

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Воронежский государственный технический университет»
(ФГБОУ ВО «ВГТУ», ВГТУ)

На правах рукописи

ТИШУКОВ Борис Николаевич

**СТРУКТУРНОЕ МОДЕЛИРОВАНИЕ СЛОЖНЫХ ОБЪЕКТОВ
НА ОСНОВЕ ИНТЕГРИРОВАННОЙ ПРОГРАММНОЙ СРЕДЫ
ЧИСЛЕННОЙ ОПТИМИЗАЦИИ**

Специальность 05.13.18 – «Математическое моделирование, численные методы
и комплексы программ»

ДИССЕРТАЦИЯ

на соискание ученой степени
кандидата технических наук

Научный руководитель: доктор технических наук, профессор,
Заслуженный деятель науки РФ
Львович Яков Евсеевич

Воронеж 2019

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	4
1. АНАЛИЗ МЕТОДОВ ЧИСЛЕННОЙ ОПТИМИЗАЦИИ СЛОЖНОСТРУКТУРИРОВАННЫХ ОБЪЕКТОВ И ВОЗМОЖНОСТИ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ПРОЦЕССА ПОИСКА С ИСПОЛЬЗОВАНИЕМ ИМИТАЦИОННОГО И НЕЙРОСЕТЕВОГО МОДЕЛИРОВАНИЯ.....	14
1.1. Характеризация класса сложноструктурированных объектов	14
1.2. Особенности применения численных методов структурной оптимизации	18
1.3. Особенности применения нейросетевого моделирования для решения задач прогнозирования.....	27
1.4. Возможность использования аппарата имитационного моделирования для обработки статистических данных и проверки полученных решений на оптимальность.....	35
1.5. Необходимость построения интегрированной среды численной оптимизации, нейросетевого и имитационного моделирования для повышения эффективности вычислительного процесса.....	41
2. РАЗРАБОТКА ОПТИМИЗАЦИОННОЙ МОДЕЛИ И ПРОЦЕДУР ЧИСЛЕННОЙ ОПТИМИЗАЦИИ СЛОЖНОСТРУКТУРИРОВАННЫХ ОБЪЕКТОВ.....	46
2.1. Разработка и описание особенностей оптимизационной модели сложноструктурированных объектов	47
2.2. Формирование процедур численной оптимизации в рандомизированной вычислительной среде на основе популяционных алгоритмов.....	55
2.3. Ориентация популяционных алгоритмов на оптимизацию объектов исследуемого типа.....	59
3. ИНТЕГРАЦИЯ ПРОЦЕДУР ОПТИМИЗАЦИИ, НЕЙРОСЕТЕВОГО И ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ В ЕДИНУЮ ВЫЧИСЛИТЕЛЬНУЮ СРЕДУ.....	65
3.1. Структура интегрированной вычислительной среды численной оптимизации, нейросетевого и имитационного моделирования.....	66
3.2. Процедуры генерации и трансформации последовательностей псевдослучайных чисел.....	70

3.3. Синхронизация режимов нейросетевого и имитационного моделирования и численной оптимизации в рамках вычислительной среды.....	79
4. АНАЛИЗ ЭФФЕКТИВНОСТИ РАЗРАБОТАННЫХ СРЕДСТВ ОПТИМИЗАЦИИ И МОДЕЛИРОВАНИЯ НА ОСНОВЕ РЕЗУЛЬТАТОВ ВЫЧИСЛИТЕЛЬНОГО И НАТУРНОГО ЭКСПЕРИМЕНТОВ.....	87
4.1. Характеристика программного комплекса и его структура.....	87
4.2. Особенности проведения и результаты вычислительного эксперимента.....	90
4.3. Особенности проведения и результаты натурального эксперимента в рамках информационной системы управления.....	102
ЗАКЛЮЧЕНИЕ.....	111
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ.....	113
ПРИЛОЖЕНИЕ 1.....	123
ПРИЛОЖЕНИЕ 2.....	125
ПРИЛОЖЕНИЕ 3.....	126

ВВЕДЕНИЕ

Актуальность темы. На современном этапе ускоренного развития промышленных и технических систем, связанного с возрастанием их сложности и преобладания модульности построения, распространение получили инфокоммуникационные, технологические и транспортные системы, эффективность функционирования которых достигается за счет целенаправленного варьирования их структурными компонентами. Поэтому возникает необходимость в выделении сложных объектов, относящихся к классу сложноструктурированных. Проблематика, связанная с обеспечением высокого качества функционирования таких объектов связана с необходимостью развития математических средств структурного моделирования и оптимизации.

Вопросы моделирования и численной оптимизации в данной области рассматриваются в рамках ряда концепций: структурной адаптации, структурной и многоальтернативной оптимизации (Батищев Д.И., Цыпкин Я.З., Растрингин Л.А., Львович Я.Е., Поляк Б.Т., Курейчик В.Г., Пшеничный Б.Н. и другие). В этих исследованиях преимущественно используются два подхода. Один основан на переходе к эквивалентным задачам параметрической оптимизации и применении ориентированных на них всего арсенала численных методов поиска экстремума. Однако такое преобразование приводит к значительному увеличению размерности задач и повышению трудоемкости вычислительного процесса. Для второго подхода характерен переход к эквивалентной задаче за счет механизмов рандомизации и сглаживания с сохранением размерности исходных переменных, заданных на конечном дискретном множестве. Здесь трудоемкость вычислительного процесса в ряде случаев возрастает за счет недостаточных возможностей управления переборными схемами стохастического поиска. При этом возможности применения в задачах исследуемого класса имитационного моделирования (Советов Б.Я., Самарский А.А., Рыжиков Ю.И., Емельянов А.А., Юсупов Р.М., Шрейдер Ю.А., Соболев И.М., Бусленко Н.П. и другие), нейросетевого

моделирования (Розенблатт Ф., Кохоннен Т., Галушкин А.И., Горбань А.Н., Румельхарт Д.Е., Дунин-Барковский В.Л., Малинецкий Г.Г., Головкин В.А., Куссуль Э.М.), а также совместного использования имитационного и нейросетевого подходов в условиях структурной оптимизации исследуемого класса объектов в полной мере не представлены.

Это обуславливает необходимость обеспечения возможности интеграции перечисленных выше двух подходов моделирования и численной оптимизации сложноструктурированных объектов на множествах дискретных и непрерывных переменных. Кроме того, при построении единой вычислительной среды, требуется расширение механизмов рандомизации и сглаживания, комплексирования стохастических и популяционных алгоритмов.

Таким образом, актуальность темы определяется необходимостью интеграции перечисленных подходов структурного моделирования в рамках интегрированной программной среды численной оптимизации сложноструктурированных объектов на множествах дискретных и непрерывных переменных путем расширения механизмов рандомизации и сглаживания, комплексирования стохастических и популяционных алгоритмов.

Работа выполнена в соответствии с основным научным направлением Воронежского государственного технического университета «Интеллектуальные информационные системы» (Приказ по ВГТУ от 25.04.2017г. № 182).

Цель и задачи исследования. Целью диссертационной работы является повышение эффективности структурного моделирования сложных объектов на основе комплекса имитационных и нейросетевой моделей, интегрированных в программную среду численной оптимизации.

Для достижения поставленной цели необходимо решить следующие задачи:

- разработать формализованное описание процессов анализа и оптимизации сложноструктурированных объектов с использованием аппарата нейросетевого и имитационного моделирования;

- разработать нейросетевую модель для решения задачи прогнозирования параметров сложноструктурированных объектов;
- разработать комплекс имитационных моделей для проведения анализа оценки эффективности функционирования объектов со сложной структурой;
- сформировать комплекс моделей и адаптивных алгоритмов численной оптимизации сложноструктурированных объектов;
- разработать средства интеграции разработанных моделей анализа и прогнозирования, а также численной оптимизации в вычислительную среду;
- разработать интегрированную программную среду моделирования и численной оптимизации сложноструктурированных объектов и оценить его эффективность в режиме вычислительного и натурального экспериментов.

Объект исследования: сложноструктурированные объекты из различных объектных областей.

Предмет исследования: математические модели (нейросетевые и имитационные) и модели численной оптимизации, обеспечивающие эффективность функционирования сложноструктурированных объектов.

Методы исследования. Для решения поставленных в работе задач используются основные положения теории математического моделирования и численной оптимизации, теории вероятностей и математической статистики, теории информации, методы имитационного и нейросетевого моделирования, стохастического и популяционного подходов к решению экстремальных задач.

Положения, выносимые на защиту:

1. Показано, что использование предложенного комплекса имитационных и нейросетевой моделей позволяет повысить эффективность структурного моделирования функционирования объектов и прогнозирования их параметров.

2. Доказано, что разработанный комплекс моделей и адаптивных процедур численной оптимизации на основе популяционных алгоритмов позволяет расширить возможность учета особенностей функционирования сложноструктурированных объектов.

3. Показано, что базовыми элементами вычислительной среды структурного моделирования и поиска оптимального варианта структуры сложного объекта являются процедуры трансформации параметров распределений дискретных и непрерывных псевдослучайных чисел с различными законами распределения в процессе численного решения экстремальной задачи путем комплексирования адаптивного алгоритма рандомизированного поиска и популяционного алгоритма обезьяньего поиска.

4. Доказано, что математическая модель синхронизации режимов нейросетевого и имитационного моделирования и численной оптимизации лежит в основе условий останова основных процессов интегральной вычислительной среды.

5. Показано, что разработанная интегрированная среда структурного моделирования и оптимизации сложноструктурированных объектов имеет возможность интеграции в автоматизированные системы проектных организаций, занимающихся проектированием сложноструктурированных промышленных систем.

Тематика работы. Соответствует следующим пунктам специальности 05.13.18: п.4. Реализация эффективных численных методов и алгоритмов в виде комплексов проблемно-ориентированных программ для проведения вычислительного эксперимента; п.5. Комплексные исследования научных и технических проблем с применением современной технологии математического моделирования и вычислительного эксперимента; п.8. Разработка систем компьютерного и имитационного моделирования.

Научная новизна. В диссертационной работе получены следующие результаты, характеризующиеся научной новизной:

- способ формализованного описания сложноструктурированных объектов с использованием комплексной системы имитационного и нейросетевого моделирования, отличающийся математическими условиями синхронизации методов моделирования при их включении в единый поисковый

цикл численной оптимизации и позволяющий оценивать эффективность альтернативных вариантов структуры исследуемого объекта;

- комплекс моделей сложноструктурированных объектов, отличающийся многовариантностью формализации экстремальных и граничных требований к их функционированию и позволяющий выбрать адекватный вариант модели для проведения вычислительного эксперимента;

- адаптивные процедуры численной оптимизации, отличающиеся комплексированием алгоритма рандомизированного поиска и популяционного алгоритма и позволяющие сформировать единую вычислительную среду структурного моделирования и поиска оптимального варианта структуры сложного объекта;

- структура интегрированной программной среды, отличающаяся формой обмена данными между пакетами имитационного и нейросетевого моделирования с ядром программного комплекса программой численной оптимизации и позволяющая реализовать взаимодействие с промышленной информационной системой при проведении вычислительного и натурального экспериментов.

Практическая значимость и результат внедрения. Разработанные нейросетевая и имитационные модели, а также модели и алгоритмы численной оптимизации позволяют:

- выполнять анализ имеющейся структуры сложных объектов, а также определять проблемные элементы их структуры;

- осуществлять переход от общего вида задачи для сложноструктурированного объекта к частному;

- обрабатывать статистические данные об объекте исследования и на основе их анализа формировать модель объекта, учитывающую возможность варьирования типами и параметрами структурных компонентов;

- формировать с использованием программной среды множество перспективных вариантов функционирования сложноструктурированных

объектов в информационной системе транспортной сети и осуществлять проверку эффективности полученных вариантов.

Полученные в работе теоретические результаты могут быть использованы в научно-исследовательских и проектных организациях, занимающихся проектированием сложноструктурированных промышленных систем.

Результаты работы внедрены в автоматизированную систему проектирования дорожной и транспортной сети г.Воронежа, а также использованы в учебном процессе кафедры систем автоматизированного проектирования и информационных систем в рамках курсов «Методы оптимизации», «Моделирование систем», «Моделирование процессов и систем». Результаты внедрения подтверждены соответствующими актами.

Апробация работы. Результаты представленной работы докладывались и обсуждались на научных мероприятиях различного уровня: Всероссийской молодежной конференции «Интеллектуальные технологии будущего. Естественный и искусственный интеллект» (Воронеж, 2011), всероссийской конференции «Математическое моделирование в технике и технологии» (Воронеж, 2011), всероссийской научной конференции «Интеллектуальные информационные системы» (Воронеж, 2011), всероссийской молодежной научной школы «Инженерия знаний. Представление знаний: состояние и перспективы» (Воронеж, 2012), международной молодежной конференции в рамках фестиваля науки «Математические проблемы современной теории управления системами и процессами» (Воронеж, 2012), научной конференции студентов, аспирантов и молодых ученых «Инновационные технологии на базе фундаментальных научных разработок – прорыв в будущее» (Воронеж, 2012), межвузовском конкурсе инновационных проектов «Кубок инноваций» (Воронеж, 2013), международной лектории, посвященном 30-летию кафедры САПРИС и памяти ведущих ученых в области САПР (Воронеж, 2014), международной научно-практической конференции «Виртуальное моделирование, прототипирование и промышленный дизайн» (Тамбов, 2014), II

международной научно-практической конференции «Антропоцентрические науки: инновационный взгляд на образование и развитие личности» (Воронеж, 2015), всероссийской конференции «Интеллектуальные информационные системы» (Воронеж, 2015), II международной научно-практической конференции «Виртуальное моделирование, прототипирование и промышленный дизайн» (Тамбов, 2015), межвузовском конкурсе инновационных проектов «Кубок инноваций - 2016» (Воронеж, 2016), региональной научно-практической конференции студентов, аспирантов и молодых ученых «Информационные технологии, программные продукты и телекоммуникационные системы на базе фундаментальных научных разработок» (Воронеж, 2016), III Международной конференции «Виртуальное моделирование, прототипирование и промышленный дизайн» (Тамбов, 2016), всероссийской научной конференции с международным участием «Интеллектуальные информационные системы» (Воронеж, 2017), IV Международной конференции «Виртуальное моделирование, прототипирование и промышленный дизайн» (Тамбов, 2017), Международная конференция «Человеческий капитал в формате цифровой экономики», посвященная 90-летию С.П. Капицы (Москва, 2018).

Результаты работы представлялись на конкурсы и выставки различных уровней, где были отмечены дипломами победителя и призера.

Публикации. По результатам диссертационного исследования опубликовано 15 научных работ, в том числе 4 – в изданиях, рекомендованных ВАК РФ; 1 свидетельство о государственной регистрации программ для ЭВМ. В работах, опубликованных в соавторстве и приведенных в автореферате, лично автором предложены адаптивные алгоритмы для решения задач численной оптимизации сложноструктурированными объектами [1,2,10], математическая модель синхронизации режимов численной оптимизации, нейросетевого и имитационного моделирования [3], инвариантная модель сложноструктурированного объекта и ее модификации для задачи оптимизации [4], алгоритмы и модели проведения имитационного моделирования [7,13,15],

структура вычислительной среды, лежащая в основе программного комплекса [9,11], программная реализация разработанных оптимизационных алгоритмов [5].

Структура и объем работы. Диссертация состоит из введения, четырех глав, заключения, списка литературы и приложений. Основной текст работы изложен на 112 страницах машинописного текста, включая 25 рисунков, 2 таблицы и 64 формулы. Список литературы содержит 104 наименования.

Во введении диссертационной работы обоснована актуальность темы исследования, сформулированы цель и задачи, научная новизна и практическая значимость полученных результатов, изложено краткое содержание диссертации.

В первой главе сформулировано понятие сложноструктурированного объекта, рассмотрены его особенности и свойства. На основе полученных результатов разработана схема цикла формирования модели для оптимизационной задачи и поиска ее решения. Представлены результаты анализа численных методов структурной оптимизации и возможности их применения для задач оптимизации объектов исследуемого типа. Рассмотрены особенности применения нейросетевого моделирования для задач прогнозирования. Описываются практические аспекты и алгоритмизация процедур использования имитационного моделирования для обработки статистических данных и оценки эффективности полученных вариантов решений перед их оценкой лицом, принимающим окончательное решение.

На основе анализа полученных результатов и, исходя из особенностей сложноструктурированных объектов, сделано заключение о необходимости разработки интегрированной вычислительной среды численной оптимизации, нейросетевого и имитационного моделирования.

Вторая глава посвящена разработке оптимизационной модели и процедур численной оптимизации сложноструктурированных объектов. В частности, осуществляется разработка оптимизационной модели сложноструктурированных объектов в виде инвариантной модели μ_l и

возможности ее модификаций в частные модели $\overline{\mu_2, \mu_5}$ на основе свойств объекта рассматриваемого типа. Представлены результаты разработки схемы формирования комплекса оптимизационных моделей в зависимости от свойств сложноструктурированного объекта и структурной схемы обобщенного алгоритма компенсирования метода многоальтернативной оптимизации и вариационного моделирования при численной оптимизации сложноструктурированных объектов. Описывается возможность решения поставленной задачи для сложноструктурированных объектов в статическом и динамическом режимах, приводятся результаты формирования оптимизационных моделей для каждого из режимов. Осуществляется формирование процедуры численной оптимизации на основе популяционных алгоритмов, а затем происходит их ориентация для решения поставленной задачи. Приводятся результаты построения адаптивных алгоритмов моделирования объектов в виде структурных схем для статического и динамического режимов.

Исходя из полученных результатов, делается вывод о том, что наиболее приемлемым методом численной оптимизации в предлагаемой вычислительной среде является метод, основанный на сочетании алгоритмов многоальтернативной оптимизации и популяционных алгоритмов.

В третьей главе описывается разработка модели вычислительной среды, а также описываются базовые процедуры, входящие в структуру среды (погружение в рандомизированную среду, генерация случайных последовательностей, нейросетевое и имитационное моделирование, трансформационные процедуры и синхронизация режимов интегрированной вычислительной среды).

Описывается алгоритмическая процедура дополнения механизмов рандомизации и сглаживания предложенными в работе механизмами комплексного моделирования, синхронизации режимов и итерационной трансформации стохастического процесса направленного поиска экстремума.

Доказывается необходимость объединения всех процедур между модулями генерация случайных последовательностей и трансформации параметров генераторов.

Приводятся результаты разработки алгоритмов интеграции описанных процедур в единую вычислительную среду и осуществления их взаимосвязи.

Четвертая глава посвящена описанию модульной структуры программной среды, основу которой составляет разработанная вычислительная среда, и оценке ее эффективности на основе результатов проведения вычислительного и натурального экспериментов.

Описывается методика взаимодействия разработанной программной среды с системами предприятий, основанная на использовании XML-файлов.

Предложена процедура проведения вычислительного эксперимента на основе разработанных алгоритмов функционирования предложенной системы, а также приводятся результаты эксперимента на натуральных объектах транспортной сети г.Воронежа, соответствующих исследуемому типу объектов.

Разработанная вычислительная среда позволяет сформировать множество перспективных вариантов по изменению структуры объекта с целью повышения эффективности его функционирования. Окончательный вариант выбирается экспертным путем из сформированного множества полученных вариантов на основе анализа полученных результатов и соответствия их начальным условиям.

В заключении приведены основные научные и практические результаты исследования. Полученные результаты доказывают теоретическую и практическую значимость разработанной вычислительной среды, а также входящих в нее моделей и алгоритмов, в процессе решения задачи повышения эффективности использования сложноструктурированных объектов.

1 АНАЛИЗ МЕТОДОВ ЧИСЛЕННОЙ ОПТИМИЗАЦИИ СЛОЖНОСТРУКТУРИРОВАННЫХ ОБЪЕКТОВ И ВОЗМОЖНОСТИ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ПРОЦЕССА ПОИСКА С ИСПОЛЬЗОВАНИЕМ ИМИТАЦИОННОГО И НЕЙРОСЕТЕВОГО МОДЕЛИРОВАНИЯ

1.1 Характеризация класса сложноструктурированных объектов

В настоящее время во многих сферах стали определяться классы объектов, для которых становятся характерны новые формы и принципы организации управления ими. Их особенностями является то, что использование существующих форм и методов управления ими в стандартном виде в большинстве случаев не позволяет решить поставленные задачи и достичь необходимого эффекта. Для этого требуется проводить эволюцию форм и методов управления объектами, скрещивать уже существующие и тем самым, выводить новые формы, отвечающие возникшим условиям.

Именно это и послужило причиной выбора в качестве объекта исследования класс сложноструктурированных. Чтобы сформулировать определение рассматриваемого класса объектов, дадим понятие структурной и вариативной формам управления.

Структурная форма управления – это такая форма управления, особенностью которой является то, что при сформированной структуре объекта изменяется её архитектура, типы и параметры компонентов. Большое внимание описанию и характеристике структурной формы управления объектами уделено в работах [46].

Вариативная форма управления – это такая форма, при которой решение, позволяющее повысить эффективность процесса управления выбранным объектом, определяется путем выбора из множества альтернативных вариантов наиболее подходящего для конкретного условия.

На основании вышеизложенного сформулируем определение для структурновариативной формы управления.

Структурновариативная форма управления сложными объектами – это такая форма, при которой эффективность функционирования объекта достигается при помощи поиска структуры объекта исследования, которая будет являться наиболее рациональной и эффективной в рамках существующих условий путем выбора из множества альтернативных вариантов наиболее подходящего для конкретного условия.

Исходя из рассмотренных выше свойств, следует, что отличительная особенность класса сложноструктурированных объектов является то, что решение задач оптимизации ими требует поиск рациональной (оптимальной) их структуры за счет перебора возможных вариантов изменения типов и параметров компонентов с учетом заданных ограничивающих факторов и допустимых параметров [96].

Таким образом, сложноструктурированный объект – это такой объект, эффективность функционирования которого достигается при помощи поиска структуры объекта исследования, которая будет являться наиболее рациональной и эффективной в рамках существующих условий путем выбора из множества альтернативных вариантов наиболее подходящего для конкретного условия.

На основе проведенного анализа характеристики объектов различного типа [8,13,14,21,38,46,86] сформулируем свойства, которыми обладают сложноструктурированные объекты:

1. Варьируемые структурные компоненты $s = \overline{1, S}$ в формализованном виде представляются вектором X , включающем 3 подмножества X_1, X_2, X_3 переменных:

- $X_1 = (x_1, \dots, x_j, \dots, x_J)$ - подмножество альтернативных переменных, при этом

$$x_j = \begin{cases} 1, & \text{если некоторый } j\text{-й элемент или связь включаются} \\ & \text{в структуру объекта,} \\ 0, & \text{в противном случае, } j = \overline{1, J}; \end{cases}$$

- $X_2 = (x_1, \dots, x_l, \dots, x_L)$ - подмножество переменных, принимающих дискретные значения и характеризующих типы структурных компонентов;

$$x_l = (x_{l_1}, \dots, x_{l_m}, \dots, x_{l_M}), x_{l_m} \geq 0, m = \overline{1, M};$$

- $X_3 = (x_1, \dots, x_n, \dots, x_N)$ - подмножество непрерывных переменных, изменение которых ограничено некоторым интервалом $x_n^{MIN} \leq x_n \leq x_n^{MAX}, x_n^{MIN} \geq 0, n = \overline{1, N}$, и характеризует параметры структурных компонентов.

2. Следует рассматривать динамический, зависящий от момента времени $\tau \in [\tau_0; \tau_g]$ и статический режимы функционирования объекта.

3. Адекватное отражение зависимости показателей натурального объекта $y = (y_1, \dots, y_i, \dots, y_I)$, характеризующих заданные цели, от варьируемых структурных компонентов идентифицируется некоторой вычислительной средой, позволяющей определять значения составляющей вектора y при зафиксированном варианте структуры s^0 и соответствующем значении вектора X^0 :

$$y = f_{ec}(X^0) \sim f_n(s^0),$$

где знаком \sim обозначаем подобие.

4. Достижение заданных целей определяется набором экстремальных

$$Y_{i1} = f_{i1}(x) \rightarrow \max(\min), i_1 = \overline{1, I_1}$$

и граничных

$$y_{i2} = f_{i2}(x) \rightarrow b_{i2}, i_2 = \overline{1, I_2}$$

требований, где b_{i2} – допустимый уровень показателя y_{i2} .

5. В качестве сложноструктурированных объектов для вычислительных и натуральных экспериментов будем рассматривать сетевые инфокоммуникационные, технологические и транспортные системы.

После того, как мы провели формализованное описание класса сложноструктурированных объектов в виде $f_{ec}(X)$, перейдем к задачам оптимизации таких объектов [8,13,14].

Очевидно, что оптимальность структуры объекта будет определяться по какому-либо из критериев, в наибольшей степени влияющих на нее. Именно

этот критерий и будет являться критерием оптимальности структуры объекта и будет выражаться целевой функцией в математической модели задачи [97].

Схематически цикл работы с оптимизационной задачей для сложноструктурированного объекта можно представить в виде схемы на рисунке 1.1.



Рис. 1.1. Схема формирования оптимизационной модели и численного решения задачи оптимизации

Проанализируем возможность применения численных методов структурной оптимизации для сложноструктурированных объектов, моделирование которых осуществляется с использованием имитационного и нейросетевого подходов.

1.2 Особенности применения численных методов структурной оптимизации

Рассмотрим особенности применения численных методов оптимизации для исследуемого класса объектов и их основные виды [3,4,8,13,14,21,53,88,101]. Для этого сначала определимся, что будем понимать под структурной оптимизацией.

Структурная оптимизация заключается в подборе такой структуры исследуемого объекта, которая с точки зрения управления им будет являться наиболее эффективной в условиях поставленной цели и заданных допустимых ограничениях [23].

Исходя из количества рассматриваемых в задаче критериев оптимальности, задача может являться однокритериальной или многокритериальной. Если в задаче имеются ограничения, то такая задача будет называться условной оптимизационной задачей [26].

В случае решения задач, связанных с оптимизацией объектов со сложной структурой, мы будем иметь дело с многокритериальной условной оптимизацией.

Как правило, для решения задач структурной оптимизации многокритериального условного типа используются итеративные численные методы дискретного математического программирования. Подробное описание методов такого типа представлено в работах [8,13,14,26,40,100].

Суть указанных выше численных методов оптимизации заключается в организации вычислительного процесса до тех пор, пока не станет понятно, что улучшение найденного на предыдущих шагах решения невозможно. При этом из области допустимых решений выбирается так называемый начальный вариант решения поставленной задачи, от которого начинается перемещение по ней в соответствии с действиями, указанными в выбранном численном методе (оптимизационном алгоритме).

Рассмотрим возможность классификации выбранной группы методов по способу определения оптимального решения. По данному классификационному признаку мы можем группу итеративных численных методов дискретного математического программирования (численных методов решения задач структурной оптимизации) разделить на три подгруппы [68]. Остановимся на каждой из них более подробно.

О методах (алгоритмах), относящихся к первой подгруппе, можно сказать, что при их применении на каждом итерационном шаге решение поставленной задачи улучшается. После определенного числа выполненных итераций дальнейшее улучшение искомого решения становится невозможным.

Ко второй подгруппе методов относятся алгоритмы, обеспечивающие улучшение решения на последовательных итерациях, но гарантирующие получение оптимального решения как предела некоторого бесконечного вычислительного процесса.

Третья подгруппа численных методов (алгоритмов) основана на принципе работы метода проб и ошибок. При выборе для решения методов, относящихся к третьей подгруппе, последовательные поиски позволяют улучшить искомый результат, но монотонное улучшение оптимального решения решаемой задачи не гарантируется.

Наиболее распространенными методами рассматриваемых подгрупп являются метод ветвей и границ, метод отсечений, а также генетические (популяционные) методы.

Рассмотрим более основательно каждый из названных выше численных методов решения задач структурной оптимизации дискретного типа (более подробно дискретный тип объектов и систем рассматривается в [13,14]).

Первым является метод ветвей и границ. Этот метод является общим методом поиска оптимального решения задач дискретного программирования. Характерной особенностью рассматриваемого метода является использование в его вычислительном алгоритме идеи, которая заключается в последовательном использовании конечного числа множества вариантов решения исходной

задачи, что в свою очередь приводит к переходу от полного перебора возможных вариантов решения к сокращенному [24].

Он начинает свою работу с определения нижней и верхней границ для условия начальной задачи. Если верхняя и нижняя границы совпадают, то полученный результат является оптимальным значением, и алгоритм прекращает работу. Иначе, множество переменных разбивается на несколько собственных подмножеств, объединение которых совпадает с исходным множеством. Эти подзадачи становятся потомками исходной. Далее алгоритм применяется рекурсивно к каждой из подзадач, создавая дерево подзадач. Если оптимальное решение найдено для некоторой подзадачи, то оно является достижимым для исходной задачи (не обязательно оптимальным), но так как оно достижимо, его можно использовать для обрезания ветвей у исходного дерева. Процесс поиска продолжается до тех пор, пока каждая из подзадач не будет решена или выкинута или до тех пор, пока не будет достигнут заданный порог между лучшим из найденных решений и нижней границей $f(x)$ для всех нерешенных задач.

Алгоритм решения задачи структурной оптимизации, основанный на методе ветвей и границ можно представить в следующем виде:

1⁰. Для стандартной оптимизационной задачи дискретного типа осуществляем вычисление верхней (v_0) и нижней (n_0) оценок. При этом $v_0 = F(x^0)$, а n_0 обеспечивается значением некоторого решения задачи x^0 .

