. таблицу 2.1):  $K_1$  – коэффициент текущей ликвидности, отн. ед.,  $K_2$  – коэффициент обеспеченности собственными оборотными средствами, отн. ед.,  $K_{\text{АЛ}}$  – коэффициент абсолютной ликвидности, отн. ед.

Пусть по предпочтению ЛПР и/или в результате предыдущих действий имеется  $k$  различных сообщений об относительной важности критериев,  $k = 2$ , т. е. известно, что в результате анализа векторов  $u^1 = (1,0; 1,5; 1,1)$ ,  $v^1 = (1,2; 1,2; 1,0)$ ,  $u^2 = (1,5; 1,0; 1,2)$ ,  $v^2 = (1,1; 1,1; 1,1)$  было принято: 1)  $u^1 \succ v^1$ ; 2)  $u^2 \succ v^2$ . При этом  $u^1 - v^1 = (-0,2; 0,3; 0,1)$ ,  $u^2 - v^2 = (0,4; -0,1; 0,1)$ .

Для проверки совместности векторов  $u^1, v^1, u^2, v^2$  в соответствии с алгебраическим критерием непротиворечивости [69] составляется однородная система линейных уравнений:

$$\lambda_1 - 0,2\mu_1 + 0,4\mu_2 = 0, \quad (6.14)$$

$$\lambda_2 + 0,3\mu_1 - 0,1\mu_2 = 0, \quad (6.15)$$

$$\lambda_3 + 0,1\mu_1 + 0,1\mu_2 = 0. \quad (6.16)$$

Числа  $\lambda_3, \mu_1, \mu_2$  являются неотрицательными, поэтому из уравнения (6.16) следует:

$$\lambda_3 = 0, \mu_1 = 0, \mu_2 = 0. \quad (6.17)$$

Тогда из (6.14) и (6.15) имеем:

$$\lambda_1 = \lambda_2 = 0. \quad (6.18)$$

Следовательно, в системе линейных уравнений (6.14), (6.15), (6.16) отсутствуют ненулевые неотрицательные решения, что означает совместность двух пар векторов  $u^1, v^1, u^2, v^2$ .

Для построения множества недоминируемых векторов  $\text{Ndom } Y$  составляется система линейных уравнений [69] для векторов  $y' = y^1$  и  $y'' = y^2$ :

$$\begin{aligned}\lambda_1 - 0,2\mu_1 + 0,4\mu_2 &= -0,1, \\ \lambda_2 + 0,3\mu_1 - 0,1\mu_2 &= 0,15, \\ \lambda_3 + 0,1\mu_1 + 0,1\mu_2 &= 0,1.\end{aligned}\tag{6.19}$$

Система уравнений (6.19) имеет ненулевое неотрицательное решение  $\lambda_1 = \lambda_2 = \mu_2 = 0$ ,  $\lambda_3 = 0,05$ ;  $\mu_1 = 0,5$ . Следовательно, выполняется отношение предпочтения  $y^1 \succ_M y^2$ , а значит вектор  $y^2 \notin \text{Ndom } Y$ .

Аналогично система линейных уравнений (6.19) для векторов  $y' = y^4$  и  $y'' = y^3$ :

$$\begin{aligned}\lambda_1 - 0,2\mu_1 + 0,4\mu_2 &= 0,2, \\ \lambda_2 + 0,3\mu_1 - 0,1\mu_2 &= -0,05, \\ \lambda_3 + 0,1\mu_1 + 0,1\mu_2 &= 0,1.\end{aligned}\tag{6.20}$$

Система (6.20) имеет ненулевые неотрицательные решения  $\lambda_1 = \lambda_2 = \mu_1 = 0$ ,  $\lambda_3 = 0,05$ ;  $\mu_2 = 0,5$ . Поэтому вектор  $y^3 \notin \text{Ndom } Y$ .

Пара систем линейных уравнений (6.19) для векторов  $y' = y^1$ ,  $y'' = y^4$  и  $y' = y^4$ ,  $y'' = y^1$ :

$$\begin{aligned}\lambda_1 - 0,2\mu_1 + 0,4\mu_2 &= -0,4, & \lambda_1 - 0,2\mu_1 + 0,4\mu_2 &= 0,4, \\ \lambda_2 + 0,3\mu_1 - 0,1\mu_2 &= 0,3, & \lambda_2 + 0,3\mu_1 - 0,1\mu_2 &= -0,3, \\ \lambda_3 + 0,1\mu_1 + 0,1\mu_2 &= 0; & \lambda_3 + 0,1\mu_1 + 0,1\mu_2 &= 0.\end{aligned}\tag{6.21}$$

Решения для систем (6.21):

$$\begin{aligned}\lambda_1 = -0,26; \lambda_2 = 0,13; \lambda_3 = -0,05; & \lambda_1 = 0,45; \lambda_2 = -0,4; \lambda_3 = 0,004; \\ \mu_1 = 0,55; \mu_2 = -0,08; & \mu_1 = -0,36; \mu_2 = 0,05.\end{aligned}$$



Так как отсутствуют ненулевые неотрицательные решения, то ни одно из отношений предпочтения  $y^1 \succ_M y^4$  и  $y^4 \succ_M y^1$  не выполняется. В результате множество недоминируемых векторов представлено следующими элементами:

$$\text{Ndom}_M Y = \{y^1, y^4\}.$$

Множество  $\text{Ndom}_M Y = \{y^1, y^4\}$  является оценкой сверху для множества  $\text{Se-}IY$ , т. е.  $\text{Se}IY \subset \{y^1, y^4\}$ . Так как  $y^2, y^3 \notin \text{Ndom}_M Y$  и следовательно  $y^2, y^3 \notin \text{Se}IY$ , то  $y^2, y^3$  не могут быть выбранными.

Разработанные программные средства позволяют моделировать стратегии управления на основе базовой имитационной модели с формированием основных показателей деятельности предприятия и принимать решение на основе сформированной целевой функции.

#### **6.5.4 Реализация планирования производственной программы и управления запасами ресурсов на промышленном предприятии**

Задача планирования выпуска продукции в производственном цехе состоит в разработке графика ее изготовления, где для продукции  $A_i$  определяется объем  $V_i$  и дата  $\text{ProdDate}_i$  выпуска готовой продукции. Для начала выпуска продукции  $A_i$  должны быть в наличии все комплектующие и материалы  $PC_i$  в соответствии с объемом  $V_i$ . График производства продукции должен быть составлен так, чтобы максимально загрузить производственные мощности  $\sum_i \text{ProdCap}_i \rightarrow \max$  и выполнить к заданному сроку  $\text{SaleDate}_i$  наибольшее количество заказов ( $\text{ProdDate}_i \leq \text{SaleDate}_i$ ) при минимальных затратах в производстве продукции (например, благодаря сокращению числа переналадок оборудования при смене типа продукции).

В том случае, когда в одном производственном цехе размещено несколько технологических линий, выпускающих однородные изделия, либо в структуре предприятия имеются цеха с выпуском одинаковой продукции, но с разными затратами или ограничениями на производство единицы продукции, актуальной

становится задача планирования производственной программы для минимизации себестоимости продукции.

Планирование цеховой производственной программы при известной максимальной производственной мощности  $\max Cap$  каждой производственной линии цеха выполняется в соответствии с алгоритмом (6.22):

$$\forall A_i \left\{ \begin{array}{l} PC_i := true \\ \text{иначе} \end{array} \right. \left\{ \begin{array}{l} \sum_i V_i < \max Cap \\ \text{иначе} \end{array} \right. \left\{ \begin{array}{l} \sum_i V_i := \sum_i V_i + ProdCap_i; \\ Date_i := prodDate_i; \\ \sum_i V_i = 0; \\ prodDate_i := prodDate_i + 1; \end{array} \right. \quad (6.22)$$

$$\text{иначе} \{ Date_i := Date_i + waitTime. \}$$

Базовая имитационная модель, разработанная для производственных предприятий, является основой программного комплекса (ПК) ShagoVitaPro [139, 154], который предназначен для поддержки принятия решений в производстве детской обуви (рисунок 6.25).

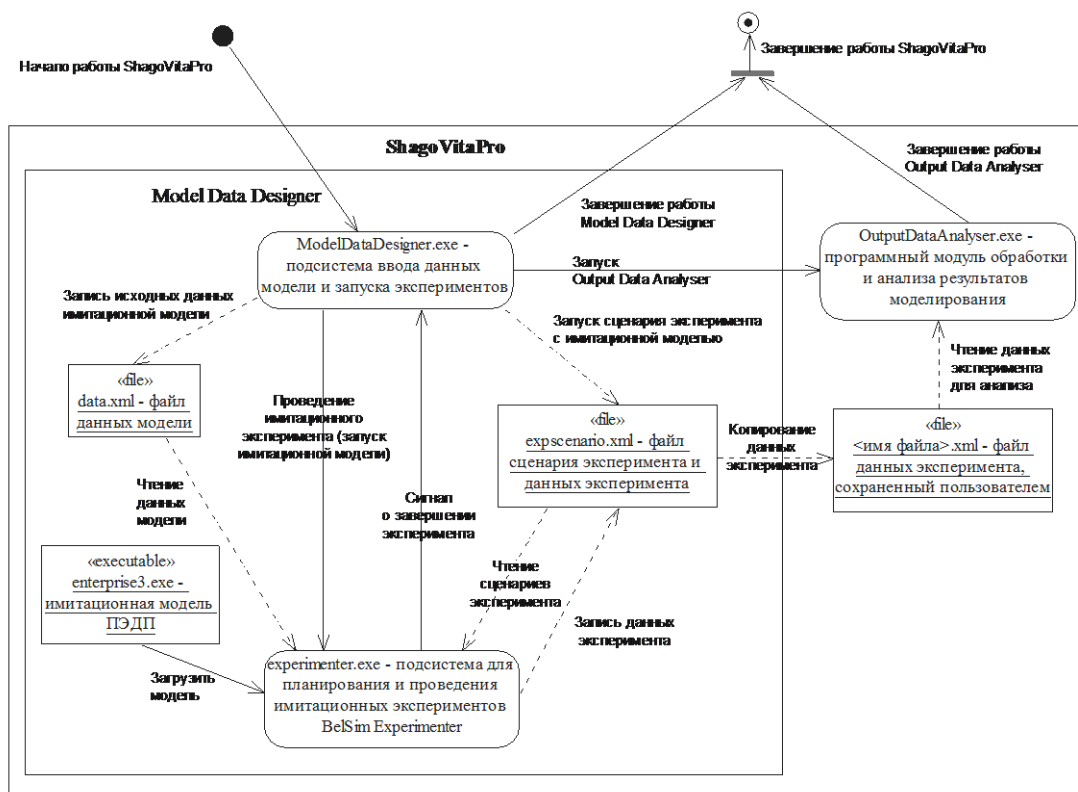


Рисунок 6.25 – Схема взаимодействия компонентов ShagoVitaPro

В ПК ShagoVitaPro для обеспечения производственного процесса материалами и комплектующими ( $PC_i:=true$ ) оценивается эффективность механизмов управления запасами материальных ресурсов. Например, для управления складскими запасами материальных ресурсов с постоянным объемом заказа минимально допустимый объем складского запаса может быть принят на нескольких уровнях: 420 ед., 840 ед., 1260 ед. После проведения модельных экспериментов получены итоговые результаты (таблица 6.7).

**Таблица 6.7 – Исследование процесса управления складскими запасами материальных ресурсов с постоянным объемом заказа**

Наименование параметра	Объем заказа, ед.			Максимум
	420	840	1260	
Название откликов модели	420	840	1260	
Чистая прибыль, тыс. р.	360835	357611	325014	360835
Коэффициент оборачиваемости собственного капитала, отн. ед.	10,29	10,50	11,36	11,36
Коэффициент обеспеченности собственными средствами, отн. ед.	0,83	0,79	0,76	0,83

Полученные в результате имитационных экспериментов значения откликов нормируются делением на максимальное значение (таблица 6.8).

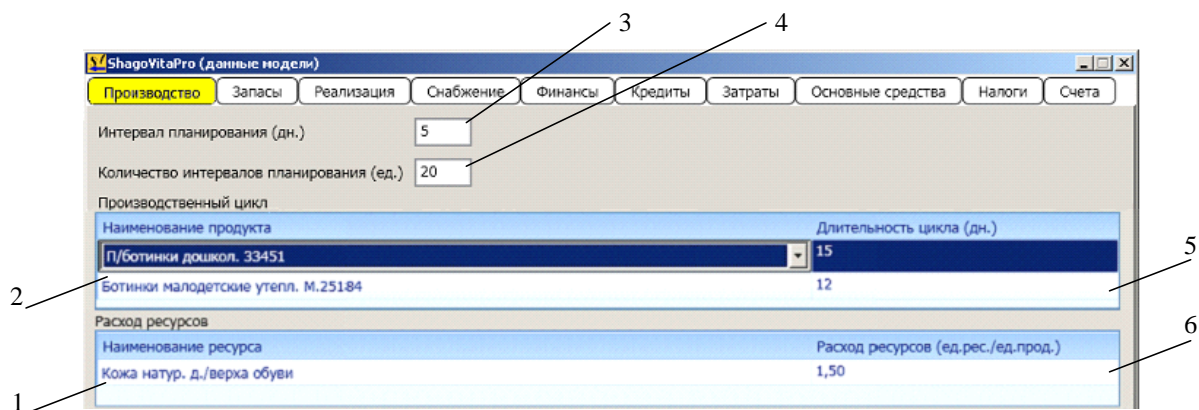
**Таблица 6.8 – Нормированные значения откликов имитационной модели**

Наименование параметра	Объем заказа, ед.		
	420	840	1260
Название откликов модели	420	840	1260
Чистая прибыль, отн. ед.	1,00	0,99	0,90
Коэффициент оборачиваемости собственного капитала, отн. ед.	0,91	0,92	1,00
Коэффициент обеспеченности собственными средствами, отн. ед.	1,00	0,96	0,92
Значение интегрального критерия	<b>2,91</b>	2,83	2,82

Интегральный критерий определяется линейной моделью путем суммирования значений нормированных откликов модели. В соответствии с таблицей 6.8 рациональное значение объема заказа ресурсов принято равным 420 ед.

Для эксплуатации базовой имитационной модели (БИМ) самими специалистами предприятия в ПК ShagoVitaPro разработан блок для ввода исходных данных модели (БВДМ) Model Data Designer, который предоставляет пользователю графический интерфейс для задания исходных значений параметров и позволяет сохранять входные и выходные данные модели в файле с расширением .SVT или в виде XML-файла. Для автоматизации планирования и проведения имитационных экспериментов с исполняемым файлом БИМ при исследовании производственно-экономической деятельности предприятия (ПЭДП) ПК ShagoVitaPro использует подсистему BelSim Experimenter по определенному сценарию на основе данных, задаваемых пользователем (см. рисунок 6.25).

Задание входных данных модели реализовано на десяти закладках. Например, параметры планирования производства представлены на закладке «Производство» с указанием используемых ресурсов, производимой продукции, длительности интервала планирования (см. п. 3.5) и др. (рисунок 6.26).



**Рисунок 6.26 – Пользовательский интерфейс с вкладкой «Производство»**

На рисунке 6.26 используются следующие обозначения: 1 – множество используемых материальных ресурсов; 2 – множество отдельных видов производимой продукции; 3 – задаваемая исследователем длительность интервала планирования, дней; 4 – количество интервалов планирования за период исследова-

ния; 5 – длительность производственного цикла единицы  $i$ -го вида продукции, дней/ед.; 6 – нормативный показатель расхода  $r$ -ого вида материального ресурса на изготовление единицы  $i$ -ого вида продукции.

На вкладке «Реализация» в ПК ShagoVitaPro вводятся временные интервалы между поставками изготовленной продукции потребителю, сроки платежа, объемы отгруженной продукции, цены по каждому типу продукции, данные для учета инфляционных процессов. Вкладка «Кредиты» позволяет вводить процентную ставку по кредиту, интервалы между обращениями за кредитом, срока возврата кредита, а указание (да/нет) на использование кредитов для покрытия нехватки денежных средств. На вкладке «Снабжение» (рисунок 6.27) указываются цены на материальные ресурсы, срок платежа за полученные ресурсы. Вкладка «Затраты» предназначена для ввода постоянных ежемесячных затрат и относительной доли заработной платы в них. По каждому типу продукции требуется ввести прочие переменные затраты и относительную долю заработной платы. На вкладках «Снабжение» и «Затраты» вводятся также показатели инфляционных процессов соответственно для ресурсов и затрат.

The screenshot shows the 'Снабжение' (Supply) tab in the ShagoVitaPro software. The interface is divided into several sections:

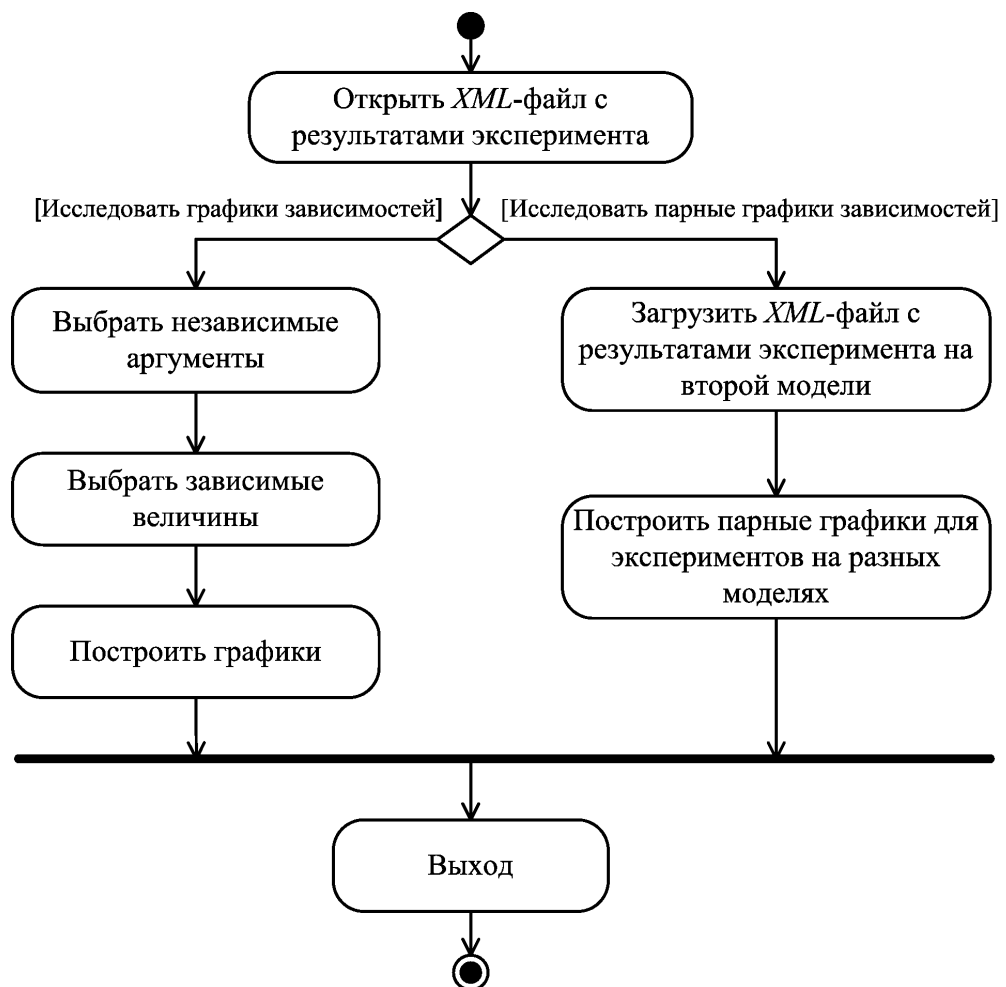
- Срок платежа (Payment Term):**
  - Среднее (дн.) (Average (days)): 15
  - Стандартное отклонение (дн.) (Standard deviation (days)): 5
  - Вид функции плотности распределения (Distribution function type): Равномерное (Uniform)
- Коэффициент изменения цен (Price change coefficient):**
  - Среднее (отн.ед./мес.) (Average (rel. unit./month)): 1,0250
  - Стандартное отклонение (отн.ед.) (Standard deviation (rel. unit.)): 1,0250
  - Вид функции плотности распределения (Distribution function type): Равномерное (Uniform)
- Интервал между изменениями цен (Price change interval):**
  - Среднее (дн.) (Average (days)): 18
  - Стандартное отклонение (дн.) (Standard deviation (days)): 9
  - Вид функции плотности распределения (Distribution function type): Равномерное (Uniform)
- Цены ресурсов (Resource prices):**

Наименование ресурса (Resource name)	Цена (руб./ед.) (Price (rub./unit.))
Кожа натур. д./верха обуви (Natural leather/upper of shoes)	1 500,00р.

**Рисунок 6.27 – Пользовательский интерфейс с вкладкой «Снабжение»**

Компонентами ПК ShagoVitaPro, кроме БВДМ и подсистемы BelSim Experimenter, являются файл с данными эксперимента и блок для обработки и анализа результатов моделирования Output Data Analyser (см. рисунок 6.25), что дает возможность пользователю оперативно подготовить исходные данные мо-

дели, провести имитационный эксперимент, выполнить обработку и анализ полученных результатов. Использование Output Data Analyser в ПК ShagoVitaPro позволяет упростить предварительную обработку и анализ результатов при исследовании имитационных моделей самими специалистами предприятия. ПК ShagoVitaPro с использованием Output Data Analyser (рисунок 6.28) предназначен для поддержки принятия решения основному пользователю программного комплекса – лицу, определяющему и принимающему решение. ПК обеспечивает исследование динамики изменения основных показателей (рентабельность, кредиторская и дебиторская задолженность и др.) производственно-экономической деятельности предприятия для разных вариантов изменений в производственной структуре или отдельных параметров (стоимость используемых ресурсов, переменные и постоянные затраты и др.) [39].



**Рисунок 6.28 – Схема технологии использования блока обработки результатов эксперимента Output Data Analyser**

Программный комплекс ShagoVitaPro был разработан при финансовой поддержке ОАО «Обувь» (г. Могилев, Республика Беларусь) в соответствии с договором ХД0877 и принят в опытную эксплуатацию (Приложение Е).

#### **6.5.5 Эксплуатация имитационной модели в ООО «СМИТ-Ярцево»**

Кроме белорусских предприятий компоненты базовой имитационной модели внедрены на российском предприятии ООО «СМИТ-Ярцево» (Смоленская обл.) с годовым экономическим эффектом, подтвержденным документами о внедрении (приложение Е).

В ООО «СМИТ-Ярцево» корпоративная информационная система применяется для разработки и построения производственной программы технологических линий в цехе прямой трубы. Производственная программа разрабатывается при известной производственной мощности оборудования, фактического запаса материальных ресурсов на складе. Требуется разработать программу производственного цеха для обеспечения функционирования технологических линий с наименьшими переналадками оборудования, необходимыми при переходе на новый тип продукции.

Задача планирования производства в ООО «СМИТ-Ярцево» имеет ряд особенностей и ограничений. Например, для изготовления ПИ-трубы установлены две технологические линии с разной производительностью и временным ресурсом, ограничивающим их использование в течение рабочего дня. Первая технологическая линия производит ПИ-трубы с диаметром от 25 мм до 133 мм. Вторая линия производит ПИ-трубы с диаметром от 25 мм до 133 мм и выше. Известной величиной является производительность технологических линий для изготовления ПИ-трубы заданного диаметра. Определенный тип ПИ-трубы, например оцинкованной трубы, изготавливается на первой технологической линии. Для заказчиков из Москвы и Московской области ПИ-трубы изготавливаются на второй технологической линии. Для неукомплектованной ПИ-трубы (на складе отсутствуют материальные ресурсы для производства) дата изготовления переносится на определенное количество рабочих дней.

Постановка задачи: спланировать производство ПИ-труб по технологическим линиям для минимизации простоев, вызванных переналадками оборудования при переходе от изготовления ПИ-трубы с одним диаметром на ПИ-трубу с другим диаметром. Ограничением решения задачи является необходимость производства ПИ-труб в определенной последовательности с учетом поставки продукции заказчику на дату, предусмотренную контрактом.

Из-за особенностей и ограничений исследуемого производства актуальной является задача создания алгоритма планирования для доопределения базовой имитационной модели с учетом специфики производства ООО «СМИТ-Ярцево». Для решения поставленной задачи используются следующие параметры:

$D_f$  – внешний диаметр ПИ-трубы, определенной на позиции  $f$  :

$$D_f = \{25, 28, 32, 42, 45, 57, 60, 76, 89, 108, 133, 159, 219, 273, 325, 377, 426, 530, 630, 720, 820, 1020\};$$

$Material_f$  – материальный ресурс для производства ПИ-трубы позиции  $f$

$$Material_f = \{Ст, Ц\}, \text{ Ст – стальная, Ц – оцинкованная};$$

$Quantity_f$  – длина ПИ-трубы позиции  $f$ ;

$Time_f$  – период изготовления ПИ-трубы позиции  $f$  :

$$Time_f = quantity_f \cdot W(D_f),$$

где  $W(D_f)$  – производительность изготовления ПИ-трубы с диаметром  $D_f$  позиции  $f$ , мин/м.п.:  $W(D_f < 159) = 1,2$  мин/м.п.;  $W(D_f = 219) = 1,33$  мин/м.п.;  $W(D_f = 273) = 2,14$  мин/м.п.;  $W(D_f = 325) = 2,31$  мин/м.п.;  $W(D_f = 377) = 2,4$  мин/м.п.;  $W(D_f = 426) = 2,5$  мин/м.п.;  $W(D_f = 530) = 3$  мин/м.п.;  $W(D_f = 630) = 3,5$  мин/м.п.;  $W(D_f = 720) = 4$  мин/м.п.;  $W(D_f = 820) = 6$  мин/м.п.;  $W(D_f = 1020) = 10$  мин/м.п.;

$customerType_f$  – характеристика заказчика ПИ-трубы позиции  $f$  :

$customerType_f = \{М, РФ, РБ\}$  (М – заказчик из Москвы, РФ – из остальных регионов России, РБ – из Беларуси);



$Nreset_L$  – некоторое число переналадок  $L$ -ой технологической линии ( $L=1..2$ ) на определенную дату;

$maxNreset_L$  – наибольшее возможное число переналадок  $L$ -ой технологической линии ( $L=1..2$ ), устанавливается в качестве ограничения;

$maxSumTime_L$  – максимальное время работы  $L$ -ой технологической линии в течение суток ( $L=1..2$ ), час – вводится пользователем;

$maxQuantity_f$  – максимальная длина ПИ-труб позиции  $f$ , произведенных при полной загрузке линии в течение рабочего дня:

$$maxQuantity_f = maxSumTime_L / W(D_f);$$

$currentPlanDate_L$  – планируемая дата выпуска продукции на линии  $L$ .