2⁰. Будем считать, что $F^* = F(x^0)$, а $x^* = x^0$. В данном случае в качестве F^* обозначим оптимальное значение искомой функции, а x^* - оптимальное решение исходной задачи.

3⁰. Выполним проверку равенства $v_0 = n_0$. Если это равенство верно, то переходим к шагу 4 алгоритма; в противном случае осуществляем переход к шагу 5.

4⁰. Поставленная задача структурной оптимизации объекта исследования решена. При этом x^* является оптимальным решением.

5⁰. Задаем $m = 0$, где m – номер вершины, в которую следует произвести ветвление, а 0 присваивается так называемому корню.

6⁰. Осуществление ветвления в вершине m . На этом этапе происходит включение получаемых вершин в массив финальных вершин на данном этапе решения задачи и исключение самой вершины m из массива.

7⁰. Вычисление нижних и верхних оценок в полученных на предыдущем шаге работы алгоритма финальных вершинах.

8⁰. Проверка условия на наличие новой вершины в дереве решений, для которой нижняя оценка при некотором решении x' больше F^* . В случае выполнения условия осуществляется переход к шагу 9 данного алгоритма; в противном случае переходим к шагу 10.

9⁰. Будем считать, что $F^* = F(x')$, а $x^* = x'$.

10⁰. Проводим анализ оценок по текущим финальным вершинам: если вершина считается неперспективной, то мы ее отбрасываем; в случае, если нижняя и верхняя оценки для вершины совпадают, то мы выделяем эти вершины.

11⁰. Строим массив по полученным результатам для так называемых претендентов на ветвление.

12⁰. Если сформированный на шаге 12 данного алгоритма массив оказывается пустым, то переходим к шагу 4 алгоритма. В противном случае переходим к шагу 13.

13⁰. Осуществляем выбор вершины, из которой будем производить дальнейшее ветвление. Номер такой вершины заносим в переменную K .

14⁰. Значение K присваиваем переменной m , после чего переходим к выполнению всех шагов алгоритма, начиная с шага 6.

Более детально метод ветвей и границ рассмотрен в [5,8,21,40].

Эффективность описанного выше метода имеет прямую зависимость от количества отброшенных на начальной стадии его выполнения вершин. В качестве ключевых этапов алгоритма, реализующего метод ветвей и границ, можно выделить следующие:

- получение верхних оценок (шаги 1 и 7);
- получение нижних оценок (шаги 1 и 7);
- процедура ветвления (шаг 6);
- определение номера вершины для осуществления дальнейшего ветвления по дереву возможных решений (шаг 13).

Следующим методом, который мы рассмотрим, будет метод отсечений. Полное описание этого метода представлено в [5,24,26,31]. Опишем принципы, на которых построен данный метод, его ключевые достоинства и алгоритм работы метода отсечений.

Алгоритм реализации метода отсечений (округлений) будет иметь следующий вид:

1⁰. Производится исключение условия целочисленности из требований к решению задачи, после чего осуществляется поиск оптимального решения с помощью симплекс-метода.

2⁰. В случае, если найденное на шаге 1 данного алгоритма решение является целочисленным, то искомое решение задачи структурной оптимизации решена. В противном случае переходим к шагу 3.

3⁰. Методом округления нецелочисленных позиций в векторе оптимального решения до ближайших целых чисел осуществляем построение дерева возможных целочисленных решений. При этом округление может осуществляться как в большую, так и в меньшую сторону.

4⁰. Из полученного на шаге 3 множества возможных целочисленных решений осуществляем выбор тех, которые удовлетворяют области допустимых целочисленных решений.

5⁰. Проводим расчет значений целевой функции F для каждого из выбранных на шаге 4 данного алгоритма возможных решений. Оптимальным считаем такое решение (значение вектора X), для которого значение функции F принимает максимальное (минимальное) значение (в зависимости от условия исходной задачи).

Главным недостатком метода отсечений, по мнению авторов [26], являются два фактора: в общем случае данный метод не всегда гарантирует получение истинного оптимального решения поставленной задачи или не позволяет определить допустимое решение, принадлежащее заданной области решений.

Следующим видом численных методов решения задач структурной оптимизации, которые мы рассмотрим, будут популяционные алгоритмы. Данные методы подробно отражены в работах [2,6,17,58].

Популяционные алгоритмы – это численные методы (алгоритмы) поиска, построенных на основе механизмов естественного отбора и принципов наследования. В алгоритмах данного типа используется эволюционный принцип, в соответствии с которым выживают наиболее приспособленные особи. Алгоритмы, построенные на генетических принципах, имеют ряд отличительных особенностей от численных методов оптимизации, рассмотренных в данной главе немного раньше. В частности, генетические алгоритмы обладают рядом отличительных свойств:

1. Кодирование параметров - генетические алгоритмы обрабатывают не значения параметров самой задачи, а их закодированную форму.
2. Операции на популяции – генетические алгоритмы осуществляют поиск решения исходя из единственной точки (начальное приближение) а из некоторой популяций.
3. Использование минимума информации о функций – генетические алгоритмы используют только целевую информацию, а не производные либо дополнительную информацию.
4. Рандомизация информации – генетические алгоритмы применяют вероятностные, а не детерминированные правила выбора.

Доказано [29], что методы оптимизации, в основе которых лежат популяционные алгоритмы, демонстрируют высокие результаты при решении сложных задач, заключающихся в переборе возможных вариантов решения на конечных дискретных множествах (большинство из которых NP-полные, т.е. не

решаются полным перебором за полиномиальное время). Примером такого типа задач является задача коммивояжера.

Для работы с популяционными алгоритмами используются термины, заимствованные из генетики в упрощенном виде и основные понятия линейной алгебры [7,14,31,59,99,101].

Вектор – упорядоченный набор чисел, называемых компонентами вектора.

Булев вектор – вектор, компоненты которого принимают значения из двух элементного (булева) множества значений, как правило $\{0, 1\}$.

Популяция – конечное множество особей.

Особь (индивидуум, организм) – набор хромосом с закодированными в них множествами параметров задачи, то есть решений, которые иначе называются точками в пространстве поиска (search points).

Хромосома (цепочка или кодовая последовательность) – это вектор генов.

Хромосома может быть представлена в виде булева вектора, полученного с помощью двоичного кодирования либо кода Грея. Хромосома обозначается, как правило, A .

Ген (свойство, знак или детектор) – атомарный элемент генотипа, в частности, хромосомы. Несет в себе наследственную информацию. Обозначается x .

Связь хромосомы и гена изображена на рисунке 1.2.

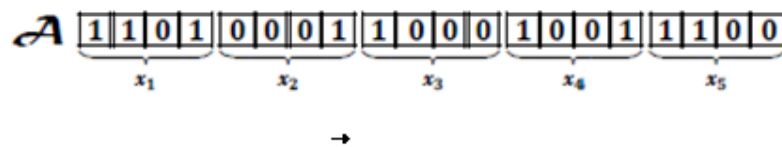


Рис. 1.2. Распределение наследственной информации по длине хромосомы.

Генотип (структура) – набор хромосом данной особи. Следовательно, особями популяции могут быть генотипы либо единичные хромосомы (в самом простом случае, генотип состоит из одной хромосомы).

Фенотип – набор значений, соответствующих данному генотипу, то есть

это декодированная структура или множество параметров задачи (решение, точка пространства поиска).

Аллель – значение конкретного гена.

Локус – позиция, указывающая место размещения данного гена в хромосоме.

Множество позиций генов – локи.

Одним из ключевых понятий в теории генетических алгоритмов является так называемая функция приспособленности (fitness function), или по-другому функция оценки.

Фитнесс-функция играет значительную роль, так как позволяет определить степень приспособленности определенных особей в заданной популяции и выбрать из них наиболее приспособленных (то есть имеющие максимальные значения фитнес-функции) в соответствии с эволюционным принципом выживания сильнейших [31].

Функция приспособленности так же получила свое название из генетики.

В задачах оптимизации функция приспособленности, как правило, оптимизируется, (точнее сказать, максимизируется) и называется целевой функцией. В задачах максимизации целевая функция преобразуется, и проблема сводится к максимизации.

В теории управления функция приспособленности может принимать вид функции погрешности, а в теории игр – стоимостной функции.

На каждой итерации генетического алгоритма приспособленность каждой особи данной популяции оценивается при помощи функции приспособленности, и на этой основе создается следующая популяция особей, составляющих множество потенциальных решений проблемы.

Очередная популяция в генетическом алгоритме называется поколением, а к вновь создаваемой популяции особей применяется термин «новое поколение» или «поколение потомков» [36].

Генетические операторы необходимы для того, чтобы применить принципы наследственности и изменчивости к популяции. Помимо

отличительных особенностей, о которых будет рассказано ниже, для всех операторов определено такое свойство как вероятность. То есть описываемые операторы не обязательно применяются ко всем скрещиваемым особям, что вносит дополнительный элемент неопределенности в процесс поиска решения. В данном случае, неопределенность не подразумевает негативный фактор, а является своеобразной «степенью свободы» работы генетического алгоритма.

В отличие от эволюции, происходящей в природе, генетический алгоритм только моделирует те процессы в популяции, которые являются существенными для развития.

Наиболее приспособленные особи получают возможность «воспроизводить» потомство с другими особями популяции, что приводит к появлению новых особей, сочетающих в себе некоторые характеристики, наследуемые ими от родителей. Менее приспособленные особи с меньшей вероятностью смогут воспроизвести потомков, так что те свойства, которыми они обладали, будут постепенно исчезать из популяции в процессе эволюции.

Таким образом, воспроизводится вся новая популяция допустимых решений, путем выбора лучших представителей предыдущего поколения, скрещивания их и получения множества новых особей. Это новое поколение будет содержать более высокое соотношение характеристик, которыми обладают хорошие члены предыдущего поколения. В итоге, хорошие характеристики распространяются по всей популяции. Скрещивание наиболее приспособленных особей приводит к тому, что исследуются наиболее перспективные участки пространства поиска. В конечном итоге, популяция будет сходиться к оптимальному решению задачи.

Имеется много способов реализации идеи биологической эволюции в рамках генетического алгоритма. Схема популяционного алгоритма представлена на рисунке 1.3.

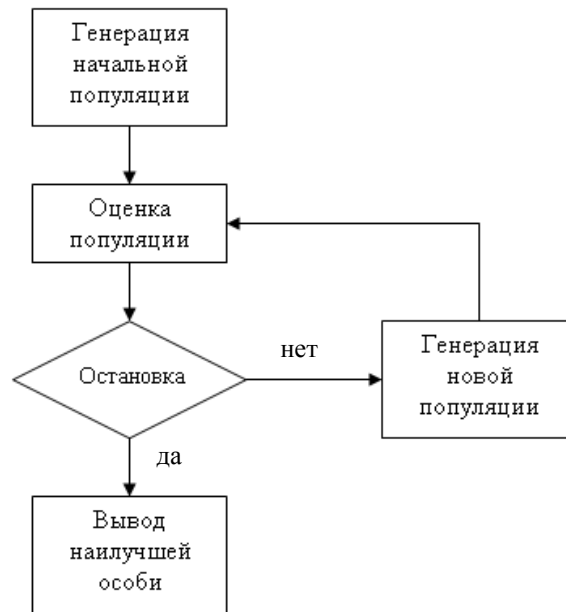


Рис. 1.3. Схема популяционного алгоритма оптимизации

1.3 Особенности применения нейросетевого моделирования для решения задач прогнозирования

В настоящее время различные области человеческой деятельности имеют необходимость в разрешении ситуаций, когда по имеющимся статистическим данным (вектор X), необходимо выполнить прогнозирование некоторой величины Y , зависящую от X (то есть X и Y имеют некоторую закономерность $Y = f(X)$), но при этом четко определить или задать эту зависимость мы не можем, так как значения вектора X обозначают значение исследуемого параметра в настоящий момент времени или представляет собой статистическую выборку за прошедший период времени t , а Y – значение этого параметра в будущий момент времени.

Одним из наиболее мощных и гибких математических аппаратов, позволяющих успешно решать задачи прогнозирования, является нейросетевое моделирование [35]. В процессе определения параметров прогнозирования с помощью нейронных сетей необходимо задавать те переменные, для которых имеется возможность их анализа и предсказания. Важным критерием является требуемый при этом уровень детализации. Выбранный уровень детализации зависит от ряда факторов. К ним можно отнести точность и доступность

обрабатываемых данных, точность результатов прогнозирования и др. В тех случаях, если оптимальный набор переменных определить трудно или невозможно, то прибегают к использованию различных альтернатив и осуществляют выбор одного из вариантов из многоальтернативного набора, который позволяет достичь наилучшего результата. Подробное описание технологий нейросетевого моделирования и их преимущества приведено в работах [17,20,32,35,39,61,102].

Следующий этап при разработке нейросетевой прогнозирующей модели заключается в определении следующих параметров: прогнозирование периода, горизонта прогнозирования, а также интервала прогнозирования. Прогнозируемый период является основной единицей времени, на которую осуществляется прогноз. Горизонт прогнозирования – это количество периодов в будущем, которые попадают в поле прогноза. Если необходимо осуществить прогноз на предстоящие две недели (14 дней) с предоставлением данных на каждый из дней исследуемого периода, тогда период прогнозирования равен 1 дню, а горизонт – 14 дней.

Интервал прогнозирования – это частота, с которой выполняется следующий (новый) прогноз. В большинстве случаев значение этой величины и значение периода прогнозирования совпадают. Задание горизонта и периода прогнозирования зависят от условий принятия решений в той предметной области, для которой производится решение задачи прогнозирования [85,99].

Выбор рассмотренных параметров прогнозирования является не самым сложным в процессе построения нейросетевой модели. Для того, чтобы выполняемое прогнозирование имело смысл, значение величины горизонта прогнозирования должно быть больше или равно времени, которое необходимо для реализации решения поставленной задачи, принятого на основе полученных в ходе прогноза результатов.

Исходя из этого, качество прогнозирования в наибольшей степени зависит от природы принимаемого управленческого решения.

В современной науке известно множество методов, позволяющих решать описанные задачи в условиях неопределенности. Их главный недостаток заключается в том, что они повышают вариацию ошибки прогнозирования. В свою очередь увеличение величины горизонта прогнозирования отрицательно влияет на точность прогноза. С этим можно справиться, если уменьшить время, которое необходимо для реализации полученного проектного решения, а также уменьшив ошибку прогнозирования и горизонт [85].

Точность выполняемого прогноза, которая требуется для решаемого класса задач, оказывает значимое влияние на прогнозирующую нейросетевую модель. Также большое влияние на качество (точность) прогноза оказывает обучающая выборка (ее объем, качество используемых значений и др.).

Рассмотрим пошагово основы использование аппарата нейронных сетей (нейросетевого моделирования) для решения задач прогнозирования [61,85].

На первом этапе необходимо осуществить подготовку данных для нейронной сети. Как показывает практика, предварительная обработка данных может являться одним из трудоемких элементов проведения процедуры нейросетевого анализа. При этом процесс подготовки статистических данных осуществляет пользователь.

В общем случае через вектор X обозначают некоторую совокупность $\{X_1, \dots, X_2, \dots, X_k\}$ наблюдаемых значений случайных величин, которые в рассматриваемом классе задач называются прогнозными переменными. Основная задача состоит в получении такой функции $\varphi(X)$, которую можно использовать для оценки прогнозируемой величины Y : $\varphi(X)=Y$; функцию типа $\varphi(X)$ называют предикторами величины Y по X . Разработка алгоритмов и методов получения оптимальных предикторов и составляет главную цель решения задачи прогнозирования.

Если совокупность величин $\{X_1, X_2, \dots, X_k\}$ представляет собой значения какого-либо динамического параметра, т.е. X зависит от t , то эту совокупность принято называть временным рядом, при этом каждое значение X_i соответствует значению параметра в конкретное время $t=\{t_1, t_2, \dots, t_k\}$. Тогда

рассматриваемая задача заключается в поиске значения измеряемой величины X в момент времени $t_{k+1}, t_{k+2}, t_{k+3}, \dots$, т.е. для получения прогноза необходимо определить закономерность исследуемого временного ряда.

Задачи прогнозирования можно разделить на два вида: одношаговый и многошаговый прогноз [39].

Одношаговый прогноз является краткосрочным, при этом чтобы получить прогнозируемую величину выбирают только фактические данные. Исходя из этого, одношаговое прогнозирование является более точным, но оно не позволяет решать задачи долгосрочного прогнозирования.

Многошаговым прогнозом является долгосрочный прогноз, целью которого определяется в поиске поведения исследуемых параметров для установленного промежутка времени в будущем. При этом нейросетевая модель использует полученные прогнозные значения временного ряда в качестве входных данных для выполнения дальнейшего прогноза.

На современном этапе развития науки искусственные нейронные сети представлены большим количеством разнообразных типов и конфигураций. Ознакомиться с ними можно в работах [16,17,61,93]. При этом все они имеют много общего.

Далее рассмотрим принципы построения функционирования различных видов нейронных сетей.

Каждый нейрон может выполнять простые процедуры распознавания. При этом мощность нейронных вычислений зависит от соединений нейронов в единую сеть.

Любая простая нейронная сеть состоит из совокупности нейронов, которые при объединении образуют слой. Через вершины сети слева происходит поступление и распределение входных сигналов. Так как вычислений эти элементы сети не производят, то мы не будем считать их за слой. Для того, чтобы отличать элементы нейронной сети, которые выполняют только обработку сигналов, от вычислительных нейронов, будем обозначать их на

схеме в виде кругов и квадратов (прямоугольников) соответственно. Пример нейронной сети представим на рисунке 1.4.

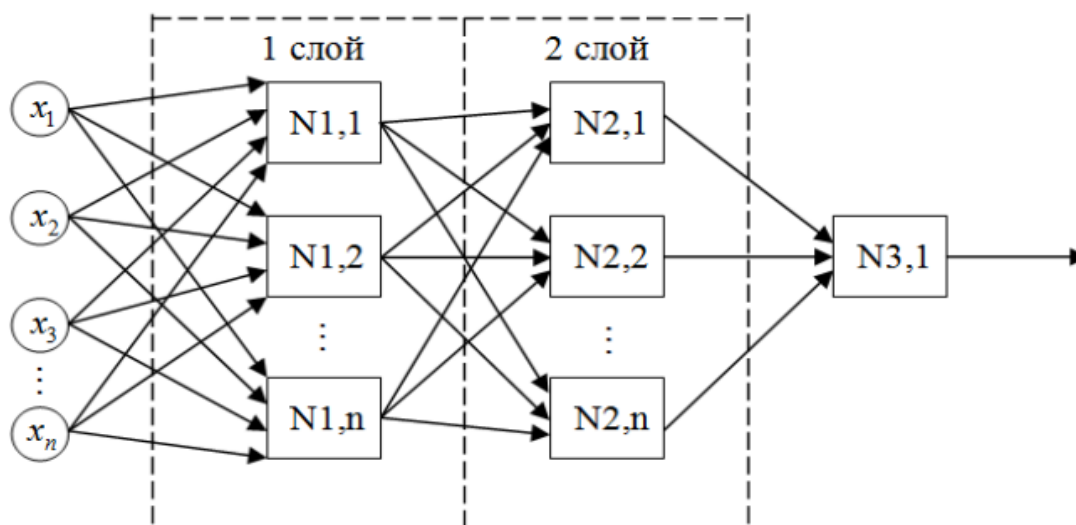


Рис. 1.4. Пример нейронной сети

Каждый элемент, входящий в множество входов нейронной сети X , соединен отдельным весом со всеми искусственными нейронами. Каждый нейрон определяет взвешенную сумму входов в построенную сеть. В биологических и искусственных нейронных сетях некоторые соединения могут отсутствовать. Также имеют место быть соединения между выходами и входами элементов в каждом слое нейронной сети [61].

Более сложные и масштабные нейронные сети в большинстве случаев обладают большими вычислительными возможностями. Несмотря на то, что созданы сети большого количества конфигураций, организация нейронных слоев в них копирует слоистые структуры различных отделов мозга.

Многослойные сети могут образовываться Объединением слоев. Выход одного из слоев может являться входом для последующего.

Использование в качестве математического аппарата многослойных нейронных сетей может позволить увеличить вычислительную мощность тогда и только тогда, если так называемая активационная функция между ее слоями будет иметь нелинейный вид. Определение значения, полученного на выходе слоя определяется при помощи умножения входного вектора на первую

весовую матрицу с дальнейшим умножением результирующего вектора на вторую весовую матрицу:

$$(X \cdot W_I) \cdot W_{II} \quad (1.1)$$

Так как умножение матриц обладает свойством ассоциативности, то (1.1) можно представить в виде:

$$X \cdot (W_I \cdot W_{II}) \quad (1.2)$$

Возможность преобразований (1.1) – (1.2) доказывает эквивалентность двухслойной линейной сети одному слою с весовой матрицей, равной произведению двух весовых матриц. Отсюда следует, что любую многослойную линейную нейронную сеть можно представить в виде эквивалентной однослойной сети. Исходя из этого, для повышения мощности использования аппарата нейронных сетей необходимо осуществить нелинейность ее активационной функции.

Рассмотренные выше примеры нейронных сетей относятся к классу сетей без обратных связей, то есть в них отсутствуют соединения, идущие от выходов некоторого слоя к входам этого же слоя или предыдущих слоев. Этот класс нейронных сетей называется сети прямого распространения. Нейронные сети более общего вида, которые имеют соединения от выходов к входам, относятся к классу сетей с обратными связями.

Следующей особенностью нейронных сетей является их способность к обучению [39]. Обучение нейронных сетей имеет сходство с процессом интеллектуального развития человека и его интеллекта. В свою очередь это свойство нейронных сетей характеризует достижение глубокого понимания этого процесса. Но возможности обучения нейронных сетей являются ограниченными. Также уже получены весомые достижения в области обучения искусственных нейронных сетей. К ним можно отнести говорящую сеть Сейновского.

Нейронная сеть подвергается процессу обучения, чтобы для определенного множества входов получать необходимое или близкое к нему множество выходов. Каждое из таких множеств (входное или выходное) рассматривается в виде вектора. Процесс обучения заключается в последовательном предъявлении входных векторов с одновременной подстройкой весов в соответствии с заданной процедурой (методикой). В процессе обучения веса нейронной сети преобразуются таким образом, что каждый входной вектор вырабатывает выходной соответственно.

Обучение нейронных сетей может осуществляться в двух режимах: с учителем и без учителя [85]. Рассмотрим отличия одного метода от другого.

Процесс обучения нейронной сети с учителем заключается в наличии для каждого входного вектора целевого вектора, который представляет собой требуемый выход. При этом пара входной-целевой векторы называется обучающей парой. Как правило, нейронная сеть при данном типе обучения имеет несколько таких пар.

Второй тип обучения нейронных сетей (без учителя) является более приближенным к естественному процессу обучения в биологической системе. Описанный в своих трудах Кохоненном и другими учеными, процесс обучения нейронных сетей без учителя не требует наличия целевого вектора, а результат на выходе не сравнивается с predetermined идеальными результатами. К обучающей выборке относятся только из входных векторов. Многие современные алгоритмы обучения нейронных сетей по принципу обучения без учителя основано на результатах работ, представленных в концепции Хэбба. Существующие алгоритмы обучения нейронных сетей представлены в работе [85].

После того, как рассмотрены все особенности построения и применения нейросетевых моделей, их виды и формы обучения, а также поэтапно представлен процесс их использования для решения прикладных задач, рассмотрим обобщенный алгоритм решения задач прогнозирования с

использованием аппарата нейросетевого моделирования. Алгоритм будет иметь следующий вид:

- 1⁰. Сбор и обработка необходимых статистических данных за выбранный временной отрезок;
- 2⁰. Формирование временного ряда с заданным интервалом;
- 3⁰. Избавление от пробелов в истории путем их заполнения;
- 4⁰. Сглаживание полученного ряда с помощью метода экспоненциальной средней;
- 5⁰. Получение ряда относительного изменения величины, для которой осуществляется прогнозирование;
- 6⁰. Формирование таблицы «окон» с глубиной погружения временных интервалов;
- 7⁰. Добавление в полученную таблицу дополнительных данных (в случае необходимости);
- 8⁰. Операция шкалирования;
- 9⁰. Определение и формирование обучающей и контрольной выборок;
- 10⁰. Подбор структуры и параметров нейронной сети в зависимости от исходных параметров решаемой задачи;
- 11⁰. Осуществление процесса обучения нейронной сети;
- 12⁰. Проверка работоспособности нейронной сети и получение результата (прогноза);
- 13⁰. Оценка полученных результатов, проверка полученной нейросетевой модели на адекватность. В случае, если погрешность ε (ошибка) принимает допустимое значение ($\varepsilon \leq \varepsilon_0$) – переход к шагу 14, в противном случае ($\varepsilon > \varepsilon_0$) возвращаемся к шагу 10 данного алгоритма.
- 14⁰. Использование полученной нейросетевой модели для решения задач прогнозирования.

1.4 Возможность использования аппарата имитационного моделирования для обработки статистических данных и проверки полученных решений на оптимальность

Для обработки статистических данных, а также проверки полученных решений на оптимальность одним из наиболее эффективных математических методов является использование аппарата имитационного моделирования.

Под имитационным моделированием принято понимать математический метод, который позволяет строить модели объекта исследования, использование которых позволяет оценить результаты влияния различных процессов на выбранный объект в условиях, наибольшим образом приближенных к реальным [54].

Аппарат имитационного моделирования относится к тем методам, которые позволяют оценить объект (систему) и ее поведение при изменении различных показателей (как внутренних, так и внешних). Применение данного метода позволяет осуществлять выбор не только различных вариантов и структур объектов исследования, но и различных форм его функционирования. Подробное описание задач, которые позволяет решать аппарат имитационного моделирования, представлено в работах [5,22,33,64,65,72,95,103].

Рассмотрим далее более подробно использование имитационного моделирования для решения рассматриваемых типов задач [34,44,54,66,69].

В частности, для решения первой задачи современные пакеты имитационного моделирования позволяют по статистическим данным (статистической выборке) определить соответствующий этой совокупности данных тип распределения. Рассмотрим решение задачи этого типа подробнее.

Поскольку одним из входных параметров имитационной модели объекта исследования является закон распределения, в соответствии с которым осуществляется процедура формирования и поступления транзактов, то для достижения эффективности решения поставленной задачи необходимо как

можно точнее задать этот параметр имитационной модели [72]. Определение типа и параметров распределения вручную является трудоемкой задачей.

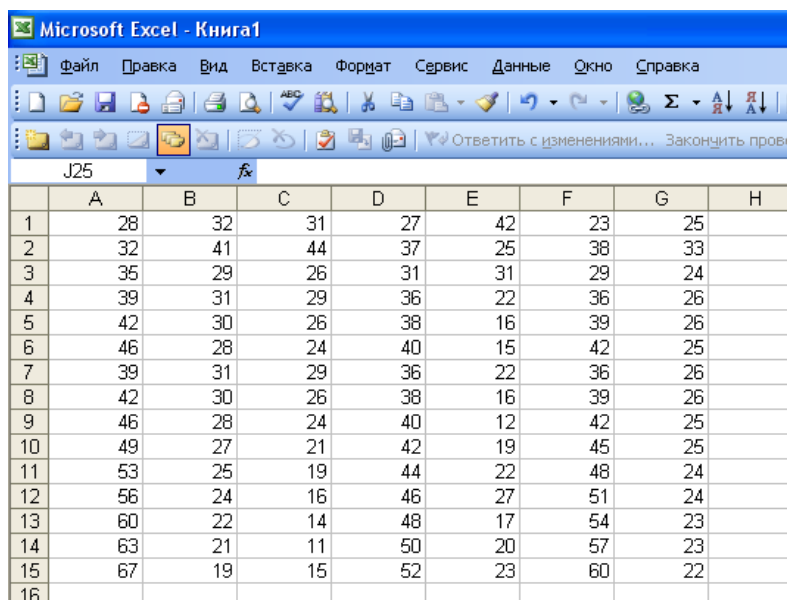
Решить эту задачу можно с помощью сбора статистических данных на исследуемом объекте рассматриваемого типа и последующей их обработки с помощью инструментария Arena Input Analyzer [95]. Рассмотрим алгоритм процесса поиска подходящего распределения с использованием заявленных методов и средств более подробно.

Для достижения поставленной цели необходимо собрать статистические данные. При этом статистическая выборка должна быть объемной для повышения точности решения.

Затем полученные статистические данные необходимо представить в виде, удобном для дальнейшей обработки. Это возможно сделать с помощью программы Microsoft Office Excel 2007. Для этого нужно создать и заполнить полученными данными таблицу по форме, представленной на рисунке 1.5.

Далее полученную таблицу статистических данных необходимо перевести в формат, соответствующий формату входных данных для дальнейшей обработки с помощью Arena Input Analyzer. Для этого сохраним созданный в Microsoft Office Excel в формате *.txt (MS-DOS). В результате получим файл, представленный на рисунке 1.6.

После того, как нужный файл был сгенерирован, запускаем Input Analyzer, создаем новый документ и загружаем в него статистические данные из подготовленного на предыдущем шаге файла. В результате чего по статистическим данным строится диаграмма, а также кривая, характеризующая наиболее подходящий под статистические данные закон распределения и его основные параметры. Результат такого подбора представим на рисунке 1.7.