Тогда задача планирования программы производственного цеха состоит в определении для всех  $f$  даты выпуска  $Date_f$  и технологической линии  $L$  для производства ПИ-трубы позиции  $f$ .

Задача решается при следующих ограничениях:

- 1)  $\square \forall f : (D_f > 133) \Rightarrow (L = 2)$ ;
- 2)  $\forall f : (customerType_f = M) \Rightarrow (L = 2)$ ;
- 3)  $\forall f : (Material_f = \Pi) \Rightarrow (L = 1)$  ;
- 4)  $\forall Date_f \forall L \sum_f Time_f \cdot Quantity_f = maxSumTime_L$ ;
- 5)  $\forall Date_f \forall L Nreset_L \leq maxNreset_L$ .

Для построения оптимального плана производства предложен алгоритм планирования производства, включающий следующую последовательность основных этапов для каждого типа продукции  $f$  :

- 1) Для типа продукции  $f$ , по которой общая длина труб превышает величину  $maxQuantity_f$  на соответствующей линии, выполняется разбиение, пока выпуск продукции  $f$  не станет меньше максимального дневного выпуска  $maxQuantity_f$ , затем выполняется добавление следующего типа продукции  $f$  с максимальным дневным выпуском  $maxQuantity_f$ .

В результате для каждого типа продукции  $f$ , для которой производство превышает  $maxQuantity_f$ , получают несколько позиций с максимальным дневным выпуском и одну позицию с выпуском, равным остатку от деления общей длины труб на максимальный дневной выпуск.

2) Для каждого типа продукции  $f$ , начиная с первого, задается дата изготовления:

2.1) Если для рассматриваемого типа продукции  $f$  не указана дата, и диаметр трубы не превышает 133 мм, характеристика заказчика не равна «М» или труба оцинкованная, производство планируется на первой линии.

2.1.1) Выполняется проверка превышения времени работы линии на текущую дату, с учетом текущего типа продукции, максимального времени ее работы. При отсутствии превышения, переходим к шагу 2.1.2, иначе – к шагу 2.1.3.

2.1.2) Оценивается число  $Nreset_L$  запланированных переналадок технологической линии на текущую дату. Если переналадки с учетом нового типа продукции  $f$  превышают заданное максимальное количество  $maxNreset_L$ , выполняется поиск типа продукции  $f$  с одним из тех диаметров, которые уже запланированы на этот день. Если  $Nreset_L$  не достигнуто – планируется тип продукции  $f$  на текущую дату. Суммарное время изготовления трубы на эту дату увеличивается на величину, требуемую для производства текущего типа продукции.

2.1.3) Текущая позиция типа продукции  $f$  разделяется на две позиции и планируется такая длина выпуска труб, которую можно произвести до конца рабочей смены, остаток планируется к выпуску на следующую рабочую смену. Возвращаются к первой позиции типа продукции  $f$  с не указанной датой выпуска.

2.2) Если для планируемой позиции  $D_f$  превышает 133 мм, или заказчик из Москвы, выпуск предусматривается на линии больших диаметров.

2.2.1) Выполняется проверка превышения длительности работы линии ее максимального допустимого времени работы. Если превышения допустимого времени работы нет – переход к шагу 2.2.2, в противном случае – к шагу 2.2.3.

2.2.2) Оценивается запланированное количество переналадок  $N_{reset_L}$ . Если  $N_{reset_L}$  с учетом новой позиции превышает  $maxN_{reset_L}$ , выполняется поиск  $f$  с одним из тех диаметров, которые уже запланированы на текущий день. При отсутствии превышения позиция  $f$  предусматривается на текущую дату, и время изготовления трубы увеличивается на величину, требуемую для изготовления позиции  $f$ .

2.2.3) Если длительность изготовления ПИ-труб с учетом планируемой позиции  $f$  превышает допустимое время работы технологической линии, тогда текущая позиция  $f$  разделяется на две позиции. На текущую дату предусматривается такая длина ПИ-труб, которую можно выпустить до конца рабочего дня, остаток планируется на следующий день.

Параметры для реализации алгоритма планирования задаются на пользовательской форме (рисунок 6.29). Вид фрагмента планируемой производственной программы производства ПИ-труб представлен на рисунке 6.30.

План выпуска ПИ-трубы с минимумом переналадок

Количество часов работы линии малых диаметров (в сутки)	20	Количество переналадок линии малых диаметров (в сутки)	3
Количество часов работы линии всех диаметров (в сутки)	20	Количество переналадок линии больших диаметров (в сутки)	3
Неукомплектованные позиции перемещать на	3	дн.	
Максимальная суммарная длина изготавливаемых труб (в сутки)	1500	метров	

Работа по субботам     Работа по воскресеньям

Спланировать выпуск труб

Рисунок 6.29 – Задание параметров для планирования производства

План производства ПИ-трубы с 01.10.2011 по 17.10.2011						Составлен 01.10.2011 15:49:00									
№ п/п	Наименование продукции	D	Примечан.	Ед. изм.	Кол-во	Приорите	Сдана	Итого, м	№ заявки	Дата отгрузки	Дата выгус	Вид	Комплект	№ линии	Труба
9	Труба Ц 48х3,5-1-ППУ-ПЭ/125 # Труба ГОСТ 3262 /Ст 20	48	ГОСТ 326	м.п.	23		0	23	1862-ПИ	изм	28.09.2011	01.10.2011	М	полная	малая Ц
2	Труба Ц 48х3,5-1-ППУ-ПЭ/125 # Труба ГОСТ 3262 /Ст 20	48	ГОСТ 326	м.п.	20		0	20	1901-ПИ		05.10.2011	01.10.2011	РФ	полная	малая Ц
1	Труба Ст 57х3,5-1-ППУ-ПЭ/125 # Труба ГОСТ 10705 /Ст 20	57	ГОСТ 107	м.п.	103		23	80	1803-ПИ		26.09.2011	01.10.2011	РФ	полная	малая Ст
3	Труба Ст 57х3,5-1-ППУ-ПЭ/125 # Труба ГОСТ 10705 /Ст 20	57	ГОСТ 107	м.п.	62		0	62	1807-ПИ		27.09.2011	01.10.2011	РФ	полная	малая Ст
1	Труба Ст 57х3,5-3х3-1-ППУ-ОЦ/200 # Труба ГОСТ 10705 /Ст	57	ГОСТ 107	м.п.	220		0	220	1836-ПИ		28.09.2011	01.10.2011	М	полная	большая Ст
1	Труба Ст 57х4-1-ППУ-ПЭ/125 # Труба ГОСТ 10705 /Ст 20	57	ГОСТ 107	м.п.	100		0	100	1907-ПИ	срочно	30.09.2011	01.10.2011	М	полная	большая Ст
1	Труба Ц 57х3-1-ППУ-ОЦ/125 # Труба ГОСТ 10705 /Ст 20	57	ГОСТ 107	м.п.	150		0	150	1866-ПИ		03.10.2011	01.10.2011	РФ	полная	малая Ц
3	Труба Ст 57х3-1-ППУ-ПЭ/125 # Труба ГОСТ 10705 /Ст 20	57	ГОСТ 107	м.п.	60		0	60	1903-ПИ		03.10.2011	01.10.2011	РФ	полная	малая Ст
3	Труба Ц 57х3-1-ППУ-ПЭ/125 # Труба ГОСТ 10705 /Ст 20	57	ГОСТ 107	м.п.	20		0	20	1901-ПИ		05.10.2011	01.10.2011	РФ	полная	малая Ц
2	Труба Ц 76х3,5-1-ППУ-ПЭ/140 # Труба ГОСТ 10705 /Ст 20	76	ГОСТ 107	м.п.	23		0	23	1862-ПИ	изм	28.09.2011	01.10.2011	М	полная	малая Ц
2	Труба Ст 76х4-1-ППУ-ПЭ/140 # Труба ГОСТ 10705 /Ст 20	76	ГОСТ 107	м.п.	100		90	10	1907-ПИ	срочно	30.09.2011	01.10.2011	М	полная	большая Ст
3	Труба Ст 76х4-1-ППУ-ПЭ-625 ЗМКв /140 L=10000 # Труба ГОСТ	76	ГОСТ 873	шт	2		0	20	1893-ПИ		04.10.2011	01.10.2011	М	полная	большая Ст
2	Труба Ст 76х4-1-ППУ-ПЭ-625 ЗМКв /140 L=10000 # Труба ГОСТ	76	ГОСТ 873	шт	2		0	20	1893-ПИ		04.10.2011	01.10.2011	М	полная	большая Ст
1	Труба Ст 76х4-1-ППУ-ПЭ/140 # Труба ГОСТ 8732 /Ст 20	76	ГОСТ 873	м.п.	205		0	205	1893-ПИ		04.10.2011	01.10.2011	М	полная	большая Ст
1	Труба Ст 76х3,5-1-ППУ-ПЭ/140 # Труба ГОСТ 10705 /Ст 20	76	ГОСТ 107	м.п.	40		0	40	1901-ПИ		05.10.2011	01.10.2011	РФ	полная	малая Ст
2	Труба Ст 76х3,5-1-ППУ-ПЭ/140 # Труба ГОСТ 10705 /Ст 20	76	ГОСТ 107	м.п.	250		0	100	1890-ПИ		10.10.2011	01.10.2011	РФ	полная	малая Ст
2	Труба Ст 108х4-1-ППУ-ПЭ/180 # Труба ГОСТ 10705 /Ст 20	108	ГОСТ 107	м.п.	24		4	20	1807-ПИ		27.09.2011	01.10.2011	РФ	полная	малая Ст
1	Труба Ст 108х4-57х3-1-ППУ-ОЦ/315 # Труба ГОСТ 10705 /Ст	108	ГОСТ 107	м.п.	20		0	20	1861-ПИ		29.09.2011	01.10.2011	РФ	полная	малая Ст
1	Труба Ст 108х4-1-ППУ-ПЭ/180 # Труба ГОСТ 10705 /Ст 20	108	ГОСТ 107	м.п.	128		0	128	1898-ПИ		29.09.2011	01.10.2011	РФ	полная	малая Ст
4	Труба Ст 108х5-1-ППУ-ПЭ/180 # Труба ГОСТ 10705 /Ст20	108	ГОСТ 107	м.п.	100		0	100	1907-ПИ	срочно	30.09.2011	01.10.2011	М	полная	большая Ст
2	Труба Ст 108х4-1-ППУ-ПЭ/180 # Труба ГОСТ 10705 /Ст 20	108	ГОСТ 107	м.п.	160		0	160	1891-ПИ		03.10.2011	01.10.2011	М	полная	большая Ст
1	Труба Ст 108х4-1-ППУ-ПЭ/180 # Труба ГОСТ 10705 /Ст 20	108	ГОСТ 107	м.п.	83		0	83	1887-ПИ		04.10.2011	01.10.2011	МСК	полная	малая Ст
1	Труба Ст 108х4-1-ППУ-ПЭ/180 # Труба ГОСТ 10705 /Ст 20	108	ГОСТ 107	м.п.	50		0	50	1892-ПИ		04.10.2011	01.10.2011	РФ	полная	малая Ст
1	Труба Ст 133х4-1-ППУ-ПЭ/225 # Труба ГОСТ 10705 /Ст 20	133	ГОСТ 107	м.п.	46		0	46	1862-ПИ	изм	28.09.2011	01.10.2011	М	полная	большая Ст
1	Труба Ст 133х4-1-ППУ-ОЦ/225 # Труба ГОСТ 10705 /Ст 20	133	ГОСТ 107	м.п.	46		0	46	1894-ПИ		29.09.2011	01.10.2011	РФ	полная	малая Ст
1	Труба Ст 133х4-1-ППУ-ПЭ/225 # Труба ГОСТ 10705 /Ст 20	133	ГОСТ 107	м.п.	866		257.73	108.27	1883-ПИ		02.10.2011	01.10.2011	МСК	полная	малая Ст
1	Труба Ст 159х4,5-1-ППУ-ПЭ/250 # Труба ГОСТ 10705 /Ст 20	159	ГОСТ 107	м.п.	115		0	115	1807-ПИ		27.09.2011	01.10.2011	РФ	полная	большая Ст
1	Труба Ст 159х4,5-1-ППУ-ОЦ/250 # Труба ГОСТ 10705 /Ст 20	159	ГОСТ 107	м.п.	420		0	50	1905-ПИ		04.10.2011	01.10.2011	М	полная	большая Ст
<b>ИТОГО за 01.10.2011</b> больших диаметров 165 м, малых диаметров 1914,27 м. Выпуск на большой линии 1046 м, на малой линии 1033,27 м.															
1	Труба Ст 38х3-1-ППУ-ПЭ/110 # Труба ГОСТ 8732 /Ст 20	38	ГОСТ 873	м.п.	30		0	30	1906-ПИ		30.09.2011	02.10.2011	М	нет трубы	большая Ст
2	Труба Ц 42,3х3,2-1-ППУ-ОЦ/125 # Труба ГОСТ 3262 /Ст 20	42	ГОСТ 326	м.п.	150		29.53	120.47	1866-ПИ		03.10.2011	02.10.2011	РФ	полная	малая Ц
2	Труба Ст 76х3,5-1-ППУ-ПЭ/140 # Труба ГОСТ 10705 /Ст 20	76	ГОСТ 107	м.п.	250		0	150	1890-ПИ		10.10.2011	02.10.2011	РФ	полная	малая Ст
5	Труба Ст 89х4-1-ППУ-ПЭ/180 # Труба ГОСТ 10705 /Ст 20	89	ГОСТ 107	м.п.	100		50	50	1907-ПИ	срочно	30.09.2011	02.10.2011	М	полная	большая Ст
4	Труба Ст 89х4-1-ППУ-ОЦ/180 # Труба ГОСТ 8732 /Ст 20	89	ГОСТ 873	м.п.	470		0	300	1896-ПИ		05.10.2011	02.10.2011	М	нет трубы	большая Ст
3	Труба Ц 89х4-1-ППУ-ПЭ/160 # Труба ГОСТ 10705 /Ст 20	89	ГОСТ 107	м.п.	146		0	146	1910-ПИ		08.10.2011	02.10.2011	М	полная	малая Ц
4	Труба мерная Ст 89х4-1-ППУ-ПЭ/160 # Труба ГОСТ 10705 /Ст	89	ГОСТ 107	м.п.	8,5		0	8,5	1900-ПИ		09.10.2011	02.10.2011	РФ	полная	малая Ст
5	Труба мерная Ст 89х4-1-ППУ-ПЭ/160 # Труба ГОСТ 10705 /Ст	89	ГОСТ 107	м.п.	8,5		0	8,5	1900-ПИ		09.10.2011	02.10.2011	РФ	полная	малая Ст
6	Труба мерная Ст 89х4-1-ППУ-ПЭ/160 # Труба ГОСТ 10705 /Ст	89	ГОСТ 107	м.п.	4		0	4	1900-ПИ		09.10.2011	02.10.2011	РФ	полная	малая Ст
7	Труба мерная Ст 89х4-1-ППУ-ПЭ/160 # Труба ГОСТ 10705 /Ст	89	ГОСТ 107	м.п.	4		0	4	1900-ПИ		09.10.2011	02.10.2011	РФ	полная	малая Ст

**Рисунок 6.30 – Фрагмент разработанного плана производства ПИ-труб**

Эксплуатация базовой имитационной модели с алгоритмом планирования программы производства, учитывающим особенности и ограничения производства ООО «СМИТ-Ярцево», реализована в корпоративной информационной системе предприятия. При рациональных параметрах процесса планирования программы производства и планирования поставок материальных ресурсов получен годовой экономический эффект в сумме 3 954 080 рос. р. в 2011 году (приложение Е) за счет сокращения неуккомплектованных к моменту запуска в производство изделий на 50% и сокращения заказов, не выполненных в срок, на 80%.

### 6.5.6 Эксплуатация ПТКИ BelSim в ОАО «Моготекс»

В производстве готовой ткани и трикотажных полотен качество выпускаемой продукции напрямую зависит от используемого сырья, технологии изготовления и оборудования на предприятии. Для поиска путей повышения качества продукции, экономии энергоресурсов ставится задача построения имитационной модели (ИМ) процесса управления производством готовой продукции предприя-

тия. Готовая ткань и трикотажные полотна производятся в соответствии с требованиями технологической документации ОАО «Моготекс» (г. Могилев).

Цель процесса управления – производство готовых тканей и трикотажных полотен в соответствии с планом производства, соответствующих определенным требованиям и удовлетворяющих требованиям потребителей. Одной из стадий изготовления готовой ткани является производственный процесс крашения тканей набивным способом (ПП КТНС). Входными ресурсами производственного процесса являются: суровые ткани и трикотажные полотна с соответствующей сопроводительной документацией; химикаты и красители, прошедшие входной контроль; вода; электроэнергия; пар; газ. Соответственно, на выходе получают готовые ткани и трикотажные полотна с соответствующей сопроводительной документацией, а так же – потери в процессе производства химикатов, красителей и воды.

В себестоимость отдельных технологических режимов включаются стоимости используемых ресурсов: химикатов, красителей, энергоресурсов (пара, газа, электроэнергии), зарплаты рабочих и др.

Для каждого типа технологического оборудования отделом труда и заработной платы (ОТИЗ) как функция от расчетной скорости  $v_{ob}$  рассчитывается нормированное задание  $N_z$  – количество ткани выбранного артикула (в м.п.), которое должно обрабатываться на данном оборудовании по заданному технологическому режиму за 1 час. Ниже представлен вид одной из формул выведенных ОТИЗ для расчета  $N_z$ :

$$N_z = ROUND(kpv \cdot v_{ob} \cdot 60) \text{ (м.п./ч)}, \quad (6.23)$$

где  $kpv \in [0, 1]$  – коэффициент, вводимый ОТИЗ для соответствующего типа оборудования.

Нормативный показатель времени  $T_{Nz\_1000}$ , необходимого на обработку 1000 п.м. выбранного артикула ткани на определенном оборудовании по заданному технологическому режиму, зависит от нормированного задания  $N_z$ :

$$T_{Nz\_1000} = 1000/N_z \text{ (ч)}. \quad (6.24)$$

С учетом известных норм расхода ресурсов за 1 час работы оборудования ( $N_{RG}$  – газа, м. куб/ч;  $N_{RP}$  – пара, т/ч;  $N_{RE}$  – электроэнергии, кВт/ч), коэффициента энтальпии  $Intal$  и цен на энергоресурсы ( $Cen_G$ ,  $Cen_P$ ,  $Cen_E$  – газа, пара, электроэнергии, соответственно) рассчитываются: нормы расхода необходимых для обработки на оборудовании 1000 м.п. ткани ресурсов ( $N_{RG\_1000}$ ,  $N_{RP\_1000}$ ,  $N_{RE\_1000}$  – газа, пара, электроэнергии, соответственно), стоимости расхода отдельно взятых ресурсов ( $St_G$ ,  $St_P$ ,  $St_E$  – газа, пара, электроэнергии, соответственно) и общая стоимость энергоресурсов  $St_{Energ}$ :

$$St_G = Cen_G \cdot N_{RG\_1000} \text{ (руб.)}, \quad (6.25)$$

$$N_{RG\_1000} = N_{RG} \cdot T_{Nz\_1000} \text{ (м. куб)}, \quad (6.26)$$

$$St_P = Cen_P \cdot N_{RP\_1000} \text{ (руб.)}, \quad (6.27)$$

$$N_{RP\_1000} = Intal \cdot N_{RP} \cdot T_{Nz\_1000} \text{ (гигакал)}, \quad (6.28)$$

$$St_E = Cen_E \cdot N_{RE\_1000} \text{ (руб.)}, \quad (6.29)$$

$$N_{RE\_1000} = N_{RE} \cdot T_{Nz\_1000} \text{ (кВт)}, \quad (6.30)$$

$$St_{Energ} = St_G + St_P + St_E \text{ (руб.)}. \quad (6.31)$$

Суммарная стоимость  $St_{1000}$  расхода химикатов и энергоресурсов на обработку 1000 м.п. выбранной ткани по заданному технологическому режиму на определенном оборудовании рассчитывается по формуле:

$$St_{1000} = St_{Energ} + St_{Xim} \text{ (руб.)}, \quad (6.32)$$

где  $St_{Xim}$  – Общая стоимость расхода химикатов, необходимых на обработку 1000 м.п. по заданному технологическому режиму на определенном оборудовании.

С учетом (6.1) – (6.10)  $St_{1000}$  является функцией от расчетной скорости работы технологического оборудования  $v_{ob}$ .

В технологических проводках производства готовой ткани ОАО “Мого-текс” для каждого технологического режима указывается допустимый диапазон изменения расчетной скорости  $v_{ob}$ , м/мин:

$$v_{ob} \in [S_{MIN}, S_{MAX}]. \quad (6.33)$$

ОТИЗ на основании (6.11) определяет для дальнейших расчетов некоторое усредненное значение  $v_{ob}$ :

$$v_{ob} = 0,5 \cdot (S_{MIN} + S_{MAX}) + 1. \quad (6.34)$$

Рассматриваемый ПП КТНС является управляемой системой, находящейся в одном из состояний, изменяемых на соответствующем этапе производства управляющим воздействием (управлением). При этом эффективность управления определяется многокритериальной целевой функцией (ЦФ) в зависимости от текущего состояния производственной системы и применяемого управления.

ПП КТНС представлен в виде взвешенного ациклического орграфа [99], в котором технологические режимы – вершины графа. Нагрузки (стоимости ресурсов, временные затраты) на дуги графа, в общем случае стохастического характера, могут быть уточнены методом Монте-Карло с использованием специально разработанных имитационных моделей и/или посредством статистического анализа данных ПП КТНС. При этом для каждой технологической операции определены нормы расхода времени и общие стоимости расхода ресурсов для обработки заданного количества продукции.

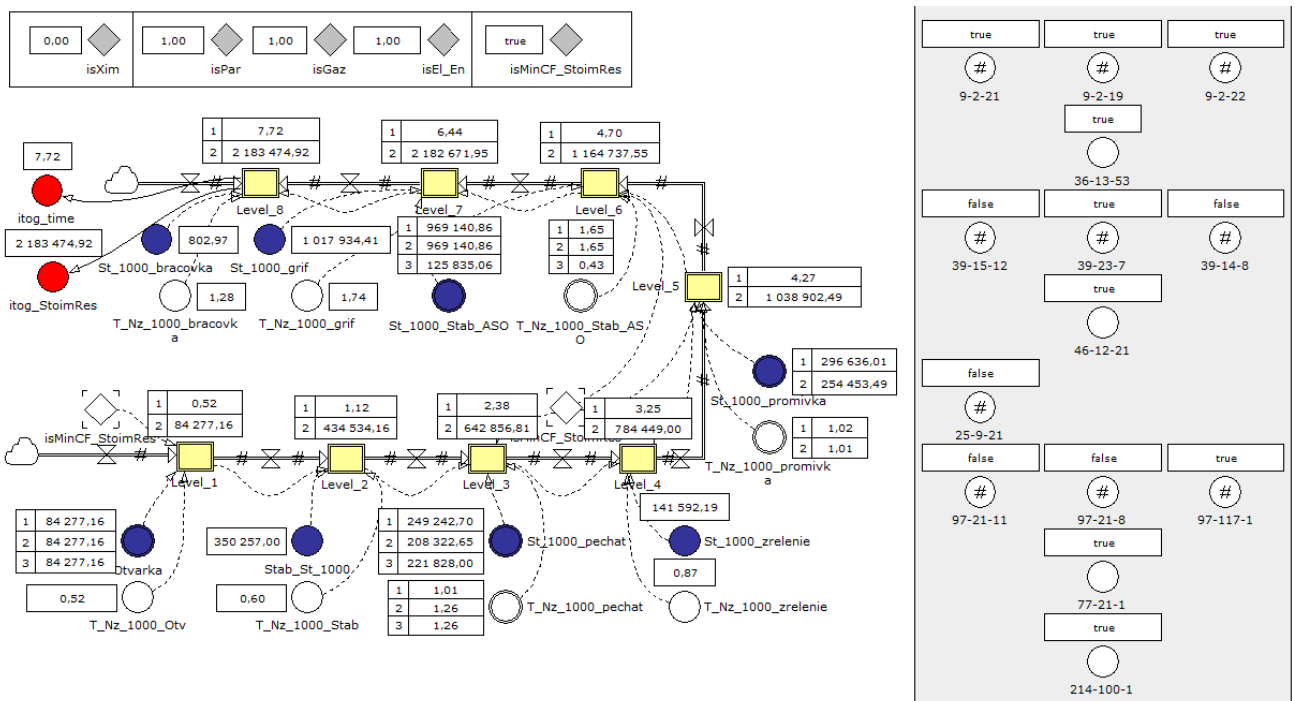
Для устранения неоднозначности выбора технологических режимов на каждом из этапов ПП КТНС поставлена задача многокритериальной оптимизации с использованием принципа оптимальности Беллмана с лексикографическим упорядочением критериев оптимизации (по стоимости ресурсов и/или временных затрат для выпуска определенного количества продукции). Требуется

найти состояния ТП, «близкие» в соответствии с заданной мерой близости к оптимальному состоянию.

Для поиска путей оптимизации распределения ресурсов в ПП КТНС построена имитационная модель (ИМ), реализующая модифицированный метод оптимизации ПП КТНС. Для разработки ИМ выбрана среда PowerSim, определены критерии оптимизации и управляющие параметры ИМ.

С учетом концептуальной модели процесса производства готовой ткани в нотации IDEF0 и формул (6.23) – (6.34) в среде математического пакета PowerSim построена имитационная модель (ИМ) для расчета норм расхода времени и энергоресурсов (пар, газ, электроэнергия) на выпуск 1000 м.п. тканей по отдельным технологическим режимам.

Каждая стадия моделируемого производства готовых тканей представлена на отдельном листе имитационной модели. Выбор оптимального сочетания технологических режимов для исследуемой проводки выполняется на специально разработанном листе итогов имитационной модели (рисунок 6.31).



**Рисунок 6.31 – Вкладка итогов имитационного моделирования**

Переменные синего цвета отражают стоимости расхода ресурсов, необходимых для обработки 1000 м.п. в соответствии с указанными режимами, а пере-



менные белого цвета – результаты по времени. Переменные уровня, выделенные желтым цветом, показывают суммарный результат на каждой стадии по времени и стоимости в соответствии с заданными критериями оптимизации (min или max). Переменные красного цвета показывают итоговый результат оптимизации по времени и стоимости. В области серого цвета – переменные, показывающие оптимальное сочетание технологических режимов для технологической проводки (путь по графу). В верхней части листа находятся константы, которые используются в качестве переключателей оптимизации – управляющие переменные.

Построенная имитационная модель позволяет быстро пересчитывать разные варианты для режимов оборудования с использованием заданных ресурсов (электроэнергия, пар, газ и др.). При этом можно выбирать критерий оптимизации (min, max) и получать промежуточные значения временных и стоимостных затрат ресурсов на каждой стадии производства продукции.

Шаги выполнения экспериментов с ИМ:

**Шаг 1.** Установка начальных значений входных параметров ИМ.

**Шаг 2.** Расчет для каждой технологической операции ПП КТНС норм расхода времени на обработку определенного количества продукции.

**Шаг 3.** Расчет для каждой операции ПП КТНС норм расхода отдельных ресурсов на обработку определенного количества продукции.

**Шаг 4.** Определение для каждой операции ПП КТНС общих стоимостей расхода ресурсов для обработки определенного количества продукции. При этом общие стоимости расхода ресурсов определяются с учётом параметров ИМ управления потоками ресурсов при имитации ПП КТНС.

**Шаг 5.** Подготовка по результатам моделирования вариантов решений для оптимизации ПП КТНС:

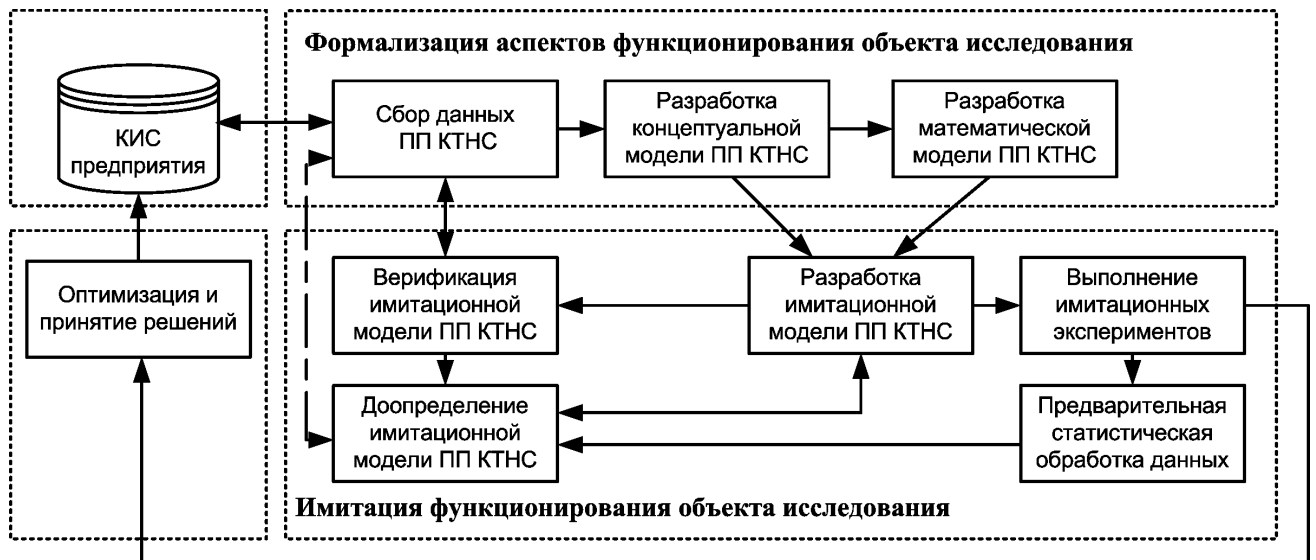
**Шаг 5.1** Определение последовательности оптимальных состояний ПП КТНС на этапах производства готовой продукции.

**Шаг 5.2** Построение оптимального сочетания технологических режимов для ПП КТНС (оптимальных управлений).

**Шаг 5.3** Расчет с учётом шагов 3, 5.2 общего расхода отдельных ресурсов, необходимых для получения оптимального сочетания технологических режимов ПП КТНС.

**Шаг 6.** Выгрузка результатов оптимизации ИМ в КИС предприятия.

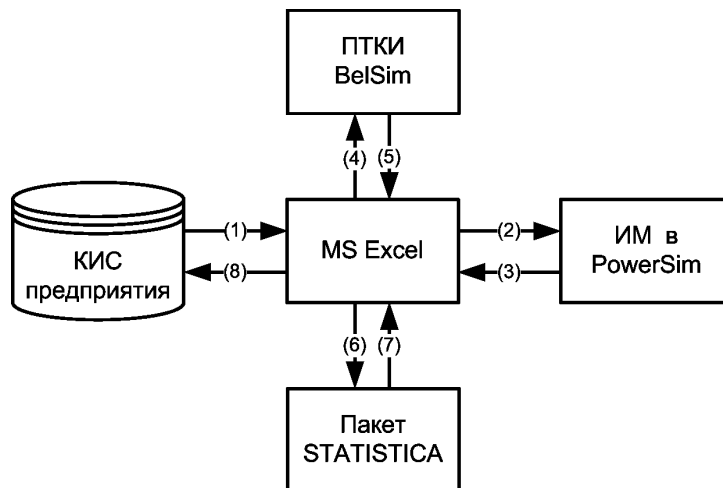
Для уточнения нагрузок на дуги графа ПП КТНС посредством применения метода Монте-Карло в ИМ предусмотрены стохастический анализ имитации норм расхода ресурсов рассматриваемых технологических режимов и оптимизация ПП КТНС с учётом уточнённых (в результате статистической обработки накопленных результатов моделирования) общих стоимостей ресурсов (см. рисунок 6.31). При этом на рисунке 6.32 приведена схема реализации имитационных экспериментов.



**Рисунок 6.32 – Схема подготовки и проведения имитационных экспериментов**

Для реализации метода оптимизации распределения ресурсов в ПП КТНС разработан программный комплекс – комплексное использование программных и информационных средств (рисунок 6.33), включающих MS Excel, пакет математического моделирования PowerSim, программно-технологический комплекс имитации (ПТКИ) сложных систем BelSim (ПО STATISTICA), интегрированный в корпоративную информационную систему (КИС) предприятия, что обеспечивает решение задачи оптимального распределения ресурсов в реальных условиях

корпоративной информационной системы предприятия.



**Рисунок 6.33 – Комплекс программных и информационных средств интеграции ИМ в КИС предприятия**

Алгоритм работы программного комплекса:

Шаг 1. Получение, передача и запись необходимых данных из источника КИС предприятия в табличный процессор MS Excel.

Шаг 2. Установка начальных значений переменных ИМ в PowerSim; запуск имитационного эксперимента.

Шаг 3. Запись результатов имитационного моделирования из PowerSim в табличный процессор MS Excel.

Шаги 4, 6. Передача результатов моделирования из табличного процессора MS Excel в подсистему Data Processing ПТКИ BelSim (или пакет STATISTICA) для анализа,

Шаги 5, 7. Запись данных о результатах анализа в табличный процессор MS Excel.

Шаг 8. Передача и сохранение полученных результатов анализа в СУБД КИС предприятия.

В результате разработки и применения имитационной модели получены оптимальные сочетания режимов для проводки с минимальными и максимальными затратами на энергоресурсы, соответственно:

1) Время: 7,72 час; Суммарная стоимость электроэнергии, пара, газа: 2183474,92 р.;

2) Время: 8,7 час; Суммарная стоимость электроэнергии, пара, газа: 3109883,29 р.;

Для исследованной технологической проводки 1-3775 ПП КТНС определенного артикула ткани моделирование показало: итоговая разница между наилучшим и наихудшим вариантами сочетаний режимов составила 926 тыс. р., а разница по времени составила около часа на производство 1000 м.п. ткани. Экономический эффект от оптимизации (при производстве 1000 м. п. тканей) в ценах 2012-го года составил порядка 684 тыс. белор. р. (14%), сокращение времени на 0,64 час (9%), соответственно для стоимостных и временных затрат.

## 6.6 Выводы по шестой главе

В шестой главе диссертационной работы получены следующие основные результаты.

1. Приведены результаты апробации метода поиска рациональных решений в системе управления промышленным предприятием. На апробацию было выбрано решение следующих задач эксплуатации имитационной модели для системы распределения ресурсов предприятия: оптимизация распределения производственной нагрузки (задача 1); выявление свойств имитационных моделей контрактов продажи, купли, на дилерское обслуживание и товарному обмену для обоснования применения аналитических моделей оценки эффективности этих контрактов (задача 2); выбор типа координации планирования производственной программы (задача 3); создание библиотеки имитационных моделей исследования информационной системы промышленного предприятия (задача 4).

2. ПТКИ *BelSim* позволяет решать задачи моделирования как для производственных проблем предприятия, так и для других социально-экономических систем. Типовой оптимизационной задачей является задача о планировании грузоперевозок на автотранспортном предприятии с минимальными затратами. Для решения этой задачи в среде ПТКИ *BelSim* использована реализация генетического алгоритма, что позволило значительно сократить пространство решений за счет исключения недопустимых вариантов грузоперевозок. Задача о грузопере-

возках использовалась также для оценки эффективности распределенных вычислений в вычислительной сети.

3. В ходе апробации метода поиска рациональных решений была решена задача оптимизации производственного объединения ОАО «Могилевхимволокно», состоящего из нескольких заводов. Имитационная модель производственного процесса этого объединения была реализована на основе процессного способа имитации. Проведено исследование имитационной модели непрерывного производственного процесса. По результатам имитационных экспериментов были построены зависимости себестоимости продукции одновременно от выработки двух типов ресурсов и зависимости количества выпущенной продукции одновременно от выработки тех же двух типов ресурсов. В результате применения разработанного алгоритма планирования процесса производства диметилтерефтолата годовой экономический эффект составил 100 млн белор. р. в ценах 2005г. за счет снижения удельных затрат в себестоимости выпускаемой продукции.

5. Для исследования контрактов в базовую имитационную модель предприятия дополнительно включены процессы, моделирующие различные типы контрактов по соответствующим алгоритмам. Количественная оценка эффективности контракта ИМ исследуется на двух уровнях (0 – прогон модели без контракта, 1 – прогон модели с контрактом). Для ввода исходных данных модели и построения плана эксперимента используется специальная программа – *ExperimentDesigner*.

6. Для решения задачи оптимального планирования на уровне производственного цеха разработан программный комплекс (ПК) ShagoVitaPro на основе базовой имитационной модели. ПК позволяет упростить анализ и обработку результатов экспериментов при эксплуатации моделей самими специалистами предприятия, обеспечивает исследование динамики изменения основных показателей (рентабельность, кредиторская и дебиторская задолженность и др.) производственно-экономической деятельности предприятия для разных вариантов

изменений в производственной структуре или отдельных параметров (стоимость используемых ресурсов, переменные и постоянные затраты и др.).

7. Разработаны математические модели для прогнозирования эффективности контрактов, наиболее часто используемых на промышленных предприятиях (продажи, купли, дилерского обслуживания, по товарному обмену и др.), отличающиеся наличием интервальных оценок рентабельности, которые реализованы в среде ПТКИ BelSim с применением *процессного способа имитации при высоком уровне детализации* процессов в системе управления предприятия, на основе которых разработан программный комплекс *Contract Analyzer*, реализующий методики прогнозирования эффективности и оптимизации параметров контрактов с интервальными оценками их рентабельности

8. Эксплуатация базовой имитационной модели с алгоритмом планирования программы производства, учитывающим особенности и ограничения производства ООО «СМИТ-Ярцево», реализована в корпоративной информационной системе предприятия. Внедрение разработанных программных средств и алгоритмов в ООО «СМИТ-Ярцево» позволило получить годовой экономический эффект в сумме 3 954 080 рос. р. в 2011 году (приложение Е) за счет сокращения неукомплектованных к моменту запуска в производство изделий на 50% и сокращения заказов, не выполненных в срок, на 80%.

9. При использовании методики решения задачи определения последовательности оптимальных управлений и состояний производственного процесса способом пошаговой оптимизации на примере данных ОАО «Моготекс» (Республика Беларусь, г. Могилев) экономический эффект от оптимизации (при производстве 1000 метров погонных тканей) в ценах 2012-го года составил порядка 684 тыс. белор. р. (14 %), сокращение времени на 0,64 час (9 %), соответственно для стоимостных и временных затрат.

## ЗАКЛЮЧЕНИЕ

### Основные выводы и результаты диссертации

1. Разработаны теоретические положения технологии имитационного моделирования и принятия решений в информационных системах управления промышленного предприятия, основанные на представлении информационной системы в виде многоуровневой иерархической социально-экономической системы и введении понятия ресурсов  $f$ - и  $p$ -типа, решении многокритериальной оптимизационной задачи управления проектированием имитационных моделей внутрифирменного управления предприятием, *отличающихся* комплексным применением информационных технологий корпоративной информационной системы, что в совокупности является существенным научным достижением в теории управления социально-экономическими системами и имеет важное социально-экономическое и хозяйственное значение для повышения эффективности управления производственными фирмами.

2. Предложен метод построения имитационных моделей (ИМ), основанный на процессном способе моделирования системы управления ресурсами промышленного предприятия, отличающийся использованием комплекса взаимосвязанных имитационных моделей с высоким уровнем детализации, включает *новые подходы* к составлению *вербальной и концептуальной модели* процессов функционирования промышленного предприятия на основе IDEF0-концепции; что обеспечивает построение модели *процесса* управления ресурсами предприятия в сокращенные сроки.

3. Исследован метод решения задачи многокритериального управления построением имитационных моделей системы управления промышленного предприятия с использованием комплекса информационных технологий и ресурсов  $f$ - и  $p$ -типа (трудовых ресурсов и программных средств), основанный на использовании принципа оптимальности Беллмана, обеспечивающий эффективное использование материальных, финансовых и трудовых ресурсов предприятия с корпоративной информационной системой.

4. Разработан метод поиска рациональных решений по распределению финансовых, материальных и трудовых ресурсов промышленного предприятия, основанный на имитации системы управления ресурсами с помощью базовой имитационной модели, *отличающийся* использованием постулата совместимости координирующих решений и специальной процедуры адаптации модели алгоритма случайного поиска оптимума целевой функции с применением кластерного анализа для определения наилучшего решения в системе распределения ресурсов.

5. Предложен алгоритм количественной оценки эффективности координирующих решений в системе управления промышленного предприятия, основанный на использовании базовой имитационной модели, постулата совместимости координирующих решений и применении аксиоматического принципа Парето.

6. Исследован программно-технологический комплекс имитации сложных систем (ПТКИ) BelSim с открытой архитектурой для автоматизации задач проектного моделирования процессов управления предприятиями, использующими корпоративную информационную систему, и программное средство реализации метода поиска рациональных решений BelSim Optimizer, реализованное в среде ПТКИ BelSim.

7. Адекватность базовой имитационной модели проверена при исследовании бизнес-процессов КИС в ОАО «Обувь» (Республика Беларусь, г. Могилев); исследование имитационной модели производственного процесса выполнено в ОАО «Могилевхимволокно» (Республика Беларусь) с получением годового экономического эффекта 100 млн белор. р. в ценах 2005 г. за счет снижения удельных затрат в себестоимости выпускаемой продукции; внедрение разработанных программных средств и алгоритмов в ООО «СМИТ-Ярцево» (г. Ярцево, Смоленская обл., Российская Федерация) позволило получить годовой экономический эффект в сумме около 4 млн рос. р. в ценах 2011 г.; методика решения задачи определения последовательности оптимальных управлений и состояний производственного процесса способом пошаговой оптимизации апробирована на



примере данных ОАО «Моготекс» (Республика Беларусь, г. Могилев) со значительным экономическим эффектом.

8. Теоретические основы и практическая реализация результатов исследования (имитационные модели, программные продукты, технологии их использования и исследования) используются в процессе обучения специалистов в области внутрифирменного управления, системного анализа и имитационного моделирования сложных систем (Белорусско-Российский университет, Белорусский национальный технический университет).

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Авдулов, П. В. Экономико-математические методы и модели для руководителя / П. В. Авдулов, Э. И. Гойзман, В. А. Кутузов. – М.: Экономика, 1984. – 232 с.: ил.
2. Аверченков, В. И. Концепция оценки эффективности процессов управления в корпоративных информационных системах предприятий / В. И. Аверченков, С. К. Крутолевич, А. И. Якимов, К. В. Захарченков // Вестник Брянского государственного технического университета. – 2016. – № 1 (49). – С. 113–119.
3. Аверченков, В. И. Многокритериальное управление технологическим процессом с использованием принципа оптимальности Беллмана / В. И. Аверченков, А. И. Якимов, Е. М. Борчик // Известия Волгоградского государственного технического университета. – 2014. – № 3(22). – С. 94–101.
4. Аверченков, В. И. Обработка многомерных данных несколькими методами кластерного анализа / В. И. Аверченков, А. И. Якимов, Е. М. Борчик, В. В. Башаримов // Вестник Воронежского государственного технического университета, серия: Системный анализ и информационные технологии. – 2016. – № 1. – С. 110–119.
5. Аверченков, В. И. Структура программно-технологического комплекса имитации в корпоративной информационной системе предприятия / В. И. Аверченков, А. И. Якимов // Вестник Белгородского государственного технологического университета им. В. Г. Шухова. – 2016. – № 6. – С. 176–182.
6. Алиев, Р. А. Методы и алгоритмы координации в промышленных системах управления / Р. А. Алиев, М. И. Либерзон. – М: Радио и связь, 1987. – 208 с.
7. Алтунин, А. Е. Модели и алгоритмы принятия решений в нечетких условиях: монография / А. Е. Алтунин, М. В. Семухин. – Тюмень: Изд-во Тюменского гос. ун-та, 2000. – 352 с.: ил.
8. Альховик, С. А. Прогнозирование финансово-экономического состояния предприятия на основе имитационного моделирования / С. А. Альховик, А. И.

Якимов // Проблемы самоорганизующихся производственных и социально-экономических систем: материалы II-й междунар. науч.-практ. конф., 15–16 нояб. 2001 г. – Минск: Гипросвязь, 2001. – С. 64–67.

9. Альховик, С. А. Программный комплекс оценки эффективности контрактов «Contract Analyzer 2» : свидетельство о регистрации компьютерной программы № 026 / С. А. Альховик, А. И. Якимов. – Минск: НЦИС, 2008. – Заявка № С20080011. – Дата подачи: 04.04.2008.

10. Альховик, С. А. Моделирование маркетинговой деятельности промышленных предприятий для прогнозирования эффективности контрактов : дис. ... канд. техн. наук : 05.13.18 : защищена 13.02.04 : Альховик Сергей Александрович. – Гомель, 2004. – 325 с. – Библиогр. : с. 162–172.

11. Аналитический обзор "Российский рынок B2B" [Электрон. ресурс] / вып. III, 2002. - Режим доступа : <http://www.rocit.ru/inform/index.php3?path=ibpartners>.

12. Багатурова, О. С. Математические модели маркетинга (аналитический обзор) / О. С. Багатурова, А. Т. Мамиконов // Автоматика и телемеханика. – 1991. – № 8. – С. 3–35.

13. Барсегян, А. А. Методы и модели анализа данных: OLAP и Data Mining / А. А. Барсегян, М. С. Куприянов, В. В. Степаненко, И. И. Холод. – СПб.: БХВ-Петербург, 2004. – 336 с.: ил.

14. Башаримов, В. В. Выбор функции качества при исследовании генетического алгоритма / В. В. Башаримов, А. И. Якимов // Изв. Белорус. инженерной академии. – 2004. – № 1(17)/1. – С. 164–166.

15. Башаримов, В. В. Математические модели для автоматизированного проектирования рационального раскрытия с применением численных методов / В. В. Башаримов, А. И. Якимов, С. А. Альховик // Современные методы проектирования машин: Респ. межведомств. сб. науч. трудов. – Минск: Технопринт, 2004. – Т. 6, Вып. 2. – С. 90–94.

16. Башаримов, В. В. Решение задач рационального выбора параметров систем в ПТКИ BelSim на основе алгоритма случайного поиска / В. В. Башари-

мов, А. И. Якимов // Интеллектуальные системы (AIS'06) и Интеллектуальные САПР (CAD'06): тр. междунар. науч.-техн. конф. в 3 т., Дивноморск (Россия), 3–10 сент. 2006 г. – М. : Физматлит, 2006. – Т. 1. – С. 358–362.

17. Бигель, Дж. Управление производством / Дж. Бигель : пер. с англ. – М.: Финансы и статистика, 1983. – 248 с.

18. Бодров, В. И. Математические методы принятия решений // В. И. Бодров, Лазарева Т. Я., Мартемьянов Ю. Ф.: Учебное пособие. – Тамбов: Изд-во Тамб. гос. техн. ун-та, 2004. – 124 с.

19. Бодрунов, С. Д. Маркетинг информационных услуг: модели управления / С. Д. Бодрунов, И. В. Максимей. – М.: Луч, 1992. – 127 с.: ил.

20. Борчик, Е. М. Построение плотностей распределения Пирсона для многомодальных выборок с применением кластеризации / Е. М. Борчик, А. И. Якимов, А. И. Степанов, В. В. Башаримов // Информационные технологии. – 2012. – № 12. – С. 30–35.

21. Бригхэм, Ю. Энциклопедия финансового менеджмента / Ю. Бригхэм ; пер. с англ. – М.: Издательство «Дело и сервис», 1998. – 816 с.

22. Букатов, А. А. Программирование многопроцессорных вычислительных систем / А. А. Букатов, В. Н. Дацюк, А. И. Жегуло. – Ростов-на-Дону: Издательство ООО «ЦВВР», 2003. – 208 с.

23. Бурков, В.Н. Теория активных систем: состояние и перспективы / В. Н. Бурков, Д. А. Новиков. – М.: Синтег, 1999. – 128 с.

24. Венцов, Н. Н. Исследование и разработка генетических алгоритмов и автоматов адаптации для повышения эффективности доступа к данным САПР СБИС : автореф. дис. ... канд. техн. наук. – Таганрог: 2006. – 16 с.

25. Воеводин, В. В. Параллельные вычисления / В. В. Воеводин, Вл. В. Воеводин. – СПб.: БХВ-Петербург, 2002. – 608 с.

26. Гаврилов, Д. А. Управление производством на базе стандарта MRP II. Принципы и практика / Д. А. Гаврилов. – СПб.: Питер, 2002. – 352 с.

27. Гладков, Л. А. Генетические алгоритмы / Л. А. Гладков, В. В. Курейчик, В. М. Курейчик / Под ред. В. М. Курейчика. – 2-е изд., испр. и доп. – М.: Физматлит, 2010. – 368 с.

28. Гламаздин, Е. С. Управление корпоративными программами: информационные системы и математические модели / Е. С. Гламаздин, Д. А. Новиков, А.В. Цветков. – М.: ИПУ РАН, 2003. – 159 с.

29. Голубков, Е. П. Прибыль как индикатор эффективности: необходимость комплексного подхода / Е. П. Голубков // Маркетинг в России и за рубежом. – 1998. – № 3. – С.: 37–43.

30. Горчаков, А. А. Компьютерные экономико-математические модели / А. А. Горчаков, И. В. Орлова. – М.: ЮНИТИ, 1995. – 134 с.

31. Гранберг, А. Г. Результаты экспериментальных расчетов по оценке эффективности инвестиционных проектов с использованием межотраслевых межрегиональных моделей / А. Г. Гранберг [и др.] // Регион: экономика и социология. 2010. № 4. С. 45–72.

32. Григорьева, А. А. Математические модели задачи определения конкурентоспособности продукции / А. А. Григорьева, Ю. М. Осипов // Автоматизация и современные технологии. – 1999. – № 4. – С. 36–39.

33. Губко, М. В. Математические модели оптимизации иерархических структур / М. В. Губко. – М.: ЛЕНАНД, 2006. – 264 с.