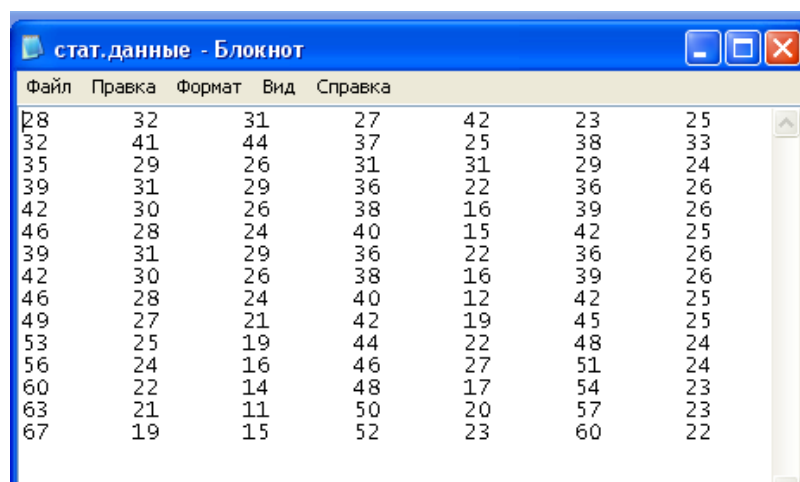
Microsoft Excel - Книга1

Файл Правка Вид Вставка Формат Сервис Данные Окно Справка

J25

	A	B	C	D	E	F	G	H
1	28	32	31	27	42	23	25	
2	32	41	44	37	25	38	33	
3	35	29	26	31	31	29	24	
4	39	31	29	36	22	36	26	
5	42	30	26	38	16	39	26	
6	46	28	24	40	15	42	25	
7	39	31	29	36	22	36	26	
8	42	30	26	38	16	39	26	
9	46	28	24	40	12	42	25	
10	49	27	21	42	19	45	25	
11	53	25	19	44	22	48	24	
12	56	24	16	46	27	51	24	
13	60	22	14	48	17	54	23	
14	63	21	11	50	20	57	23	
15	67	19	15	52	23	60	22	
16								

Рис. 1.5. Форма для создания исходного пакета для инструментария Arena Input Analyzer



стат. данные - Блокнот

Файл Правка Формат Вид Справка

28	32	31	27	42	23	25
32	41	44	37	25	38	33
35	29	26	31	31	29	24
39	31	29	36	22	36	26
42	30	26	38	16	39	26
46	28	24	40	15	42	25
39	31	29	36	22	36	26
42	30	26	38	16	39	26
46	28	24	40	12	42	25
49	27	21	42	19	45	25
53	25	19	44	22	48	24
56	24	16	46	27	51	24
60	22	14	48	17	54	23
63	21	11	50	20	57	23
67	19	15	52	23	60	22

Рис. 1.6. Файл для загрузки в инструментарий Arena Input Analyzer

Таким образом, из полученного на рисунке 1.7 решения мы можем проанализировать результаты подбора типа распределения, определить наиболее подходящий для исходных статистических данных тип и его параметры.

Для решения задач, относящихся ко второму рассматриваемому типу, применение аппарата имитационного моделирования позволяет осуществить синтез следующих компонентов объекта (системы) исследования: методов, алгоритмов функционирования, параметров, а также его структуры. С помощью анализа результатов имитационного моделирования имеется

возможность проверки структуры и параметров исследуемого объекта на оптимальность по установленным критериям эффективности [38]. Если рассматривать этот процесс по частям, то он будет складываться из следующих элементов.

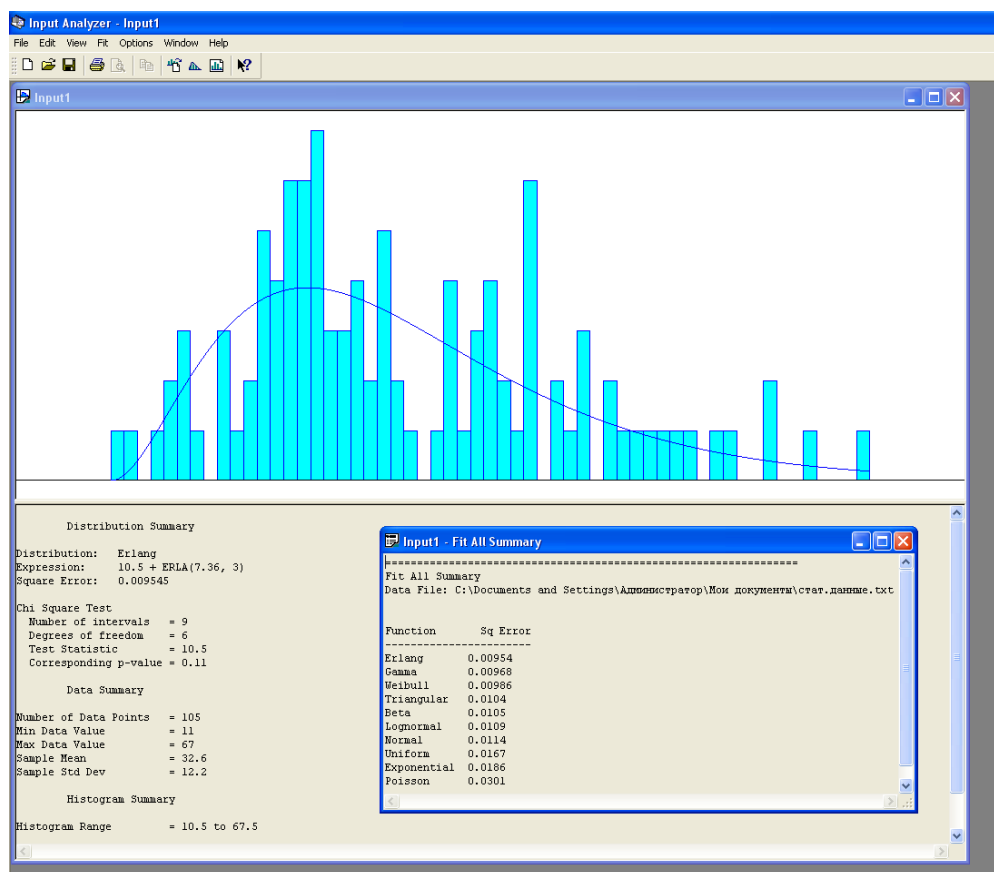


Рис. 1.7. Результат подбора параметров распределения для входных значений имитационной модели

Первым таким элементом будет являться имитационная система, которая позволяет осуществить процесс исследования имитационной модели, которая задается в виде конечной совокупности определенного количества блоков и связей между ними, обеспечивающей процесс взаимодействия во времени и пространстве. В большинстве случаев имитационная модель объекта состоит из блоков трех типов [94]:

- блоки, отвечающие за процесс функционирования объекта исследования S ;
- блоки, характеризующие ключевые параметры внешней среды ε , которые оказывают влияние на исследуемый объект;

– блоки, позволяющие организовать взаимодействие между первыми двумя типами блоков, получать и обрабатывать результаты моделирования.

Еще одной особенностью имитационной системы является тот факт, что с помощью заданного набора переменных имеется возможность управления исследуемыми объектами и процессами над ними, а именно, начальными условиями, условиями проведения имитационного эксперимента, условиями внешней среды и т.д. [52,67].

Следующим элементом является математическое обеспечение для рассматриваемой имитационной системы. Под математическим обеспечением в данном случае будем понимать совокупность законов и алгоритмических процедур [74].

Далее следует рассмотреть такой элемент, как программное обеспечение. В него включены программные средства, позволяющие осуществлять планирование и проведение вычислительного эксперимента, в основе которого лежит имитационное моделирование, а также программные средства, дающие возможность обработки и интерпретации результатов моделирования. Требования к проведению процедуры вычислительного эксперимента на основе аппарата имитационного моделирования представлены в работах [37,46,72]

Под таким элементом, как информационное обеспечение, будем рассматривать методы и средства организации базы данных, а также документы, описывающие процесс моделирования и анализа его результатов [42,48,72,90].

Для того, чтобы успешно решать поставленную задачу проверки структуры объекта на оптимальность, необходимо на основе вышеизложенных теоретических основ разработки имитационной модели построить имитационную модель объекта исследования, определить параметры генерации входных транзактов для нее с учетом всех особенностей исследуемого объекта и перед тем, как приступить к ее использованию, провести проверку ее эффективности. Рассмотрим обобщенный алгоритм разработки и

использования имитационной модели для решения управленческих задач [72]. Его можно представить в следующем виде:

1⁰. Определение и формулировка задачи моделирования. В нашем случае основной задачей организации и проведения процесса моделирования является определение степени эффективности структуры объекта исследования, полученной в ходе решения задач оптимизации, в зависимости от влияния на нее различных факторов (как внешних, так и внутренних).

2⁰. Выбор структуры объекта исследования из допустимых вариантов и определение ее основных характеристик. На данном этапе происходит описание всех элементов, входящих в структуру объекта, и их характеристик. Также определяется тип допустимых значений, которые могут принимать эти характеристики.

3⁰. Определение входных параметров. На этом шаге происходит определение входных параметров разрабатываемой имитационной модели для объекта исследования, а также законов, характеризующих некоторые из них.

4⁰. Определение выходных показателей и их анализ. Здесь происходит определение наиболее значимых выходных показателей имитационной модели, анализ которых позволит нам принять решение об эффективности предлагаемой структуры или о переходе к дальнейшему процессу ее доработки (поиска новых оптимальных решений).

5⁰. Программирование имитационной модели объекта. На данном этапе происходит выбор системы имитационного моделирования, а затем осуществляется процесс ее программирования согласно соответствующему синтаксису языка пакета, планирование процесса моделирования и проведения вычислительного эксперимента.

Значение эффективности используемой имитационной модели должно стремиться к максимуму.

Чаще всего эффективность имитационной модели оценивается одним или несколькими критериями [41]:

- достоверность и точность результатов имитационного моделирования;

- время разработки имитационной модели объекта;
- ресурсные затраты.

Исходя из этого, наилучшим показателем эффективности разработанной имитационной модели является сравнение результатов моделирования с результатами реальных исследований. В случае, если разница (погрешность) между этими показателями будет принимать значение, которое в абсолютном значении будет меньше либо равным установленному в качестве допустимого, то такая модель объекта исследования будет являться эффективной. В противном случае такая модель не позволит с необходимой точностью решать поставленную задачу проверки структуры объекта на оптимальность.

1.5 Необходимость построения интегрированной среды численной оптимизации, нейросетевого и имитационного моделирования для повышения эффективности вычислительного процесса

Эффективность любого вычислительного процесса, который организовывается для решения задач, разбиваемых на несколько подзадач, зависит от ряда параметров [5]. К таким параметрам можно отнести: организацию обмена информацией, характеристиками используемого оборудования для проведения вычислений и обработки информации, организации интерфейса, характеристиками процессора, используемых вычислительных методов и алгоритмов и др. В данном случае под интерфейсом будем понимать совокупность алгоритмов, вычислительных характеристик и конструктивных требований, которые в совокупности обеспечивают интеграцию взаимодействующих между собой элементов вычислительного процесса в единую вычислительную систему (среду) [37].

Как было сказано выше, в основе каждого вычислительного процесса лежит определенный алгоритм или несколько алгоритмов (вычислительных методов) решения поставленной задачи. Для повышения эффективности вычислительного процесса необходимо правильно подобрать или разработать

алгоритм, лежащий в его основе. Чтобы справиться с этим, рассмотрим понятие алгоритма и основные принципы их построения [44].

Алгоритм является одним из ключевых понятий при организации и проведении вычислительного процесса, который, в свою очередь, не обладает формальным определением, выраженным через терминологию более простых определений, а опирается на полученные непосредственно из опытов [69].

Под алгоритмом следует понимать любое точное пошаговое описание, которое определяет протекание вычислительного процесса. Каждый алгоритм начинается с исходных для него данных и набора действий и операций над ними, четкое выполнение которых позволяет достичь необходимого результата.

Если вычислительный алгоритм является сложным, то обычно он содержит на одном из своих шагов разветвление вычислительного процесса, которые задаются определенными условиями, повторение вычислительных процедур, для него задается определенный показатель точности вычислений и др.

Эффективность реализуемых в ходе решения задач управления вычислительных процессов оценивается с помощью критериев: среднее время ответа и средневзвешенное время обращения [97].

Величина среднего времени ответа характеризует время пребывания решаемой задачи в вычислительной системе

$$U = \frac{1}{n} \sum_{i=1}^n (t_i^3 - t_i^n), \quad (1.3)$$

где n – число подзадач, входящих в решаемую задачу; t_i^3 - время завершения решения i -ой подзадачи; t_i^n - время поступления i -ой подзадачи.

Среднее взвешенное время обращения

$$W = \frac{1}{n} \sum_{i=1}^n \frac{t_i^3 - t_i^n}{\eta_i}, \quad (1.4)$$

W - определяет зависимость величины времени ответа от времени выполнения (t обслуживания). η_i - суммарное (общее) время выполнения процесса решения

i-ой подзадачи. При этом установлена зависимость: чем меньше время выполнения, тем выше загрузка ресурсов.

Также, одним из методов, позволяющих повышать эффективность процесса управления для рассматриваемого класса задач, т.е. задач, состоящих из нескольких подзадач, при этом для решения каждой подзадачи необходимо использовать определенный вычислительный метод или алгоритм, является разработка интегрированной вычислительной среды численной оптимизации, нейросетевого и имитационного моделирования. При этом особенностью такой вычислительной среды является интеграция в единое целое каждой из рассматриваемых подзадач с соответствующими им методами, моделями и алгоритмами. Особенности построения и использования таких вычислительных сред подробно описаны в работах [21,28,37,45,48].

Исходя из анализа применяемых в вычислительной среде математических методов и моделей (в частности, оптимизационные и имитационные модели), можно сказать, что они имеют собственную внутреннюю структуру и принципы функционирования, а также в основе их работы лежат разные типы алгоритмов. Но, при этом, использование этих видов моделей в единой вычислительной среде позволяет повысить эффективность вычислительного процесса, а, следовательно, и качество, и эффективность принимаемых управленческих решений [48]. Это становится возможным за счет того, что эти модели имеют общие механизмы взаимодействия между собой и общие элементы. Подробнее такие алгоритмы описаны в работах [34,43,47,48,90].

Рассмотрим каждый из элементов разрабатываемой вычислительной среды более подробно.

На этапе сбора и обработки данных осуществляется сбор статистических данных, которые будут характеризовать необходимые для осуществления процесса управления параметры исследуемого объекта. После этого данные обрабатываются и представляются в виде, удобном для дальнейшего применения.

После того, как статистические данные готовы для дальнейшего использования, они делятся на две части: обучающая выборка и данные для проверки результата. По данным, относящимся к категории обучающей выборки, осуществляется обучение смоделированной нейронной сети.

Формализованная характеристика компонентов сложноструктурированных объектов с вариативным управлением заключается в формулировке и представлении основных характеристик сложноструктурированного объекта в виде математических закономерностей и математических моделей.

Далее происходит прогнозирование исследуемых параметров для рассматриваемого объекта. Затем, когда получены результаты прогноза, их сравнивают с результатами статистических данных за этот же период времени. На основании результатов проведенного сравнения определяется адекватность нейросетевой модели, а, следовательно, и результатов, полученных с помощью нее.

На этапе поиска оптимального решения построенная и описанная оптимизационная модель приводится к определенному виду, отвечающему параметрам и техническим характеристикам исследуемого объекта заданного типа. После чего происходит решение полученной задачи для него с использованием популяционных алгоритмов и алгоритмов многоальтернативной оптимизации с привлечением экспертной информации [48,92].

Следующим этапом является проверка эффективности полученного решения по преобразованию структуры объекта исследования при помощи имитационного моделирования. На данном этапе эффективность полученной структуры объекта исследования будет оцениваться по принципу, описанному в п. 1.4 данной работы. Оценка эффективности также осуществляется на основе привлечения экспертной информации.

В случае, если полученный вариант решения признается эффективным, происходит его сохранение как окончательного варианта исследуемого объекта,

отвечающего требованиям оптимальности поставленной задачи. В противном случае осуществляется процедура организации режима нейросетевого и имитационного моделирования и поисковой оптимизации, по результатам которой формируется множество перспективных вариантов решений. Затем осуществляется проверка каждого из вариантов полученного множества на эффективность и с помощью экспертного оценивания выбирается наиболее эффективное для заданных условий. Полученное решение будет представлять из себя решение о структурной реорганизации объекта исследования с целью повышения эффективности его функционирования.

2 РАЗРАБОТКА ОПТИМИЗАЦИОННОЙ МОДЕЛИ И ПРОЦЕДУР ЧИСЛЕННОЙ ОПТИМИЗАЦИИ СЛОЖНОСТРУКТУРИРОВАННЫХ ОБЪЕКТОВ

Основой формирования интегрированной среды численной оптимизации и моделирования является комплекс оптимизационных моделей, отражающих разнообразие переменных и особенности сложноструктурированных объектов, которые охарактеризованы в п.1.1. Именно оптимизационные модели определяют объединение в единую математическую конструкцию формализованного описания и экстремальных, и граничных требований к вариациям структуры объектов. Учет всех перечисленных требований приводит к формированию инвариантной модели, модификации которой дает возможность учесть частные ситуации и построить схему выбора определенного компонента комплекса в зависимости от сочетания конкретных свойств исследуемого объекта.

Поскольку ряд моделей, входящих в комплекс близок по своей структуре к задачам многоальтернативной оптимизации, целесообразнее рассматривать схемы преобразования исходных задач к каноническим, для которых возможно комплексирование ряда алгоритмических модулей поиска оптимального решения.

Для однокритериальной задачи на множестве дискретных переменных возникает возможность исследования объекта как в статическом, так и в динамическом режимах. Этот класс задач позволяет использовать при их решении более простые численные методы, чем алгоритмы многоальтернативной оптимизации. К таким численным методам относятся популяционные алгоритмы, и среди них приемлемым является алгоритм, основанный на поведении обезьян, так как позволяет организовать движение агентов, входящих в популяцию, аналогично алгоритму многоальтернативной оптимизации в рандомизированной среде, что создает предпосылки общего

подхода к численной оптимизации по всему комплексу оптимизационных моделей сложноструктурированных объектов.

2.1 Разработка и описание особенностей оптимизационной модели сложноструктурированных объектов

Исходя из свойств 1 и 4 (см. п.1.1), характеризующих особенности сложноструктурированных объектов, задачу выбора оптимального решения предлагается в общем случае представить следующей инвариантной моделью (μ_1) [81]:

$$f_{ij}(x_1, x_2, x_3) \rightarrow \max_{x_1, x_2, x_3}, i \in \overline{1, I}, j \in \overline{1, 2}$$

$$\begin{cases} \beta_i \cdot x_i \leq b_i, i \in \overline{1, I}, \\ x_{1_j} = \begin{cases} 1, & j = \overline{1, J}, \\ 0, & \end{cases} \\ x_{2_l} = (x_{2_{l1}}, \dots, x_{2_{lm}}), l = \overline{1, L}, x_{2_{lm}} \geq 0, m = \overline{1, M}, \\ x_{3_n}^{\min} \leq x_{3_n} \leq x_{3_n}^{\max}, n = \overline{1, N}, x_{3_n}^{\min} \geq 0, n = \overline{1, N}. \end{cases} \quad (2.1)$$

Модель (μ_1) имеет целый ряд модификаций в частные модели $\overline{\mu_2}, \overline{\mu_5}$:

- однокритериальная оптимизационная модель (μ_2):

$$\psi_{l2}(x_1, x_2, x_3) \rightarrow \max_{x_1, x_2, x_3}$$

$$\begin{cases} f_{l2}(x_1, x_2, x_3) \leq b_{i2}, i_2 = \overline{1, I_2}, \\ x_{1_j} = \begin{cases} 1, & j = \overline{1, J}, \\ 0, & \end{cases} \\ x_{2_l} = (x_{2_{l1}}, \dots, x_{2_{lm}}), l = \overline{1, L}, x_{2_{lm}} \geq 0, m = \overline{1, M}, \\ x_{3_n}^{\min} \leq x_{3_n} \leq x_{3_n}^{\max}, n = \overline{1, N}, x_{3_n}^{\min} \geq 0, n = \overline{1, N} \end{cases}. \quad (2.2)$$

- многокритериальная без ограничений (μ_3):

$$f_{ij}(x_1, x_2, x_3) \rightarrow \max_{x_1, x_2, x_3}, i \in \overline{1, I}, j \in \overline{1, 2}$$

$$\begin{cases} x_{1_j} = \begin{cases} 1, & j = \overline{1, J}, \\ 0, & \end{cases} \\ x_{2_l} = (x_{2_{l1}}, \dots, x_{2_{lm}}), l = \overline{1, L}, x_{2_{lm}} \geq 0, m = \overline{1, M}, \\ x_{3_n}^{\min} \leq x_{3_n} \leq x_{3_n}^{\max}, n = \overline{1, N}, x_{3_n}^{\min} \geq 0, n = \overline{1, N}. \end{cases} \quad (2.3)$$

- включающая альтернативные и дискретные переменные (μ_4):

$$\begin{aligned}
 f_{ij}(x_1, x_2) &\rightarrow \max_{x_1, x_2}, \quad i_1 = \overline{1, I_1} \\
 f_{i_2}(x_1, x_2) &\leq b_{i_2}, \quad i_2 = \overline{1, I_2} \\
 x_{1j} &= \begin{cases} 1, & j = \overline{1, J} \\ 0, & \end{cases} \\
 x_{2l} &= (x_{2l_1}, \dots, x_{2l_m}), \quad l = \overline{1, L}, x_{2lm} \geq 0, m = \overline{1, M}
 \end{aligned} \tag{2.4}$$

- однокритериальная без ограничений, включающая только дискретные переменные (μ_5).

$$\begin{cases}
 f(x_k, x_l, \dots, x_m) \rightarrow \max_{x_k, x_l, \dots, x_m}, \\
 x_k = \{x_{k_1}, x_{k_2}, \dots, x_{k_K}\}, x_k \in K', \\
 x_l = \{x_{l_1}, x_{l_2}, \dots, x_{l_L}\}, x_l \in L', \\
 \dots \\
 x_m = \{x_{m_1}, x_{m_2}, \dots, x_{m_M}\}, x_m \in M', \\
 x_k > 0, x_l > 0, \dots, x_m > 0.
 \end{cases} \tag{2.5}$$

где K', L', \dots, M' - множества допустимых вариантов параметров исследуемого объекта (множество вариантов решений).

Из (2.5) имеем, что целевая функция, выраженная в виде $f(x_k, x_l, \dots, x_m)$, будет отображать зависимость критерия оптимальности от основных параметров объекта.

Ограничения на переменные в построенной оптимизационной модели будут задаваться в дискретном виде, при этом их значения будут выбираться из конечных множеств K', L', \dots, M' соответственно и могут принимать значения строго больше 0.

Решение поставленной задачи будем искать в виде вектора X (2.6):

$$X = \begin{Bmatrix} x_k^{n1} \\ x_l^{n2} \\ \dots \\ x_m^{nN} \end{Bmatrix} \tag{2.6}$$

В дальнейшем основные исследования будем проводить с использованием наиболее распространенной при решении прикладных задач практической направленности модели (μ_5).

Предлагается схема выбора структуры оптимизации модели, исходя из конкретных особенностей реализации свойств 1 и 4 сложноструктурированного объекта (рис. 2.1).

Наиболее приемлемыми для решения задач $\overline{\mu_1, \mu_5}$ являются методы многоальтернативной оптимизации [49] и вариационного моделирования [37].

Путем комплексирования отдельных модулей указанных методов предлагается обобщенный алгоритм, структурная схема которого приведена на рис. 2.2.

Для его использования требуется предварительные преобразования исходных задач $\overline{\mu_1, \mu_5}$ и входящих в них переменных.

1. Переход от моделей $\overline{\mu_1, \mu_4}$ к модели μ_2 путем введения следующей эквивалентной аддитивной функции [8]:

$$\psi(x_1, x_2, x_3) = \sum_{i_j=1}^{I_1} \alpha_{i_j} \psi_{i_j}(x_1, x_2, x_3),$$

где α_{i_1} - весовые коэффициенты,

$$0 \leq \alpha_{i_1} \leq 1, \quad i_1 = \overline{1, I_1}, \quad \sum_{i_1=1}^{I_1} \alpha_{i_1} = 1.$$

2. Переход от моделей μ_1, μ_2, μ_4 к модели μ_3 с одним критерием $\psi(x_1, x_2, x_3)$ путем формирования следующей функции [30]:

$$F(x_1, x_2, x_3, y) = \psi(x_1, x_2, x_3) + \sum_{i_2=1}^{I_2} y_{i_2} (b_{i_2} - f_{i_2}(x_i)),$$

где y_{i_2} - коэффициенты эквивалентной функции, $y_{i_2} \geq 0$.

3. Переход от дискретных переменных x_i^2 к альтернативным переменным одним из следующих способов:

а) Заменяется набором альтернативных переменных, соответствующих номерам l_m координат вектора x_i^2 :

$$x_{lm}^2 = \begin{cases} 1, & \text{если при поиске используется значение } x_l^2 = x_{lm}^2, l = \overline{1, L}, m = \overline{1, M} \\ 0, & \text{в противном случае} \end{cases}$$

б) в случае целочисленных значений координат x_{lm}^2 они приравняются в двоичном исчислении

$$x_{lm}^2 = 1 + 2z_{1l} + 4z_{2l} + \dots,$$

где $z_{1l}, z_{2l}, \dots = \begin{cases} 1 \\ 0 \end{cases}$.

Индексы $1, 2, \dots$ характеризуют разряды двоичного числа. Учитывая, что при выборе определенного разряда появляются числа $x_{lm}^2 > x_{lm}$, их необходимо в поисковой процедуре исключать алгоритмически [9, 60, 86].

4. Переход от моделей $\overline{\mu_1, \mu_5}$ к эквивалентным с использованием механизмов рандомизации и сглаживания. Рандомизация множества переменных – переход в вычислительной среде к псевдослучайным реализациям их значений

$$\tilde{x}_i^1, j = \overline{1, J}, \quad \tilde{x}_i^2, j = \overline{1, L}, \quad \tilde{x}_n^3, j = \overline{1, N}.$$

Поскольку рандомизация приводит к случайным реализациям функций $\tilde{f}_{i1}, \tilde{f}_{i2}$, то для перехода к эквивалентам задаче оптимизации используют механизм сглаживания, то есть перехода к математическим ожиданиям на множествах случайных реализаций $\hat{f}_{i1}, \hat{f}_{i2}$.

Процесс решения поставленной задачи классическими оптимизационными методами является трудоемким, а в некоторых случаях и невозможным. Это обусловлено рядом факторов: нелинейность, многоэкстремальность, высокая вычислительная сложность оптимизируемой функции, большая размерность области поиска решений и др.

Задача (2.5) – (2.6) является оптимизационной моделью сложноструктурированных объектов в статическом режиме. Помимо статического режима решения поставленной задачи, необходимо осуществлять ее решение возможно и в динамическом режиме. Рассмотрим далее

отличительные особенности формализации этой задачи для динамического режима поиска.