34. Де Роза, К. Планирование ресурсов, синхронизированное с покупателем (CSRP) // Оценка промышленных инвестиционных проектов [Электрон. ресурс] С.-Петербург / К. Де Роза. Режим доступа: <http://www.cfin.ru/vernikov/mrp/csrp.shtml>.

35. Емельянов, А. А. Компьютерное моделирование / А. А. Емельянов, Е. А. Власова. – М.: МГУЭСИ, 2002. – 90 с.: ил.

36. Жданов, С. А. Экономические модели и методы в управлении / С. А. Жданов. – М.: Издательство «Дело и сервис», 1998. – 176 с.

37. Заложнев, А.Ю. Внутрифирменное управление. Оптимизация процедур функционирования / А. Ю. Заложнев. – М.: ЗАО «ПМСОФТ», 2005. – 290 с.

38. Замятина, Е. Б. Интеграция информационных систем и систем имитационного моделирования на основе многоуровневых моделей / Е. Б. Замятина, Л. Н. Лядова, А. И. Миков, А. И. Якимов // Математика программных систем: межвуз. сб. науч. ст. / Перм. гос. ун-т. – Пермь, 2008. – С. 12–23.

39. Захарченков, К. В. Разработка метода, моделей и технологии оценки эффективности процессов управления в корпоративных информационных системах : автореф. дис. канд. техн. наук : 15.13.10 : защищена 16.12.14 : Захарченков Константин Васильевич. – Брянск, 2014. – 238 с. – Библиогр. : с. 168–183.

40. Згуровський, М. З. Основи системного аналізу / М. З. Згуровський, Н. Д. Панкратова. – Київ: ВНУ, 2007. – 544 с. : іл.

41. Ильягуева, М. А. Теория и методология управления развитием промышленных предприятий на основе применения методов моделирования и информационных технологий : дис. ... д-ра экон. наук : 08.00.05, 08.00.13 / Махачкала : Дагестан. гос. ун-т.-, 2006. – 405 с.

42. Интеграция систем имитационного моделирования и информационных систем на основе онтологий и многоуровневых моделей предметных областей : отчет о НИР (заключ.) / Беларус.-Рос. ун-т; рук. А. И. Якимов ; исполн. : С. А. Альховик [и др.]. – Могилев, 2010. – 84 с. – Библиогр. : с. 65–70. – № ГР 20082063. – Инв. № ХД0894.

43. Иозайтис, В. С. Финансово-экономическое управление исполнением бюджета доходов и расходов в условиях нормативного планирования / В. С. Иозайтис // Управленческий учет и финансы. – 2014. – № 4 (40). – С.258–271.

44. Канаев, К. А. Сравнительный анализ форматов обмена данными, используемых в приложениях с клиент-серверной архитектурой / К. А. Канаев, Е. В. Фалеева, Ю. В. Пономарчук // Фундаментальные исследования. – 2015. – № 2–25. – С. 5569-5572; URL: <https://www.fundamental-research.ru/ru/article/view?id=38464> (дата обращения: 05.07.2016).

45. Кельтон, В. Имитационное моделирование. Классика CS / В. Кельтон, А. Лоу; 3-е изд. – СПб.: Питер; Киев: Издательская группа ВНУ, 2004. – 847 с.: ил.

46. Кириличев, Б. В. Моделирование систем / Б. В. Кириличев. – М.: МГИУ, 2009 – 274 с.
47. Ковалевич, А. А. Исследование стохастических алгоритмов оптимизации для применения в имитационном моделировании систем / А. А. Ковалевич, А. И. Якимов, Д. М. Албкеират // Информационные технологии. – 2011. – №8. – С. 55–60.
48. Колесник, В. Д. Курс теории информации / В. Д. Колесник, Г. Ш. Полтырев. – М.: Наука, 1982. – 416 с.
49. Кугаенко, А. А. Методы динамического моделирования в управлении экономикой / А. А. Кугаенко. – М.: Университетская книга, 2005. – 456 с.
50. Кустарев, В. П. Информационные технологии и системы в управлении социально-экономическими объектами / В. П. Кустарев // Научно-технический вестник информационных технологий, механики и оптики. – 2007. – № 2. – С. 213–215.
51. Ладутько, Н. И. Бухгалтерский учет / Н. И. Ладутько, П. Е. Борисевский, Н. П. Дробышевский, Е. Н. Ладутько; Под общ. ред. Н. И. Ладутько. – 3-е изд., перераб. и доп. – Мн.: «ФУАинформ», 2003. – 824 с.
52. Ланкастер, П. Теория матриц / П. Ланкастер; пер. с англ. – М.: Наука, 1982. – 272 с.
53. Ларман, К. Применение UML и шаблонов проектирования. Введение в объектно-ориентированный анализ и проектирование / К. Ларман. – М.: Издательский дом «Вильямс», 2001. – 496 с.
54. Левчук, В. Д. Программно-технологические комплексы имитации сложных дискретных систем: монография / В. Д. Левчук, И. В. Максимей. – Гомель, ГГУ им. Ф. Скорины, 2006. – 263 с.
55. Леоненков, А. В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose / А. В. Леоненков. – М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2006. – 320 с.: ил.

56. Лешко, В. Анализ показателей бухгалтерской отчетности / В. Лешко // Финансовый директор. – 2006. – № 7. – С. 56–62.
57. Литвинюк, А. Введение в интегрированную среду разработки Eclipse [Электрон. ресурс] / А. Литвинюк. – 2007. – Режим доступа : [http://www.javaportal.ru/java/ide/intro\\_eclipse.html](http://www.javaportal.ru/java/ide/intro_eclipse.html).
58. Макаров, С. С. Разработка методов моделирования и информационного обеспечения при управлении медицинскими учреждениями : автореф. дис. ... д-ра техн. наук. – Таганрог: 2006. – 38 с.
59. Максимей, И. В. Имитационное моделирование на ЭВМ / И. В. Максимей. – М.: Радио и связь, 1988. – 232 с.: ил.
60. Месарович, М. Теория иерархических многоуровневых систем. / М. Месарович, Д. Мако, И. Такахара; пер. с англ. под ред И.Ф. Шахнова. – М.: Мир, 1973. – 344 с.: ил.
61. Михайлов, А. В. Что такое MRP, MRP II, ERP, ERP II, CRM, SCM, CSRP, B2C, B2B? / А. В. Михайлов [Электрон. ресурс]. – 2003. – Режим доступа : <http://www.bcons.ru>.
62. Мишин, С. П. Оптимальные иерархии управления в экономических системах / С. П. Мишин. – М.: ПМСОФТ, 2004. – 190 с.
63. Монахов, В. Язык программирования Java и среда NetBeans / В. Монахов. – СПб.: «БХВ-Петербург», 2011. – 704 с.
64. Нешитой, В. В. Статистический анализ и регулирование технологических процессов на базе обобщенных распределений с параметром сдвига: метод. рекомендации / В. В. Нешитой. – Мн.: БелГИСС, 2001. – 40 с.
65. Новиков, Д. А. Математические модели формирования и функционирования команд / Д. А. Новиков. – М.: Издательство физико-математической литературы, 2008. – 184с.
66. Новиков, Д. А. Механизмы управления динамическими активными системами / Д. А. Новиков, И. М. Смирнова, Т. Е. Шохина. – М.: ИПУ РАН, 2002. – 124 с.



67. Новиков, Д. А. Теория управления организационными системами / Д. А. Новиков; 3-е изд., испр. и дополн. – М.: Издательство физико-математической литературы, 2012. – 604 с.
68. Новиков, Д. А. Структура теории управления социально-экономическими системами / Д. А. Новиков // Управление большими системами. – 2009. – № 24. – С. 216–237.
69. Ногин, В. Д. Принятие решений в многокритериальной среде: количественный подход / В. Д. Ногин. – 2-е изд., испр. и доп. – М.: Физматлит, 2004. – 176 с.
70. Норенков, И. П. Эвристики и их комбинации в генетических методах дискретной оптимизации / И. П. Норенков // Информационные технологии. – 1999. – № 1. – С. 2–5.
71. Олзоева, С. И. Особенности автоматизированного распределения вычислительного процесса для имитационного моделирования систем / С. И. Олзоева // Высокопроизводительные параллельные вычисления на кластерных системах: материалы четвертого междунар. науч.-практич. семинара. – Самара, СГАУ, 2004. – С. 210–214.
72. Орешенков, А. А. Механизм управления предприятием и его развитие в условиях рыночной экономики (на примере предприятий легкой промышленности) : автореф. дис. ... канд. экон. наук. – Мн.: 1995. – 16 с.
73. Орлов, С. А. Технологии разработки программного обеспечения / С. А. Орлов. – СПб.: Питер, 2002. – 464 с.
74. Ортнер, Г. Применение ЭВМ на предприятиях. Критерии оптимальной организации / Г. Ортнер ; пер. с нем. под ред. В. М. Савинкова. – М.: Изд-во «Прогресс», 1975. – 310 с.: ил.
75. Пелих, А. С. Экономико-математические методы и модели в управлении производством / А. С. Пелих, Л. Л. Терехов, Л. А. Терехова. – Ростов н/Д: Феникс, 2005. – 248 с.
76. Подиновский, В. В. Парето-оптимальные решения многокритериальных задач / В. В. Подиновский, В. Д. Ногин. – М.: Наука, 1982. – 256 с.

77. Разработка методов и программных средств интеллектуальной информационно-аналитической системы поддержки принятия решений, интегрированной в комплексную информационную систему промышленного предприятия: отчет о НИР (заключ.) / Белорус.-Рос. ун-т; рук. А. И. Якимов ; исполн. : Е. А. Якимов [и др.]. – Могилев, 2014. – 146 с. – Библиогр. : с. 139–146. – № ГР 20110379. – Инв. № ГБ1112ф.

78. Растрингин, Л. А. Адаптация сложных систем. / Л. А. Растрингин. - Рига: Зинатне, 1981. – 375 с.: ил.

79. Резников, В. Б. Разработка и исследование метода построения программного комплекса моделирования для распределенных систем с многоуровневым представлением сложных объектов : автореф. дис. ... канд. техн. наук. – Таганрог: 2006. – 16 с.

80. Рихтер, Дж. Программирование на платформе Microsoft .NET Framework / Дж. Рихтер ; пер. с англ. – 2-е изд., испр. – М.: Издательско-торговый дом «Русская Редакция», 2003. – 512 с.: ил.

81. Розанова, А. А. К вопросу о пределах применимости информационных технологий в управлении социально-экономическими системами / А. А. Розанова // Современные исследования социальных проблем. – 2015. – № 8. – С. 344–352.

82. Рубцов, С. В. Опыт использования стандарта IDEF0 / С. В. Рубцов // Открытые системы. – 2003. – № 1. – С. 53–56.

83. Рыжиков, Ю. И. Имитационное моделирование: теория и технологии / Ю. И. Рыжиков. – М.: Альянс, 2004. – 384 с.

84. Савицкая, Г. Анализ деловой активности предприятия / Г. Савицкая // Финансовый директор. – 2005. – № 7. – С. 57–65.

85. Савицкая, Г. Анализ интенсивности и эффективности использования основных средств / Г. Савицкая // Финансовый директор. – 2006. - № 10. – С. 38–44.

86. Саймон, Г. Науки об искусственном / Г. Саймон ; пер. с англ. Э. Л. Наппельбаума. – М.: Мир, 1972. – 148 с.: ил.

87. Светуньков, С. Г. Модели спроса и предложения в пространстве цена-объем-доход / С. Г. Светуньков. – Ульяновск: Изд-во УлГУ, 1999. – 173 с.

88. Сенченко, С. А. Совершенствование процесса управления предприятием на основе внедрения и адаптации комплексных информационно-ориентированных систем : автореф. дис. ... канд. экон. наук. – Таганрог: 2006. – 28 с.

89. Сильнова, С. В. Моделирование реализации продукции в условиях рынка / С. В. Сильнова, Г. Р. Валеева // Автоматизация и современные технологии. – 2004. – № 7. – С. 40–46.

90. Симанков, В. С. Генетические алгоритмы и поиск оптимальных решений / В. С. Симанков, В. А. Частикова // Автоматизация и современные технологии. – 2003. – №6. – С. 39–45.

91. Тельнов, Ю. Ф. Реинжиниринг предприятия на основе имитационной модели / Ю. Ф. Тельнов // Корпоративный менеджмент. – 2001. – 4 кв. – С. 37–50.

92. Теория и практика проектирования интегрированных автоматизированных систем управления как многоуровневых иерархических структур: отчет о НИР (заключ.) / МогГТУ; рук. А. И. Якимов ; исполн. Н. И. Подскробко [и др.]. – Могилев, 2001. – 57 с. – Библиогр. : с. 45–49. – № ГР 19962036. – Инв. № ГБ 9618.

93. Терехин, В. И. Эффективность моделирования денежных средств на российских предприятиях / В. И. Терехин, Н. А. Подгорнова // Финансовый менеджмент. – 2011. – №2. – С. 23–31.

94. Троелсен, Э. Язык программирования С# 5.0 и платформа .NET 4.5 / Э. Троелсен. – 6-е изд. – М.: Вильямс, 2013. – 1312 с.

95. Форрестер, Дж. Основы кибернетики предприятия (индустриальная динамика) / Дж. Форрестер : пер. с англ : общ. ред. и предисл. Д. М. Гвишиани. – М.: Изд-во «Прогресс», 1971. – 340 с.: ил.

96. Харламов, А. И. Общая теория статистики: Статистическая методология в изучении коммерческой деятельности / А. И. Харламов [и др.]; под ред. А. А. Спирина, О.Э. Башиной. – М.: Финансы и статистика, 1996. – 296 с.: ил.

97. Холзнер, С. XML. Энциклопедия / С. Холзнер : 2-е изд. – СПб.: Питер, 2004. – 1101 с.: ил.

98. Черноруцкий, И. Г. Методы оптимизации в теории управления . И. Г. Черноруцкий. – СПб. : Питер, 2004. – 256 с.

99. Шилейко, А. В. Введение в информационную теорию систем / А. В. Шилейко, В. Ф. Кочнев, Ф. Ф. Химушин : под ред. А. В. Шилейко. – М.: Радио и связь, 1985. – 280 с.: ил.

100. Шпаковский, Г. И. Программирование многопроцессорных систем в стандарте MPI / Г. И. Шпаковский, Н. В. Серикова. – Мн.: БГУ, 2002. – 323 с.

101. Щепкин, А. В. Внутрифирменное управление (модели и методы) / А. В. Щепкин. – М.: ИПУ РАН, 2001. – 80 с.

102. Эддопс, М. Методы принятия решений / М. Эддопс, Р. Стэнсфилд ; пер. с англ. под ред. И. И. Елисеевой. – М.: Аудит, ЮНИТИ, 1997. – 590 с.

103. Юхневич, И. Н. Оптимизация производственных запасов в условиях перехода к рынку Республики Беларусь: автореф. дис. ... канд. экон. наук. – Мн.: 1995. – 20 с.

104. Якимов, А. И. Автоматизация эксперимента на примере исследования генетического алгоритма / А. И. Якимов, В. В. Башаримов, С. А. Альховик // Автоматизация и современные технологии. – 2006. – № 5. – С. 3–9.

105. Якимов, А. И. Автоматизированная технология прогнозирования эффективности контрактов / А. И. Якимов, С. А. Альховик // Перспективные технологии, материалы и системы: сб. науч. тр. / МГТУ; редкол.: И. С. Сазонов (гл. ред.) [и др.]. – Могилев, 2001. – С. 502–507.

106. Якимов, А. И. Анализ методов построения имитационных моделей корпоративных информационных систем / А. И. Якимов, О. М. Демиденко, Н. Н. Ивкина // Информационные системы и технологии. – 2016. – № 2(94). – С. 40–50.

107. Якимов, А. И. Анализ эффективности контрактов с использованием программного комплекса Contract Analyzer / А. И. Якимов, С. А. Альховик // Маркетинг в России и за рубежом. – 2001. – № 2. – С. 26–33.

108. Якимов, А. И. Выбор стратегии межуровневой координации в сложной производственно-экономической системе на основе имитационного моделирования / А. И. Якимов, С. А. Альховик // Интеллектуальные системы (AIS'06) и

Интеллектуальные САПР (CAD'06): тр. междунар. науч.-техн. конф. в 3 т., Дивноморск (Россия), 3–10 сент. 2006 г. – М. : Физматлит, 2006 – Т. 2. – С. 239–245.

109. Якимов, А. И. Имитационное моделирование в ERP-системах управления / А. И. Якимов, С. А. Альховик. – Минск: Белорус. наука, 2005. – 197 с. : ил.

110. Якимов, А. И. Имитационное моделирование в комплексных информационных системах / А. И. Якимов // Изв. Гомельс. гос. ун-та им. Ф. Скорины. – 2003. – № 3(18). – С. 106–110.

111. Якимов, А. И. Имитационное моделирование в системе принятия решений на промышленном предприятии / А. И. Якимов // Проблемы управления и приложения (техника, производство, экономика): тр. междунар. конф. в 3 т., 16–20 мая 2005 г. – Минск: Ин-т математики НАН Беларуси, 2005. – Т. 3. – С. 161–165.

112. Якимов, А. И. Имитационное моделирование процессным способом эффективности контракта продажи на основе ПТКИ BELSIM / А. И. Якимов // Изв. Гомельс. гос. ун-та им. Ф. Скорины. – 2006. – № 4(37). – С. 139–141.

113. Якимов, А. И. Информационная оценка программных средств для управления экономической деятельностью промышленного предприятия на основе имитационной модели / А. И. Якимов, К. В. Захарченков // Вестн. Брянск. гос. техн. ун-та. – 2014. – № 1(41). – С. 94–101.

114. Якимов, А. И. Информационное обеспечение прогнозирования эффективности контрактов на предприятиях текстильной промышленности / А. И. Якимов, С. А. Альховик, А. И. Степанов // Текстильная промышленность. – 2002. – № 9. – С. 34–35.

115. Якимов, А. И. Итерационная процедура обобщения результатов кластеризации многомерных данных несколькими методами / А. И. Якимов, Е. М. Борчик, В. В. Башаримов // Математическое и имитационное моделирование систем. МОДС 2011 : тез. докл. шестой междунар. науч.-практич. конф., 27–30 июня 2011 г. – Чернигов: ИПММС НАН Украины, 2011. – С. 419–423.

116. Якимов, А. И. К вопросу о поиске рациональных решений в ERP-системе управления промышленным предприятием / А. И. Якимов // Динамические системы: устойчивость, управление, оптимизация: тез. докладов междунар. конф. к 90-летию со дня рождения академика Е. А. Барбашина, Минск, 29 сент.– 4 окт. 2008 г. – Минск: Ин-т математики НАН Беларуси, 2008. – С. 168–170.

117. Якимов, А. И. К вопросу об интеграции программных информационных систем / А. И. Якимов, А. И. Степанов, О. Г. Добромислин // Изв. Белорус. инженерной академии. – 2004. – № 1(17)/2. – С. 60–62.

118. Якимов, А. И. Кластеризация состояний объекта при решении задачи выбора оптимальных технологических режимов / А. И. Якимов, Е. М. Борчик, Е. М. Максимов // Информационные технологии и вычислительные системы. – 2016. – № 4. – С. 33–43.

119. Якимов, А. И. Конструирование интегральной целевой функции производственно-экономической деятельности предприятия / А. И. Якимов // Инфраструктурное обеспечение бизнес-процессов в региональных социально-экономических системах: материалы Всерос. науч.-практ. конф.; Кострома, 15 – 16 апреля 2016 г. – Кострома: КГУ им. Н. А. Некрасова, 2016. – С. 92–96.

120. Якимов, А. И. Математическое и программное обеспечение для поддержки принятия решений при планировании производства продукции на промышленном предприятии / А. И. Якимов, К. В. Захарченков // Современные проблемы науки и образования. – 2014. – № 2. – URL: <http://www.science-education.ru/116-12770> (дата обращения: 16.04.2014).

121. Якимов, А. И. Метод имитационного моделирования многоуровневых иерархических систем / А. И. Якимов // Электронное моделирование. – 2008. – № 5. – С. 69–80.

122. Якимов, А. И. Метод моделирования координирующих решений в многоуровневой иерархической системе / А. И. Якимов // Математическое и имитационное моделирование. МОДС 2007: тез. докл. II-й науч.-практ. конф. с междунар. участием, 25–29 июня 2007 г. – Киев, Ин-т проблем математ. машин и систем НАН Украины, 2007. – С. 111–114.

123. Якимов, А. И. Методика кластеризации целевых функций в ПТКИ BelSim / А. И. Якимов, В. В. Башаримов, Е. М. Борчик // Информационные технологии в профессиональной деятельности и научной работе: материалы Всерос. науч.-практ. конф. с междунар. участием, 18–19 апр. 2008 г.: в 2 ч. – Йошкар-Ола, Марийский гос. техн. ун-т, 2008. – Ч. 2. – С. 203–207.

124. Якимов, А. И. Методы и средства имитационного моделирования производственно-экономических систем / А. И. Якимов // Математичні машини і системи. – 2007. – № 1. – С. 62–66.

125. Якимов, А. И. Моделирование в системе управления химическим предприятием. Задача распределения производственной нагрузки на уровне объединения / А. И. Якимов // Компьютеризация в химической и смежных отраслях промышленности: Сборник материалов междунар. науч.-техн. семинара. – Новополюк: ПГУ, 2006. – С. 52–57.

126. Якимов, А. И. Моделирование в системе управления химическим предприятием. Задача оптимизации распределения производственной нагрузки / А. И. Якимов, С. А. Альховик // Компьютеризация в химической и смежных отраслях промышленности: Сборник материалов междунар. науч.-техн. семинара. – Новополюк: ПГУ, 2006. – С. 44–46.

127. Якимов, А. И. Модернизация программно-технологического комплекса имитации сложных систем BelSim для организации распределенных вычислений / А. И. Якимов // Информатика. – 2008. – №2(18). – С. 137–142.

128. Якимов, А. И. Настройка параметров генетического алгоритма для решения практических оптимизационных задач / А. И. Якимов, С. А. Альховик, В. В. Башаримов // Изв. Белорус. инженерной академии. – 2003. – №1(15)/2. – С. 332–334.

129. Якимов, А. И. О совместном использовании методов кластерного анализа многомерных данных / А. И. Якимов, Е. М. Борчик, В. В. Башаримов // Доклады БГУИР. – 2011. – № 5(59). – С. 95–102.

130. Якимов, А. И. Об исследовании длительности переходного процесса в имитационной модели производственной деятельности промышленного пред-

приятия / А. И. Якимов // Труды БГТУ. Сер. VI, Физ.-мат. науки и информ. – 2008. – Вып. XVI. – С. 116–120.

131. Якимов, А. И. Об одной методике принятия решений в иерархической структуре предприятия / А. И. Якимов // Изв. Гомельс. гос. ун-та им. Ф. Скорины. – №5(44). – 2007. – С. 132–137.

132. Якимов, А. И. Определение классов целевых функций методами кластерного анализа / А. И. Якимов, Е. М. Борчик, В. В. Башаримов // Веснік Гродзенскага дзяржаўнага ўніверсітэта імя Янкі Купалы. – 2012. – № 1(126). – С. 80–89.

133. Якимов, А. И. Оптимизационная задача управления построением имитационной модели в корпоративной информационной системе / А. И. Якимов, Е. А. Якимов, В. И. Аверченков, Н. Н. Ивкина // Вестник Брянского государственного технического университета. – 2016. – № 2 (50). – С. 207–214.

134. Якимов, А. И. Особенности интеграции имитационной модели объекта с информационной системой предприятия / А. И. Якимов // Изв. Гомельс. гос. ун-та им. Ф. Скорины. – №5(32). – 2005. – С. 90–92.

135. Якимов, А. И. Применение имитационного моделирования в современных концепциях управления предприятием / А. И. Якимов // Изв. Гомельс. гос. ун-та им. Ф. Скорины. – 2004. – № 4(25). – С. 77–80.

136. Якимов, А. И. Применение математической теории систем / А. И. Якимов // Весці АН Беларусі. Сер. фіз-мат. навук. – № 3. – 1999. – С. 132.

137. Якимов, А. И. Программно-технологический комплекс имитации сложных систем «BelSim 2» : свидетельство о регистрации компьютерной программы № 025 / А. И. Якимов, С. А. Альховик, В. В. Башаримов, К. В. Захарченков. – Минск: НЦИС, 2008. – Заявка № С20080010. – Дата подачи: 04.04.2008.

138. Якимов, А. И. Программный комплекс для прогнозирования финансово-экономического состояния и оптимизации управления предприятием / А. И. Якимов, С. А. Альховик // Наука и инновации в регионах Беларуси: Материалы республ. науч.-практ. конф. / Сост. В. С. Драгун, В. К. Пивоваров, О. О. Станюленис. – Могилев: ИТМ НАН Беларуси, 2002. – С. 146–149.



139. Якимов, А. И. Программный комплекс имитации производственно-экономической деятельности «ShagoVitaPro» : свидетельство о регистрации компьютерной программы № 308 / А. И. Якимов, К. В. Захарченков, С. А. Альховик, И. А. Емельянов, И. И. Мельников. – Минск: НЦИС, 2011. – Заявка № С20110027. – Дата подачи: 14.04.2011.

140. Якимов, А. И. Процедура выделения классов целевых функций с обобщением результатов кластеризации несколькими методами / А. И. Якимов, Е. М. Борчик, В. В. Башаримов // Вычислительный интеллект (результаты, проблемы, перспективы): материалы 1-й Междунар. науч.-техн. конф. (10–13 мая 2011 г., Черкассы). – Черкассы: Маклаут, 2011. – С. 502–503.

141. Якимов, А. И. Разработка технологии решения задач оптимизации моделей различных объектов машиностроения / А. И. Якимов, В. В. Башаримов, С. А. Альховик // Вестн. Могилевс. гос. техн. ун-та. – №1(6). – 2004. – С. 196–199.

142. Якимов, А. И. Статистический анализ в программно-технологическом комплексе имитации производственно-экономических систем / А. И. Якимов // Математические методы в технике и технологиях – ММТТ-29: сб. трудов XXIX Междунар. науч. конф.; Санкт-Петербург, 31 мая – 3 июня 2016 г.: в 12 т. Т.3. / под общ. ред. А.А. Большакова. – Саратов: Саратов. гос. техн. ун-т; Санкт-Петербург: СПбГТИ(ТУ), СПбПУ, СПИИРАН; Самара: Самарск. гос. техн. ун-т, 2016. – С. 91–93.

143. Якимов, А. И. Технология имитационного моделирования систем управления промышленных предприятий : монография / А. И. Якимов. – Могилев: Беларус.-Рос. ун-т, 2010. – 304 с.

144. Якимов, А. И. Технология имитационного моделирования сложных многоуровневых систем / А. И. Якимов // Моделирование-2006: Сб. трудов междунар. конф. – К.: Ин-т проблем моделир. в энергетике им. Г. Е. Пухова НАН Украины, 2006. – С. 455–460.

145. Якимов, Е. А. Исследование временных рядов с равномерным, нормальным и экспоненциальным шумом с помощью SSA-метода / Е. А. Якимов,

В. Г. Замураев, А. И. Якимов // Вестн. Брест. гос. техн. ун-та. – 2010. – № 5(65). – С. 100–104.

146. Якимов, Е. А. Метод и средство повышения эффективности эксплуатации имитационных моделей функционирования предприятия : автореф. дис. канд. техн. наук : 15.13.18 : защищена 24.05.12 : Якимов Евгений Анатольевич. – Гомель, 2012. – 270 с. – Библиогр. : с. 140–154.

147. Якимов, Е. А. О преобразовании методом сингулярного спектрального анализа последовательностей данных с равномерным шумом / Е. А. Якимов, В. Г. Замураев, А. И. Якимов // Информатика. – 2011. – № 1(29). – С. 52–61.

148. Якимов, Е. А. Построение модели числовых данных методом сингулярного спектрального анализа в имитационном моделировании / Е. А. Якимов, Е. М. Борчик, А. И. Якимов // Вестн. Иркутск. гос. техн. ун-та. – 2013. – № 2 (73). – С. 45–51.

149. Якимов, Е. А. Программный модуль анализа сингулярного спектра числовых последовательностей данных «BelSim2#.SSA» : свидетельство о регистрации компьютерной программы № 518 / Е. А. Якимов, Д. М. Албкеират, Н. М. Чапаров, А. И. Якимов. – Минск: НЦИС, 2013. – Заявка № С20130042. – Дата подачи: 23.05.2013.

150. Якимов, Е. А. Сингулярный спектральный анализ последовательностей данных на этапе эксплуатации имитационной модели / Е. А. Якимов, О. М. Демиденко, А. И. Якимов // Вестн. Брянск. гос. техн. ун-та. – 2013. – № 1 (37). – С. 95–101.

151. Якимов, И. М. Имитационное моделирование вероятностных объектов в системе Flexsim / И. М. Якимов, А. П. Кирпичников // Вестник Казанского технологического университета. – 2016. – Т. 19. – № 21. – С. 170–173.

152. Al-Aomar R. A Robust Simulation-Based Multicriteria Optimization Methodology / R. Al-Aomar // Proceedings of 2002 Winter Simulation Conference. – San Diego: IEEE Press, 2002. – pp. 1931–1939.

153. Al-Khamis, T. M. Simulation-based Optimization for Repairable Systems Using Particle Swarm Algorithm / T. M. Al-Khamis, M. A. Ahmed // Proceedings of 2005 Winter Simulation Conference. – Orlando: IEEE Press, 2005. – pp. 857–861.

154. Alkhovik, S. A. Software package for industrial and economic activity simulation “ShagoVitaPro” / S. A. Alkhovik, I. I. Melnikov, A. I. Yakimov, K. V. Zakharchenkov // Belarus-Korea science and technology seminar; June 27, 2011. – Minsk: BNTU, 2011. – P. 103.

155. Al-Shihabi, S. Subcontracting in a Make-to-stock Production System, IPA Gradients for a SFM / S. Al-Shihabi // Proceedings of 2005 Winter Simulation Conference. – Orlando: IEEE Press, 2005. – pp. 819–824.

156. April, J. Practical Introduction to Simulation Optimization / J. April, F. Glover, J. P. Kelly, M. Laguna // Proceedings of 2003 Winter Simulation Conference. – New Orleans: IEEE Press, 2003. – pp. 71–78.

157. Averchenkov, V. Hierarchical Deep Learning: A Promising Technique for Opinion Monitoring and Sentiment Analysis in Russian-Language Social Networks / V. Averchenkov, D. Budylskii, A. Podvesovskiy, A. Averchenkov, M. Rytov, A. Yakimov // Proceedings of the First Conference, CIT&DS 2015: Creativity in intellectual technologies & data science. Volgograd, Russia, September 15 – 17, 2015. – Springer International Publishing, Switzerland. – 2015. – pp. 583–592.

158. Banks, J. The Future of the Simulation Industry / J. Banks, J. C. Hagan, P. Lendermann, C. McLean, E. H. Page, C. D. Pegden, O. Ulgen, J. R. Wilson // Proceedings of 2003 Winter Simulation Conference. – New Orleans: IEEE Press, 2003. – pp. 2033–2043.

159. Barton, R. R. Panel: Simulation – Past, Present and Future / R. R. Barton, P. A. Fishwick, J. O. Henriksen, R. G. Sargent, J. M. Twomey // Proceedings of 2003 Winter Simulation Conference. – New Orleans: IEEE Press, 2003. – pp. 2044–2050.

160. Bengtsson, N. Input Data Management Methodology for Discrete Event Simulation / N. Bengtsson, G. Shao, B. Johansson, T. Lee, S. Leong, A. Skoogh, C. McLean [Электрон. ресурс] // Proceedings of 2009 Winter Simulation Conference. –

2009. – P. 1335–1344. – Режим доступа : <http://www.informs-sim.org/wsc09papers/125.pdf>.

161. Boesel, J. Future of simulation optimization / J. Boesel, R. O. Bowden, F. Glover, J. P. Kelly, E. Westwig // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 1466–1469.

162. Buchholz, P. Enhancing Evolutionary Algorithms with Statistical Selection Procedures for Simulation Optimization / P. Buchholz, A. Thümmler // Proceedings of 2005 Winter Simulation Conference. – Orlando: IEEE Press, 2005. – pp. 842–852.

163. Buss, A. Component based Simulation Modeling with Simkit / A. Buss // Proceedings of 2002 Winter Simulation Conference. – San Diego: IEEE Press, 2002. – pp. 243–249.

164. Carson, II J. S. Introduction to Modeling and Simulation / J. S. Carson, II // Proceedings of 2003 Winter Simulation Conference. – New Orleans: IEEE Press, 2003. – pp. 7–13.

165. Carson, II J. S. Model Verification and Validation / J. S. Carson, II // Proceedings of 2002 Winter Simulation Conference. – San Diego: IEEE Press, 2002. – pp. 52–58.

166. De Jong, K. A. An analysis of behavior of a class of genetic adaptive systems / K. A. De Jong // Doctoral dissertation, University of Michigan, University Microfilms № 76–9381, 1975.

167. Ding, H. A Simulation-Optimization Approach Using Genetic Search for Supplier Selection / H. Ding, L. Benyoucef, X. Xie // Proceedings of 2003 Winter Simulation Conference. – New Orleans: IEEE Press, 2003. – pp. 1260–1267.

168. Drew, S. Quasi-Monte Carlo Strategies for Stochastic Optimization / S. Drew, T. Homem-de-Mello // Proceedings of 2006 Winter Simulation Conference. – Monterey: IEEE Press, 2006. – pp. 774–782.

169. Drewry, D. T. An Optimization-Based Multi-Resolution Simulation Methodology / D. T. Drewry, P. F. Reynolds Jr., W. R. Emanuel // Proceedings of 2002 Winter Simulation Conference. – San Diego: IEEE Press, 2002. – pp. 467–475.

170. Eskandari, H. Multiobjective Simulation Optimization Using an Enhanced Genetic Algorithm / H. Eskandari, L. Rabelo, M. Mollaghasemi // Proceedings of 2005 Winter Simulation Conference. – Orlando: IEEE Press, 2005. – pp. 833–841.

171. Fishwick, P. A. Using XML for Simulation Modeling / P. A. Fishwick // Proceedings of 2002 Winter Simulation Conference. – San Diego: IEEE Press, 2002. – pp. 616–622.

172. Fowler, Martin. UML distilled: a brief guide to the standard object modeling language / Martin Fowler, Kendall Scott. – 2nd ed. – Addison Wesley Longman, 2000. – 195 p.

173. Fu, M. C. Simulation Optimization: A Review, New Developments, and Applications / M. C. Fu, F. Glover, J. April // Proceedings of 2005 Winter Simulation Conference. – Orlando: IEEE Press, 2005. – pp. 83–95.

174. Geany: интегрированная среда разработки приложений [Электрон. ресурс] – 2009. – Режим доступа : <http://www.interface.ru/home.asp?artId=21809>.

175. Gehlsen, B. A Framework for Distributed Simulation Optimization / B. Gehlsen, B. Page // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 508–514.

176. Golenko-Ginzburg, D. Hierarchical control models of man-machine production systems, Vol. 1: Fundamentals: Monograph/ Scientific editor Prof. V. N. Burkov. - Science Book Publishing House, Lorman, MS, USA, 2012. – 268 p.

177. Golenko-Ginzburg, D. Hierarchical control models of man-machine production systems, Vol. 2: Control algorithms and practical applications: Monograph / Scientific editor Prof. V. N. Burkov. – Science Book Publishing House, Lorman, MS, USA, 2013. – 328 p.

178. Hasan, H. Information systems development as a research method / H. Hasan // Australasian Journal of Information Systems. - Special Issue 2003/2004. – pp. 4–13.

179. Hedlund, H. E. A Genetic Algorithm and an Indifference-Zone Ranking and Selection Framework for Simulation Optimization / H. E. Hedlund, M. Molaghasemi // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 417–421.

180. Hierarchical mathematics: theory of sway / S. Novikava, S. Gancharova, A. Burawkin, M. Mahoniok, W. Nowik, A. Yakimaw, K. Miatliuk // Proceedings of 8th IFAC / IFORS / IMACS / IFIP Symposium on Large Scale Systems: Theory and Ap-

plications, LSS'98., Rio Patras (Greece), July 15–17, 1998. – Rio Patras: University of Patras, 1998. – pp. 480–487.

181. Hu J. Stochastic Optimization Using Model Reference Adaptive Search / J. Hu, M. C. Fu, S. I. Marcus // Proceedings of 2005 Winter Simulation Conference. – Orlando: IEEE Press, 2005. – pp. 811–818.

182. Hybinette, M. SASSY: A Design for a Scalable Agent-Based Simulation System Using a Distributed Discrete Event Infrastructure / M. Hybinette, E. Kraemer, Y. Xiong, G. Matthews, J. Ahmed // Proceedings of 2006 Winter Simulation Conference. – Monterey: IEEE Press, 2006. – pp. 926–933.

183. IDEF0. “FIPS Integration Definition for Function Modeling (IDEF0),” Federal Information Processing Standards Publication 183, Computer Systems Laboratory, National Institute of Standards and Technology. 1993.

184. Jakimow, A. I. Kompleks programowo-technologiczny BelSim ds. realizacji funkcji modelowania w systemach ERP / A. I. Jakimow [Электрон. ресурс] Białorusko-Rosyjski Państwowy Uniwersytet, Mogilew. – 2004. Режим доступа : <http://www.belembassy.org/poland/pol/index.php?id=575>.

185. Kirkpatrick, S. Optimization by Simulated Annealing / S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi // Science. – №220. – 1983. – pp. 671–680.

186. Law, A. M. Simulation-Based Optimization / A. M. Law, M. G. McComas // Proceedings of 2002 Winter Simulation Conference. – San Diego: IEEE Press, 2002. – pp. 41–44.

187. L'Ecuyer, P. Quasi-Monte Carlo Methods for Simulation / P. L'Ecuyer // Proceedings of 2003 Winter Simulation Conference. – New Orleans: IEEE Press, 2003. – pp. 81–89.

188. Leonard, J. A model of project and organizational dynamics / J. Leonard // Australasian Journal of Information Systems. – Vol. 17. – No 2. – 2012. – pp. 5–22.

189. Lobo, F. G. The parameter-less genetic algorithm: Rational and automated parameter selection for simplified genetic algorithm operation / F. G. Lobo . – IlliGAL Report No 2000030, July 2000 [Электрон. ресурс]. – 2001. Режим доступа : <http://www.illigal.ge.uiuc.edu>.

190. Maryak, J. L. Global Random Optimization by Simultaneous Perturbation Stochastic Approximation / J. L. Maryak, D. C. Chin // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 307–312.

191. McDonnel, L. Business driving in hierarchical systems / L. McDonnel, K. Miatliuk, S. Gancharova // Proceedings of 8th IFAC / IFORS / IMACS / IFIP Symposium on Large Scale Systems: Theory and Applications, LSS'98. – Greece, Rio Patras, University of Patras, 1998. – P.: 766–771.

192. McLean, C. Simulation Standards: Current Status, Needs, and Future Directions / C. McLean, S. K. Leong, C. Harrell, P. M. Zimmerman, R. F. Lu // Proceedings of 2003 Winter Simulation Conference. – New Orleans: IEEE Press, 2003. – pp. 2019–2026.

193. Mehta, N. Information technology and knowledge in software development teams: The role of project uncertainty / N. Mehta, D. Hall, T. Byrd // Information & Management. – 2014. – Vol. 51. – № 4. – pp. 417–429

194. Modarres, M. Predicting and Improving Complex Business Processes: Values and Limitations of Modeling and Simulation Technologies / M. Modarres // Proceedings of 2006 Winter Simulation Conference. – Monterey: IEEE Press, 2006. – pp. 598–603.

195. Nordgren, W. B. Flexsim Simulation Environment / W. B. Nordgren // Proceedings of 2002 Winter Simulation Conference. – San Diego: IEEE Press, 2002. – pp. 250–252.

196. Novikov, D. Control methodology / D. Novikov. – New York: Nova Science Publishing, 2013. – 76 p.

197. Ólafsson, S. Towards a Framework for Black-Box Simulation Optimization / S. Ólafsson, J. Kim // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 300–306.

198. Onggo, S. A Dynamic Business Model for Component-Based Simulation Software / S. Onggo, D. Soopramanien, M. Pidd // Proceedings of 2006 Winter Simulation Conference. – Monterey: IEEE Press, 2006. – pp. 954–959.

199. O'Reilly, J. Introduction to AweSim / J. O'Reilly [Электрон. ресурс] Proceedings of 2002 Winter Simulation Conference. – 2002. – pp. 221–224. – Режим доступа : <http://www.informs-sim.org/wsc02papers/028.pdf>.

200. Petredis, V. Varying quality function in genetic algorithms and the cutting problem / V. Petredis, S. Kasarlis // Proceedings of 8th IFAC / IFORS / IMACS / IFIP Symposium on Large Scale Systems: Theory and Applications, LSS'98. – Greece, Rio Patras, University of Patras, 1998. – P. 766–771.

201. Pichitlamken, J. Selection-of-the-Best Procedures for Optimization Via Simulation / J. Pichitlamken, B. L. Nelson // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 401–407.

202. Pllana, S. UML based Modeling of Performance Oriented Parallel and Distributed Applications / S. Pllana, T. Fahringer // Proceedings of 2002 Winter Simulation Conference. – San Diego: IEEE Press, 2002. – pp. 497–505.

203. Quesnel, F. Sheduling of Large-scale Virtualized Infrastructures : Toward Cooperative Management [Электрон. ресурс] / F. Quesnel. – E-book: John Wiley&Sons, 2014. – Режим доступа : <https://books.google.by/books>.

204. Reeves, C. R. Genetic algorithms for the Operations Researcher / C. R. Reeves // Journal on Computing. – 1997. – № 9. – P. 231–250.

205. Reeves, C. R. Genetic algorithms. Modern heuristic techniques for combinatorial problems / C. R. Reeves // Blackwell Scientific Publications. – Oxford, 1993. – pp. 151–196.

206. Robinson, S. A Statistical Process Control Approach for Estimating the Warm-up Period / S. Robinson // Proceedings of 2002 Winter Simulation Conference. – San Diego: IEEE Press, 2002. – pp. 439–446.

207. Robinson, S. Automated Analysis of Simulation Output Data / S. Robinson // Proceedings of 2005 Winter Simulation Conference. – Orlando: IEEE Press, 2005. – pp. 763–770.

208. Robinson, S. Conceptual modeling for simulation: issues and researches requirements / S. Robinson // Proceedings of 2006 Winter Simulation Conference. – Monterey: IEEE Press, 2006. – pp. 792–800.



209. Robinson, S. Modelling and Improving Human Decision Making with Simulation / S. Robinson, T. Alifantis, R. Hurrion, J. Edwards, J. Ladbrook, T. Waller // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 913–920.

210. Roehl, M. Flexible Integration of XML into Modeling and Simulation Systems / M. Roehl, A. M. Uhrmacher // Proceedings of 2005 Winter Simulation Conference. – Orlando: IEEE Press, 2005. – pp. 1813-1820.

211. Saad, S. M. Simulation of Distributed Manufacturing Enterprises: A New Approach / S. M. Saad, T. Perera, R. Wickramarachchi // Proceedings of 2003 Winter Simulation Conference. – New Orleans: IEEE Press, 2003. – pp. 1168–1173.

212. Schmeiser, B. W. Some Myths and Common Errors in Simulation Experiments / B. W. Schmeiser // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 39–46.

213. Schriber, T. J. Inside Discrete-Event Simulation Software: How it Works and Why it Matters / T. J. Schriber, D. T. Brunner // Proceedings of 2005 Winter Simulation Conference. – Orlando: IEEE Press, 2005. – pp. 167–177.

214. Shapiro, A. Monte Carlo Simulation Approach to Stochastic Programming / A. Shapiro // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 428–431.

215. Sturm, J. Developing XML Solutions [Электрон. ресурс] / J. Sturm. – Microsoft Press, 2000. – 1 CD ROM.

216. Szechtman, R. Constrained Monte Carlo and the Method of Control Variates / R. Szechtman, P. W. Glynn // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 394–400.

217. Wainer, G. A. Modeling and Simulation of Complex Systems with Cell-DEVS / G. A. Wainer [Электрон. ресурс] Proceedings of 2004 Winter Simulation Conference. – 2004. – pp. 49–60. – Режим доступа : <http://www.informs-sim.org/wsc04papers/007.pdf>.

218. Weisfeld, M. The Object-Oriented Thought Process / M. Weisfeld. – 4th ed. – Addison-Wesley Professional, 2013. – 336 p.

219. Whitman, L. A Taxonomy of a Living Model of the Enterprise / L. Whitman, K. Ramachandran, V. Ketkar // Proceedings of 2001 Winter Simulation Conference. – New Jersey: IEEE Press, 2001. – pp. 848–855.

220. Wolpert, D. H. [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=1198599](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1198599)No Free Lunch Theorems for Optimization / D. H. Wolpert, W. G. Macready // Evolutionary Computation, IEEE Transactions on. – V. 1, № 1. – 1997. – pp. 67–82.

221. Yakimov, A. I. Computer Data Analysis in Modeling and Optimization of Manufacturing Process Control System / A. I. Yakimov, S. A. Alkhovik // Computer Data Analysis and Modeling : Complex Stochastic Data and Systems : Proc. of the Eighth Intern. Conf., Minsk, Sept. 11-15, 2007. In 2 vol. Vol 2. – Minsk : Publ. center BSU, 2007. – pp. 204–207.

222. Yamane, S. Meta-Level Control Architecture for Massively Multiagent Simulations / S. Yamane // Proceedings of 2006 Winter Simulation Conference. – Monterey: IEEE Press, 2006. – pp. 889–896.

223. Zhao, W. A Framework for Configurable Hierarchical Simulation in a Multiple-User Decision Support Environment / W. Zhao, A. Verbraeck // Proceedings of 2005 Winter Simulation Conference. – Orlando: IEEE Press, 2005. – pp. 327–335.

224. Zhao, Y.C.H.Q.C. [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=1198599](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1198599) The No Free Lunch Theorems: Complexity and Security / Y.C.H.Q.C. Zhao, D. L. Pepyne // Automatic Control, IEEE Transactions on. – V. 48, № 5. – 2003. – pp. 783–793.

225. Zülch, G. Modelling and Simulation of Human Decision-Making in Manufacturing Systems / G. Zülch // Proceedings of 2006 Winter Simulation Conference. – Monterey: IEEE Press, 2006. – pp. 947–953.

## ПРИЛОЖЕНИЕ А. Аксиоматический принцип Парето для оценки эффективности принятия решений

Принцип Парето применяется не для всех многокритериальных задач. В. Д. Ногиным определен класс задач многокритериального выбора, для которого применение принципа Парето является обоснованным [69].

Задача многокритериального выбора решается в том случае, если имеется множество решений  $X$  (вариантов) и необходимо выбрать такое решение, которое будет принято для реализации в системе управления. Это решение, указанное среди элементов множества  $X$ , называют выбранным решением. Если выбирается более одного решения, то формируется множество таких решений  $Sel X$ :

$$Sel X \subset X. \quad (A.1)$$

В задаче многокритериального выбора должны быть определены функции  $f_1, f_2, \dots, f_m \mid m \geq 2$  на области определения  $X$ . Такие функции называют целевыми функциями. В зависимости от предметной области их могут называть критериями оптимальности или критериями эффективности. Иногда их именуют показателями или количественными критериями качества. Так как имеется несколько функций  $f_1, f_2, \dots, f_m$  по условию задачи, то их можно представить в виде векторного критерия  $f = (f_1, f_2, \dots, f_m)$ . Важно отметить, что областью значений векторного критерия  $f = (f_1, f_2, \dots, f_m)$ , являются действительные числа в  $m$ -мерном пространстве  $R^m$ , которое называют критериальным пространством или иначе пространством оценок. Всякое действительное значение критерия  $f$  для некоторого решения  $x$  из области определения  $X$  именуют векторной оценкой  $f(x) = (f_1(x), f_2(x), \dots, f_m(x)) \in R^m \mid x \in X$ . Такая оценка может быть определена для любых решений  $x \in X$ , образуя множество оценок [69]:

$$Y = f(X) = \{y \in R^m \mid y = f(x), x \in X\}. \quad (A.2)$$

При выборе нескольких решений из множества  $X$ , образующих множество  $Sel X \subset X$ , можно рассматривать оценки для такого подмножества, соответственно получаем:

$$\text{Sel } Y = f(\text{Sel } X) = \{y \in R^m \mid y = f(x), x \in \text{Sel } X\}. \quad (\text{A.3})$$

Рассмотрим два возможных решения  $x'$  и  $x''$ . Предположим, что координатор (лицо, принимающее решение (ЛПР)) выбирает (отдает предпочтение) первое из них, тогда записывают:  $x' \succ_X x''$ . Знак  $\succ_X$  именуют отношением предпочтения [69]. Отношение предпочтения обладает следующими свойствами:

антирефлексивности

$$\forall x \in X \neg \exists x \succ_x x; \quad (\text{A.4})$$

антисимметричности

$$\forall x', x'' \in X \ x' \succ_x x'' \& x'' \succ_x x' \Rightarrow x' = x''; \quad (\text{A.5})$$

и транзитивности

$$\forall x', x'', x''' \in X \ x' \succ_x x'' \& x'' \succ_x x''' \Rightarrow x' \succ_x x'''. \quad (\text{A.6})$$

Поскольку множество решений  $X$  целевой функцией  $f$  отображается в множество векторов  $Y$ , то на этом множестве  $Y$  также можно рассматривать отношение предпочтения, обозначаемое  $\succ_Y$ . Тогда имеем:

$$f(x') \succ_Y f(x'') \Leftrightarrow x' \succ_X x'' / \forall x', x'' \in X. \quad (\text{A.7})$$

Соответственно можно утверждать, что вектор  $y' = f(x')$  является более предпочтительным, чем вектор  $y'' = f(x'')$  :

$$y' \succ_Y y'' \quad (\text{A.8})$$

тогда и только тогда, если решение  $x'$  является предпочтительнее решения  $x''$ :

$$x' \succ_X x''. \quad (\text{A.9})$$

В. Д. Ногиным на основе аксиоматического подхода строго сформулирован принцип Парето и установлено, при выполнении каких требований *применение этого принципа оправдано* [69].

*Аксиома 1 исключения доминируемых решений* утверждает, что всякое множество  $\text{Sel } X \subset X$  не должно иметь ни одного  $x'' \in X$ , которое предпочтительнее решения  $x' \in X$ :

$$\forall x', x'' \in X \exists x' \succ_X x'' \Rightarrow x'' \notin X. \quad (\text{A.10})$$

Таким образом, при выполнении требования аксиомы 1 из множества возможных решений  $X$  должны быть исключены все доминируемые решения, образующих подмножество  $\text{Dom } X \subset X$ . Тогда множество  $\text{Ndom } X = X \setminus \text{Dom } X$  называют множеством недоминируемых решений, которое формально определяется следующим равенством:

$$\text{Ndom } X = \{x^* \in X \mid \neg \exists x \in X, x \succ_X x^*\}. \quad (\text{A.11})$$

Таким образом, для множества решений  $\text{Sel } X$  при выполнении требования аксиомы 1, становится верным включение:

$$\text{Sel } X \subset \text{Ndom } X, \quad (\text{A.12})$$

которое определяет, что решения в соответствии с аксиомой 1 должны выбираться только из множества  $\text{Ndom } X$ .