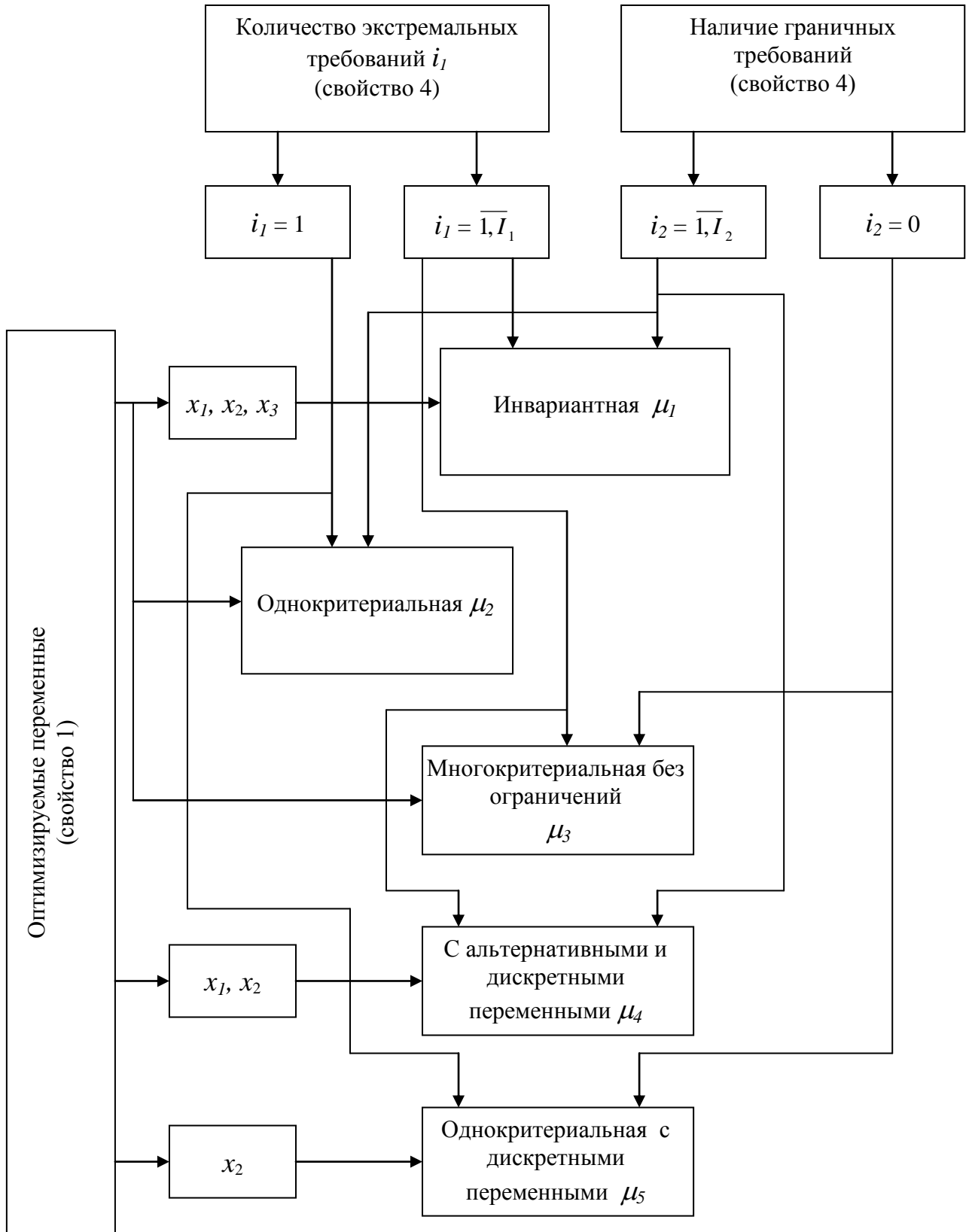


Рис. 2.1. Схема формирования комплекса оптимизационных моделей в зависимости от свойств сложноструктурированного объекта

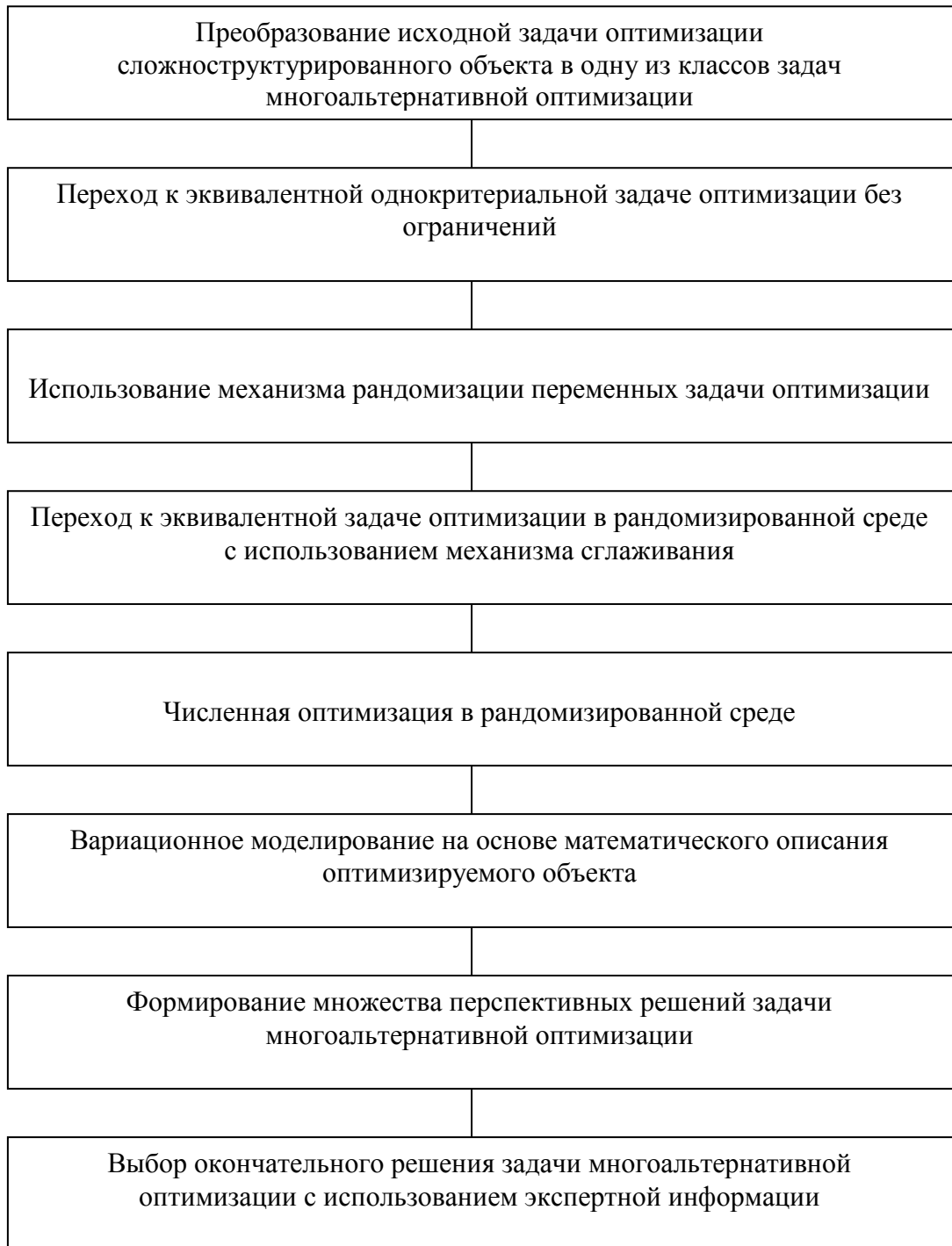


Рис. 2.2. Структурная схема обобщенного алгоритма комплексирования метода многоальтернативной оптимизации и вариационного моделирования при выборе решения для сложноструктурированных объектов

Опишем ряд изменений, которые нам необходимо внести в нашу модель (2.5) и форму поиска оптимального решения (2.6) для возможности осуществлять решение в динамическом режиме. Очевидно, что раз мы говорим о динамике изменений, то в нашей модели должен появиться параметр, который будет характеризовать время.

Параметры динамической постановки задачи, в отличие от (2.5), мы будем разделять на две группы: постоянные и варьируемые во времени. К постоянным параметрам мы отнесем те, которые не подлежат оперативному управлению и изменению их характеристик. К варьируемым отнесем ряд параметров, которые имеют свойство осуществлять мгновенное параметрическое управление их основными характеристиками.

Особенность решения рассматриваемой задачи в динамическом режиме будет заключаться в разбиении ее на две подзадачи.

Первая подзадача – это задача определения оптимальной структуры исследуемого объекта в статическом режиме.

Вторая подзадача – это задача поиска оптимальной структуры объекта в динамическом режиме. Основными отличиями от задачи, представленной в виде (2.5) – (2.6) являются начальные условия, а также вид и количество ограничений.

Рассмотрим эти изменения более подробно.

Будем считать, что решение поставленной задачи (подзадачи 2) будет осуществляться на временном отрезке $\tau \in [\tau_0; \tau_g]$.

Тогда в нашей модели появятся новые характеристики:

- $x_k(\tau), x_l(\tau), \dots, x_v(\tau)$ – вектор состояния основных характеристик варьируемого типа исследуемого объекта в зависимости от момента времени $\tau \in [\tau_0; \tau_g]$;

- τ - время; для упрощения поиска решения будем полагать, что моменты времени в рамках заданного отрезка принимают дискретные значения, т.е. $\tau = \{\tau^1, \tau^2, \dots, \tau^U\}$.

Заметим, что параметры модели исследуемого объекта, относящиеся к категории постоянных, а именно $x_{v+1}, x_{v+2}, \dots, x_m$, будут входить в нее как константы, значение которых будет получено в результате решения подзадачи 1.

Сама модель (2.5) после модификации примет вид (2.7).

Решение задачи будет заключаться в поиске такого вектора (2.8), который в каждый момент времени $\tau_u = \{\tau^1, \tau^2, \dots, \tau^U\}$ приносит функции $f(x_k(\tau), x_l(\tau), \dots, x_v(\tau), x_{v+1}, x_{v+2}, \dots, x_m, \tau)$ оптимальное значение.

$$\begin{cases}
 f(x_k(\tau), \dots, x_v(\tau), x_{v+1}, x_{v+2}, \dots, x_m, \tau) \rightarrow \max, \\
 x_k(\tau) = \{x_k^1(\tau), x_k^2(\tau), \dots, x_k^K(\tau)\}, x_k(\tau) \in K', \\
 x_l(\tau) = \{x_l^1(\tau), x_l^2(\tau), \dots, x_l^L(\tau)\}, x_l(\tau) \in L', \\
 \dots \\
 x_v(\tau) = \{x_v^1(\tau), x_v^2(\tau), \dots, x_v^V(\tau)\}, x_v(\tau) \in V', \\
 x_{v+1} = x_{v+1}^{opt}, x_{v+2} = x_{v+2}^{opt}, \dots, x_m = x_m^{opt}; \\
 x_k(\tau) > 0, x_l(\tau) > 0, \dots, x_m(\tau) > 0.
 \end{cases} \quad (2.7)$$

$$X^{opt}(\tau_u) = \begin{pmatrix} x_k^{n1}(\tau) \\ x_l^{n2}(\tau) \\ \dots \\ x_v^{mN}(\tau) \\ x_{v+1} \\ x_{v+2} \\ \dots \\ x_m \end{pmatrix} \quad (2.8)$$

После того, как оптимизационная модель сложноструктурированного объекта разработана для обоих (статического и динамического) режимов решения поставленной задачи, степень критичности ограничений полученной модели на результат решения задачи определена, критерии расположены по степени важности, можно переходить к выбору алгоритмов ее решения, а также описанию процедуры численной оптимизации для разрабатываемой вычислительной среды.

2.2 Формирование процедур численной оптимизации в рандомизированной вычислительной среде на основе популяционных алгоритмов

Процедуры численной оптимизации для разрабатываемой вычислительной среды будут основываться на использовании популяционных алгоритмов оптимизации [30]. Выбор в пользу этого математического аппарата был сделан из-за их гибкости и возможности быть использованными в задачах, для которых практически невозможно составить аналитические зависимости, имеющих высокую трудоемкость необходимых вычислений и т.д.

Основным достоинством популяционных методов численной оптимизации является использование в них системы, состоящей из так называемых агентов. Под агентом понимается некая точка с определенными начальными координатами, а сам процесс поиска оптимального решения поставленной задачи заключается в перемещении выбранного агента в допустимой области по заданным правилам с установленной величиной шага перемещения. Исходя из этого, наиболее важными преимуществами популяционных методов являются гибкость к условию задачи и возможность за сравнительно короткие промежутки времени получать решения задачи, близкие к оптимальным.

Рассмотрим методику популяционного алгоритма в обобщенном виде:

1⁰. Инициализация популяции (в заданной области поиска решений создаем произвольным образом несколько начальных приближений к искомому решению поставленной задачи). Этот этап можно назвать как создание популяции агентов.

2⁰. Миграция агентов популяции (с помощью определенного набора миграционных операторов распределяем агентов популяции так, чтобы в конечном счете подобраться к искомому оптимальному значению заданной функции).

3⁰. Завершающий этап (Проверка условий окончания итерационных процедур алгоритма. Если они достигнуты – завершаем процедуру алгоритма и лучшее положение агентов считаем в качестве приближенного решения поставленной задачи. В противном случае возвращаемся к шагу 2).

Для дальнейшей работы с популяционными алгоритмами введем следующие обозначения:

$f(X)$ – Заданная целевая функция (ц.ф.);

$f(X^{opt})$ – значение ц.ф., соответствующее оптимальному значению;

S_0 – начальная популяция;

S – популяция, имеющаяся на данный момент;

S' – дальнейшие популяции;

s^{best} – лучший агент популяции;

p^{max} – предельное значение количества итерационных процедур (популяций);

p – номер текущего поколения (итерации);

X_j – текущий вектор параметров оптимизационной модели;

X_{j+1} – вектор параметров оптимизационной модели на следующем шаге итерационного процесса;

X^{opt} – лучшее положение агента за весь период итерационного процесса;

$\lambda(X)$ – функция приспособленности агента (фитнесс-функция);

λ^{best} – оптимальное значение фитнес-функции.

Считаем, что целевая функция поставленной оптимизационной задачи достигает максимального (минимального) значения, при этом соответствующая ей «фитнесс-функция» также должна достигать максимального (минимального) значения.

Для процесса формирования начальной популяции в большинстве случаев распределяют по случайному принципу. При этом распределение должно быть равномерным по всей области поиска.

Условием окончания поиска в популяционных алгоритмах является достижение предельного значения (верхней границы) количества поколений

заданной популяции или на протяжении нескольких итерационных шагов (их количество оговаривается в начальных условиях работы алгоритма) значение целевой функции остается неизменным.

В современных литературных источниках [7,18,30,31,61] описаны несколько десятков видов популяционных алгоритмов. К ним относятся алгоритмы пчелиного роя, косяка рыб, муравьиной колонии и многие другие.

Проведя сравнительный анализ популяционных алгоритмов оптимизации, для решения задачи оптимизации сложноструктурированных объектов нами был выбран алгоритм, основанный на поведении обезьян. Выбранный алгоритм относится к числу эволюционных и хорошо адаптирован для решения оптимизационных задач, модели которых включают в себя нелинейность, недифференцируемость, высокую размерность пространства поиска с большой скоростью схождения. Ещё одним преимуществом алгоритма обезьяньего поиска является то, что данный алгоритм регулируется малым числом параметров.

Рассмотрим в общем виде основные особенности данного алгоритма более подробно.

Данный алгоритм основан на поведении обезьян в процессе их миграции по горам с целью поиска пищи. Полагается, что обезьяны считают «чем выше гора, тем больше пищи на ее вершине».

Из начального положения каждая обезьяна, входящая в сформированную популяцию, перемещается по склону горы, пока не достигнет горной вершины. После серии таких подъемов обезьяна совершает перемещение на соседние горы, после чего процедура повторяется (локальный способ).

В случае, если на локальном этапе все окрестности начального положения обезьяны исследованы, но поставленная цель при этом не достигнута, происходит переход к глобальному способу.

Глобальный способ заключается в более длинном перемещении обезьяны в новую область исследования.

Описанная процедура будет иметь количество итераций, которое задает пользователь перед началом работы алгоритма.

Оптимальным решением задачи при использовании описанного алгоритма будет являться самая высокая из просмотренных вершин, исследованных выбранной популяцией.

Пусть имеется популяция S , состоящая из i агентов. Тогда локальный способ будет иметь следующую формулировку:

1. Исходя из начального (текущего) положения агента X_i определяем его возможное новое положение X'_i в виде (2.9):

$$x'_{i,j} = W_1((x_i - b); (x_i + b)), i \in [1 : |S|], j \in [1 : |X|], \quad (2.9)$$

где b – максимально возможное значение длины прыжка агента в заданном направлении, $b > 0$.

2. Если точка X'_i является допустимой, при этом $\mu(X'_i) = \mu(X_i)$, тогда присваиваем X_i значение X'_i . При этом для агента i заданной популяции S завершаем процедуру локальных прыжков (локальный способ). В противном случае возвращаемся к 1 шагу локального способа заданное кол-во раз.

Далее рассмотрим глобальный способ (глобальные прыжки):

1. По равномерному закону распределения выбираем случайное вещественное число τ из интервала $[\tau_{\min}; \tau_{\max}]$. Это значение будет характеризовать значение величины глобального прыжка (2.10):

$$\tau = W(\tau_{\min}; \tau_{\max}). \quad (2.10)$$

Значения τ_{\min} и τ_{\max} могут иметь разные знаки, поэтому значение τ может быть как отрицательным, так и положительным.

2. На следующем этапе глобального способа осуществляем поиск возможного нового положения X'_i для агента s_i в виде (2.11):

$$x'_{i,j} = x_{i,j} + \tau(x_j^c - x_{i,j}), i \in [1 : |S|], j \in [1 : |X|], \quad (2.11)$$

где x_j^c – текущее положение центра тяжести агентов заданной популяции по j -ому координатному направлению (2.12), при этом:

$$x_{i,j}^c = \frac{1}{|S|} \cdot \sum_{i=1}^{|S|} x_{i,j}. \quad (2.12)$$

3. В случае, если положение X'_i является допустимым по заданным условиям, то присваиваем X_i значение X'_i . При этом для агента i заданной популяции S завершаем процедуру глобальных прыжков. В противном случае возвращаемся к 1 шагу глобального способа.

В связи с тем, что с помощью рассмотренного оптимальное решение может быть получено на промежуточном итерационном шаге, то необходимо сохранять найденные значения на всех итерационных шагах.

2.3 Ориентация популяционных алгоритмов на оптимизацию объектов исследуемого типа

На основе анализа теоретических сведений о популяционных алгоритмах оптимизации, описанных в пункте 2.2 данной работы, а также особенностей оптимизационных моделей (2.5) - (2.6), (2.7) - (2.8) для оптимизации сложноструктурированных объектов в статическом и динамическом режимах соответственно, произведем разработку популяционного алгоритма для решения поставленной задачи в статическом режиме, а затем его адаптацию под условия динамической задачи. Основываться разрабатываемые алгоритмы будут на принципах алгоритма обезьяньего поиска.

Основными особенностями использования алгоритма для решения задачи (2.5) – (2.6) будет являться наличие и исследование возможных значений переменных, входящих в вектор искомого решения X , представленного в виде (2.6).

Также для каждого из параметров (2.6) задано конкретное значение шага при «локальных прыжках» b_m .

Сам алгоритм в модифицированном виде под исследуемый класс объектов представим в следующем виде:

1⁰. Задаем m начальных популяций с соответствующими значениями $b = \{b_1, b_2, \dots, b_m\}$. При этом значение шагов рассчитывается с учетом возможных решений.

2⁰. В соответствии с установленными значениями шагов и начальными популяциями совершаем локальные прыжки. Для каждого i -го шага вычисляем значение фитнес-функции.

3⁰. Проводим сравнение значений фитнес-функций для i -го и $i-1$ итерационных шагов. После вычислений проводим проверку условий: $\lambda_i > \lambda_{i-1}$.

4⁰. В случае выполнения условия полученное на i -ом шаге решение запоминаем как оптимальное и продолжаем итерационную процедуру, переместившись в точку на величину шага b .

5⁰. После того, как все возможные точки пройдены, проводим сравнение результатов, полученных на каждом из локальных шагов, и выбираем из них наилучшее.

Далее проведем адаптацию разработанного алгоритма под динамические условия задачи. Главными его отличительными чертами будут являться наличие и исследование возможных значений переменных, входящих в вектор искомого решения $X^{opt}(\tau)$, представленного в виде (2.8).

Также для каждого из параметров (2.8) задано конкретное значение шага при «локальных прыжках» b_m .

Адаптированный вид алгоритма под динамический характер решаемой задачи можно представить в следующем виде:

1⁰. Задаем m начальных популяций с соответствующими значениями $b = \{b_1, b_2, \dots, b_m\}$. При этом значение шагов рассчитывается с учетом возможных решений.

2⁰. В соответствии с установленными значениями шагов и начальными популяциями совершаем локальные прыжки. Для каждого i -го шага вычисляем значение фитнес-функции.

3⁰. Проводим сравнение значений фитнес-функций для i -го и $i-1$ итерационных шагов. После вычислений проводим проверку условий: $\lambda_i > \lambda_{i-1}$.

4⁰. В случае выполнения условия полученное на *i*-ом шаге решение запоминаем как оптимальное и продолжаем итерационную процедуру, переместившись в точку на величину шага *b*.

5⁰. После того, как все возможные точки пройдены, проводим сравнение результатов, полученных на каждом из локальных шагов и выбираем из них наилучшее.

6⁰. По полученному решению определяем соответствующий ему вектор параметров (2.6) исследуемой модели (2.5).

7⁰. Осуществляем разбиение параметров (2.6) на категории варьируемых и статических. После чего формируем оптимизационную модель для решения подзадачи 2 (динамического типа) в виде (2.7) – (2.8).

8⁰. Задаем временной отрезок $[\tau_0; \tau_g]$, а затем представляем его в дискретной форме $\{\tau^J, \tau^2, \dots, \tau^U\}$.

9⁰. В соответствии с установленными значениями шагов и начальными популяциями совершаем локальные прыжки. Для каждого *i*-го шага в момент времени τ_u вычисляем значение фитнес-функции.

10⁰. Проводим сравнение значений фитнес-функции для *i*-го и *i-1* итерационных шагов. После вычислений проводим проверку условий: $\lambda_i(\tau_u) > \lambda_{i-1}(\tau_u)$.

11⁰. Если заданное условие выполнено, то полученное на *i*-ом шаге решение запоминаем как оптимальное для момента времени τ_u и продолжаем итерационную процедуру, переместившись в точку на величину заданного шага *b*.

12⁰. Как только все возможные точки, допустимые для момента времени τ_u пройдены, проводим сравнение результатов, полученных на каждом из локальных шагов, и выбираем из них оптимальное для момента времени τ_u .

13⁰. Изменяем значение временного параметра (от значения τ_u к значению τ_{u+1}) и переходим к шагу 3 данного алгоритма.

Структурную схему численной процедуры для статического и динамического режимов представим на рис. 2.3 и 2.4 соответственно [81].

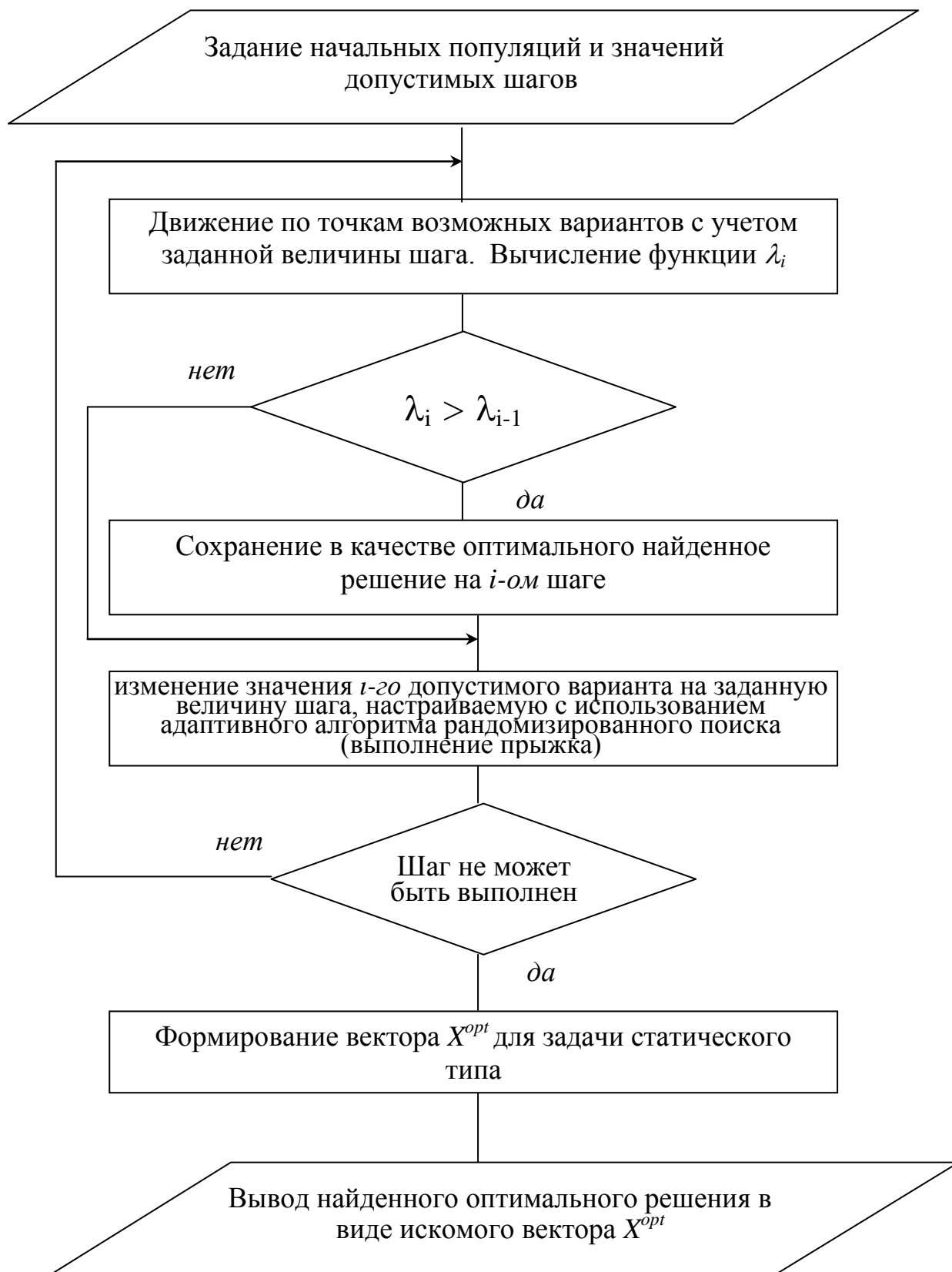


Рис. 2.3. Блок-схема численной процедуры для задачи статического типа на основе адаптивного алгоритма поведения обезьян.

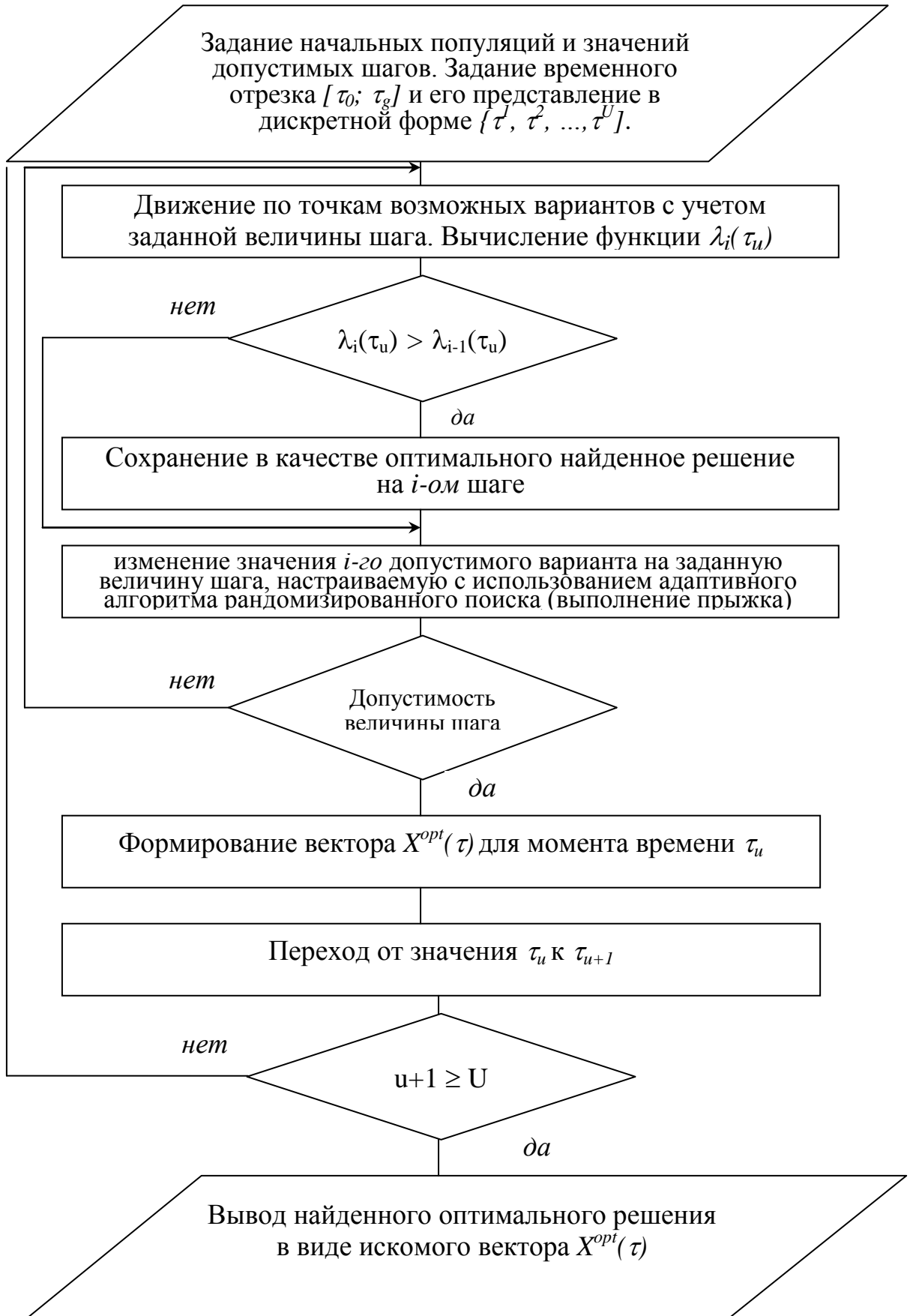


Рис. 2.4. Блок-схема численной процедуры для задачи динамического типа на основе адаптивного алгоритма поведения обезьян.

Выводы второй главы:

1. Необходимость учета разнообразия реализаций свойств сложноструктурированного объекта приводит к формированию комплекса оптимизационных моделей, среди которых инвариантная обобщает выполнение свойств в полном объеме, а частные представляют ее модификациями.

2. С целью построения численных процедур решения оптимизационных задач целесообразным является предварительный переход к эквивалентным задачам за счет построения аддитивных функций специального вида, перехода от дискретных к альтернативным переменным, использование механизмов рандомизации и сглаживания.

3. Наиболее приемлемый метод численной оптимизации на базе эквивалентных задач основан на сочетании алгоритмов многоальтернативной оптимизации и популяционных алгоритмов.