*Аксиома 2 продолжения отношения предпочтения* утверждает, что если существует отношения  $\succ_X$  предпочтения на множестве  $X$ , то оно справедливо и в пространстве  $R^m$  критериев, причем это продолжение  $\succ_Y$  обладает свойствами антирефлексивности и транзитивности.

Требование в соответствии с аксиомой 2 заключается в том, что у ЛПР имеется возможность сравнивать оценки по отношению предпочтительности. При этом может выполняться только одно соотношение из следующих трех возможных:

$$\text{а) } y' \succ y'';$$

$$\text{б) } y'' \succ y';$$

$$\text{в) } \neg \exists y' \succ y'' \vee y'' \succ y'.$$

Поскольку в задаче многокритериального выбора отношение предпочтения и критерии качества представляют интересы одного и того же ЛПР, то они должны согласовываться между собой. Критерий  $f_i$  полагают согласованным с

отношением предпочтения  $\succ_Y$ , если для  $\forall y', y'' \in R^m$  справедливо следующее утверждение:

$$\begin{aligned} y' = (y'_1, \dots, y'_{i-1}, y'_i, y'_{i+1}, \dots, y'_m), y'' = (y''_1, \dots, y''_{i-1}, y''_i, y''_{i+1}, \dots, y''_m), y' > y'' \Rightarrow \\ \Rightarrow y' \succ y''. \end{aligned} \quad (\text{A.14})$$

Взаимосвязь отношения предпочтения данного ЛПР с критериями качества выражается следующим требованием.

*Аксиома 3 согласования критериев с отношением предпочтения* утверждает, что каждый из критериев  $f_1, f_2, \dots, f_m$  является согласованным с отношением  $\succ_Y$  предпочтения.

ЛПР заинтересован в возможно больших значениях всех компонент критерия  $f$ , что представляется в виде аксиомы Парето.

*Аксиома Парето относительно решений* утверждает, что для любых двух решений  $x', x'' \in X$  таких, что справедливо неравенство  $f(x') \geq f(x'')$ , является справедливым соотношением  $x' \succ_X x''$ .

При этом неравенство  $f(x') \geq f(x'')$  означает, что выполняются отношения  $f_i(x') > f_i(x'') \vee f_i(x') = f_i(x'')$  для всех компонент  $i = 1, 2, \dots, m$ , при условии  $f(x') \neq f(x'')$ .

Если для любых двух решений из множества  $X$  ( $x', x'' \in X$ ) справедливо неравенство  $f(x') \geq f(x'')$ , то в соответствии с аксиомой Парето первое решение  $x'$  предпочтительнее второго решения  $x''$ , т. е.  $x' \succ_X x''$ . Тогда по требованию аксиомы 1 решение  $x''$  не должно выбираться и его необходимо удалить из перечня вариантов для принятия решений. Удаление подобных решений приводит к множеству Парето-оптимальных решений  $P_f(X)$ :

$$P_f(X) = \{x^* \in X / \neg \exists x \in X, f(x) > f(x^*)\}. \quad (\text{A.15})$$

Если выполняются требования аксиом 2 и 3, то для множества  $\text{Ndom } X$  справедливо включение  $\text{Ndom } X \subset P_f(X)$ .

Если выполняются требования аксиом 1 - 3 для множества решений  $\text{Sel } X$  таких, что  $\text{Sel } X \neq \emptyset$ , то справедливо следующее включение:  $\text{Sel } X \subset P_f(X)$ . Это

включение выражает принцип Парето: если действия ЛПР соответствуют аксиомам 1 - 3, то выбранные решения являются Парето-оптимальными.

В этом случае вектор  $f(x^*)$  для решения  $x^*$ , являющегося Парето-оптимальным, также является Парето-оптимальным. Парето-оптимальные вектора  $f(x^*)$  образуют множество  $P(Y)$  такое, что выполняется равенство

$$P(Y) = f(P_f(X)) = \{f(x^*) \in Y / \forall x^* \in P_f(X)\}. \quad (\text{A.16})$$

Множество возможных векторов  $Y$  на практике, как правило, является конечным, т. к. конечным является множество решений  $X$ . Тогда, при выполнении условия  $X \neq \emptyset$ , найдется хотя бы одно решение  $x^*$ , являющееся Парето-оптимальным, и, следовательно, определится один вектор из множества  $Y$ , также являющийся Парето-оптимальным.

**ПРИЛОЖЕНИЕ Б. Алгоритмы компонентов базовой имитационной модели**

Отдельные процессы, соответствующие основным компонентам базовой имитационной модели, рассмотрены более подробно.

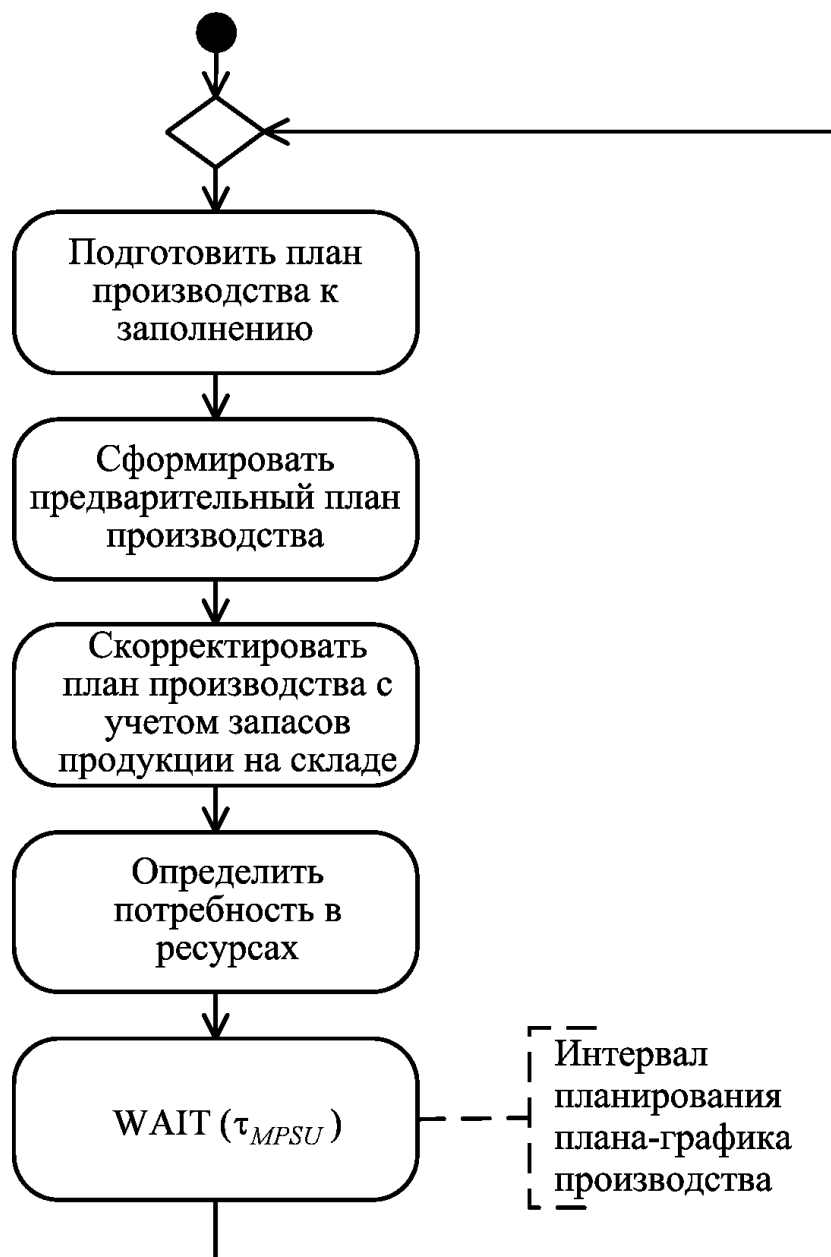
*Процесс «Планирование производства»* реализует последовательность функциональных действий, необходимых для формирования плана-графика производства заказанной продукции. Планирование выполняется на следующий временной период  $\tau_{MPS}$  и отдельно по интервалам  $\tau_{MPSU}$  периода  $\tau_{MPS}$ . В начале моделирования подготавливаются структуры данных разрабатываемого плана производства: вводится новая ячейка и обнуляются данные за предшествующий период ( $\tau_{MPS} - \tau_{MPSU}$ ). Затем создается новый план производства для следующего интервала планирования, определяющий выпуск определенного вида продукции с учетом объемов отгрузок готовой продукции в соответствии с условиями контрактов. Сформированный план подвергается коррекции – сокращаются объемы выпуска на величину объемов готовой продукции, хранящейся на складе. Далее определяются объемы материальных ресурсов, требуемых для выполнения плана в каждом интервале  $\tau_{MPSU}$ . В завершение процесса выполняется оператор синхронизации WAIT( $\tau_{MPSU}$ ), который назначает время следующей активизации, соответствующей следующему временному интервалу  $\tau_{MPSU}$  планирования [109].

Схема алгоритма процесса «Планирование производства» приведена на рисунке Б.1.

*Процесс «Управление производством»* (рисунок Б.2) моделирует действия по определению времени и объемов начала производства по каждому виду продукции. На основе плана производства и длительности технологического цикла с учетом объемов незавершенного производства определяется количество продукции, изготовление которой необходимо начать в текущем периоде. В случае ненулевого значения этой величины выполняется оператор CREATE\_PROCESS

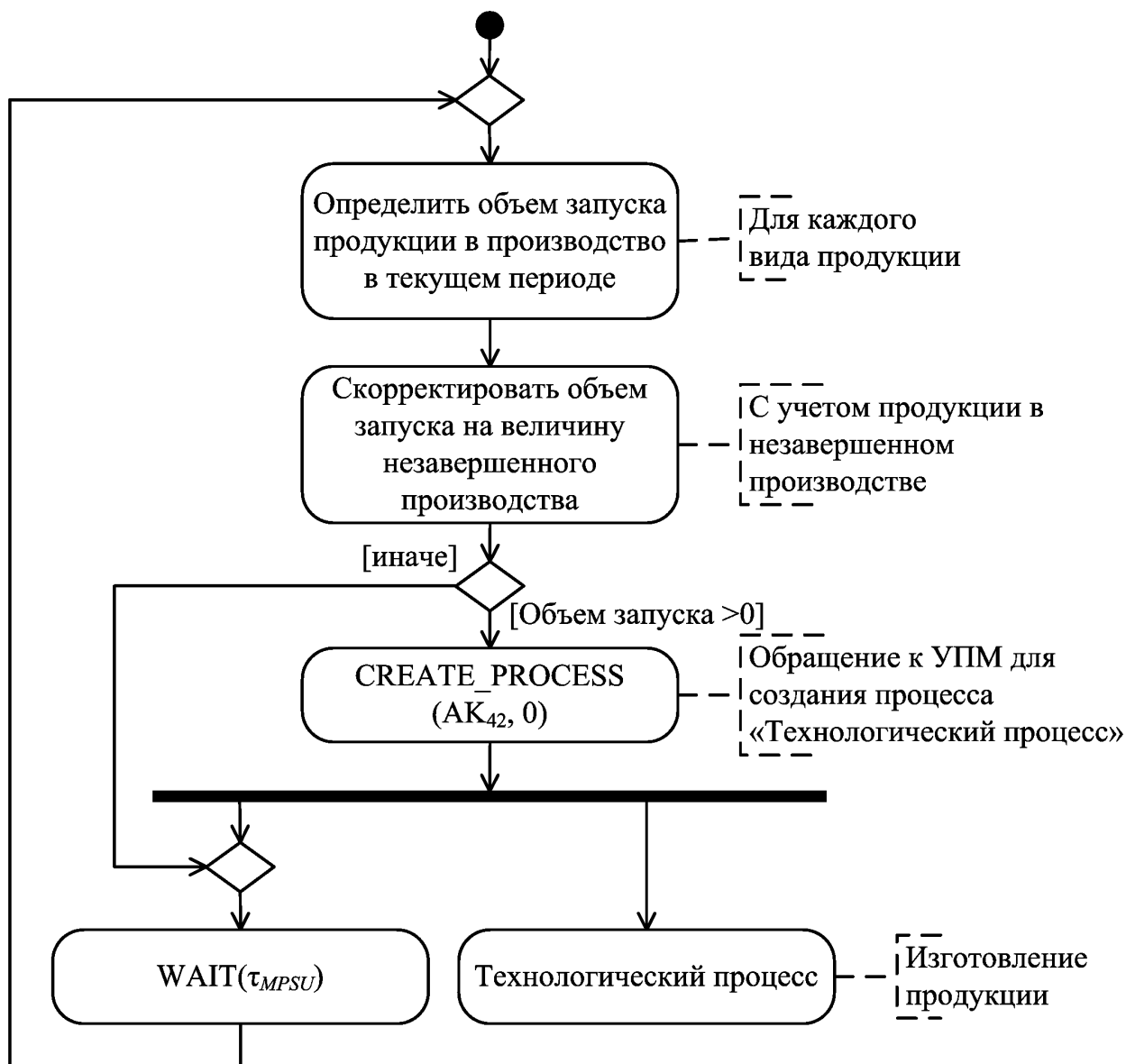


( $AK_{42}, 0$ ), посредством которого осуществляется обращение к УПМ с целью создания процесса «Технологический процесс» [125].



**Рисунок Б.1 – UML-диаграмма процесса по планированию производства**

В результате вызова оператора `CREATE_PROCESS` создается новый процесс с начальной активностью  $AK_{42}$  – «Запуск партии продукции в производство» и назначается момент его первой активизации, равный значению локального времени текущего процесса. Описанная выше последовательность действий выполняется для всех видов продукции, после чего вызывается оператор синхронизации процесса `WAIT( $\tau_{MPSU}$ )` [109].



**Рисунок Б.2 – UML-диаграмма процесса управления производством**

Процесс ПК<sub>42</sub> «Технологический процесс» имитирует выполнение технологических операций для производства заданной партии продукции (рисунок Б.3).

Первым оператором процесса (см. рисунок Б.3) является проверка наличия на складе ресурсов, необходимых для производства. При недостаточном количестве ресурсов процесс переводится в состояние ожидания до тех пор, пока не будет выполнено условие

$$RS_i \geq O_j \cdot RCR_{ij}, i = \overline{1, n_R}. \quad (\text{Б.1})$$



**Рисунок Б.3 – UML-диаграмма процесса, имитирующего технологический процесс**

В соответствии с (Б.1) состояние ожидания длится до тех пор, пока запасы ресурсов будут не меньше количества, необходимого для производства партии продукции объемом  $O_j$ , исходя из нормы  $RCR_{ij}$  расхода  $i$ -го ресурса на изготов-

ление единицы продукции  $j$ -го вида. Для этого вызывается оператор синхронизации WAIT\_UNTIL с соответствующим условием в качестве параметра. Затем размер запасов ресурсов на складе уменьшается на величину  $(O_j \cdot RCR_{ij}), i = \overline{1, n_R}$ , а объем незавершенного производства увеличивается на величину  $O_j$ . Вызовом оператора WAIT( $T_{Cj}$ ) процесс переводится в состояние ожидания на время  $T_{Cj}$ , равное длительности технологического процесса для  $j$ -го вида продукции. При следующей активизации процесса выполняются операторы, уменьшающие объем незавершенного производства, соответственно увеличивающие запас готовой продукции на складе и сохраняющие необходимые данные для бухгалтерского учета. Процесс завершается вызовом оператора END для удаления его из списка процессов УПМ [109].

*Алгоритм процесса «Управление поставками ресурсов»* имеет вид, аналогичный алгоритму процесса «Управление производством» (см. рисунок Б.2). В данном случае цикл выполняется для всех типов ресурсов. Исходя из графика потребностей в ресурсах, который формируется процессом «Планирование производства», и сроков поставки, а также учитывая размеры складских запасов и уже размещенных заказов, определяется объем заказа, который необходимо сделать в текущем периоде. Если этот требуемый объем заказа больше нуля, то размещается заказ. Для этого посредством вызова оператора CREATE\_PROCESS( $AK_{22}, 0$ ), создается процесс «Поставка ресурсов». Затем цикл повторяется для следующего типа ресурса. По завершении обработки всех ресурсов вызывается оператор синхронизации WAIT( $\tau_{MPSU}$ ).

*Алгоритм процесса «Поставка ресурсов»* (рисунок Б.4) описывает функциональные действия при выполнении контракта на поставку необходимых ресурсов. При этом можно выделить три основных блока: предварительная оплата, поставка ресурсов и оплата по факту, отсрочка платежа.

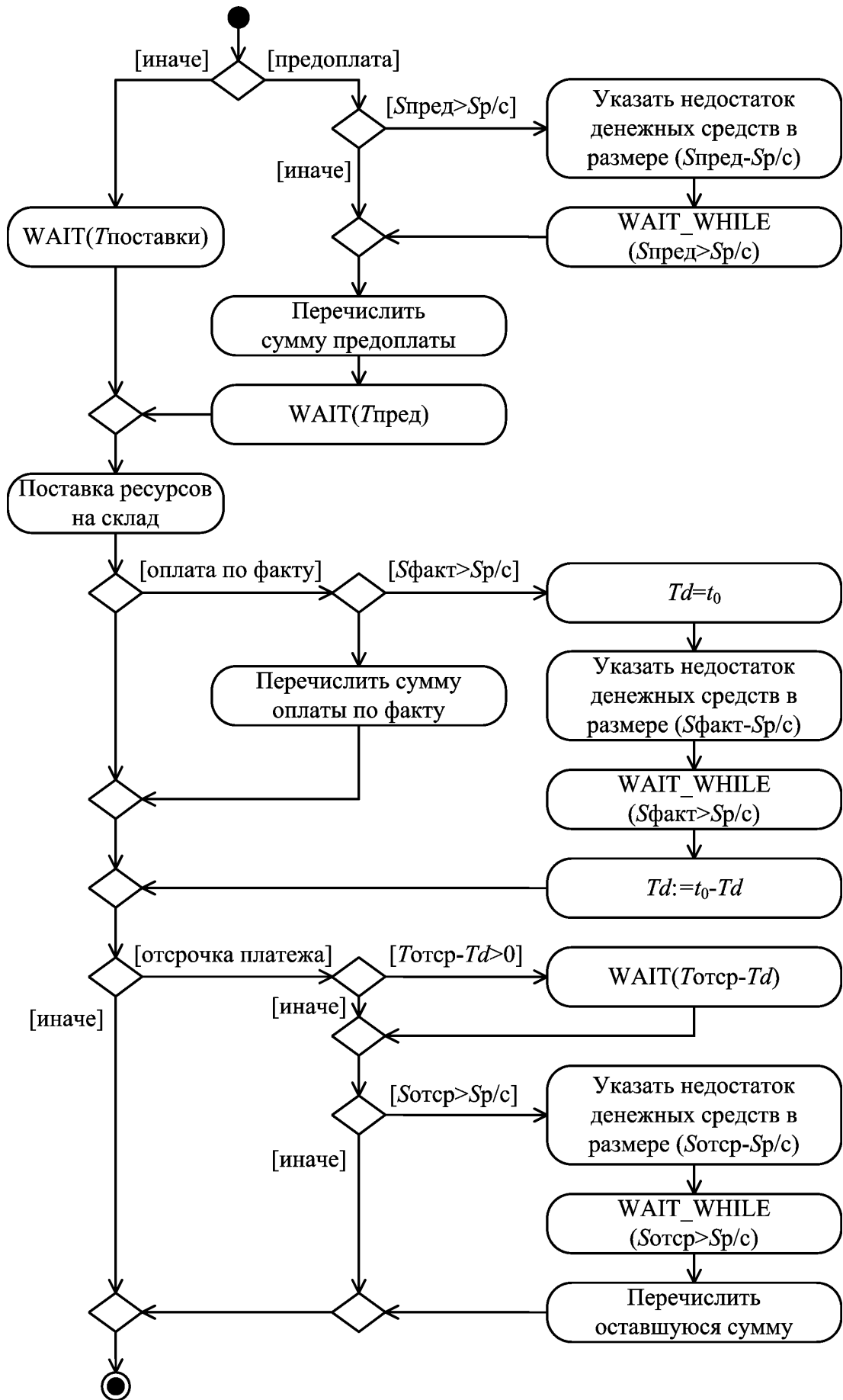


Рисунок Б.4 – UML-диаграмма процесса поставки ресурсов

В том случае, если контрактом предусматривается предварительная оплата, осуществляется проверка наличия необходимых денежных средств на расчетном счете предприятия ( $S_{\text{пред}} > S_{\text{р/с}}$ ). Если необходимая сумма отсутствует, фиксируется недостаток денежных средств в размере ( $S_{\text{пред}} - S_{\text{р/с}}$ ) и выполнение процесса приостанавливается посредством оператора WAIT\_WHILE ( $S_{\text{пред}} > S_{\text{р/с}}$ ).

При следующей активизации процесса необходимая сумма снимается с расчетного счета и управление вновь передается УПМ вызовом оператора WAIT ( $T_{\text{пред}}$ ), где  $T_{\text{пред}}$  – срок предварительной оплаты. Если же контракт не предусматривает предварительную оплату, то выполняется оператор WAIT ( $T_{\text{пост}}$ ), где  $T_{\text{пост}}$  – период времени с момента подачи заявки до поставки ресурсов. Далее следуют операторы, модифицирующие величину складских запасов, что соответствует поставке ресурсов. Группы операторов, реализующие оплату по факту и отсрочку платежа с синхронизацией по объему наличных денежных средств, соответствуют группе операторов для представления предварительной оплаты. Последним следует оператор END, который завершает данный процесс и удаляет его из списка процессов.

*Алгоритм процесса «Реализация продукции»* (рисунок Б.5) в целом аналогичен алгоритму процесса «Поставка ресурсов» и описывает функциональные действия при выполнении контракта на реализацию продукции. Основное отличие заключается в сделанном предположении платежеспособности заказчика и, как следствие, отсутствии соответствующих операторов синхронизации. Кроме того, осуществляется проверка наличия необходимого количества отгружаемой продукции, а не выплачиваемой суммы денежных средств.

Процессы по уплате налогов, выдаче заработной платы, оплате постоянных затрат и издержек представлены схожими алгоритмами (рисунок Б.6). Первыми следуют операторы, с помощью которых рассчитывается сумма денежных средств, подлежащая к уплате: сумма налоговых отчислений, заработной платы производственным рабочим и постоянных затрат. Далее следуют операторы, осуществляющие проверку наличия достаточной суммы денежных средств на расчетном счете.