4. Для эффективного решения оптимизационных задач в случае учета статического и динамического режимов функционирования сложноструктурированного объекта целесообразным является модификация алгоритма обезьяньего поиска.

3 ИНТЕГРАЦИЯ ЧИСЛЕННЫХ ПРОЦЕДУР ОПТИМИЗАЦИИ, НЕЙРОСЕТЕВОГО И ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ В ЕДИНУЮ ВЫЧИСЛИТЕЛЬНУЮ СРЕДУ

Построение комплекса оптимизационных моделей сложноструктурированных объектов и обоснование адекватных им численных методов моделирования, основанных на стохастических и популяционных алгоритмах, позволяет перейти к созданию интегрированной вычислительной среды. Для того, чтобы объединить в общий цикл поисковые процедуры с использованием имитационного моделирования и процедуры активного обучения в процессе поиска с последующей верификации нейросетевой модели, недостаточно основываться только на механизмах рандомизации и сглаживания, использованных в п. 3.1 для перехода от канонических оптимизационных моделей к вероятностным эквивалентам. Требуется дополнение и расширение указанных механизмов.

Во-первых, включение в единый процесс вычислений имитационного моделирования с параллельным формированием нейросетевой модели приводит к необходимости обоснования механизма комплексного моделирования моделирования.

Во-вторых, перенесение оптимизационного поиска из множества исходных переменных сложноструктурированных объектов в множество рандомизированных аналогов диктует необходимость построения трансформационных процедур, позволяющих на каждом шаге настраивать вероятностные характеристики в зависимости от прогностических оценок направления движения к зоне экстремума.

И, наконец, многопроцедурность интегрированной вычислительной среды требует синхронизации основных режимов ее функционирования в соответствие с определенными формализованными правилами.

3.1 Структура интегрированной вычислительной среды численной оптимизации, нейросетевого и имитационного моделирования

Для разработки структуры интегрированной вычислительной среды сформулируем ряд базовых позиций, определяющих реализацию единого цикла преобразований исходной информации в управленческие решения.

1. Погружение в рандомизированную среду.

В качестве основы численных процедур оптимизации, нейросетевого и имитационного моделирования предлагается использовать их погружение в рандомизированную среду. Такая возможность определяется следующими предпосылками:

- использование схемы рандомизированного поиска;
- введение рандомизации для формирования популяций агентов алгоритма, основанного на поведении обезьян;
- численным определением показателей y^0 во всех поисковых процедурах при заданных значениях вектора оптимизируемых переменных x^0 осуществляется на основе вычислительной среды имитационной модели исследуемого класса объектов

$$y^0 = f_{им}(x^0),$$

основу которой составляет преобразование случайных потоков однородных событий.

2. Трансформационные процедуры.

Необходимость перехода к очередному этапу численной оптимизации путем перехода в новую точку поиска требует инициализации $(k+1)$ - й итерации в рандомизированной среде. С этой целью осуществляется трансформация параметров генераторов псевдослучайных чисел на основе прогнозных оценок, вычисленных специальным образом:

$$x^{k+1} = x^k + \pi(f_{им(нм)}(\tilde{x}^k)), \quad (3.1)$$

где $f_{им(нм)}$ – обозначение возможности вычисления показателя объекта в режиме моделирования;

$\pi(\cdot)$ – прогнозная оценка движения поиска в зону экстремума.

Преобразование (3.1) будем называть трансформационной процедурой.

3. Комплексирование имитационного и нейросетевого моделирования.

В работе [51] введен классический принцип дуального управления, обосновывающий возможность изучения объекта в процессе управления им, тем самым, обеспечивая улучшение управления, стремясь к оптимальному. В нашем случае в качестве способа изучения объекта, представляемого имитационной моделью, рассмотрим нейросетевое моделирование, а в качестве способа управления – численные процедуры оптимизации. Обоснованность комплексного подхода в рандомизированной среде определяются исследованиями [25], которые в теории дуального управления с активным накоплением информации привели к получению оптимальной рандомизированной стратегии управления.

Поскольку численный процесс оптимизационного поиска позволяет фиксировать за k_0 итераций k_0 выборочных значений вектора переменных сложноструктурированного объекта x^k и вектора показателей $y^k = f_{\text{им}}(x^k)$, то параллельно с имитационным моделированием появляется возможность обучения и верификации более простой по реализации нейросетевой модели. При этом для настройки вектора параметров нейросетевой модели в ходе ее обучения предлагается с целью включения этой процедуры в единую вычислительную среду провести их рандомизацию на основе нормального закона распределения.

Процесс формирования параллельной модели за счет рандомизированной настройки ее параметров в цикле численной оптимизации будем называть дуальным моделированием. Дополнительно предлагается после k_0 итераций за k_g итераций осуществлять верификацию нейросетевой модели в рандомизированной среде по статистическому критерию Фишера [39].

Использование комплексного подхода в процессе моделирования позволяет на следующих после $k_0 + k_b$ итераций. $k > k_0 + k_g$ вычислять значения

показателей с применением обученной и верифицированной нейросетевой модели:

$$y^k = f_{\text{нм}}(x^k), \quad k > k_0 + k_в.$$

4. Синхронизация режимов интегрированной вычислительной среды.

Наличие двух уровней оптимизации: итерационный поиск, позволяющий сформировать множество перспективных вариантов рассматриваемого объекта [42], и окончательный выбор оптимального варианта на этом множестве в экспертной среде, введение дуального моделирования требует выработки и обоснования некоторых условий для синхронизации всех режимов, реализуемых в рамках интегрированного вычислительного комплекса. При этом выбор в экспертной среде включается в рандомизированную схему на основе алгоритмов, рассмотренных в [42].

Перечисленные позиции предложено обобщить в рамках единой структуры вычислительной среды, приведенной на рисунке 3.1.



Рис. 3.1. Структура интегрированной вычислительной среды оптимизации, нейросетевого и имитационного моделирования

3.2 Процедуры генерации и трансформации последовательностей псевдослучайных чисел

Погружение в рандомизированную среду процессов численной оптимизации и моделирования сложноструктурированных объектов требует формирования ряда процедур генерации последовательностей дискретных и непрерывных случайных чисел с последующей трансформацией параметров генераторов в зависимости от выполнения экстремальных и граничных требований в ходе итерационного поиска.

Источником генерации случайных чисел с заданными законами распределения является генератор псевдослучайных чисел ξ , равномерно распределенных на интервале $[0,1]$.

Путем преобразования последовательности псевдослучайных чисел ξ строятся генераторы для случайных величин с другими законами распределения, используемыми в задачах моделирования и оптимизации исследуемого класса объектов (таблица 3.1). Так, генераторы дискретных случайных чисел используются при моделировании структуры объекта [42], заданного имитационной моделью, и решении задач $\overline{\mu_1, \mu_5}$ методами, рассмотренными в п.2.2, генераторы непрерывных случайных чисел необходимы в следующих случаях:

- при имитационном моделировании потоков однородных событий;
- при включении непрерывных переменных x_n^3 , $n = \overline{1, N}$ в оптимизационную модель;
- при реализации популяционного алгоритма;
- при настройке параметров в процессе обучения нейросетевой модели.

Обратимся к построению трансформационных процедур, направленных на изменение параметров генераторов дискретных и непрерывных случайных чисел, приведенных в таблице 3.1, где:

Процедуры генерации последовательностей псевдослучайных чисел в интегрированной вычислительной среде

Тип генератора	Параметры закона распределения	Метод генерации	Вычислительная процедура
<i>Дискретные случайные величины</i>			
Множество случайных булевых величин $\tilde{x}_j = \begin{cases} 1, & j = \overline{1, J} \\ 0, & \end{cases}$	$p_{x_j} = P(\tilde{x}_j = 1)$ $q_{x_j} = 1 - p_{x_j} = P(\tilde{x}_j = 0)$	Сравнительный со случайной величиной ξ	$\tilde{x}_j = \begin{cases} 1, & \xi \leq p_{x_j} \\ 0, & \xi > p_{x_j} \end{cases}$
Целочисленная дискретная случайная величина $\tilde{i} = \{1, 2, \dots, j, \dots, J\}$	$P_i = P(\tilde{j} = i)$ $\sum_{i=1}^J p_i = 1$	Сравнительный со случайной величиной ξ	$\xi_1 \leq P_1, j = 1;$ $\xi_1 > P_1, \xi_2 \leq P_2 + P_1, j = 2;$...
<i>Непрерывные случайные величины</i>			
Равномерно распределенная на интервале [a, b]	a b	Функциональное преобразование случайной величины ξ	$a + (b - a) \cdot \xi$
Распределенная по показательному закону	λ	Функциональное преобразование случайной величины ξ	$-\frac{1}{\lambda} \cdot \ln \xi$
Распределенная по закону Эрланга	λ $\nu = 2$	Суммирование двух случайных величин ξ_1, ξ_2 , одинаково распределенных по показательному закону с параметром λ	$-\frac{1}{\lambda} \cdot \ln(\xi_1 \xi_2)$
Распределенные по нормальному закону	m σ	Суммирование случайных величин ξ	$m + \sigma \left(\sum_{i=1}^{12} \xi_i - \sigma \right)$

$P(\cdot)$ – обозначение вероятности события;

a, b – соответственно минимальное и максимальное значение случайной величины;

λ – параметр показательного закона распределения (в случае потока однородных событий его интенсивность);

ν – параметр распределения Эрланга (порядок потока Эрланга);
 m – математическое ожидание случайной величины, распределенной по нормальному закону;
 σ – среднеквадратичное отклонение случайной величины, распределенной по нормальному закону.

Для генератора совокупности булевых случайных величин необходимо трансформировать параметры p_{x_j} на $(k+1)$ -й итерации поиска, исходя из прогнозной оценки величины и направления движения, вычисленной с использованием имитационной или нейросетевой модели:

$$P_{x_j}^{k+1} = P_{x_j}^k + \pi(P_{x_j}^k, f_{\text{им (ннм)}}(\tilde{x}^k)). \quad (3.2)$$

где $p_{x_j}^1 = P(\tilde{x}_j^1 = 1)$ - задается при формировании начальных условий численной оптимизации (обычно $p_{x_j}^1 = 0,5$);

$\pi(\cdot)$ – прогностическая оценка, которая определяется из условия локального улучшения сглаженной функции \hat{f} следующим образом [42]:

$$\pi(P_{x_j}^k, f_{\text{им (ннм)}}(\tilde{x}^k)) = P_{\alpha}^{k+1}[q_{x_j}^k \chi(\Delta_1 f) - p_{x_j}^k \chi(-\Delta_1 f)]; \quad (3.3)$$

$\tilde{\alpha}^{k+1}$ - случайная булева переменная, характеризующая величину шага первого уровня поиска на $(k+1)$ -й итерации и имеющая распределение:

$$P_{\alpha}^{k+1} = P_{\alpha}^k + P_j[q_{\alpha}^k \chi(\Delta_1 f \Delta_2 f) - p_{\alpha}^k \chi(-\Delta_1 f \Delta_2 f)], q_{\alpha}^k = 1 - p_{\alpha}^k; \quad (3.4)$$

$\tilde{\gamma}$ = случайная булева переменная, характеризующая величину шага второго уровня, распределение которой задается при формировании начальных условий:

$$p_{\gamma} = P(\tilde{\gamma} = 1), \quad q_{\gamma} = 1 - p_{\gamma};$$

$\Delta_1 f, \Delta_2 f$ - вариации оптимизируемой функции $f_{\text{им (ннм)}}$ первого и второго уровней поиска

$$\Delta f = f(\tilde{x}_1^k / x_j = 0) - f(\tilde{x}_1^k / x_j = 1), \quad (3.5)$$

$\tilde{x}_1^k = (\tilde{x}_1^k, \dots, \tilde{x}_\nu^k, \dots, \tilde{x}_{j-1}^k)$, $\nu = \overline{1, J-1}$, $\nu \neq j$ - вектор случайных реализаций булевых величин \tilde{x}_ν , сгенерированных на k -й итерации;

$\chi(\cdot)$ – трансформационная функция, которая в случае движения к максимуму сглаженной оптимизируемой функции позволяет увеличивать $P_{x_j}^{k+1}$ при положительном значении вариации первого уровня $\Delta_1 f$ и совпадении по знаку вариаций первого и второго уровня:

$$\chi(a) = \begin{cases} 1, & \text{если } a > 0, \\ 0, & \text{если } a < 0. \end{cases}$$

В случае генератора целочисленной дискретной случайной величины \tilde{i} на первом этапе осуществляется переход к генератору совокупности булевых случайных величин \tilde{v}

$$b_i(\tilde{v}) = \begin{cases} 1, & \tilde{v} = i, \\ 0, & \tilde{v} \neq i, i = \overline{1, J}, \end{cases}$$

где $b(\cdot)$ – функция, осуществляющая деление значений i на два подмножества $v=i$ и $v \neq i$, на втором этапе используется трансформационная процедура, аналогичная (3.2), которая в случае случайной реализации $\tilde{i} = j$ имеет вид:

$$P_j^{k+1} = P_j^k + \pi(P_j^k, f_{\text{им (ннм)}}(\tilde{v}_j^k)), \quad (3.6)$$

где p_j^1 - задается при формировании начальных условий численной оптимизации (обычно $p_j^1 = \frac{1}{J}$, $j = \overline{1, J}$);

$\pi(\cdot)$ – прогностическая оценка, аналогичная (3.3):

$$\pi(P_j^k, f_{\text{им (ннм)}}(\tilde{v}_j^k)) = P_\beta^{k+1} [q_j^k \chi(\Delta f_1) - p_j^k \chi(-\Delta f_1)], \quad (3.7)$$

P_β^{k+1} - определяется в соответствие с (3.4);

$$q_j^k = 1 - p_j^k = \sum_{i=1}^{J-1} p_i^k, i \neq j.$$

Однако, в отличие от обычной совокупности булевых случайных величин \tilde{x}_j для \tilde{v}_i необходимо сохранить исходное для рассматриваемого генератора (см. табл. 3.1) условие нормировки на соседних итерациях

$$\sum_{i=1}^J p_i^k = 1, \sum_{i=1}^J p_i^{k+1} = 1.$$

Поэтому для значений дискретной величины $i \neq j$ необходимо провести следующий пересчет значений вероятностей:

$$p_i^{k+1} = p_i^k \cdot \frac{q_i^{k+1}}{q_i^k}, \quad \forall i \in \overline{1, J}, i \neq j. \quad (3.8)$$

Процедуру (3.6) – (3.8) предполагается использовать в качестве третьего уровня процедуры (3.2) – (3.4) для номеров переменных $x_j, j = \overline{1, J}$, вводя прогнозную оценку

$$\pi(P_j^k, f_{\text{им}}(\tilde{x}_j^k)) = P_j^{k+1} [q_j^k \chi(\Delta f_3) - p_j^k \chi(-\Delta f_3)],$$

где Δf_3 - третья реализация вариации Δf , вычисляемая по формуле (3.5).

Теперь перейдем к формированию трансформационных процедур, позволяющих изменять параметры генераторов последовательностей псевдослучайных чисел для непрерывных случайных величин, используя идеи рандомизации и сглаживания при построении поисковых алгоритмов оптимизации [42, 50].

В первую очередь это касается параметра λ для случайных величин, распределенных по показательному закону или закону Эрланга, который с одной стороны меняет числовые характеристики потоков однородных событий при имитационном моделировании, а с другой стороны – характеризует составляющие вектора непрерывных переменных $x_n^3, n = \overline{1, N}$ при решении задач оптимизации $\overline{\mu_1, \mu_3}$.

Если основным параметром для имитации является интенсивность потока однородных событий λ , то плотность распределения случайной величины $\tilde{x} \in x^3$ имеет вид:

$$\omega(\tilde{x}, \lambda) = \lambda \cdot e^{-\lambda \cdot \tilde{x}}.$$

Для построения трансформационной процедуры необходимо провести сглаживание показателя, зависящего от рандомизированной в области Ω величины \tilde{x}

$$u(\lambda) = \int_{\Omega} f(\tilde{x}) \lambda \cdot e^{-\lambda \cdot \tilde{x}} dx. \quad (3.9)$$

Переход к сглаженной функции (3.9) позволяет вычислить ее производную по параметру λ :

$$\frac{du}{d\lambda} = \int_{\Omega} f(\tilde{x})(1 - \lambda\tilde{x}) \cdot e^{-\lambda\tilde{x}} dx.$$

Обозначим через $\tilde{\eta}$ случайную величину $(\lambda\tilde{x} - 1)$, а затем выразим через нее производную:

$$\frac{du}{d\lambda} = -\frac{1}{e\lambda} \int_{\Omega} f\left(1 + \frac{\tilde{\eta}}{\lambda}\right) \tilde{\eta} \cdot e^{-\tilde{\eta}} d\eta.$$

Обратим внимание на то, что под интегралом осуществляется сглаживание по $\tilde{\eta}$, распределенной по экспоненциальному закону при $\lambda = 1$. В этом случае случайная реализация производной показателя по параметру λ вычисляется по формуле:

$$\frac{d\tilde{f}}{d\lambda} = -\left(\frac{\tilde{\eta}}{e\lambda}\right) f\left(1 + \frac{\tilde{\eta}}{\lambda}\right).$$

Тогда трансформационная процедура имеет вид:

$$\lambda^{k+1} = \lambda^k + \pi(\lambda^k, f_{\text{им(нм)}}(\bar{\eta})), \quad (3.10)$$

где λ^1 - задается при формировании начальных условий численной оптимизации исходя из особенностей натурального объекта.

$$\pi(\lambda^k, f_{\text{им(нм)}}(\bar{\eta}^k)) = x^{k+1} (\eta^{-k} / e\lambda^k) f_{\text{им(нм)}}(1 + \bar{\eta}^k / \lambda^k), \quad (3.11)$$

x^1 - величина шага на первой итерации задается при формировании начальных условий численной оптимизации.

$$x^{k+1} = x^k \exp(\gamma^k \text{sign}[f_{\text{им(нм)}}(1 + \bar{\eta}^k / \lambda^{k-1}) f_{\text{им(нм)}}(1 + \bar{\eta}^k / \lambda^k)]), \quad (3.12)$$

γ^k - величина шага, которые выбирается исходя из условий сходимости адаптивной процедуры Роббинса Монро [12]

$$\gamma^k = \frac{a}{K}, \quad (3.13)$$

a - некоторое постоянное число, которое задается при формировании начальных условий численной оптимизации, sign - обозначение функции, характеризующий знак произведения в выражении (3.12).

Особенностью трансформационной процедуры (3.10) - (3.13) является изменение величины шага x в зависимости от знака произведения реализации функции f на двух соседних шагах. Если знаки совпадают и точки поиска (k) и ($k-1$) находятся по одну сторону от экстремального значения сглаженной функции (3.9), то его величина экспоненциально растет, что увеличивает вероятность перешагнуть зону экстремума. В том случае, если знаки различаются и точки поиска (k) и ($k-1$) находятся на разных сторонах от экстремального значения сглаженной функции (3.9), величина шага уменьшается и поиск фиксируется в зоне экстремума.

В случае популяционного алгоритма рассмотренного в п.2.2 требуется выбирать в соответствии с п.2.10 случайное вещественное число τ из интервала $[\tau_{\min}, \tau_{\max}]$ с использованием равномерного закона распределения. Предполагается при погружении этого алгоритма в настраиваемую рандомизированную среду выбирать значения τ на ($k+1$)-ой итерации путем следующей трансформации параметров равномерного распределения

$$x = \frac{\tau_{\max} + \tau_{\min}}{2}, g = \tau_{\max} - x = x - \tau_{\min}.$$

Через эти параметры функцию распределения случайной величины τ представим следующим образом:

$$\varphi(\bar{\tau}) = \frac{1}{2g} [I(x-g) - I(x+g)], \quad (3.14)$$

где $I(x)$ - единичная функция Хевисайда. Плотность распределения представляет собой производную от функции (3.14).

$$u(\bar{\tau}, x, g) = \frac{1}{2g} [\delta(x-g) - \delta(x+g)],$$

где $\delta(x)$ - δ -функция.

Для построения трансформационной процедуры перейдем к сглаживаемому значению показателей f

$$u(x, g) = \frac{1}{2g} \int_{\Omega} f(\bar{\tau}) [\delta(x-g) - \delta(x+g)] dx. \quad (3.15)$$

Определим частную производную по x функции (3.15) с учетом свойства δ -функции

$$\frac{\delta u(x, g)}{\delta x} = \frac{f(x - g) - f(x + g)}{2g}, \quad (3.16)$$

тогда трансформационная процедура

$$x^{k+1} = x^k + \pi(x^k, f_{\text{им(нм)}}(x^k, g)), \quad (3.17)$$

где x^1, g - задается при формировании начального условия (2.10) популяционного алгоритма.

Прогностическая оценка, исходя из (3.16), определяет

$$\pi(x^k, f_{\text{им(нм)}}(x^k, g)) = \alpha^{k+1} \frac{f(x^k - g) - f(x^k + g)}{2g}, \quad (3.18)$$

где α^1 - величина шага на первой итерации задается при формировании начальных условий численной оптимизации, которая на последующих шагах пересчитывается оптимально (3.12) с изменяющимися на (3.13) шагом α^{12}

$$\alpha^{k+1} = \alpha^k \exp(\gamma^k \text{sign}[(f(x^{k-1} - g) - f(x^{k-1} + g)) * (f(x^k - g) - f(x^{k-1} + g))]). \quad (3.19)$$

Модернизация популяционного алгоритма, изложенного в п.2.2, составляет в дополнение его в рамках интегрированной вычислительной среды трансформационной процедуры (3.17) – (3.19). При этом на каждой итерации, этого алгоритма $\tau_{\text{max}}, \tau_{\text{min}}$ не является постоянными в соответствие с (2.10), а меняются

$$\tau_{\text{max}}^k = x^k + g, \tau_{\text{min}}^k = x^k - g.$$

Следующая трансформационная процедура вводится в интегрированной среде как дополнения настроечного алгоритма параметров нейросетевой модели, обеспечивающего минимизацию обучения

$$E = \sum_{k=1}^k (f_{\text{нм}}(a^k, x^k) - f_{\text{нм}}(a^k))^2.$$

На k -й итерации процесса обучения осуществляется рандомизацией каждого параметра $a_n^k, n = 1, N$ с использованием нормального закона распределения с математическим ожиданием $m = a_n^k$ и дисперсией σ при плотности распределения

$$\omega(\bar{a}_n, a_n^k, \sigma) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(\bar{a}_n - a_n^k)^2}{2\sigma^2}}$$

Осреднённая функция, по которой ведется настройка параметров нейросетевой модели, имеет вид

$$U(a_n, \sigma) = \frac{1}{\sqrt{2\pi\sigma}} \int_{\Omega} E(\bar{a}) e^{-\frac{(\bar{a}_n - a_n^k)^2}{2\sigma^2}} da_n. \quad (3.20)$$

Используя (3.20), определяем частную производную по параметру a^k

$$\frac{\delta U(a_n^k, \sigma)}{\delta a_n^k} = \frac{1}{\sqrt{2\pi\sigma}} \int_{\Omega} E(\bar{a}) e^{-\frac{(\bar{a}_n - a_n^k)^2}{2\sigma^2}} \frac{\bar{a}_n - a_n^k}{\sigma^2} da_n, \quad (3.21)$$

Введем случайную величину $\bar{\eta} = \frac{\bar{a}_n - a_n^k}{\sigma}$.

Она распределена по нормальному закону с нулевым математическим ожиданием и единичной дисперсией. Будем интегрировать её с использованием параметра гауссовского распределения, и с новыми параметрами. Тогда (3.21) применим следующим образом

$$\frac{\delta U(a_n^k, \sigma)}{\delta a_n^k} = \frac{1}{\sigma} \int_{\Omega} E(a_n^k + \sigma \bar{\eta}) \bar{\eta} d\eta. \quad (3.22)$$

На основании (3.22) получим трансформационную процедуру

$$a_n^{k+1} = a_n^k + \pi(a^k, E(a_n^k, \sigma)), \quad (3.23)$$

где a^1, σ - задается при формировании начальных условий;

$$\pi(a_n^k, E(a_n^k, \sigma)) = \alpha^{k+1} \frac{1}{\sigma} E(a_n^k + \sigma \bar{\eta}^k) \eta^k, \quad (3.24)$$

$$\alpha^{k+1} = \alpha^k \exp \left[\text{sign} \left[\frac{1}{\sigma} (E(a_n^{k-1} + \sigma \bar{\eta}^{k-1}) \eta^{k-1} E(a_n^k + \sigma \bar{\eta}^k) \eta^k) \right] \right]. \quad (3.25)$$

Именно трансформационная процедура (3.23) – (3.25) введена дополнительно к алгоритму настройки параметров нейросетевой модели в интегрированной вычислительной среде.

Таким образом, мы рассмотрели все трансформационные процедуры, обеспечивающие изменения параметров генераторов псевдослучайных

последовательностей, приведенных в таблице 3.1, ориентируя их на соответствующие поисковые схемы в рандомизированной среде.

3.3 Синхронизация режимов нейросетевого и имитационного моделирования и численной оптимизации в рамках вычислительной среды

В интегрированной вычислительной среде, предложенной в п. 3.1, возможны следующие режимы объединения процедур нейросетевого и имитационного моделирования и процедур и поисковой оптимизации:

- 1) Поисковая оптимизация на основе имитационной модели;
- 2) Поисковая оптимизация на основе имитационной модели с обучением нейросетевой модели;
- 3) Поисковая оптимизация на основе имитационной модели с верификацией нейросетевой модели;
- 4) Поисковая оптимизация на основе нейросетевой модели;
- 5) Окончательный выбор варианта сложноструктурированного объекта с использованием имитационной модели;
- 6) Окончательный выбор варианта сложноструктурированного объекта с использованием нейросетевой модели.

Для синхронизации процедур, составляющих перечисленные выше режимы необходимо установить условия останова ряда процессов, определяющих переходы между ними:

условие останова итерационного процесса численной оптимизации для перехода к процедурам окончательного выбора варианта сложноструктурированного объекта;

условие останова процесса обучения нейросетевой модели для перехода к процессу верификации;

условие останова процесса верификации нейросетевой модели для перехода к ее использованию в процедуре численной оптимизации и выбора.

С целью обоснования перечисленных условий построим математическую модель синхронизации режимов численной оптимизации и моделирования.

В первую очередь охарактеризуем состояние рандомизированной вычислительной среды на начальном этапе. Поскольку согласно трансформационным процедурам, построенным в пункте 3.2, на первом шаге задается равномерное распределение всех рандомизированных переменных, то равномерным является распределение $t = \overline{1, T}$ вариантов сложноструктурированных объектов

$$P^1 = (P_1^1, \dots, P_t^1, \dots, P_T^1),$$

$$P_t^1 = \frac{1}{T}, t = \overline{1, T}$$

с начальной неопределенностью [91] $\chi^1 = \log T$.

Преобразование параметров начальных распределений с использованием трансформационных процедур приводит на k -ой итерации к неравномерному распределению

$$P^k = (P_1^k, \dots, P_t^k, \dots, P_T^k),$$

$$P_t^k, \quad t = \overline{1, T}$$

$$\sum_{t=1}^T P_t^k = 1$$

что характеризуется относительной неопределенностью по отношению к равномерному распределению [73]

$$\chi^k \left(\frac{P^1}{P^k} \right) = - \sum_{t=1}^T P_t^k \log P_t^1. \quad (3.26)$$

В этом случае переход от начальных параметров рандомизированной среды к трансформационным количественно характеризуется величиной полезной информации [10]

$$I^k = \log T - \chi^k \left(\frac{P^1}{P^k} \right)$$

В зависимости от средств моделирования, на основе которых вычисляются прогнозные оценки движения поиска в зону экстремума, определяется $\omega = \overline{1, w}$ вариантов значений координат вектора P^k :

$$P_{t\omega}^k; t = \overline{1, T}; \omega = \overline{1, \omega};$$

$$\sum_{t=1}^T P_{t\omega}^k = 1; \forall \omega = \overline{1, \omega}.$$

При этом появление каждого ω -го варианта характеризуется вероятностью

$$P_{\omega}^k; \omega = \overline{1, \omega};$$

$$\sum_{\omega=1}^W P_{\omega}^k = 1.$$

Учитывая (3.26), рассчитаем взвешенную неопределенность

$$\chi^k \left(\frac{P^1}{P^k} \right) = - \sum_{\omega=1}^w P_{\omega}^k \sum_{t=1}^T P_t^1 \log P_{t\omega}^k. \quad (3.27)$$

У нас появилась возможность выделить запас полезной информации, который образуется как за счет итерационного процесса численной оптимизации, так и выбора варианта моделирования в интегрированной вычислительной среде, который принято называть скрытой информацией [10]

$$I_{\text{скр}} = \log L - \chi \left(\frac{P^1}{q^k} \right) \quad (3.28)$$

$$q^k = (q_1^k, \dots, q_t^k, \dots, q_T^k),$$

$$q_t^k = (1/\omega) \sum_{\omega=1}^w P_{t\omega}^k \quad (3.29)$$

Разделим в (3.27) под знаком логарифма значения $P_{t\omega}^k$ на величину W_{qt}^k и преобразуем в начале к виду

$$\chi^k \left(\frac{P^1}{P^k} \right) = - \sum_{t=1}^T P_t^1 \sum_{\omega=1}^W P_{\omega}^k (\log q_t^k + \log \omega + \log \left(\frac{P_{t\omega}^k}{\omega \cdot q_t^k} \right)),$$

а затем, учитывая $\sum_{t=1}^T P_t^1 = 1, \sum_{\omega=1}^W P_{\omega}^k = 1$, получим

$$\chi^k \left(\frac{P^1}{P^k} \right) = - \log \omega - \sum_{t=1}^T P_t^1 (\log q_t^k + \sum_{\omega=1}^W \log \left(\frac{P_{t\omega}^k}{\omega \cdot q_t^k} \right)) \quad (3.30)$$

С учетом (3.28), которое можно представить следующим образом:

$$\sum_{\omega=1}^W \left(\frac{P_{t\omega}^k}{\omega \cdot q_t^k} \right) = 1$$

и соотношения, связывающие в теории информации [3.4] оценки неопределенности и энтропии предпочтений вариантов при выборе наилучшего из них $H(P_\omega^k)$

$$\chi\left(\frac{P^1}{P_{t\omega}^k}\right) \geq H(P_\omega^k),$$

получаем оценку снизу для выражения

$$-\sum_{\omega=1}^W P_\omega \log\left(\frac{P_{t\omega}^k}{\omega \cdot q_t^k}\right) \geq -\sum_{\omega=1}^W P_\omega \log P_\omega = H(P_\omega^k) \quad (3.31)$$

Используя (3.31) и преобразование (3.30) и соотношение (3.28), имеем

$$\chi\left(\frac{P^1}{P_{t\omega}^k}\right) \geq H(P_\omega^k) - \log \omega + \log T - I_{скр}^k \quad (3.32)$$

Соотношение (3.32) удастся проверить на каждой k -ой итерации, принимая во внимание следующую особенность рандомизированного поиска по булевым переменным x_j . Поскольку каждому t -му варианту сложноструктурированного объекта соответствует некоторый набор значений булевых величин x_j , равным 1 либо 0, то случайное событие, связанное с появлением этих наборов, является сложным событием с вероятностью [15]

$$P_t^k = \prod_{\forall x_j^k=1} P_{x_j}^k \prod_{\forall x_j^k=0} q_{x_j}^k. \quad (3.33)$$

Аналогичным образом определяются $P_{t\omega}^k$ для каждого ω -го варианта. Таким образом, удастся определить $P_{x_j\omega}^k, q_{x_j\omega}^k$ в зависимости от структуры интегрированной вычислительной среды.

Вычислимость всех компонентов выражения (3.32) позволяет ввести понятие ξ -баланса ($\xi > 0$) между его левой и правой частью, которое определяет степень приближения итерационного процесса к зоне экстремума. В этой зоне формируется множество доминирующих (перспективных) вариантов [49], поэтому распределение P_t^k и $P_{t\omega}^k$ становятся существенно неравномерными, что приводит к увеличению величины $I_{скр}$ и снижению неопределенности выбора. Возникает возможность выбора на ограниченном пространстве вариантов с

использованием рандомизированных схем экспертного оценивания [42], что дает возможность их ограниченного включения в интегрированную вычислительную среду (см. рисунок 3.1).

Процесс уменьшения величины

$$\xi^k = \chi\left(\frac{P^1}{P_{t\omega}^k}\right) - H(P_{\omega}^k) + \log \omega - \log T + I_{скр}^k$$

от k -ой итерации к $(k+1)$ -й является релаксационным (3.8), так как в [49] доказано: H - релаксационность информационных характеристик рандомизированного поиска, что позволяет в качестве одного из параметров начальных условий задать некоторую малую величину $\xi^* > 0$ и определить условие останова поиска в интегрированной вычислительной среде

$$\xi^k \leq \xi^*. \quad (3.34)$$

Условие (3.34) и соотношение (3.32) представляют собой математическую модель синхронизации, введенных выше, режимов численной оптимизации, имитационного и нейросетевого моделирования из этой модели следует условие останова итерационного процесса численной оптимизации

$$\xi^{koon} \leq \xi^{*onm} \quad (3.35)$$

с учетом вычисления вероятности вариантов объектов по формуле (3.33).

Для режима обучения нейросетевой модели в рандомизированной среде условие останова определяется так же из математической модели синхронизации. При этом в качестве вариантов степени настройки модели будем рассматривать $t = \overline{1, T}$ вариантов вектора параметров $a^k = (a_1^k, \dots, a_n^k, \dots, a_N^k)$ значения которых на k -ой итерации определяет их с использованием трансформационной процедуры (3.23) - (3.25) степень обеспечения каждым вариантом минимизации.

Критерий обучения нейросети E характеризуется распределением вероятности

$$P_t^{ob}, t = \overline{1, T}, \sum_{t=1}^T P_t^{ob} = 1 \quad (3.36)$$

Каждой итерации настройки параметров в условиях с рандомизацией на основе нормального закона соответствует вероятности того, что случайная реализация параметра \bar{a}_n находится в заданном диапазоне $[a_n^{\min}, a_n^{\max}]$ [15]

$$P_{a_n}^k = P(a_n^{\min} \leq \bar{a}_n \leq a_n^{\max}) = \Phi\left(\frac{a_n^{\max} - a_n^k}{\sigma}\right) - \Phi\left(\frac{a_n^{\min} - a_n^k}{\sigma}\right), \quad (3.37)$$

где $\Phi(\cdot)$ - нормированная функция Лапласа. На основе значений (3.37) вычисляются вероятности случайного события соответствующее тому, что все параметры $a_n, n=1, N$ укладываются в заданный диапазон, которые и определяют распределение (3.36)

$$P_t^{об} = \prod_{n=1}^N P_{a_n}^k. \quad (3.38)$$

Выражение (3.38) аналогично (3.33), поэтому по соотношению (3.32) вычисляется ξ^{koo} и устанавливается условие останова процесса обучения:

$$\xi^{koo} \leq \xi^{*об}. \quad (3.39)$$

Для синхронизации режимов интегрированной вычислительной среды помимо условий останова итерационных процессов (3.34), (3.39) предполагается условие останова процесса верификации нейросетевой модели по статистическому критерию сравнения дисперсий ошибок - критерию Фишера, расчетное значение которого определяется как

$$F_{\text{расчетное}} = \frac{D_{\text{оep}}}{D_{\text{об}}},$$

где $D_{\text{оep}}$ - дисперсия ошибки модели в процессе верификации:

$$D_{\text{оep}} = \frac{\sum_{k=1}^{k_g} (f_{nm}^k - f_{nm}^k)^2}{k_g - 1};$$

$D_{\text{об}}$ - дисперсия ошибки модели в процессе обучения:

$$D_{\text{об}} = \frac{\sum_{k=1}^{k_0} (f_{nm}^k - f_{nm}^k)^2}{k_0 - 1}.$$

Останов процесса верификации осуществляется при условии $F_{расчети} \leq F_{табл}$ [39], где $F_{табл}$ - табличное значение критерия Фишера при числе степени свободы $(k_s - 1)$ и $(k_0 - 1)$.

Выводы третьей главы:

1. В качестве концептуальной основы интеграции численных процедур оптимизации и моделирования в единый цикл обоснованным является их погружение в рандомизированную вычислительную среду. При этом механизмы рандомизации и сглаживания дополняются предложенными в работе механизмами нейросетевого и имитационного моделирования, синхронизации режимов и итерационной трансформации стохастического процесса направленного поиска экстремума.

2. При построении интегрированной вычислительной среды целесообразно объединить все процедуры между двумя основными модулями: генерации последовательностей дискретных и непрерывных чисел, инициализирующим последующий процесс вычислений, и трансформации параметров генераторов, определяющих переход к новой итерации поиска либо останова и формирования множества перспективных вариантов сложноструктурированных объектов.

3. Возможность реализации в рамках единой вычислительной среды целого ряда режимов объединения процедур численной оптимизации, нейросетевого и имитационного моделирования определяет необходимость обоснованных переходов между ними. Приемлемыми оценками таких переходов являются теоретико-информационные характеристики, позволяющие разработать математическую модель синхронизации режимов и на ее основе обосновать условия останова итерационного процесса поиска, обучения и верификации нейросетевой модели и перехода к выбору окончательного варианта структуры объекта исследования с использованием экспертной информации.

4 АНАЛИЗ ЭФФЕКТИВНОСТИ РАЗРАБОТАННЫХ СРЕДСТВ ОПТИМИЗАЦИИ И МОДЕЛИРОВАНИЯ НА ОСНОВЕ РЕЗУЛЬТАТОВ ВЫЧИСЛИТЕЛЬНОГО И НАТУРНОГО ЭКСПЕРИМЕНТОВ

В данной главе рассмотрим структуру программного комплекса, в основе которого будет лежать разработанная интегрированная среда численной оптимизации, нейросетевого и имитационного моделирования, а затем на одном из сложноструктурированных объектов в рамках информационной системы, обеспечивающей функционирование транспортной сети, по результатам вычислительного и натурального экспериментов проведем оценку эффективности разработанных в главах 2 и 3 данной работы математических методов, моделей и алгоритмов.

4.1 Характеристика программного комплекса и его структура

Разработанный программный комплекс предназначен для прикладного использования при осуществлении оптимизации сложноструктурированного объекта на основе обработки статистических данных об интенсивности и загруженности компонентов транспортной сети и прогнозирования этих параметров, а также проведения моделирования и решения задач оптимизации при различных вариациях компонентов структуры: схемы организации движения транспорта на нем, определения величин очередей и перегрузки. Поиск эффективного решения осуществляется в процессе управления в рамках информационной системы, обеспечивающей организацию пропускной способности объектов улично-дорожной сети городских автодорог и поддержки процесса проектирования новых и развязок для проблемных ее участков.

Для корректной работы программы необходимы следующие системные требования:

- операционная система: Microsoft Windows 7 и выше;
- процессор: Intel Pentium(R) D 3.40 GHz и выше;

- минимальный объем ОЗУ: 1024 МБ;
- видеокарта: интегрированная NVIDIA GeForce 7300 GS и выше;
- установленной дополнительно ПО: Microsoft Office v. 2007 и выше, MatLAB, AnyLogic, Rockwell Arena 14.

Рассмотрим обобщенную структуру программного комплекса и схему его интеграции с автоматизированной системой проектирования структуры элементов городской транспортной сети (рис. 4.1).

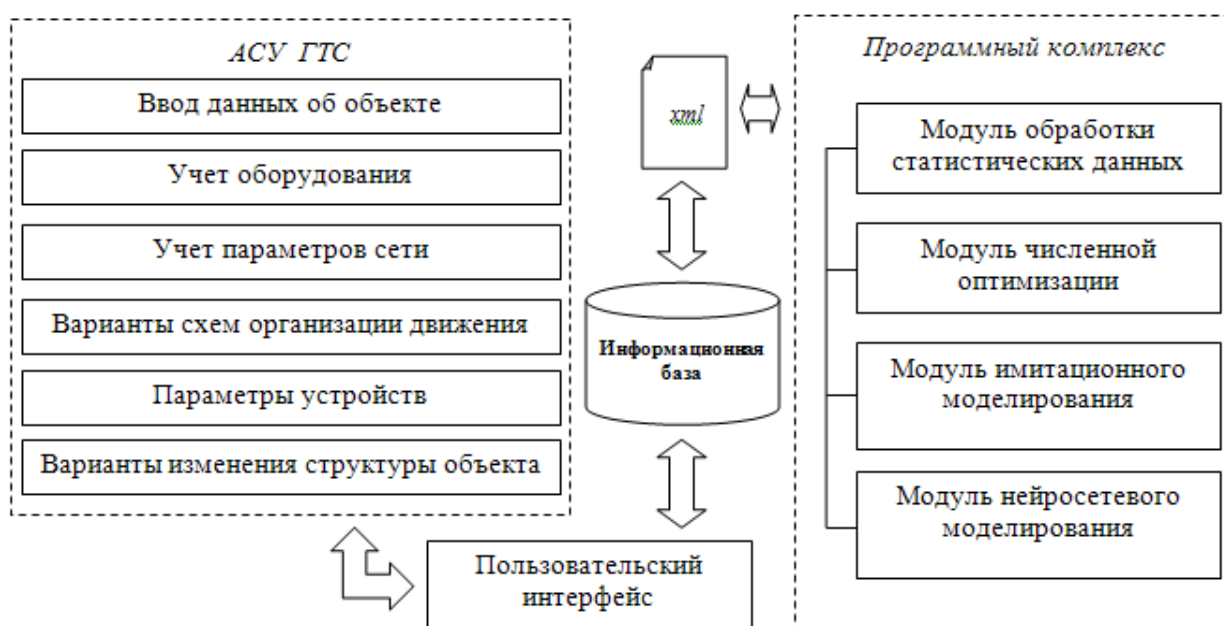


Рис. 4.1. Схема интеграции программного комплекса в АСУ транспортными потоками

Далее представим подробное описание каждого из модулей, входящих в разработанную программную среду.

Первый модуль – это модуль обработки статистических данных. Данный модуль предназначен для обработки статистики о загруженности исследуемого участка дорожной сети, а также он позволяет преобразовывать статистические данные в необходимый для дальнейшего использования формат. В нашем случае статистика будет использоваться в двух вариантах: первый – в качестве входных данных для решения задачи о прогнозировании интенсивности движения на заданный период времени, второй – для определения типа

распределения, соответствующего интенсивности движения на участке в настоящий момент времени.

Модуль оптимизационных процедур – это модуль, в котором реализованы рассмотренные и разработанные в главах 2 и 3 оптимизационные методы и алгоритмы. В частности, адаптивные алгоритмы для статического и динамического режимов управления рассматриваемым типом объектов, в основе которых лежит алгоритм поведения обезьян, и метод многоальтернативной оптимизации для формирования множества перспективных решений.

Модуль моделирования включает в себя работу с имитационными и нейросетевой моделями. В частности, с помощью этого модуля на основе особенностей решаемой задачи мы из базы разработанных имитационных моделей выбираем модель, соответствующую условиям, а затем задаем необходимые условия. После этого сохраняем полученную модификацию имитационной модели. Затем в этом же модуле осуществляем по заданным параметрам обучение нейронной сети и ее верификацию. Далее полученные результаты сохраняем и заносим в базу данных.

Модули имитационного и нейросетевого моделирования включают в себя процедуры обработки результатов моделирования, что, в свою очередь, позволяет результаты, полученные в ходе применения к исходным данным процедур численной оптимизации и алгоритмов, лежащих в основе нейросетевого и имитационного моделирования привести их к виду, удобному для дальнейшего использования (графического представления и осуществления экспертного выбора наилучшего решения).

Также реализован механизм взаимодействия предложенной вычислительной среды с программными системами предприятий, в основе которого лежит технология обмена данными при помощи Xml-файлов.

4.2 Особенности проведения и результаты вычислительного эксперимента

В качестве объекта исследования будем рассматривать базовые элементы транспортной сети и систему повышения эффективности их функционирования. Учитывая их особенности, можно сделать вывод о соответствии всех свойств этого объекта свойствам сложноструктурированных объектов, которые подробно описаны в главах 1 и 2 данной работы.

Чаще всего мы сталкиваемся с двумя видами базовых элементов – это перекрестки с простым пересечением дорог и перекрестки с круговой организацией движения. Рассмотрим для каждого из них особенности методики проведения вычислительного эксперимента с использованием предлагаемого программного обеспечения.

Исходя из процедур, лежащих в основе разработанной вычислительной среды в процессе управления рассматриваемыми объектами, можно выделить следующие этапы проведения вычислительного эксперимента:

1. Выбор параметров для исследуемого объекта.
2. Построение оптимизационной модели исследуемого объекта, формирование соответствующей оптимизационной задачи.
3. Процесс рандомизации оптимизируемых переменных и задание начальных условий.
4. Проведение структурного моделирования, в основе которого лежат процедуры имитационного и нейросетевого моделирования.
5. Решение задачи численной оптимизации на основе адаптивных популяционных алгоритмов.
6. Формирование возможных вариантов структуры исследуемого объекта.
7. Экспертный выбор наилучшего варианта объекта с учетом поставленных цели и задач.

Рассмотрим методику проведения вычислительного эксперимента более подробно.

На первом этапе определимся со структурными компонентами перекрестка и зададим зависимость, связывающую их между собой.

Первым структурным компонентом является количество полос для движения в каждом из направлений движения перекрестка. Данный компонент характеризует значение пропускной способности на данном участке улично-дорожной сети.

Следующими компонентами являются: наличие светофора на исследуемом перекрестке и схема временных характеристик его переключения. Управление этим параметром в зависимости от транспортной ситуации на участке и ее прогноза позволит понизить время ожидания транспортных средств, находящихся в очереди, что в свою очередь приведет к минимизации количества пробок и заторов.

Третьим компонентом является объектно-ориентированный вариант интенсивность движения транспортных средств в каждом из направлений движения на перекрестке. Изучение третьего компонента (сбор статистических данных) позволят провести моделирование и на его основании сделать прогноз об интенсивности движения на исследуемом участке улично-дорожной сети.

После того, как со структурными компонентами, влияющими на пропускную способность перекрестка, определились, речь пойдет о построении оптимизационной модели, состоящей из целевой функции, включающей оптимизационные переменные, и ряд ограничений, удовлетворяющих условию решаемой задачи.

В предлагаемой модели решено использовать оптимизационные переменные x_{li} – количество полос для движения на исследуемом перекрестке в i -ом направлении движения и x_{ti} – дискретные моменты времени переключения сигналов светофора в i -ом направлении движения перекрестка. При этом $x_{t_i} = \{x_{t_i}^1, x_{t_i}^2, \dots, x_{t_i}^n\}$.

Целевая функция будет иметь вид (4.1):

$$Y=f(x_{li}, x_{ti}) \rightarrow \max, \quad (4.1)$$

где Y – пропускная способность исследуемого участка. При этом данные прогноза об интенсивности движения на исследуемом участке дорожной сети города получаем с помощью нейросетевой модели. Для этого будем использовать сеть Кохонена.

Следующим компонентом оптимизационной модели являются ограничения. В нашем случае ограничения будут представлены в виде (4.2) и (4.3):

$$\sum_{i=1}^I x_{t_i} \cdot \Delta S_i \leq S_i, \quad (4.2)$$

где x_{t_i} – количество полос для движения в i -ом направлении, ΔS_i – ширина одной полосы для движения в метрах, S_i – максимально допустимый интервал ширины автодороги на исследуемом участке.

$$x_{t_i} = \{x_{t_i}^1, x_{t_i}^2, \dots, x_{t_i}^n\}, \quad (4.3)$$

где x_{t_i} – дискретные моменты времени переключения сигналов светофора в i -ом направлении движения перекрестка; величина x_{t_i} может принимать значения только из заданного множества (4.3).

Для решения задачи (4.1 – 4.3) необходимо для начала произвести ранжирование ограничений по степени критичности к состоянию экстремума с использованием алгоритма настройки вероятностей привлечения ограничений на основе аддитивной функции (4.4):

$$\Phi(x_{l_1}, x_{t_{i_2}}) = \sum_i f(x_{l_1}) + \sum_{i_2} x_{t_{i_2}} (S_{i_2} - F_{i_2}(x_{l_1})). \quad (4.4)$$

Затем построим процедуру вариационного моделирования для $i_2 \in I_2$. Она формируется на основе расчетных прогностических оценок, использующих значения вектора-градиента функции $\Phi(x, y)$, компоненты которого :

$$\frac{\partial \Phi(x_{l_1}^k, x_{t_{i_2}}^k)}{\partial x_{t_{i_2}}} = (S_{i_2} - F_{i_2}(x_{l_1}^k)).$$

Вычисление коэффициентов штрафной функции (6) на k -ой итерации будем осуществлять по формулам (4.5), (4.6):

$$P_0^{k+1} = (1 + \sum_i x_{t_{i2}}^k)^{-1}, \quad (4.5)$$

$$P_{i2}^{k+1} = x_{t_{i2}}^k (1 + \sum_{i_2=1}^{I_2} x_{t_{i_2}}^k)^{-1} \quad (4.6)$$

После того, как учет ограничений был проведен, перейдем к формированию множества перспективных решений. Базовые вариационные процедуры при этом направлены на формирование вариаций показателя Y , по которому принимаются решения, для двух случаев:

- при изменении альтернативных переменных $Z_m = 0$ либо 1 ;
- при изменении переменных x_i , где $i = \overline{1, I}$.

Вариация показателя Y по переменной Z_m на k -ом шаге определяется следующим образом (4.7):

$$\Delta_m^k Y = Y(Z^k / Z_m = 0) - Y(Z^k / Z_m = 1) \quad (4.7)$$

где Z^k – вектор случайных реализаций альтернативных переменных.

Для вычисления значения (4.7) выбирается W_l –я альтернатива m -го уровня агрегирования. Учитывая многократность реализации (9) предлагается управлять распределением (4.8), изменяя его в соответствии со значением вариации и распределением случайного шага.

$$p_{Bm} = p(B_m = 1), q_{Bm} = p(B_m = 0) = 1 - p_{Bm} \quad (4.8)$$

После этого для полного использования скрытой информации величину вероятности p_{Bm} целесообразно увеличивать для более существенного изменения p_{Zm} .

$$p_{zm}^{k+1} = p_{zm}^k + p_{Bm}^{k+1} [q_{zm}^k (\Delta_{1m} Y) - p_{zm}^k (-\Delta_{1m} Y)] \quad (4.9)$$

Далее рассмотрим особенности проведения имитационного моделирования. В данном случае имитационные модели перекрестков рассматриваемых типов будут отличаться друг от друга.

Представим в таблице 4.1 структурные компоненты системы управления структурой элементов транспортной сети. Основными компонентами будут являться перекрестки различных типов.

Структурные компоненты системы управления структурой элементов
транспортной сети

№ п/п	Топология типовых компонентов	Вариативные элементы структуры	Вариативные параметры
1	2	3	4
1	Четырехсторонняя с простым пересечением дорог	- количество полос для движения; - реверсивные полосы; - поворотные полосы; - регулировочные компоненты;	- время переключения светофоров; - режим использования реверсивных полос;
2	Четырехсторонняя с круговой организацией движения	- количество полос для движения; - реверсивные полосы; - поворотные полосы; - регулировочные компоненты;	- время переключения светофоров; - режим использования реверсивных полос;
3	Многосторонняя	- количество полос для движения; - реверсивные полосы; - поворотные полосы; - регулировочные компоненты; - эстакады и мосты;	- время переключения светофоров; - режим использования реверсивных полос;
4	Клеверообразная	- количество полос для движения; - реверсивные полосы; - поворотные полосы; - регулировочные компоненты; - эстакады и мосты; - тоннели;	- время переключения светофоров; - режим использования реверсивных полос;
5	Кольцевая	- количество полос для движения; - реверсивные полосы; - поворотные полосы; - регулировочные компоненты; - эстакады и мосты; - тоннели;	- время переключения светофоров; - режим использования реверсивных полос;

Также могут встречаться случаи с комбинированными вариантами рассмотренных топологий.

Рассмотрим перекресток с простым пересечением дорог. Через перекресток движутся потоки автомобилей. Упорядочение движения на нем определяется информационной системой управления. На рисунке 4.2 представим топологическую модель организации движения на 4-х стороннем перекрестке.

По представленной на рисунке 4.2 модели построим структурную схему процесса организации движения на перекрестке и представим ее на рисунке 4.3. Для этого обозначим каждое из направлений движений буквами А, В, С и D соответственно. На перекрестке имеется светофор, который регулирует движение автотранспорта.

Далее по рисунку 4.3 разработаем моделирующий алгоритм для одного из направлений прибытия потоков машин. С помощью разработанного алгоритма [78] решаются задачи формирования очередей в *IJ*-ом направлении, а также ведется подсчет количества машин, остающихся и добавляющихся в очереди и успешно преодолевших перекресток.

При построении имитационной модели [78] производится разбиение процесса функционирования системы на отдельные достаточно автономные подпроцессы. Блоки такой модели бывают основными и вспомогательными.

Каждый основной блок соответствует некоторому подпроцессу моделируемой системы, а вспомогательные блоки лишь представляют составную часть машинной модели, не отражая функции моделируемой системы, они нужны лишь для машинной реализации модели, фиксации и обработки результатов моделирования. В решаемой задаче блок-схема алгоритма для одного из направлений движения представлена на рисунке 4.4.

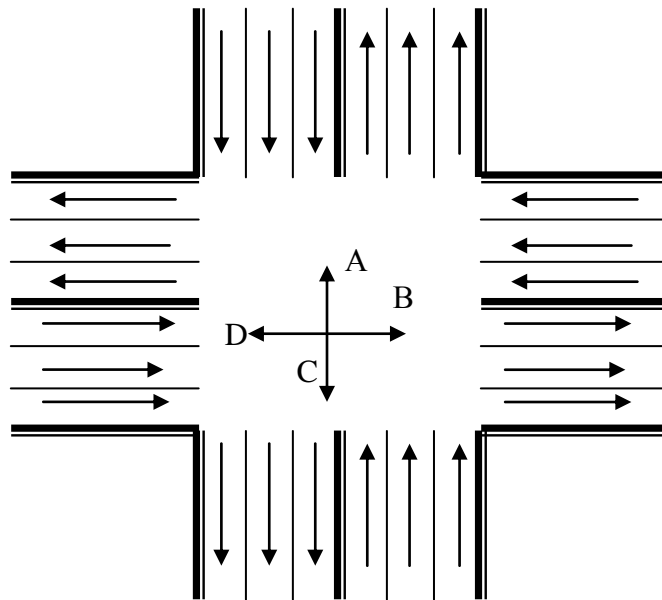


Рис. 4.2. Топологическая модель функционирования простого перекрестка

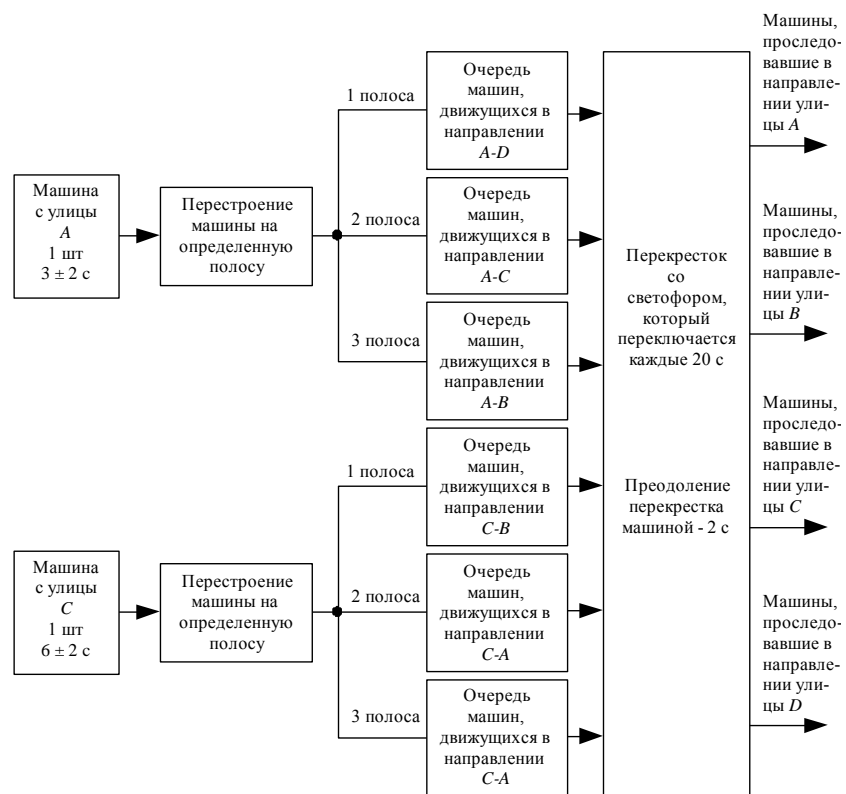


Рис. 4.3. Структурная схема процесса имитации движения транспортных средств на перекрестке

После того, как имитационная модель построена, алгоритм ее функционирования разработан и основные ее параметры для конкретной задачи

определены, переходим к процессу оценки эффективности найденных решений для структурной оптимизации исследуемого перекрестка.

Для этого запустим процесс моделирования по сформированным исходным данным, а затем проведем обработку результатов моделирования. Исходя из полученных результатов, можно будет продолжить действие по одному из трех возможных путей. Рассмотрим каждый из них более подробно.

1. Если по результатам моделирования время задержек на выбранном участке перекрестка является меньше предельно допустимого, определенного его проектной мощностью, то найденное решение является оптимальным и позволит повысить эффективность организации движения на исследуемом перекрестке.

2. Если по результатам моделирования время задержек на исследуемом участке перекрестка превышает значение предельно допустимого, определенного его проектной мощностью на величину до 15 %, то в данном случае возможно произвести корректировку варьируемых параметров (в частности, время переключения режимов светофора), при этом изменения можно внести в параметры текущей имитационной модели.

3. Если время задержек по результатам моделирования превышает значение над предельно допустимым более, чем на 15 %, то в данном случае происходит возвращение к предыдущему элементу вычислительной среды и с учетом выявленных при имитационном моделировании параметров, снижающих эффективность найденной структуры, повторяется процедура поиска оптимального решения с использованием популяционных алгоритмов.