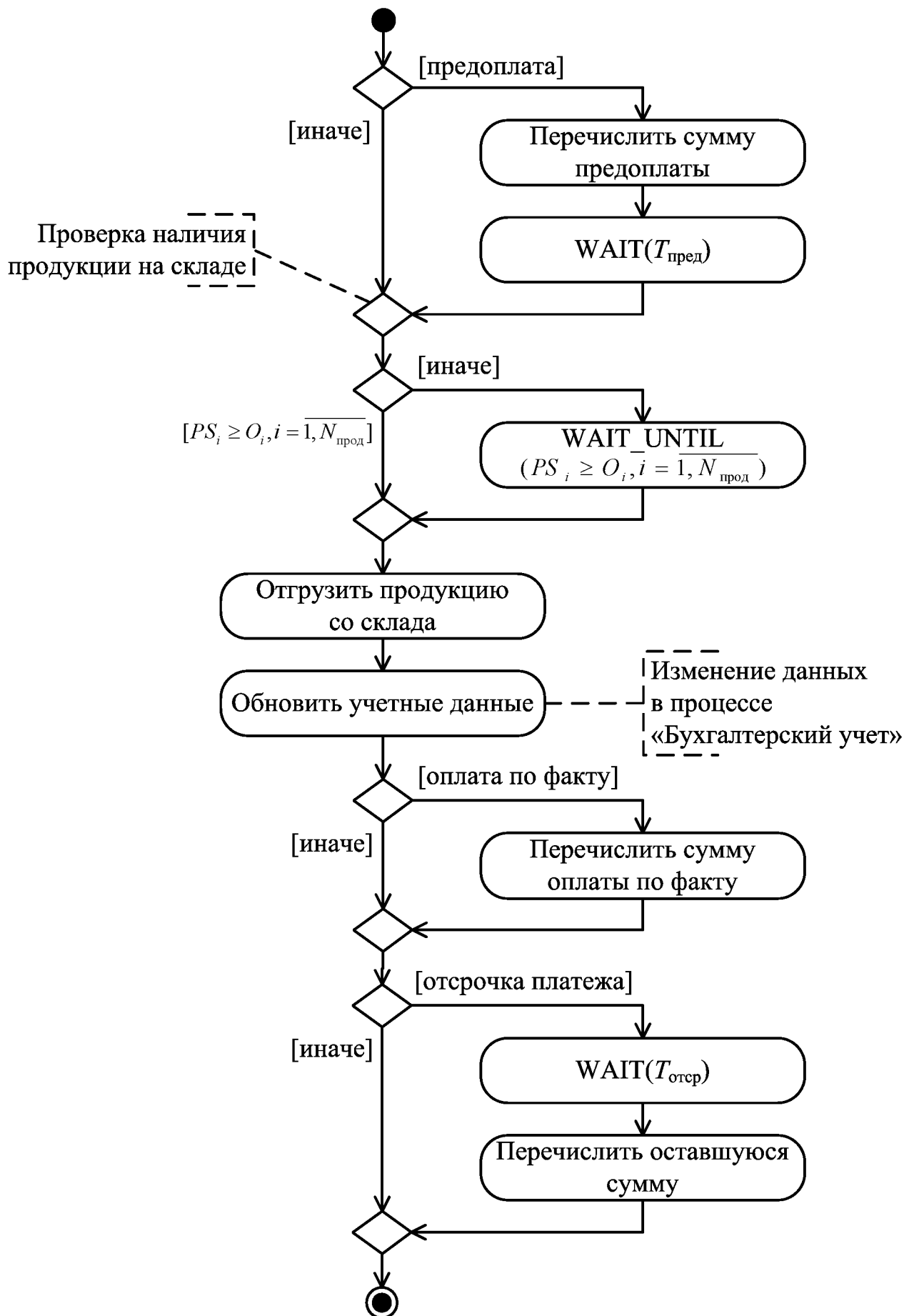
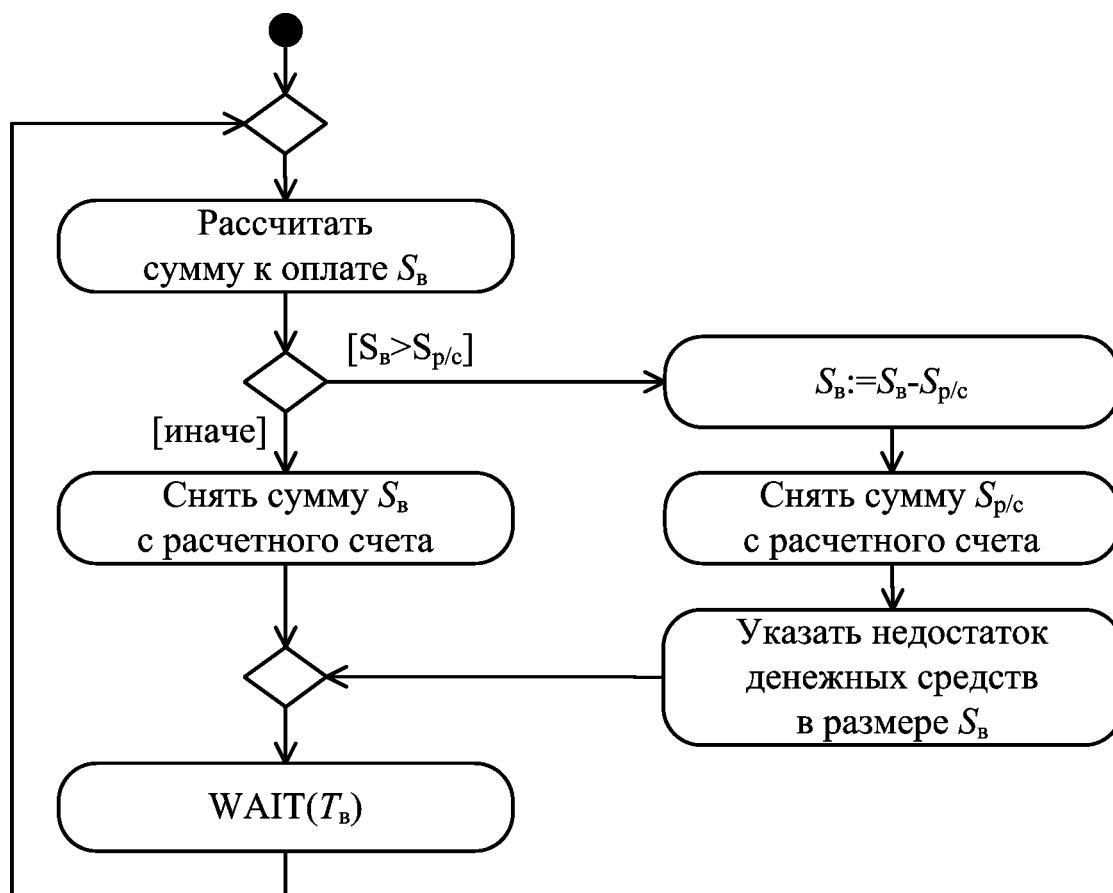


Рисунок Б.5 – UML-диаграмма процесса по реализации продукции

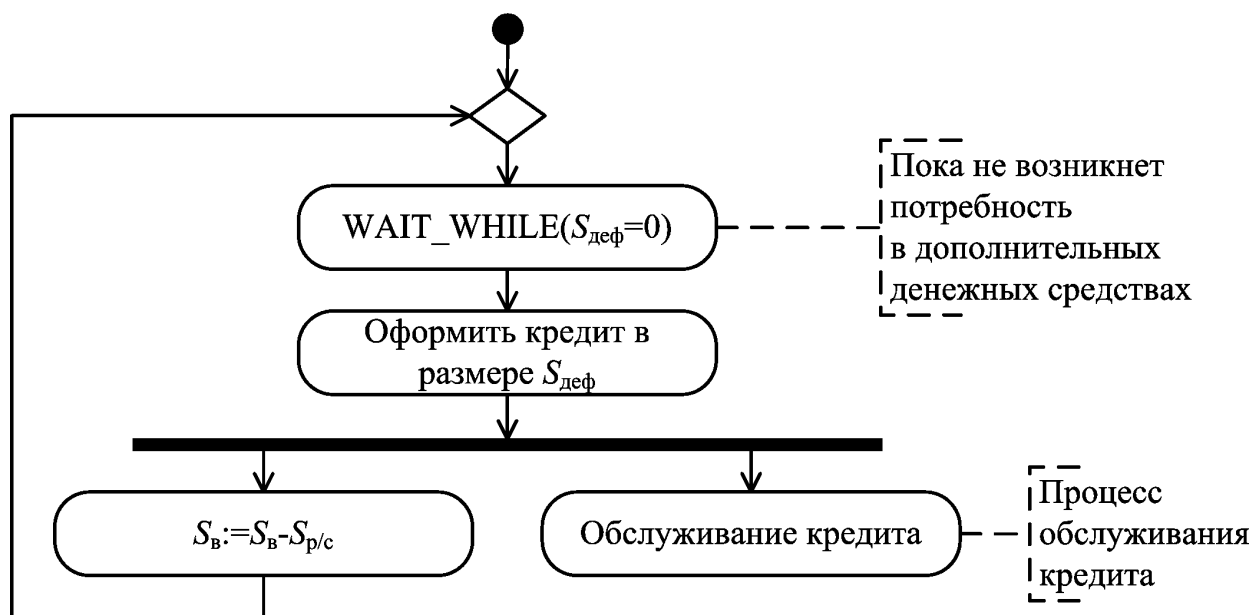


**Рисунок Б.6 – UML-диаграмма процессов по уплате налогов, выдаче заработной платы и оплате постоянных затрат и издержек**

В случае отсутствия достаточных денежных средств выплачивается имеющаяся сумма, указывается недостаток денежных средств, а остаток сохраняется (переносится на следующий период). Затем осуществляется передача управления УПМ посредством оператора  $WAIT(T_B)$ , где  $T_B = 1$  мес. – периодичность выплат. Для процесса «Уплата налогов» при недостатке денежных средств периодичность  $T_B = 1$  день, так как штрафные санкции начисляются за каждый день просрочки платежа и необходимая сумма выплачивается при первой возможности [109].

*Процесс «Получение кредитов» активизируется только в случае недостатка в денежных средствах (рисунок Б.7).*





**Рисунок Б.7 – UML-диаграмма процесса по получению кредита**

Алгоритм реализации *процесса по получению кредита* (см. рисунок Б.7) начинается с вызова оператора синхронизации  $WAIT\_WHILE(S_{\text{деф}} = 0)$ , где  $S_{\text{деф}}$  – величина, определяющая текущий дефицит наличных денежных средств. Затем группа операторов реализует оформление заявки на получения кредита в размере  $S_{\text{деф}}$ , завершается вызовом оператора  $CREATE\_PROCESS(AK_{35}, T_k)$ . В результате создается новый процесс «Обслуживание кредита» с начальной активностью  $AK_{35}$  и активизацией через время  $T_k$  получения кредита. Следующий оператор обнуляет величину  $S_{\text{деф}}$  потребности в денежных средствах, и управление снова передается на начало алгоритма.

*Алгоритм процесса «Обслуживание кредита»* (рисунок Б.8) реализует функциональные действия, соответствующие получению, выплате процентов и возврату кредита. Первый оператор переводит сумму денежных средств по кредиту на расчетный счет предприятия (получение кредита).

Последующие операторы реализуют обслуживание полученного кредита – периодическую выплату денежных средств в соответствии с условиями выданного кредита.

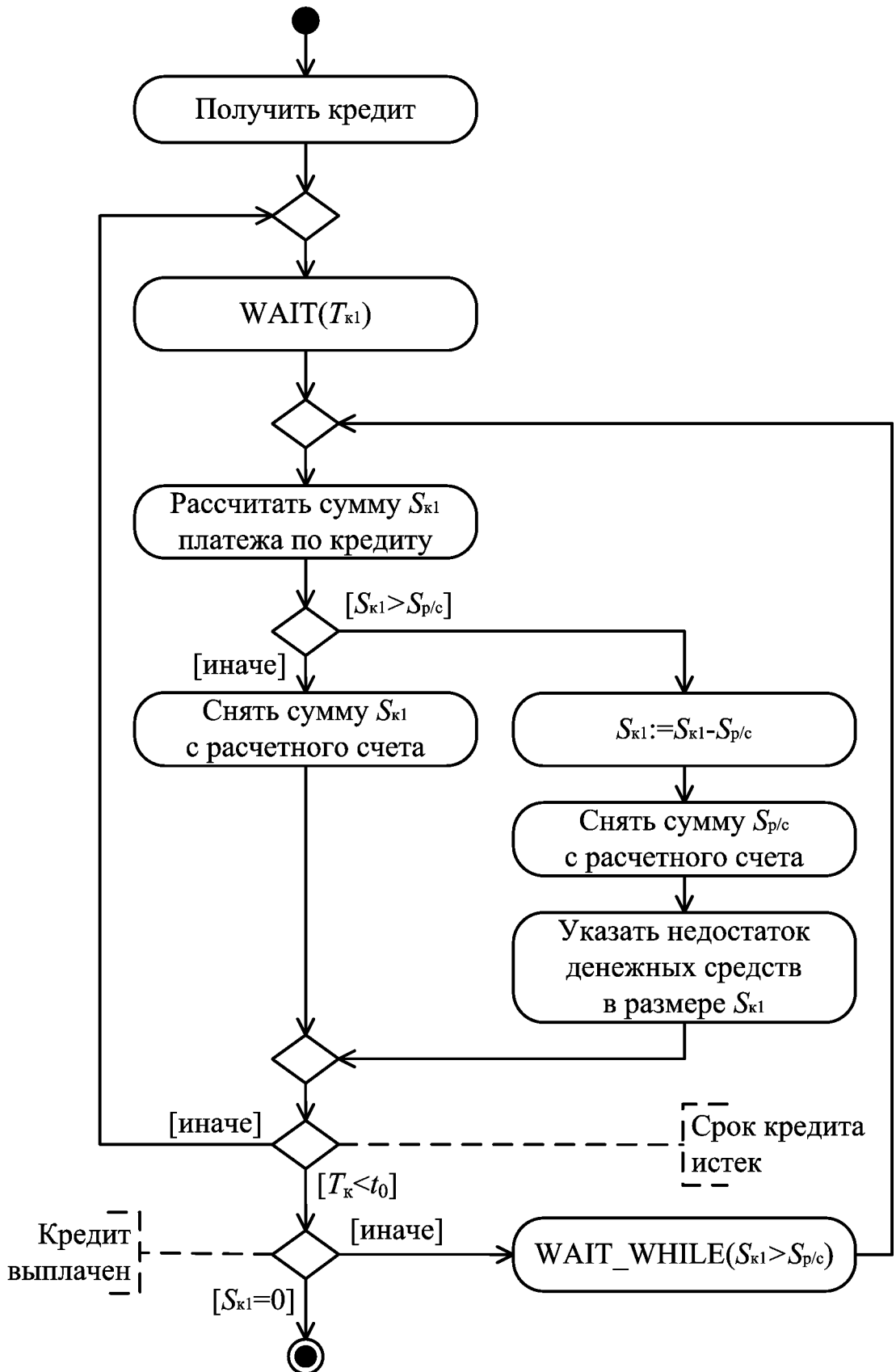


Рисунок Б.8 – UML-диаграмма процесса по обслуживанию кредита

Первым следует вызов оператора  $WAIT(T_{k1})$ , где  $T_{k1}$  – периодичность выплат денежных средств (см. рисунок Б.8). По прошествии времени  $T_{k1}$  выполняется расчет суммы требуемого платежа.

Следующий оператор сравнивает сумму к уплате с суммой денежных средств на расчетном счете. В случае недостатка выплачивается имеющаяся сумма и указывается недостаток денежных средств. Невыплаченная часть переносится на следующий период.

Процесс периодических выплат продолжается до момента окончания срока кредита  $T_k$ , после чего выплачивается оставшаяся часть кредита. При этом  $WAIT\_WHILE(S_{k1} > S_{p/c})$  реализует задержку выполнения процесса до выполнения условия наличия необходимой суммы денежных средств. Оператор END завершает процесс [109].

## ПРИЛОЖЕНИЕ В. Глобальные данные и классы базовой имитационной модели

**Таблица В.1 – Структура глобальных данных базовой имитационной модели**

Ответственность	Представляет глобальные данные модели	
Атрибуты	<i>numberOfProductTypes</i> <i>numberOfResourceTypes</i> <i>MPSTimeUnit</i>  <i>MPSPlanningHorizon</i>  <i>MPS</i> <i>launchingSchedule</i> <i>resourceSchedule</i> <i>productionCycle</i> <i>specificResourceConsumption</i>  <i>goodsInProgress</i> <i>products</i> <i>productActualCosts</i>  <i>resourcesOrdered</i> <i>resources</i> <i>shipmentIntervalDistribution</i>  <i>shipmentVolumeDistributions</i>  <i>shipmentPaymentPeriod Distribution</i>  <i>productPrices</i> <i>productPriceChangeMultiplier Distribution</i> <i>productPriceChangeInterval Distribution</i> <i>deliveryPaymentPeriods</i> <i>resourcePriceChangeMultiplier Distribution</i> <i>resourcePriceChangeInterval Distribution</i> <i>resourcePrices</i>	Количество видов продукции Количество видов ресурсов Интервал планирования основного плана-графика производства Количество интервалов в основном плане-графике производства Основной план-график производства График запуска в производство График потребности в ресурсах Длительность технологического цикла Нормы расхода ресурсов на единицу продукции Незавершенное производство Запасы продукции на складе Фактическая себестоимость продукции на складе Объем заказанных ресурсов Запасы ресурсов на складе Параметры распределения интервала между отгрузками продукции Параметры распределения объемов продукции в отгрузке Параметры распределения срока оплаты за отгруженную продукцию Цены на продукцию Параметры распределения коэффициента изменения цен продукции Параметры распределения интервала между изменениями цен продукции Сроки оплаты за полученные ресурсы Параметры распределения коэффициента изменения цен ресурсов Параметры распределения интервала между изменениями цен ресурсов Цены ресурсов

## Продолжение таблицы В.1

Ответ- ствен- ность	Представляет глобальные данные модели	
	<i>shipments</i> <i>currentAccount</i>  <i>creditRequirements</i>  <i>loanInterestRate</i> <i>loanApplicationInterval</i>  <i>loanPeriod</i> <i>cashShortageCoverage</i>  <i>fixedCosts</i>  <i>wageShareInFixedCosts</i>  <i>otherSpecificVariableCosts</i>  <i>wageShareInVariableCosts</i>  <i>costsChangeMultiplierDistribution</i>  <i>costsChangeIntervalDistribution</i>  <i>fixedAssetsReplacementCost</i>  <i>fixedAssetsWearFactor</i> <i>fixedAssetsDepreciationRate</i> <i>fixedAssetsReappraisalMultiplier</i>  <i>taxableFixedAssetsCost</i>   <i>VATRate</i> <i>receiptsTaxRate</i> <i>profitTaxRate</i> <i>profitInDisposalTaxRate</i>   <i>socialInsuranceTaxRate</i>	Список отгрузок Сумма денежных средств на расчетном счете Потребность в кредите (сумма нехватки денежных средств) Процентная ставка займа Периодичность обращения за кредитом Срок кредита Коэффициент покрытия кредитом текущей нехватки денежных средств Постоянные затраты в целом по предприятию Относительная часть заработной платы в постоянных затратах в целом по предприятию Переменные затраты в единице продукции (без стоимости ресурсов) Относительная часть заработной платы в переменных затратах Параметры распределения коэффициента изменения затрат Параметры распределения интервала между изменениями затрат Восстановительная стоимость основных средств Коэффициент износа основных средств Норма амортизации основных средств Коэффициент переоценки основных средств Стоимость основных средств, с которой уплачивается налог на недвижимость Ставка НДС Ставка налогов с выручки Ставка налогов с прибыли Ставка налогов с прибыли, остающейся в распоряжении предприятия Ставка отчислений в фонд социальной защиты

## Продолжение таблицы В.1

Ответ- ствен- ность	Представляет глобальные данные модели	
	<i>wageTaxRate</i> <i>fixedAssetsTaxRate</i> <i>ecologicalTaxRate</i>  <i>landTax</i> <i>VAT</i>  <i>quantityShipped</i> <i>quantityProduced</i> <i>resourceCosts</i>  <i>resourcesTaken</i> <i>clearReceipts</i> <i>costOfSales</i> <i>profit</i> <i>profitTax</i>  <i>dispositionOfProfits</i> <i>accounts</i>	Ставка налогов с заработной платы Ставка налога на недвижимость Сумма экологического налога на единицу продукции Сумма налога на землю Сумма НДС в реализованной продукции Количество отгруженной продукции Количество произведенной продукции Стоимость ресурсов в произведенной продукции Стоимость ресурсов, взятых со склада Выручка (за минусом налогов) Себестоимость реализации продукции Прибыль Налог на прибыль и иные обязательные платежи Использовано прибыли Счета бухгалтерского учета
Операции	<i>getResourceStockLevel</i> <i>getResourceCost</i> <i>takeProduct</i>  <i>takeResource</i>  <i>calculateProductPrice</i> <i>increaseCreditRequirements</i>  <i>decreaseCreditRequirements</i>	Объем ресурса на складе Стоимость ресурса на складе Получение вида продукции со склада (возвращает суммарные затраты ресурсов) Получение ресурса со склада (возвращает суммарную стоимость) Расчет цены продукции Увеличение суммы требуемого кредита Уменьшение суммы требуемого кредита

Перечень классов активностей процессов базовой имитационной модели представлен в таблице В.2.

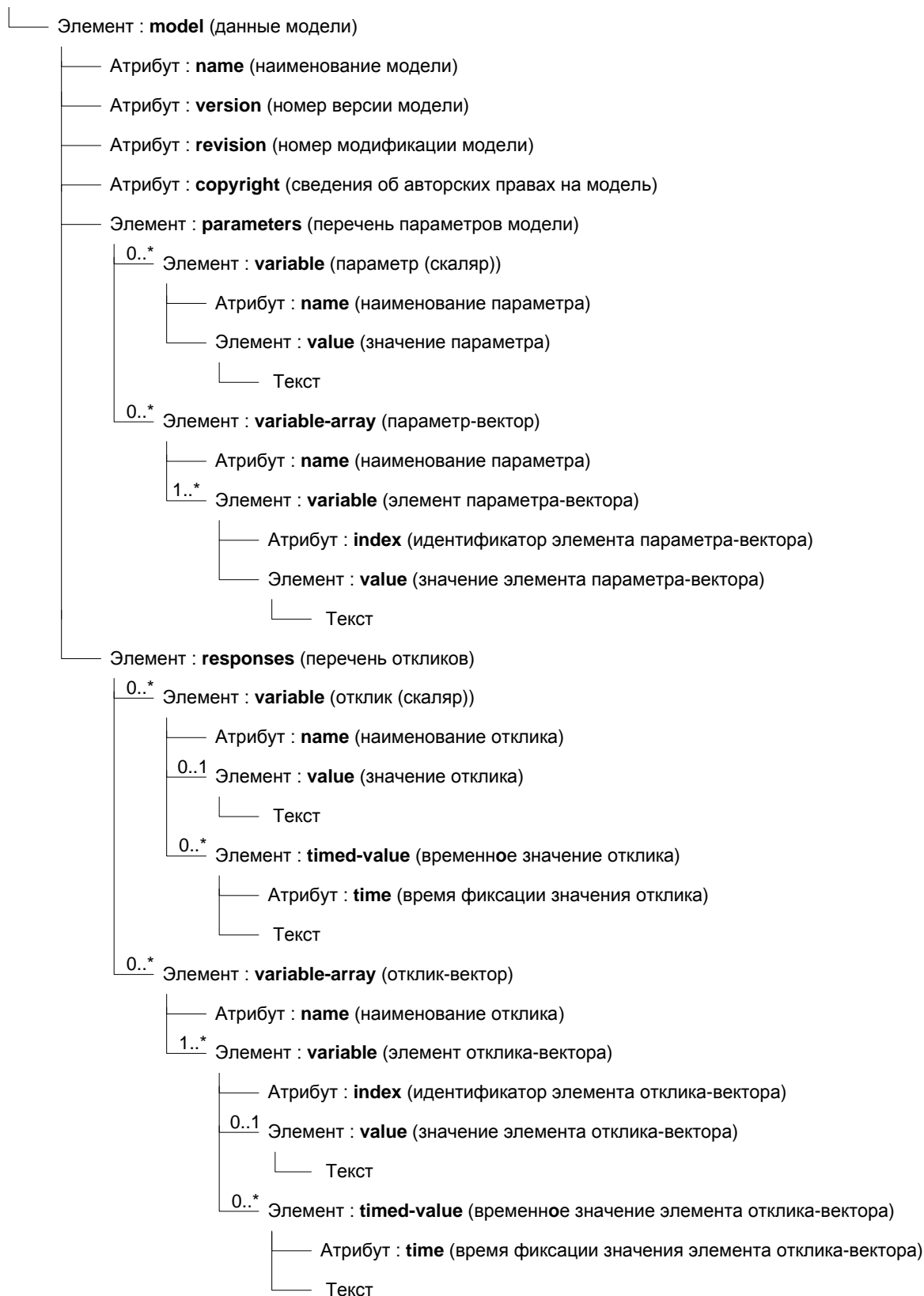
**Таблица В.2 – Перечень классов активностей процессов базовой имитационной модели**

Имя класса	Активность
<i>CShipmentGenerating</i>	Входной поток заказов (требуемых отгрузок готовой продукции)
<i>CShipmentPrepaying</i>	Предварительная оплата за продукцию
<i>CShipmentShipping</i>	Отгрузка продукции
<i>CShipmentPaying</i>	Оплата за продукцию (с отсрочкой)
<i>CMasterProductionScheduling</i>	Планирование производства
<i>CResourceOrdering</i>	Управление поставками требуемых ресурсов
<i>CDeliveryPrepaying</i>	Предварительная оплата за поставку ресурсов
<i>CDeliveryDelivering</i>	Поставка ресурсов
<i>CDeliveryPaying</i>	Оплата за ресурсы (с отсрочкой)
<i>CSalesContractRunning</i>	Выполнение контракта на реализацию продукции
<i>CProductionLaunching</i>	Запуск продукции в производство
<i>CProductionStartup</i>	Начало производства партии продукции
<i>CProductionCompletion</i>	Завершение производства партии продукции
<i>CCostChanging</i>	Изменение стоимости во времени
<i>CProductPriceChanging</i>	Изменение цены на один вид продукции
<i>CResourcePriceChanging</i>	Изменение цены ресурса
<i>CFixedCostsChanging</i>	Изменение постоянных затрат предприятия
<i>COtherVariableCostsChanging</i>	Изменение переменных затрат предприятия
<i>CAccounting</i>	Бухгалтерский учет
<i>CDebtPaying</i>	Оплата задолженности, базовый класс
<i>CWagePaying</i>	Оплата задолженности по заработной плате
<i>CTaxPaying</i>	Оплата задолженности по налогам
<i>CSocialInsuranceTaxPaying</i>	Оплата задолженности социальному страхованию
<i>CCreditPaying</i>	Оплата задолженности по кредиту (по выплате процентов)
<i>CBillPaying</i>	Оплата задолженности перед поставщиками и подрядчиками
<i>CApplyingForCredit</i>	Обращение за кредитом
<i>CCreditReceiving</i>	Получение кредита
<i>CCreditRepaying</i>	Выплата процентов и возврат кредита
<i>CStatisticsGathering</i>	Сбор статистики

## ПРИЛОЖЕНИЕ Г. Структура файла данных модели и данных эксперимента

### Структура файла данных модели:

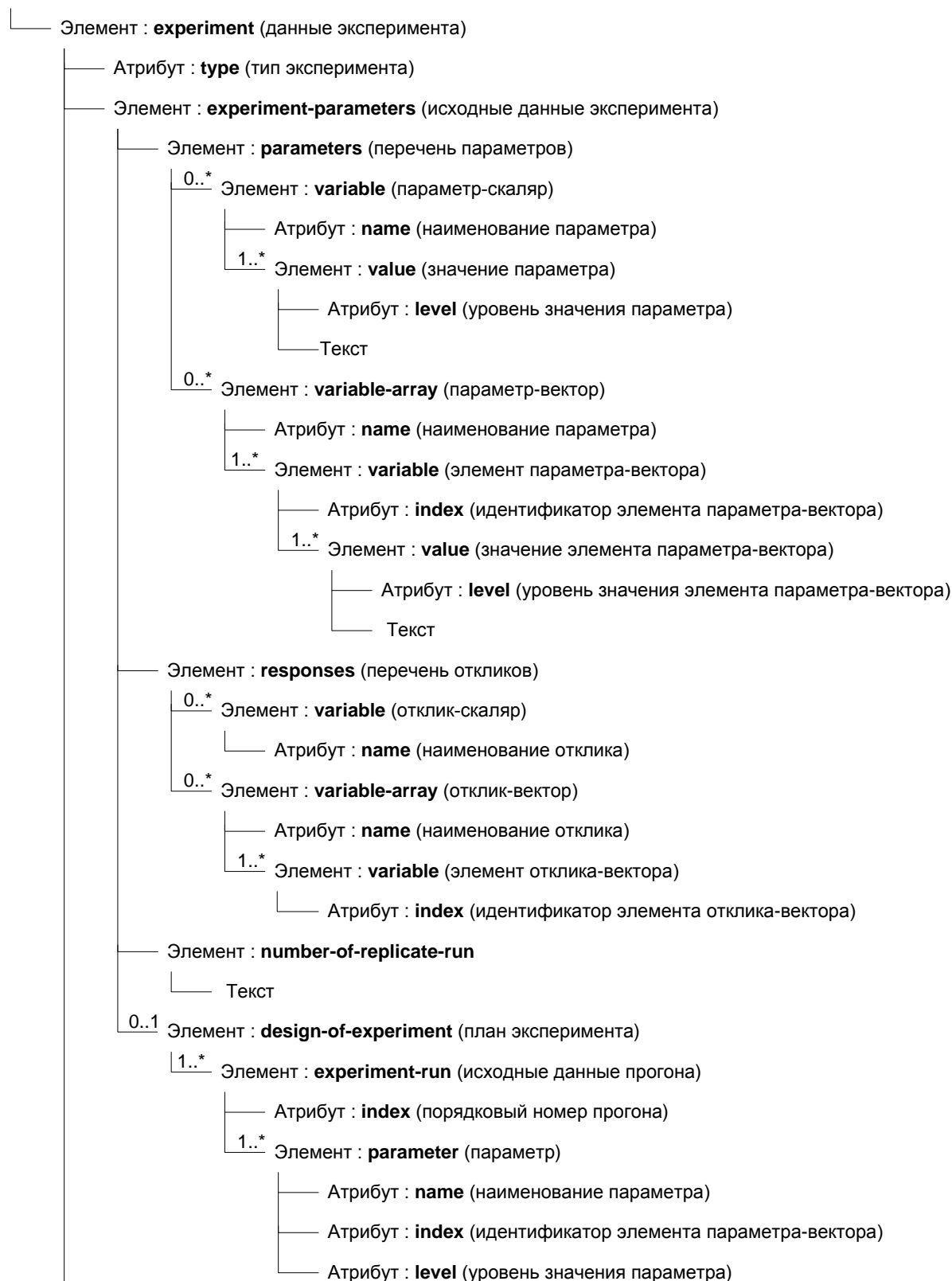
Корневой узел





## Структура файла данных эксперимента:

Корневой узел



1

Элемент : **experiment-results** (результаты эксперимента)

1..\* Элемент : **experiment-run** (результаты прогона)

Атрибут : **index** (порядковый номер прогона)

Элемент : **parameters** (перечень параметров)

0..\* Элемент : **variable** (параметр-скаляр)

Атрибут : **name** (наименование параметра)

Элемент : **value** (значение параметра)

Атрибут : **level** (уровень значения параметра)

Текст

0..\* Элемент : **variable-array** (параметр-вектор)

Атрибут : **name** (наименование параметра)

1..\* Элемент : **variable** (элемент параметра-вектора)

Атрибут : **index** (идентификатор элемента параметра-вектора)

Элемент : **value** (значение элемента параметра-вектора)

Атрибут : **level** (уровень значения элемента параметра-вектора)

Текст

1..\* Элемент : **responses** (перечень откликов)

Атрибут : **run** (порядковый номер опыта)

0..\* Элемент : **variable** (отклик-скаляр)

Атрибут : **name** (наименование отклика)

0..1 Элемент : **value** (значение отклика)

Текст

0..\* Элемент : **timed-value** (временное значение отклика)

Атрибут : **time** (время фиксации значения отклика)

Текст

0..\* Элемент : **variable-array** (отклик-вектор)

Атрибут : **name** (наименование отклика)

1..\* Элемент : **variable** (элемент отклика-вектора)

Атрибут : **index** (идентификатор элемента отклика-вектора)

0..1 Элемент : **value** (значение элемента отклика-вектора)

Текст

0..\* Элемент : **timed-value** (временное значение элемента отклика-вектора)

Атрибут : **time** (время фиксации значения элемента отклика-вектора)

Текст

## ПРИЛОЖЕНИЕ Д. Дополнение программы BelSim Experimenter для распределения вычислений

```

int wmain(int argc, wchar_t *argv[])
    clock_t start = clock(); // отметка начала проведения имитационного
эксперимента
    argvs[1] = "experimenter.exe";
    int argc = 1, MyId/*id текущего процесса*/, NumProcs/*количество про-
цессов*/;
    MPI_Init(&argc,&argvs); // установка начальных значений MPI
    MPI_Comm_size(MPI_COMM_WORLD,&NumProcs); // определяем общее
число запущенных процессов
    MPI_Comm_rank(MPI_COMM_WORLD,&MyId); // определяем иденти-
фикатор Id процесса в текущий момент времени
    if(MyId == 0) // запрет вывода на консоль повторяющихся данных
        cout << endl << g_szTitle << endl;
    if(MyId == 0) // запрет вывода на консоль повторяющихся данных
        cout << endl;
        if(MyId == 0) {
            cout << endl << L"Тип эксперимента:" << endl;
            switch (spExperimentData->getExperimentType())
            {
            case CExperimentData::AccuracyEvaluation:
                cout << L" оценка погрешности" << endl;
                break;
            case CExperimentData::TransientAndStabilityAnalysis:
                cout << L" анализ переходного процесса и устойчи-
ости" << endl;
                break;
            case CExperimentData::SensitivityAnalysis:
                cout << L" анализ чувствительности" << endl;
                break;
            case CExperimentData::FullFactorial:
                cout << L" полный факторный эксперимент" << endl;
                break;
            case CExperimentData::FractionalFactorial:
                cout << L" дробный факторный эксперимент" << endl;

```

```

if(MyId == 0) // запись результатов первого потока в основной файл
spExperimentData->save(argv[3]);
else // запись результатов других потоков во временный файл
spExperimentData->save(argv[3] + uint2wstring(myid) + wstring(L".xml"));
cout << L" OK" << endl;
MPI_Barrier(MPI_COMM_WORLD); // ожидание окончания записи ре-
зультатов всеми потоками
if(MyId == 0) { // первый поток собирает все файлы
for(int i = 1; i < numprocs; i++) {
CExperimenter::mergeXML(argv[3], argv[3] + uint2wstring(i) +
wstring(L".xml"), i);
// запись результатов i-ого потока
std::string s = wstring2string(argv[3] + uint2wstring(i) + wstring(L".xml"));
remove(s.c_str()); // удаление временного файла
}
}
float duration = (float) (clock() - start) / CLK_TCK;
cout<<"Time - "<<duration<<"sec"<<endl; // вывод длительности функци-
онирования приложения
void CExperimenter::mergeXML(wstring fileNameDist, wstring fileNameSrc,
int madeMerge)
// запись результатов из файла, именованный fileNameSrc, в файл,
именованный fileNameDist
void CExperimenter::run(SP<CModel> model, SP<CModelData> modelData,
if(runNo % NumProcs == MyId) { // запуск экспериментов, запускаемых
только нашим процессом

```

**ПРИЛОЖЕНИЕ Е. Документы об апробации, внедрении и эффективности использования**

РЕСПУБЛИКА БЕЛАРУСЬ



# СВИДЕТЕЛЬСТВО

## О РЕГИСТРАЦИИ КОМПЬЮТЕРНОЙ ПРОГРАММЫ

### № 025

Наименование компьютерной программы:  
**Программно-технологический комплекс имитации сложных систем «BelSim 2»**

Правообладатель компьютерной программы:  
**Государственное учреждение высшего профессионального образования «Белорусско-Российский университет»**

Год создания компьютерной программы:  
**2003**

Автор (авторы) компьютерной программы:  
**Якимов Анатолий Иванович  
 Альховик Сергей Александрович  
 Башаримов Вячеслав Владимирович  
 Захарченков Константин Васильевич**

Заявка № **C20080010**                      Дата подачи: **04.04.2008**

Дата внесения записи в  
 Реестр зарегистрированных  
 компьютерных программ: **07.05.2008**

Генеральный директор  
 Национального центра  
 интеллектуальной собственности

  
**Д.М.Воронетский**



Настоящее свидетельство выдано на основании сведений и материалов, представленных в заявке на регистрацию компьютерной программы.

НАЦИОНАЛЬНЫЙ ЦЕНТР ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

РЕСПУБЛИКА БЕЛАРУСЬ



# СВИДЕТЕЛЬСТВО

О РЕГИСТРАЦИИ КОМПЬЮТЕРНОЙ ПРОГРАММЫ

№ 026

Наименование компьютерной программы:

**Программный комплекс оценки эффективности  
контрактов "Contract Analyzer 2"**

Правообладатель компьютерной программы:

**Государственное учреждение высшего профессионального  
образования «Белорусско-Российский университет»**

Год создания компьютерной программы:

**2001**

Автор (авторы) компьютерной программы:

**Альховик Сергей Александрович  
Якимов Анатолий Иванович**Заявка № **C20080011**Дата подачи: **04.04.2008**Дата внесения записи в  
Реестр зарегистрированных  
компьютерных программ:**07.05.2008**Генеральный директор  
Национального центра  
интеллектуальной собственности

Л.И.Воронецкий

Настоящее свидетельство выдано на основании сведений и материалов,  
представленных в заявке на регистрацию компьютерной программы

НАЦИОНАЛЬНЫЙ ЦЕНТР ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

РЕСПУБЛИКА БЕЛАРУСЬ



# СВИДЕТЕЛЬСТВО

О РЕГИСТРАЦИИ КОМПЬЮТЕРНОЙ ПРОГРАММЫ

№ 308

Наименование компьютерной программы:

**Программный комплекс имитации производственно-экономической деятельности «Shago VitaPro»**

Правообладатель компьютерной программы:

**Государственное учреждение высшего профессионального образования «Белорусско-Российский университет»**

Год создания компьютерной программы:

**2010**

Автор (авторы) компьютерной программы:

**Якимов Анатолий Иванович  
Захарченков Константин Васильевич  
Альховик Сергей Александрович  
Емельянов Игорь Александрович  
Мельников Игорь Игоревич**

Заявка № **C20110027**

Дата подачи: **14.04.2011**

Дата внесения записи в  
Реестр зарегистрированных  
компьютерных программ:

**04.05.2011**

Генеральный директор  
Национального центра  
интеллектуальной собственности

**Д.И.Воронецкий**

Настоящее свидетельство выдано на основании сведений и материалов,  
представленных в заявке на регистрацию компьютерной программы



НАЦИОНАЛЬНЫЙ ЦЕНТР ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

РЕСПУБЛИКА БЕЛАРУСЬ



# СВИДЕТЕЛЬСТВО

О РЕГИСТРАЦИИ КОМПЬЮТЕРНОЙ ПРОГРАММЫ

№ 518

Наименование компьютерной программы:

**Программный модуль анализа сингулярного спектра числовых последовательностей данных «BelSim2#.SSA»**

Правообладатель компьютерной программы:

**Государственное учреждение высшего профессионального образования «Белорусско-Российский университет»**

Год создания компьютерной программы:

**2012**

Автор (авторы) компьютерной программы:

**Якимов Евгений Анатольевич  
Албкеират Джихад Мохаммад  
Чапаров Назар Мурадovich  
Якимов Анатолий Иванович**Заявка № **C20130042**Дата подачи: **23.05.2013**Дата внесения записи в  
Реестр зарегистрированных  
компьютерных программ:**14.06.2013**И.о. генерального директора  
Национального центра  
интеллектуальной собственности**Д.И.Недвецкий**Настоящее свидетельство выдано на основании сведений и материалов,  
представленных в заявке на регистрацию компьютерной программы

НАЦИОНАЛЬНЫЙ ЦЕНТР ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ



## Белорусский национальный технический университет

«Согласовано»  
Первый проректор Белорусско-  
Российского университета

  
Ф.И. Стовщенко  
«25» 11 2005 г.

«Утверждаю»

Первый проректор Белорусского  
национального технического университета

  
В.Л. Соломахов  
«25» 11 2005 г.

о внедрении результатов НИР в учебный процесс

Настоящий акт составлен об использовании в учебном процессе методики исследования имитационной модели контрактов купли, продажи и дилерского обслуживания, программно-технологического комплекса имитации сложных систем (ПТКИ) BelSim 2003, разработанных при выполнении темы НИР ГБ021ф № ГР 20021541: «Разработка динамических имитационных экономико-математических моделей прогнозирования в условиях неопределенности» при финансовой поддержке Государственной программы фундаментальных исследований «Исследование основных математических структур и проблем математического моделирования» (шифр «Математические структуры»).

Работа использована в учебном процессе кафедры «Экономика и управление научными исследованиями, проектированием и производством» Белорусского национального технического университета.

Результаты применяются на практических занятиях по курсам «Прогнозирование и планирование в экономике», «Экономика предприятия», «Финансы предприятия» при подготовке студентов по специальностям И-27 01 01 «Экономика и организация производства» и И-26 02 02 «Менеджмент». Внедрение результатов научных исследований позволяет организовать учебный процесс с применением современных систем имитационного моделирования и пакетов статистической обработки данных; дать обучаемым знания, необходимые для моделирования экономических процессов и систем управления.

Описание объекта внедрения прилагается и является неотъемлемой частью Акта.

От Белорусско-Российского университета

Начальник НИСа Мельников С.Ф.

Разработчики Якимов А.И.

Альховик С.А.

Захарченков К.В.

От БНТУ:

Декан факультета Шамкалович В.И.

Зав. кафедрой Гурина Е.В.

Мелюшин П.В.

Сапелкина Е.И.

**Государственное учреждение высшего профессионального образования  
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»**



Проректор по научной работе  
государственного учреждения высшего  
профессионального образования  
"Белорусско-Российский университет"  
М.Е. Лустенков  
2010 г.

**АКТ О ВНЕДРЕНИИ**  
результатов НИР в учебный процесс

Настоящий акт составлен об использовании в учебном процессе следующих научных результатов, полученных при выполнении НИР Ф08Р-231, № госрегистрации 20082063:

1) метод распределенного построения имитационной модели сложных многоуровневых иерархических систем, основанный на принципе декомпозиции информационных технологий комплексных информационных ERP-систем промышленных предприятий и реализованный в программно-технологическом комплексе имитации сложных систем (ПТКИ) BelSim 2;

2) методика перехода от вербальной и концептуальной модели исследуемой системы, представленной в нотации IDEF0, к формальной модели и описанию имитационной модели;

3) методика проведения имитационных экспериментов для сложных производственно-технических систем на основе средств распределения вычислений в ПТКИ BelSim 2, включающая процедуру оценки трудоемкости имитационного эксперимента для подтверждения правильности реализации алгоритма распределения вычислений;

4) технология автоматизации планирования, проведения и обработки результатов имитационных экспериментов, ориентированная на информационные потребности пользователей имитационной модели.

Вышеуказанные в п.п.1-4 научные результаты использованы в учебном процессе кафедры «Автоматизированные системы управления» при подготовке аспирантов по специальности 05.13.01 – «Системный анализ, управление и обработка информации» по следующим темам кандидатских диссертаций:

1) Метод и средство повышения эффективности эксплуатации имитационных моделей функционирования предприятия (утверждена Советом Бел.-Рос. ун-та 13.01.2009 г., протокол № 5);

2) Повышение функциональности среды имитационного моделирования для использования в системе управления промышленным предприятием (утверждена Советом Бел.-Рос. ун-та 13.01.2009 г., протокол № 5);

3) Имитационное моделирование и оптимизация управления предприятием общественного питания (утверждена Советом Бел.-Рос. ун-та 13.01.2009 г., протокол № 5);

4) Метод и средство оптимизации распределения ресурсов в производственном процессе крашения тканей набивным способом (утверждена Советом Бел.-Рос. ун-та 28.12.2009 г., протокол № 4);

5) Метод и средство повышения эффективности эксплуатации имитационных моделей в Интернет-сети (утверждена Советом Бел.-Рос. ун-та 28.12.2009 г., протокол № 4).

Описание объекта внедрения прилагается и является неотъемлемой частью Акта.

Начальник НИСа, канд. техн. наук

Зав. кафедрой, канд. техн. наук

Научный руководитель НИР, канд. техн. наук

С.Ф. Мельников

С.К. Крутолевич

А.И. Якимов

## ОПИСАНИЕ ОБЪЕКТА ВНЕДРЕНИЯ

ПТКИ BelSim 2, реализующий метод распределенного построения имитационной модели, методика перехода от концептуальной модели системы в нотации IDEF0-диаграмм к формальному описанию имитационной модели, методика проведения имитационных экспериментов с распределением вычислений, технология автоматизации планирования, проведения и обработки результатов имитационных экспериментов, ориентированная на информационные потребности пользователей имитационной модели.

### 1. Краткая характеристика объекта внедрения и его назначение

1.1 метод распределенного построения имитационных моделей производственно-экономических систем в составе ERP-систем управления и ПТКИ BelSim 2, реализованный на его основе, отличаются комплексным применением при программной реализации имитационных моделей технологии объектно-ориентированного анализа и проектирования, UML, XML, ADO, MPI информационных технологий, что позволяет создавать перспективные системы имитационного моделирования для комплексных систем анализа деятельности предприятий;

1.2 методика перехода от концептуальной модели системы в нотации IDEF0-диаграмм к формальному описанию имитационной модели, в соответствии с которой определяются типы компонентов модели, реализуемые функции объединяются в интерфейсы и определяются связи между компонентами на основе интерфейсов;

1.3 методика проведения имитационных экспериментов с распределением вычислений включает процедуру оценки трудоемкости имитационного эксперимента для подтверждения правильности реализации алгоритма распределения вычислений в виде регрессионного уравнения зависимости трудоемкости эксперимента от параметров модели

1.4 технология автоматизации планирования, проведения и обработки результатов имитационных экспериментов, ориентированная на информационные потребности пользователей имитационной модели, включает визуальный язык сценариев, основными компонентами которого являются элементарные операции: получение исходных данных из различных источников, статистическая обработка данных, выполнение имитационной модели, построение графиков и отчетов, экспорт данных.

2. Фамилия и инициалы научных руководителей аспирантов, использующих научные разработки: Максимей И.В., кафедра математических проблем управления УО «Гомельский государственный университет имени Франциска Скорины», заведующий кафедрой, Якимов А.И., кафедра АСУ ГУВПО «Белорусско-российский университет», доцент, Альховик С.А. , кафедра АСУ ГУВПО «Белорусско-российский университет», доцент.

3. Начало использования объекта внедрения (месяц, год): январь 2009 г.

4. Число аспирантов, пользующихся разработкой: 5 аспирантов.

Начальник НИСа, канд. техн. наук

С.Ф. Мельников

Зав. кафедрой, канд. техн. наук

С.К. Крутолевич

Научный руководитель НИР, канд. техн. наук

А.И. Якимов

## ОАО «Могилевхимволокно» Завод органического синтеза

«Согласовано»

Первый проректор Белорусско-  
Российского университетаФ.Г.Ловшенко  
2005 г.

«Утверждаю»

Главный инженер  
Завода органического синтезаМ.И.Залого  
2005 г.

### АКТ

#### о внедрении результатов НИР на заводе органического синтеза ОАО «Могилевхимволокно»

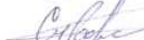
Настоящий акт составлен об использовании в производственном процессе получения диметилтерефталата (ДМТ) на заводе органического синтеза (ЗОС) математического и программного обеспечения для автоматизации расчёта оптимальной загрузки цехов ДМТ ЗОС, разработанных при выполнении соглашения о научно-производственном сотрудничестве С0501: «Разработка информационной системы для моделирования технологических процессов на ОАО «Могилевхимволокно» при финансовой поддержке Государственной программы фундаментальных исследований «Исследование основных математических структур и проблем математического моделирования» (шифр «Математические структуры»).

Выполненная работа используется при планировании распределения выпуска ДМТ между цехами в производственно-диспетчерском отделе ЗОС ОАО «Могилевхимволокно».

Внедрение результатов научных исследований: математическая модель и программное обеспечение позволяют получить оптимальное распределение загрузки цехов ДМТ для снижения себестоимости выпускаемой продукции.

Описание объекта внедрения прилагается и является неотъемлемой частью Акта.

От Белорусско-Российского университета:

Начальник НИСа  Мельников С.Ф.Разработчики  Якимов А.И.  
 Альховик С.А. Захарченков К.В. Суворов С.А.

От завода органического синтеза:

Главный технолог  Аксенов А.В.

Барбарчик А.В.

Гудына В.Е.

## ОПИСАНИЕ ОБЪЕКТА ВНЕДРЕНИЯ

**Математическая модель технологического процесса получения ДМТ, интегрированная в автоматизированную информационную систему предприятия.**

### 1. Краткая характеристика объекта внедрения и его назначение

Модель технологического процесса получения ДМТ на уровне распределения производственной нагрузки между цехами ДМТ ориентирована для применения на предприятиях с ERP-системами управления и интегрирована в корпоративную информационную систему предприятия.

Программное обеспечение получает данные из информационной системы IC: ПРЕДПРИЯТИЕ для статистического анализа в пакете STATISTICA и получения функциональных зависимостей для математической модели. При решении задачи оптимизации использован табличный процессор MS EXCEL с встроенными численными методами оптимизации SOLVER.

### 2. Фамилия и инициалы разработчиков, место работы, должность:

Якимов А.И., каф. «Автоматизированные системы управления» Белорусско-Российского университета, канд. техн. наук, доцент;

Альховик С.А., каф. «Автоматизированные системы управления» Белорусско-Российского университета, канд. техн. наук, доцент;

Захарченков К.В., каф. «Автоматизированные системы управления» Белорусско-Российского университета, аспирант.

Суворов С.А., каф. «Автоматизированные системы управления» Белорусско-Российского университета, студент.

### 3. Фамилия и инициалы работников, использующих разработку:

Аксенов А.В., главный технолог.

4. Начало использования объекта внедрения (месяц, год): июнь 2005 г.

5. Расчетный годовой экономический эффект от использования объекта внедрения: 100 000 000 (Сто миллионов) рублей.

От Белорусско-Российского университета

Начальник НИСа  Мельников С.Ф.

Разработчики  Якимов А.И.

 Альховик С.А.

 Захарченков К.В.

 Суворов С.А.

От завода органического синтеза:

Главный технолог  Аксенов А.В.

 Барбарчик А.В.

 Гудина В.Е.

«УТВЕРЖДАЮ»

Проректор по научной работе ГУВПО  
"Белорусско-Российский университет"

В. М. Пашкевич

« 2017 г.

**СПРАВКА**

о внедрении результатов диссертационной работы Якимова А.И. «Теоретические основы технологии имитационного моделирования и принятия решений в информационных системах промышленных предприятий»

1. Математическая модель технологического процесса получения диметилтерефтолата (ДМТ) на уровне распределения производственной нагрузки между цехами ДМТ интегрирована в автоматизированную информационную систему завода органического синтеза ОАО «Могилевхимволокно». Расчетный годовой экономический эффект от использования объекта внедрения 100 000 000 (Сто млн белор. рублей) (основание: акт о внедрении результатов НИР на заводе органического синтеза ОАО «Могилевхимволокно» от 20.06.2005 г.).
2. Имитационная модель динамики производственно-экономической деятельности ОАО «Обувь», программное обеспечение ShagovitaPro переданы в опытную эксплуатацию, разработаны методические рекомендации по использованию модели (основание: акт сдачи-приемки научно-технической продукции по договору № ХД 0877, составлен 31.12.2010 г. ).
3. Модели и алгоритмы управления складскими запасами, модели и алгоритмы планирования на уровне цехов и производственных операций с учетом имеющихся производственных мощностей, материалов и спецификаций продукции, реализованные в программном комплексе, интегрированном в корпоративную информационную систему ООО «СМИТ-Ярцево» (Россия, Смоленская обл.). Экономический эффект от использования программного комплекса составил 3 954 080 российских рублей в масштабе цен 2011 г. (основание: акт о практическом использовании результатов исследования, расчет экономического эффекта от 28.10.2011 г.).
4. Полученные результаты исследований (методика решения задачи определения последовательности оптимальных управлений и состояний технологического процесса способом пошаговой оптимизации) внедрены на ОАО «Моготекс» (г. Могилев, Беларусь) с экономическим эффектом от оптимизации (при производстве 1000 м.п. рассматриваемой группы тканей: артикул 10С3-КВ) за счет уменьшения стоимости расхода ресурсов (химикаты, ТВВ, красители, электроэнергия, пар, газ) на 14%, что соответствует 403 тыс. руб. в ценах 2011 г., 684 тыс. руб. в ценах 2012 г. и уменьшения времени на 9%, что соответствует 0,63 ч.; годовой экономический эффект при годовой программе 110242 м. в ценах 2011 г: 44 399 тыс. бел. руб, в ценах 2012 г: 75 370 тыс. бел. руб. (основание: акт приемки завершенной НИР «Разработка методов и программных средств интеллектуальной информационно-аналитической системы поддержки принятия решений, интегрированной в комплексную информационную систему промышленного предприятия», №ГР20110379 от 10.01.2014 г.)

Заместитель проректора по НИЧ,  
канд. техн. наук, доц.

С.Ф. Мельников

Заведующий кафедрой автоматизированных  
систем управления канд. техн. наук, доц.

С.К. Крутолевич

Доцент кафедры автоматизированных систем  
управления канд. техн. наук, доц.

А.И. Якимов