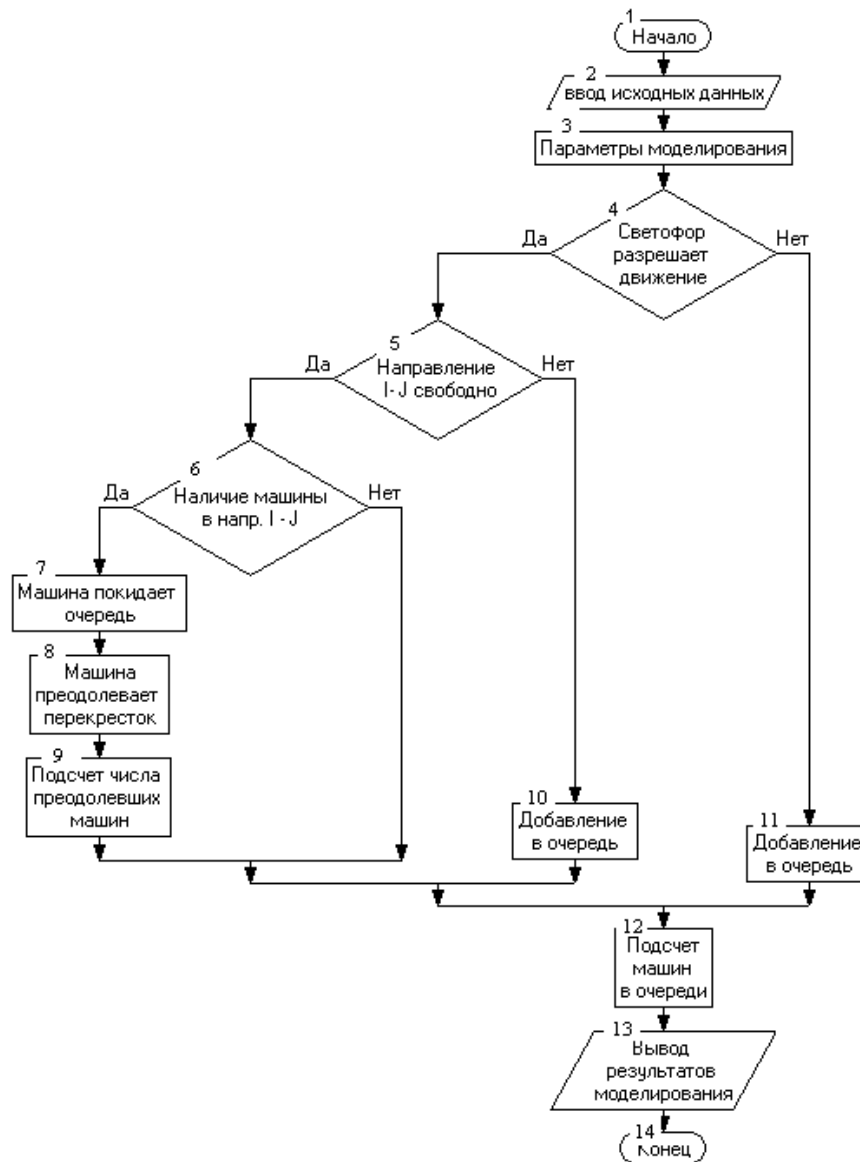


Рис. 4.4. Блок-схема алгоритма имитационного моделирования вариации компонентов элемента транспортной сети

Далее рассмотрим перекрестки дорог с кольцевой организацией движения. Также, как и в предыдущем случае, регулировку движения на нем осуществляет информационная система управления. Топологическую модель для перекрестков такого типа представим на рисунке 4.5.

Затем по рисунку 4.5 разработаем моделирующий алгоритм для одного из направлений прибытия потоков машин. С помощью разработанного алгоритма решаются задачи формирования очередей в Π -ом направлении, а также ведется подсчет количества машин, остающихся и добавляющихся в очереди и успешно преодолевших перекресток. Отличием разрабатываемого

алгоритма для данного типа перекрестков от предыдущего случая будет добавление еще одного условия. Представим его на рисунке 4.6.

Еще одной особенностью перекрестков, которую необходимо учесть в процессе повышения эффективности их функционирования, является наличие так называемых сужений ширины проезжей части. Будем называть такие задачи «горлышко». На рисунке 4.7 представим блок-схему алгоритма решения задачи «горлышко» с помощью имитационного моделирования.

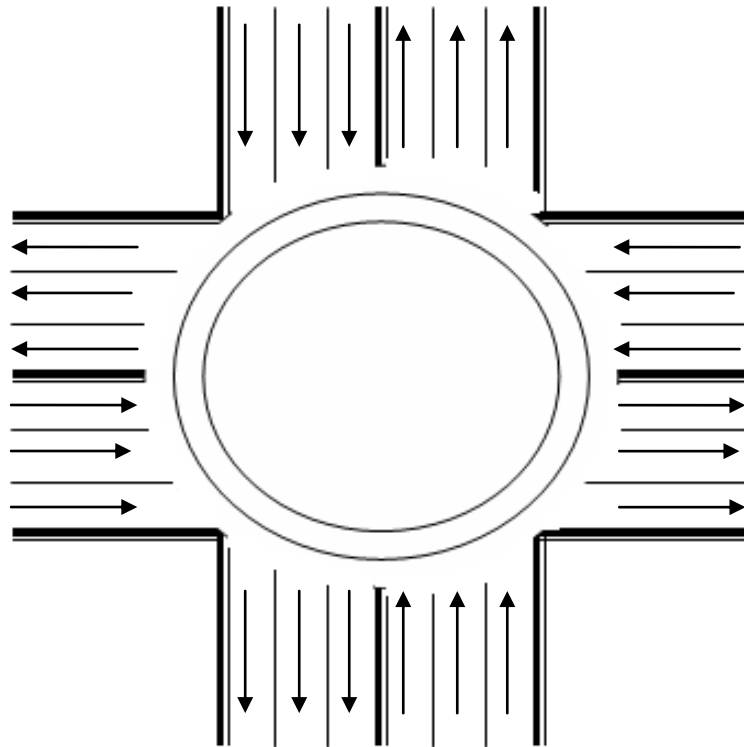


Рис. 4.5. Топологическая модель функционирования перекрестка с круговым движением

После того, как на основе оптимизационных алгоритмов и по результатам имитационного моделирования сформированы множества перспективных вариантов объекта, осуществляется экспертный выбор наилучшего из них для имеющихся условий.

Полученное решение (окончательный вариант параметров структуры исследуемого объекта) является проектным для осуществления реструктуризации перекрестка с целью повышения его пропускной способности.

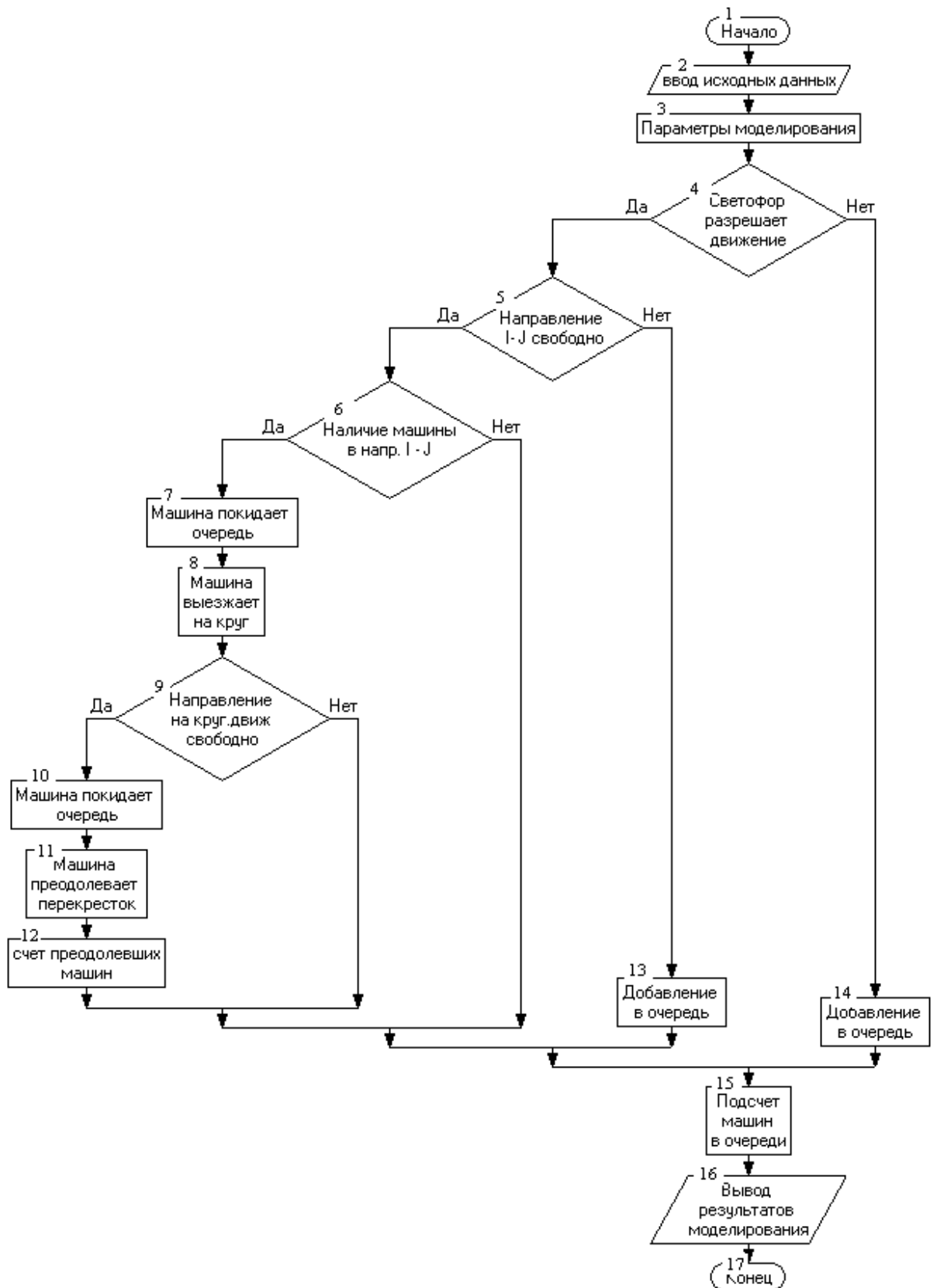


Рис. 4.6. Алгоритм функционирования модели перекрестка с круговым движением в информационной системе управления

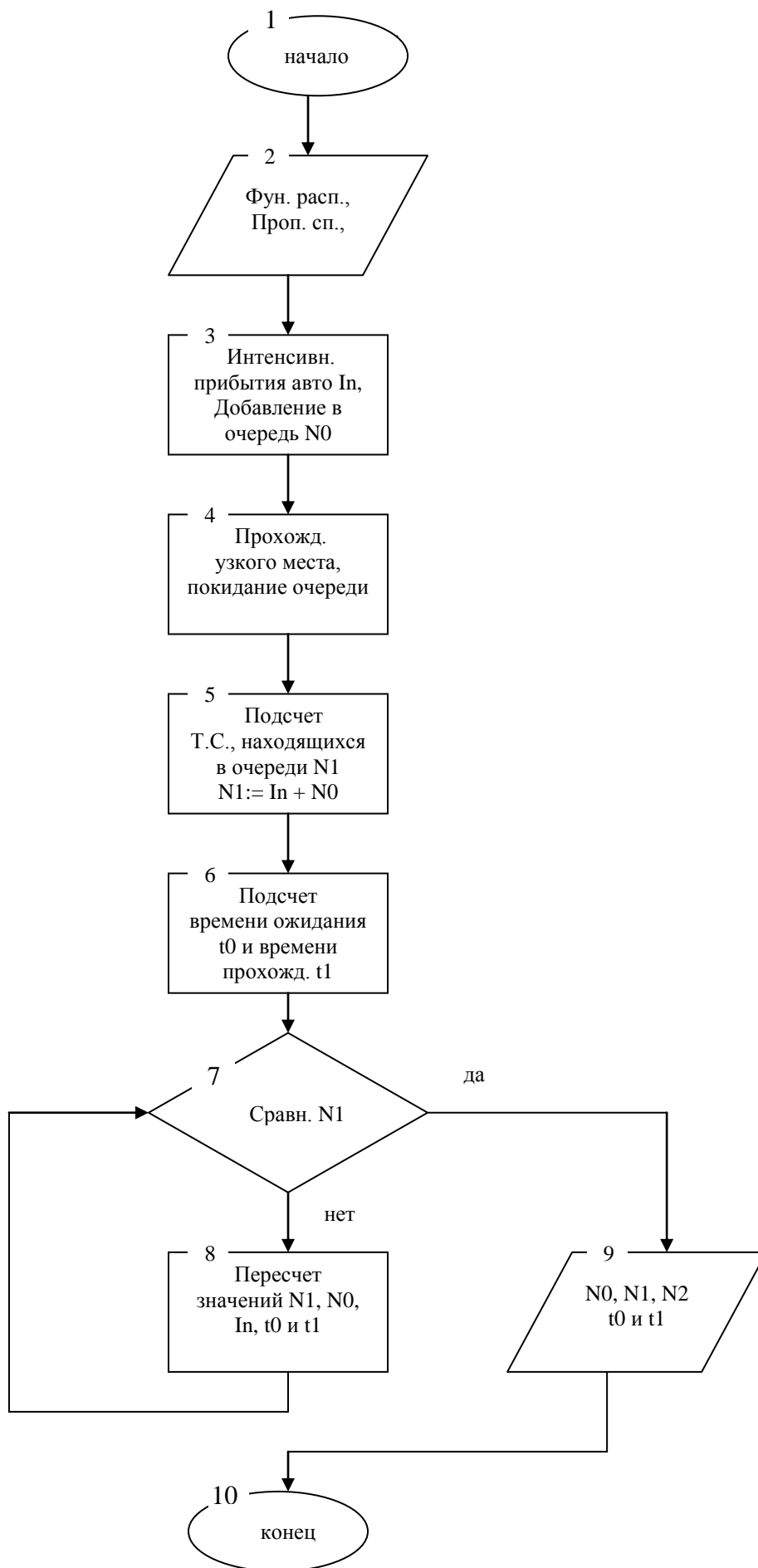


Рис. 4.7. Алгоритм решения задачи «горлышко»

4.3 Особенности проведения и результаты натурального эксперимента в рамках информационной системы управления

Для того, чтобы оценить эффективность разработанных методов, математических моделей и алгоритмической базы, воспользуемся возможностью проведения для этих целей вычислительного эксперимента. В нашем случае натуральный эксперимент будет заключаться в применении методик, описанных в п. 4.2 настоящей работы, для построения информационной системы управления конкретным участком улично-дорожной сети города, содержащим перекрестки, имеющие рассматриваемые топологические модели.

Воронеж является одним из крупнейших городов Черноземья. Население города составляет более 1 млн человек. На состояние транспортного комплекса значительное влияние оказывает интенсивное развитие парка легкового автотранспорта: с 2007 г. он увеличился примерно в 6 раз (рисунок 4.8). В настоящее время его численность составляет более 480 000 единиц. Парк личных автомобилей в обозримом будущем будет увеличиваться в среднем на 30 % каждые 10 лет.

Основной проблемой в рамках информационной системы управления городской транспортной сетью является несоответствие пропускной способности УДС реальному спросу на транспортные услуги. Если в 2005 году, в среднем по городу, были перегружены транспортными потоками (работали на пределе пропускной способности или исчерпали ее) 40 % дорог, то в настоящее время доля дорог, работающих с перегрузкой, достигла 70-77 % (рисунок 4.9).

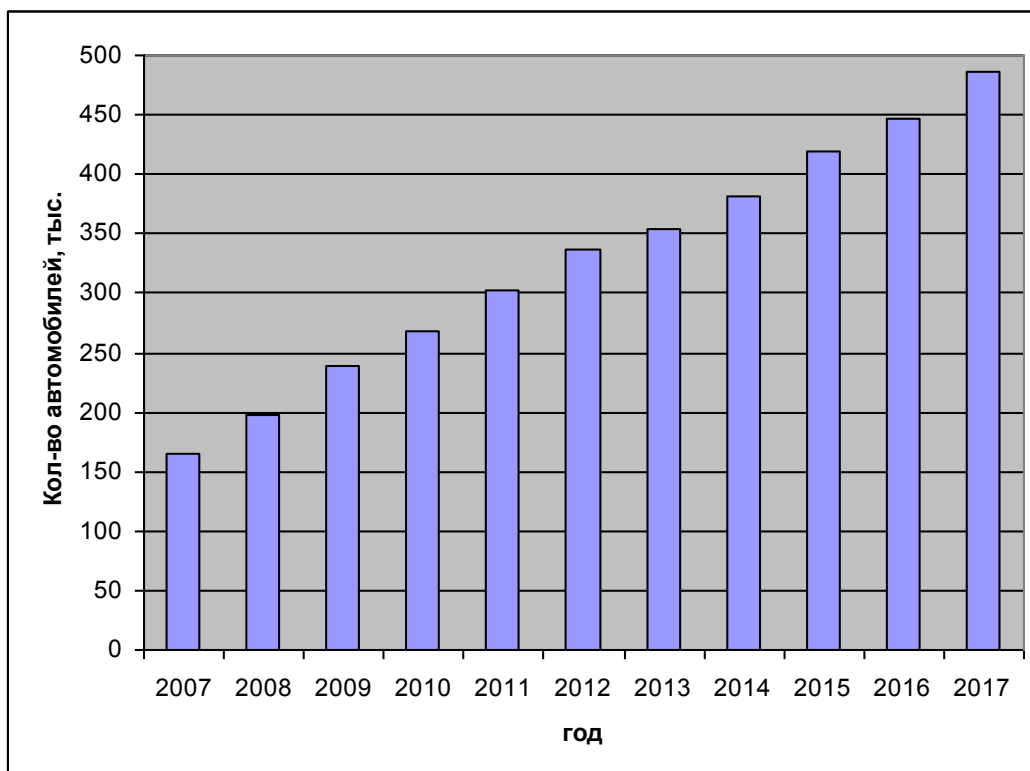


Рис. 4.8. Уровень автомобилизации населения города Воронежа за 2007 – 2017 годы

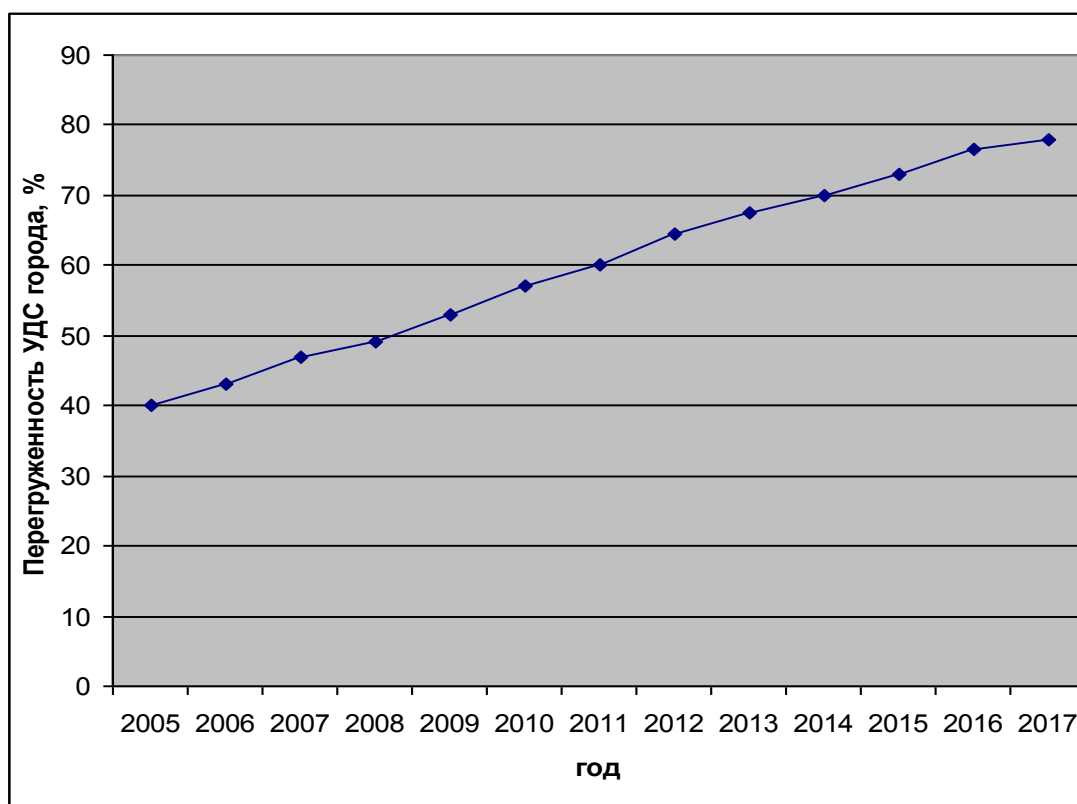


Рис. 4.9. Уровень перегрузки УДС города Воронежа за 2005 – 2017 годы

Причиной заторов на дорогах является поступление потока машин, превышающего их пропускную способность из-за увеличения потока в часы «пик» или снижения пропускной способности дороги. Наибольший процент от общего числа ежедневных заторов и пробок приходится на пересечение дорог и территории, прилегающие к нему.

В качестве объектов будем рассматривать два перекрестка – это перекресток улиц Остужева и Переверткина, имеющий тип «с простым пересечением дорог» и перекресток «Остужевское кольцо», имеющий тип «с круговой организацией движения» [79].

Для эксперимента осуществлен сбор статистических данных об интенсивности движения транспортных средств в каждом из направлений движения (для подбора типа распределения, характеризующего прогнозное значение интенсивности), а также определены технические характеристики, которыми обладает исследуемый объект.

После того, как описанные параметры определены, переходим к построению оптимизационных моделей объектов и заданию начальных условий поиска (с использованием процедур рандомизации оптимизационных переменных). Затем, применив к ним математический аппарат (4.4) – (4.9), а также адаптивные популяционные алгоритмы, описанные в главе 2 данной работы, формируем n вариантов возможного изменения топологии исследуемых перекрестков с целью улучшения эффективности их функционирования (повышения их пропускной способности) на основе разработанной вычислительной среды (рисунок 1.8).

На первом этапе сформированные n вариантов оцениваем через процедуры имитационного моделирования. Для каждого из типов исследуемых объектов алгоритмы процедур представлены на рисунках 4.3, 4.4 (для перекрестка Переверткина - Остужева) и 4.6, 4.7 (для перекрестка Остужевское кольцо) соответственно.

По результатам моделирования получаем зависимость загруженности транспортного участка от времени для каждого из найденных n вариантов структуры объекта.

Для выбора варианта структуры объекта производится сравнение загруженности перекрестка в I - J -ом направлении движения с его проектной пропускной способностью. Полученные результаты представляются визуально в виде сравнительных графиков для перекрестка «Переверткина – Остужева» на рисунке 4.10 и для перекрестка «Остужевское кольцо» на рисунке 4.11.

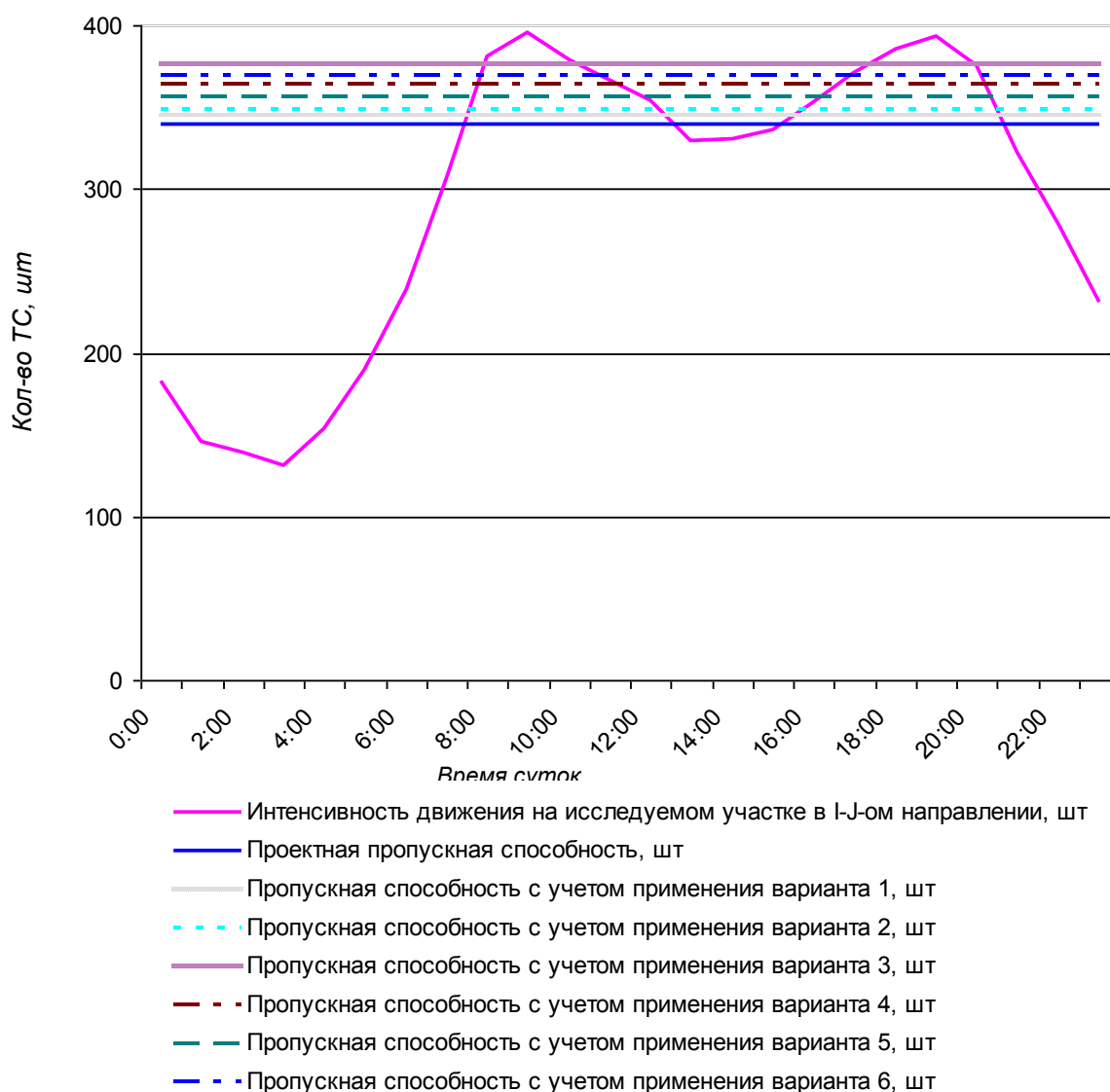


Рис. 4.10. Сравнительные графики оценки пропускной способности перекрестка «Переверткина - Остужева» с его среднесуточной загруженностью

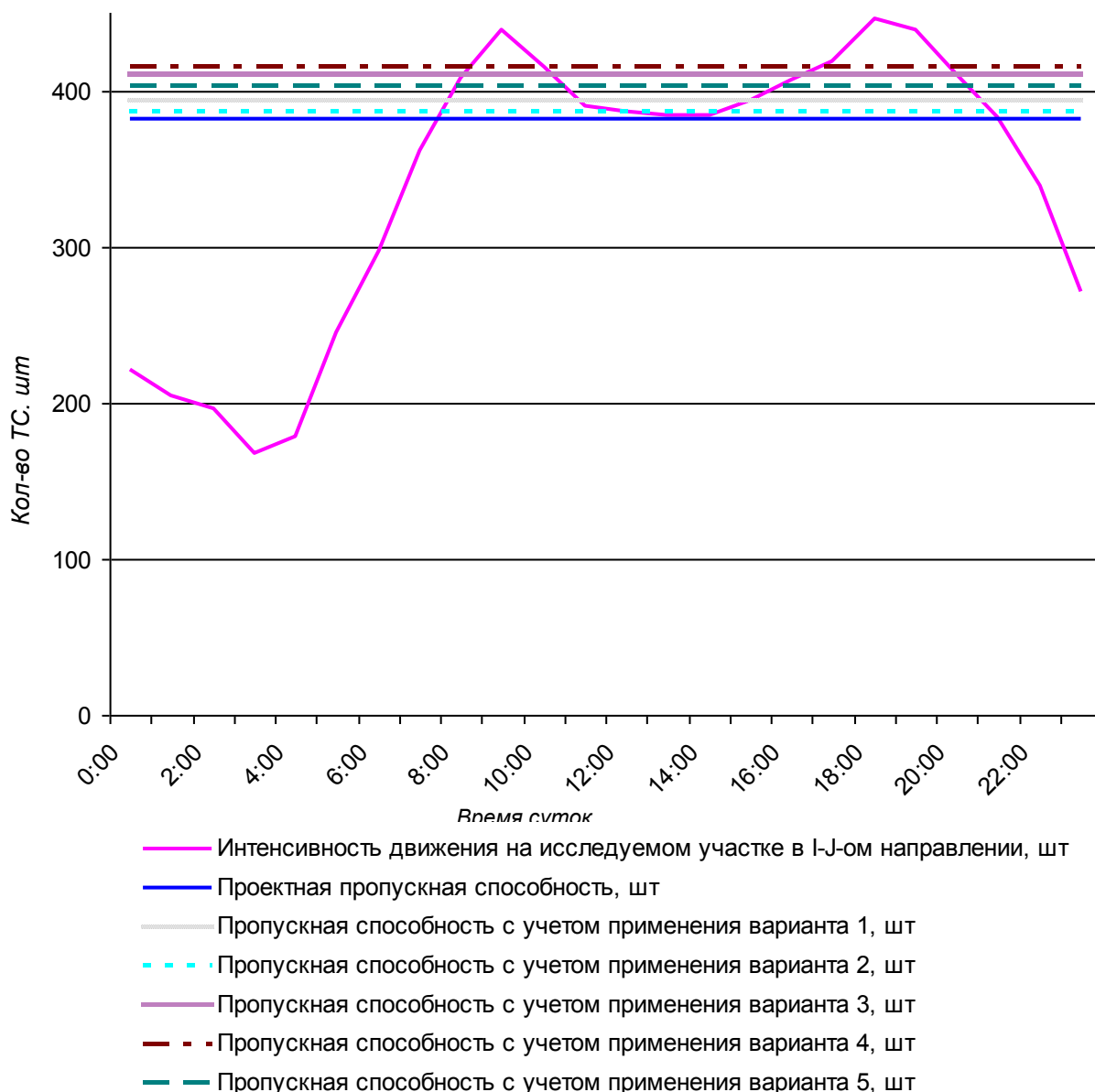


Рис. 4.11. Сравнительные графики пропускной способности перекрестка «Остужевское кольцо» с его среднесуточной загруженностью

Затем, когда сравнительный анализ полученных вариантов структуры объекта по реструктуризации объекта (в нашем случае топологии перекрестка) завершен, переходим к этапу экспертного оценивания.

На этапе экспертного оценивания по результатам сравнительного анализа (их графическому представлению) лицо, принимающее решение, выбирает наилучший вариант с учетом существующих ограничений. В нашем случае для объекта перекресток «Переверткина - Остужева» наилучшим вариантом

является вариант под номером 3, а для перекрестка «Остужевское кольцо» - вариант № 4.

Заметим, что полученные варианты для рассмотренных объектов, не позволяют полностью избавиться от их перегрузок в определенные моменты времени. В частности, для перекрестка «Переверткина - Остужева» это периоды с 8:00 до 11:00 и с 18:00 до 21:00, а для перекрестка «Остужевское кольцо» - с 7:30 до 10:30 и с 17:00 до 21:00. В связи с этим предлагается осуществить решение рассмотренной задачи еще раз, изменив начальные условия. В частности, искать решение не для суточного временного интервала, а только для тех интервалов, когда рассматриваемые перекрестки являются перегруженными, а в качестве начальной пропускной способности для них брать варианты, полученные в ходе решения на предыдущем этапе (для объекта перекресток «Переверткина - Остужева» вариант под номером 3, а для перекрестка «Остужевское кольцо» - вариант № 4). Это позволит еще больше повысить эффективность функционирования рассматриваемых объектов, но, при этом, затратив для решения задачи меньшее количество временных ресурсов, а также позволит получить экономическую выгоду при реализации найденных решений.

Полученные результаты представим в графической форме на рисунках 4.12 и 4.13 соответственно.

После этого на основании полученных результатов определяется наилучший вариант структуры объекта для проблемных временных промежутков и осуществляется для них решение задачи в динамическом режиме.

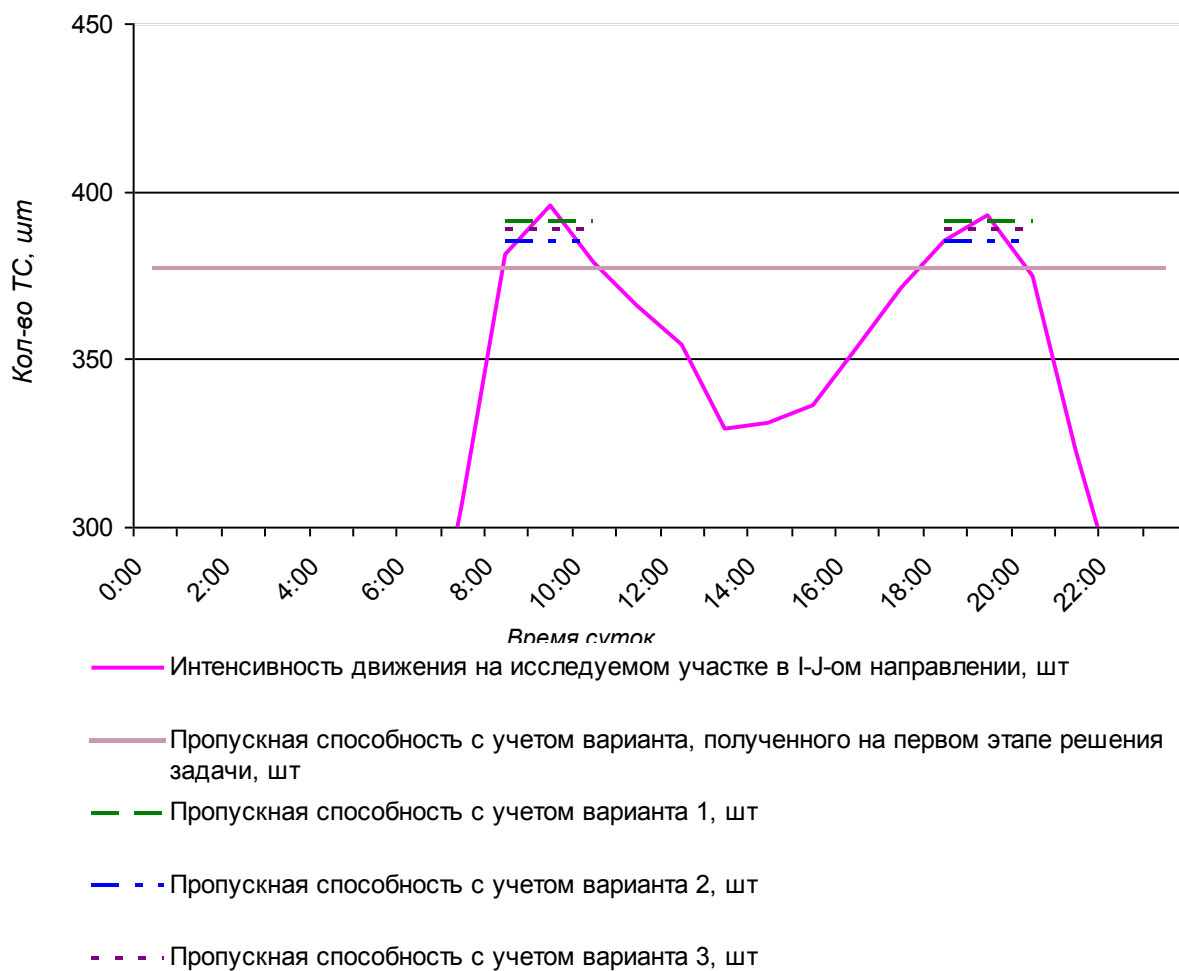


Рис. 4.12. Сравнительные графики пропускной способности перекрестка «Переверткина - Остужева» с его среднесуточной загруженностью с учетом возможности динамического режима поиска решений



Рис. 4.13. Сравнительные графики пропускной способности перекрестка «Остужевское кольцо» с его среднесуточной загруженностью с учетом возможности динамического режима поиска решений

Удалось увеличить пропускную способность в результате применения разработанной вычислительной среды в среднем на 21 %, что доказывает практическую значимость разработанной программной среды на основе нейросетевой и имитационных моделей, а также алгоритмов и процедур численной оптимизации, в процессе решения задачи повышения эффективности функционирования сложноструктурированных объектов.

Выводы четвертой главы:

1. Разработана структура программного комплекса, в основе которого лежит разработанная вычислительная среда, реализующая алгоритмы многоальтернативной оптимизации и адаптивные процедуры обезьяньего поиска, процедуры статистической обработки данных и принципы нейросетевого и имитационного моделирования для повышения функционирования сложноструктурированных объектов.

2. Разработана и описана схема проведения вычислительных экспериментов для оценки эффективности разработанных в главе 2 алгоритмов на основе метода поведения обезьян и описанных в главе 3 принципов структурного моделирования.

3. Представлены схемы и алгоритмы проведения имитационного моделирования для основных топологий сложноструктурированных объектов, а также описаны методики обработки результатов процесса моделирования.

4. На конкретных типах сложноструктурированных объектов рассмотрено проведение натуральных экспериментов, результаты которых доказывают эффективность разработанных в ходе выполнения диссертационной работы математических методов, алгоритмов и моделей, а также их взаимосвязь в рамках единой вычислительной среды.

ЗАКЛЮЧЕНИЕ

Разработка интегрированной среды на базе комплекса процедур и алгоритмов математического моделирования и численного решения экстремальных задач, введение дополнительно к механизмам рандомизации и сглаживания механизмов нейросетевого и имитационного моделирования, синхронизации режимов и итерационной трансформации параметров генераторов псевдослучайных чисел позволяет сформировать интегрированную вычислительную среду для комплексного исследования и оптимизации особого класса сложноструктурированных объектов. Каждый дополнительный механизм дает возможность адекватно отразить в численных процедурах свойства рассматриваемых в работе объектов, связанные с особенностями формирования математического описания их связей с показателями эффективности экстремальных и граничных требований, характеризующих процесс вариации структурой объекта в статическом и динамическом режимах.

Выполнение поставленной цели и задач диссертации характеризуется следующими основными результатами:

1. На основе формулировки понятия структурного моделирования и определения основных свойств класса сложноструктурированных объектов разработан способ их формализованного описания процесса анализа сложноструктурированных объектов, основой которого являются предложенные в работе имитационные и нейросетевая модели, используемые для оценки альтернативных структур исследуемого объекта.

2. Разработан комплекс имитационных моделей анализа вариантов структуры объектов исследуемого класса, позволяющий учитывать особенности конкретного объекта моделирования и ориентированных на поддержку вычислительных схем численной оптимизации.

3. Сформирован комплекс моделей оптимизации сложноструктурированных объектов, отражающих разнообразие переменных, экстремальных и граничных требований к их функционированию, за счет

варьирования которыми обеспечивается выбор адекватного варианта модели для исследуемого объекта.

4. Построены трансформационные процедуры изменения параметров генераторов дискретных и непрерывных псевдослучайных чисел в процессе итерационного формирования множества перспективных вариантов сложноструктурированных объектов.

5. Разработана структура вычислительной среды для сложноструктурированных объектов, основанная на интеграции проблемно-ориентированных процедур нейросетевого и имитационного моделирования, численной оптимизации и экспертного оценивания.

6. С использованием теоретико-информационных оценок разработана математическая модель синхронизации режимов нейросетевого и имитационного моделирования и численной оптимизации, исходя из которых получены условия останова основных процессов интегральной вычислительной среды.

7. Разработана структура интегрированной программной среды численной оптимизации сложноструктурированных объектов, а также средства ее интеграции в промышленные программные системы, эффективность применения которых подтверждена результатами проведения вычислительных и натуральных экспериментов для сложноструктурированных объектов и актами внедрения.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Авен О.И. Управление вычислительным процессом в ЭВМ: (Алгоритмы и модели). / О.Н. Авен , Я.А. Коган. М: Энергия, 1978. - 240 с.
2. Авен О.Н. Оценка качества и оптимизация вычислительных систем. / О.Н. Авен , Н.Н. Гурин, Я.А. Коган - М.: Наука, 1982. - 464 с.
3. Алексеев О.Г. Комплексное применение методов дискретной оптимизации. / О.Г. Алексеев. – М: Наука, 1987. – 279 с.
4. Алиев Т.И. Основы моделирования дискретных систем. Учебное пособие. – СПб.: СПбГУ ИТМО, 2009. - 363 с.
5. Андриющенко М.Н. Философские основы моделирования сложных систем управления / М.Н. Андриющенко, Б.Я. Советов, С.А. Яковлев и др.// Системный подход в технических науках (Методологические основы): Сб. научн. тр. — Л.: Изд АН СССР, 1989.
6. Аттетков А.В. Введение в методы оптимизации / А.В. Аттетков, В.С. Зарубин, А.Н. Канатников. - М.: Финансы и статистика, 2008. - 272 с.
7. Батищев Д.И. Применение генетических алгоритмов для решения задач дискретной оптимизации / Д.И. Батищев, Е.А. Неймарк, Н.В. Старостин // Нижегородский гос. унт-т им. Н.И. Лобачевского. – Нижний Новгород: Изд-во ННГУ им. Н.И. Лобачевского, 2007. – 85 с.
8. Батищев Д.И. Оптимизация в САПР : учебник для ВУЗов / Д.И. Батищев, Я.Е. Львович, В.Н. Фролов – Воронеж: Изд-во Воронеж. гос. ун-та, 1997. – 415 с.
9. Белецкая С.Ю. Адаптивные методы оптимального проектирования / С.Ю. Белецкая, Я.Е. Львович // Информационные технологии: приложение : М.: Новые технологии. – 2010. № 7.
10. Бонгард М.М. О понятии "полезная информация"/ М.М. Бонгард // Проблемы кибернетики. Вып.9. М.: Физматиз., 1963.
11. Бугров Д.А. Постановка задачи структурной оптимизации магистральной корпоративной телекоммуникационной сети // Информация и Космос. — 2005. - № 2. -С. 42-47.

12. Вазан М. Вазан М. Стохастическая аппроксимация: пер. с англ. / М. Вазан. - М.: Мир, 1972. - 296 с.
13. Васильев Ф.П. Методы оптимизации в 2-х книгах. Кн.1 / Ф.П. Васильев. - М.: МЦНМО, 2011. - 619 с.
14. Васильев Ф.П. Методы оптимизации в 2-х книгах. Кн.2 / Ф.П. Васильев. - М.: МЦНМО, 2011. - 433 с.
15. Вентцель Е.С. Теория вероятностей: Учеб. для вузов / Е.С. Вентцель – 6-е изд. стер. – М.: Высш. шк., 1999.— 576 с.
16. Галушкин А.И. Современные направления развития нейрокомпьютерных технологий в России. / Под ред. А.И. Галушкина. М.: Изд-во ИПРЖР, 2000. - 416 с.
17. Галушкин А.И. Теория нейронных сетей. Кн. 1: Учеб. пособия для вузов. - М.: ИПРЖР, 2000. - 416 с.
18. Гладков Л.А. Генетические алгоритмы. / Л.А. Гладков, В.М. Курейчик, В.В. Курейчик. Ростов-на-Дону: ООО «Ростиздат», - 2004.
19. Гончаров В.А. Методы оптимизации / В.А. Гончаров. – М: Высшее образование, 2009. – 191 с.
20. Горбань А. Н. Нейронные сети на персональном компьютере [Текст] / А.Н. Горбань, Д. А. Россиев. – Новосибирск : Наука, 1996. – 276 с.
21. Горбунов В.В. Эффективность использования мультипрограммной организации вычислительных процессов. / В.В. Горбунов. Журнал «Научный обозреватель». – Уфа: из-во «Инфинити», №8, 2012. – С.40 – 42.
22. Девятков В.В. Практическое применение имитационного моделирования в России и странах СНГ: обзор, анализ перспектив. // «Имитационное моделирование. Теория и практика. (ИММОД 2005)», Том II. СПб.: ФГУП ЦНИИ технологии судостроения, 2005. – С. 15 – 18.
23. Демуськов А.Б. Организация мониторинга параметров вычислительного процесса в комплексах и сетях ЭВМ при натуральных и имитационных экспериментах: Автореферат дис. канд. техн. наук: 05.13.13. – Гомель, 1999. – 22 с.

24. Дли М.И. Конфигурирование информационных и транспортных сетей в условиях неопределенности / М.И. Дли , В.В. Гимаров, С.И. Глушко // Прикладная информатика. 2012. № 6(42). С. 81-86.

25. Живоглядов В.П. Построение альтернативной теории дуального управления / Вестник КРСУ, 2012. Том 12. № 10. С. 121–124.

26. Зайцев М.Г. Методы оптимизации управления и принятия решений: примеры, задачи, кейсы / М.Г. Зайцев, С.Е. Варюхин. - М.: Дело АНХ, 2015. - 640 с.

27. Зубарев Ю.М. Специальные методы оптимизации: Учебное пособие / Ю.М. Зубарев. - СПб.: Лань, 2014. - 384 с.

28. Ивашенцев А.А. Вычислительная среда в интеллектуальных системах. Проектирование. Проблемы. / А.А.Ивашенцев. // Вестник Российского университета дружбы народов. Серия: Инженерные исследования. – М.: РУДН, № 4, 2008. – С. 61 – 67.

29. Измаилов А.Ф. Численные методы оптимизации / А.Ф. Измаилов, М.В. Солодов. - М.: Физматлит, 2008. - 320 с.

30. Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой : учебное пособие / А. П. Карпенко. — М.: Издательство МГТУ им. Н. Э. Баумана, 2014. — 446 с.

31. Карпенко А.П. Популяционные алгоритмы оптимизации. Обзор новых и малоизвестных алгоритмов / А.П. Карпенко // Приложение к журналу «Информационные технологии». – М.: из-во «Новые технологии», 2012. - №7. – 30 с.

32. Карпенко А.П. Нейросетевая аппроксимация функции предпочтений лица, принимающего решения в задаче многокритериальной оптимизации / А.П. Карпенко, Д.Т. Мухлисуллина, В.А. Овчинников // Информационные технологии, 2010. №10. - С. 2-9.

33. Карпухин И.Н. Системы имитационного моделирования учебного назначения в российской высшей школе / И.Н. Карпухин , А.А. Незнанов //

Современные проблемы информационной безопасности и программной инженерии, 2011. – С. 132-142.

34. Колесов Ю.Б. Объектно-ориентированное моделирование сложных динамических систем. СПб., Россия : СПбГПУ, 2004. - 240 с.

35. Комашинский В.И. Нейронные сети и их применение в системах управления и связи. / В.И. Комашинский , Д.А. Смирнов М.: Горячая линия-Телеком, 2003. - 94 с.

36. Кочегурова Е.А. Теория и методы оптимизации.: Учебное пособие для академического бакалавриата / Е.А. Кочегурова. - Люберцы: Юрайт, 2016. - 133 с.

37. Кочерга М.С. Построение реконфигурируемых вычислительных систем на однородных вычислительных средах. / М.С. Кочерга, В.И. Шмойлов // Вестник Южного научного центра РАН. – М.: Наука, Т.4, № 2, 2008. – С. 8 – 26.

38. Кривулин Н.К. Оптимизация сложных систем при имитационном моделировании // Вестник Ленингр. Ун-та, 1990. № 8.

39. Круглов В.В. Искусственные нейронные сети. Теория и практика., 2nd изд. / В.В. Круглов , В.В. Борисов. - М.: Телеком, 2002. - 382 с.

40. Кузнецов А.С. Теория вычислительных процессов: учебник. / А.С. Кузнецов, Р.Ю. Царев, А.Н. Князьков. – Красноярск: Сиб. Федер. Ун-т, 2015. – 184 с.

41. Лоу А.М. Имитационное моделирование. В серии «Классика Computer sciences» / А.М. Лоу, В.Д. Кельтон. – СПб.: Питер, 2004. – 846 с.

42. Лычкина Н.Н. Имитационное моделирование экономических процессов /Н.Н. Лычкина. М.: Академия АйТи, 2005. - 164 с.

43. Львович И.Я. Вариационное моделирование и оптимизация проектных решений. / И.Я. Львович. – Воронеж: изд-во ВГТУ, 1997. – 114 с.

44. Львович И.Я. Имитационное моделирование процесса выбора структуры и параметров технологических систем / И.Я. Львович, В.Н. Фролов

// Математическое моделирование: Тез. докл. Всесоюз. конф. Воронеж, 1992. С. 61.

45. Львович И.Я. Информационные технологии моделирования и оптимизации: монография / И.Я. Львович, Я.Е. Львович, В.Н. Фролов. – Воронеж, издательско-полиграфический центр «Научная книга», 2016. – 443с.

46. Львович И.Я. Структурно-параметрическая оптимизация технологических систем на основе имитационных моделей / И.Я. Львович, В.Н. Фролов // Оптимизация и моделирование в автоматизированных системах: Сб. науч. тр. Воронеж: ВПИ, 1991. С. 100-104.

47. Львович И.Я. Вариационные процедуры рационального выбора на основе оптимизационных моделей / И.Я.Львович, Э.М. Львович, В.Н. Фролов // Высокие технологии в технике, медицине и образовании: Межвуз. сб. науч. тр. ч. 1. Воронеж: ВГТУ, 1996. С. 135-139.

48. Львович Я.Е. Принятие решений в экспертно-виртуальной среде: монография / Я.Е. Львович. – Воронеж: ИПЦ «Научная книга», 2010. – 140 с.

49. Львович Я.Е. Многоальтернативная оптимизация: теория и приложения / Я. Е. Львович. - Воронеж : Кварта, 2006. – 415 с.

50. Львович Я.Е. Алгоритмизация слабоформализованных задач оптимального выбора на основе рандомизированных процедур поискового типа / Я.Е. Львович, С.Ю. Белецкая // Информационные технологии: М.: Новые технологии. – 2004. № 11.

51. Любич Ю.И. Общая теория релаксационных процессов для выпуклых функционалов / Ю.И. Любич, Г.Д. Майстровский // Успехи математических наук. - № 1, 1970. – С. 57-112.

52. Максимей И.В. Имитационное моделирование на ЭВМ. - М.: Радио и связь, 1998. – 230 с.

53. Минаков И.А. Сравнительный анализ некоторых методов случайного поиска и оптимизации / И.А.Минаков. // Известия Самарского научного центра Российской академии наук, № 2, 1999. — С. 286-293.

54. Молчанов А.А. Моделирование и проектирование сложных систем. - Киев: Вища. шк., 1988.
55. Новосельцев В.И. Системный анализ: современные концепции / В.И. Новосельцев. – Воронеж: Кварта, 2003. – 360 с.
56. Норенков И.П. Основы автоматизированного проектирования. М.: Изд-во МГТУ им. Н. Э. Баумана, 2002. 336 с.
57. Петров П.В. Технология вычислительного эксперимента / П.В. Петров, Р.А. Сунарчин, В.А. Целищев // Вестник УГАТУ. – Уфа: УГАТУ, Т.10, № 1 (26), 2008. – С. 30-35.
58. Пшеничный Б.Н. Численные методы в экстремальных задачах / Б.Н. Пшеничный, Ю.М. Данилин // М.: Наука, 1975. - 320 с.
59. Пятаев О.В. Применение генетического алгоритма для оптимизации структуры кампусной сети // Радиоэлектронные и телекоммуникационные системы и устройства: Межвузовский тематический сборник научных трудов, 2000. С. 55-61.
60. Растрингин Л.А. Адаптация сложных систем. Рига: Зинатне, 1981. — 375 с.
61. Рутковская Д. Нейронные сети, генетические алгоритмы и нечеткие системы [Текст] : Пер. с польск. / Д. Рутковская, М. Пилиньский, Л. Рутковский // М. : Горячая линия – Телеком, 2006. – 452 с.
62. Рыков А.С. Методы системного анализа: многокритериальная и нечеткая оптимизация, моделирование и экспертные оценки / А.С. Рыков. – М.: Экономика, 1999. – 192 с.
63. Рыков А.С. Методы системного анализа: оптимизация / А.С. Рыков. – М.: Экономика, 1999. – 256 с.
64. Рыжиков Ю.И. Имитационное моделирование. Теория и технология. / Ю.И. Рыжиков. - М.: Альтекс-А, 2004. - 384 с.
65. Рыжиков Ю. И. Имитационное моделирование систем массового обслуживания. Л. : ВИККИ им. А.Ф. Можайского, 1991. - 111 с.

66. Самарский А.А. Математическое моделирование: Идеи. Методы. Примеры. / А.А. Самарский, А.П. Михайлов. - М.: ФИЗМАТЛИТ, 2002. - 320 с.

67. Семенкин Е.С. Эволюционные методы моделирования и оптимизации сложных систем / Е.С. Семенкин, М.Н. Жукова, В.Г. Жуков, И.А. Панфилов, В.В. Тынченко. – Красноярск: Сибирский федеральный университет, 2007. – 198 с.

68. Сергиенко И.В. Математические модели и методы решения задач дискретной оптимизации. - Киев: Наук, дум., 1988. - 472 с.

69. Сидоренко В.Н. Имитационное моделирование в науке и бизнесе: подходы, инструменты, применение / В.Н. Сидоренко, А.В. Красносельский // Журнал «Бизнес-информатика», №2 (08), 2009. – С. 52-57.

70. Соболев И.М. Численные методы Монте-Карло / И.М. Соболев. М.: ФИЗМАТЛИТ, 1973. - 312 с.

71. Советов Б.Я. Базы данных. Теория и практика / Б.Я. Советов, В.В. Цехановский, В.Д. Чертовской // Учебник для ВУЗов. – М.: Юрайт, 2014. – 464 с.

72. Советов Б.Я. Моделирование систем (3-е изд.). / Б.Я. Советов, С.А. Яковлев – М.: Высшая школа, 2005.

73. Советов Б.Я. Теория информации / Б.Я. Советов. — Л.: Изд. ЛГУ, 1977. – 184 с.

74. Строгалев В.П. Имитационное моделирование. / В.П. Строгалев, И.О. Толкачева. - М.: МГТУ им. Баумана, 2008.

75. Тишуков Б.Н. Адаптивный подход к алгоритму обезьяньего поиска для решения задач оптимизации объектов со структурновариативной формой управления в динамическом режиме / Б.Н. Тишуков, Я.Е. Львович // Журнал «Известия Юго-Западного государственного университета». – Курск, Изд-во ЮЗГУ, № 5 (68), 2016. – С. 51-57.

76. Тишуков Б.Н. Разработка процедур численной оптимизации объектов со структурно-вариативным управлением на основе использования популяционных алгоритмов / Б.Н. Тишуков, Я.Е. Львович // Вестник

Воронежского государственного технического университета. – Воронеж, ФГБОУ ВО «ВГТУ», т.12 № 3, 2016. - С.12-16.

77. Тишуков Б.Н. Разработка математического обеспечения для синхронизации режимов поисковой оптимизации и дуального моделирования в интегрированной вычислительной среде / Б.Н. Тишуков, Я.Е. Львович // Вестник Воронежского государственного университета. Серия: Системный анализ и информационные технологии: - Воронеж, ФГБОУ ВО «ВГУ», № 2, 2017. - С.28 – 32.

78. Тишуков Б.Н. Разработка имитационной модели для проверки эффективности структуры перекрестка в рамках проведения вычислительного эксперимента / Я.Е. Львович, Б.Н. Тишуков, Э.И. Воробьев // Виртуальное моделирование, прототипирование и промышленный дизайн. Материалы III Международной научно-практической конференции.: - Тамбов, ФГБОУ ВО «ТГТУ», 2016.- с.30-35.

79. Тишуков Б.Н. Управление пропускной способностью перекрёстка на основе имитационного моделирования в информационной системе транспортной сети / Б.Н. Тишуков, Я.Е. Львович, Э.И. Воробьев // Виртуальное моделирование, прототипирование и промышленный дизайн. Материалы IV Международной научно-практической конференции: - Тамбов, ФГБОУ ВО «ТГТУ», 2017.- Т.1. с.169-175.

80. Тишуков Б.Н. Формализация задачи структурной оптимизации участка УДС в виде многоальтернативной оптимизационной модели / Б.Н. Тишуков, Я.Е. Львович // Виртуальное моделирование, прототипирование и промышленный дизайн. Материалы II Международной научно-практической конференции.: - Тамбов, ФГБОУ ВПО «ТГТУ», 2016.- с.409-411.

81. Тишуков Б.Н. Повышение эффективности функционирования объектов со структурновариативной формой управления на основе оптимизационного моделирования / Б.Н. Тишуков // Моделирование, оптимизация и информационные технологии. – 2018. – Т.6 – № 1 – URL: https://moit.vivt.ru/wp-content/uploads/2018/01/Tishukov_1_1_18.pdf.

82. Тишуков Б.Н. Знаниевые компоненты в управлении информационной системой транспортной сети / Я.Е. Львович, Б.Н. Тишуков, Д.В. Иванов, Э.И. Воробьев // Интеллектуальные информационные системы. Труды Всероссийской конференции с международным участием: - Воронеж, ФГБОУ ВО «ВГТУ», 2017.- Ч1. с.48-50.

83. Фельдбаум А.А. Основы теории оптимальных автоматических систем. - М.: Наука, 1966. - 623 с.

84. Феррари Д. Оценка производительности вычислительных систем. М.: Мир, 1981. - 576 с.

85. Хайкин С. Нейронные сети. Полный курс. М.: ООО И.Д.Вильямс, 2008. - 1104 с.

86. Хартман К. Планирование эксперимента в исследованиях технологических процессов / К. Хартман, Э. Лецкий, В. Шефер; пер. с нем. Г.Л. Фомина, Н.с. Лецкой. – М.: Мир, 1977. – 552 с.

87. Цыпкин Я.З. Адаптация и обучение в автоматических системах / Я.З. Цыпкин. – М.: Наука, 1968. – 400 с.

88. Черноруцкий И.Г. Методы оптимизации в теории управления / И. Г. Черноруцкий. – СПб.: «Питер Принт», 2004. – 361с.

89. Черноруцкий И.Г. Методы оптимизации. Компьютерные технологии / И. Г. Черноруцкий. - СПб.: ВHV, 2011. - 384 с.

90. Черноруцкий И.Г. Методы принятия решений. СПб.: БХВ-Петербург, 2005. - 416с.

91. Шеннон К. Работы по теории информации и кибернетике / К. Шеннон; пер. с англ. под ред. Р.Л. Добрашина и О.Б. Лупанова. - М.: Иностранная литература, 1963. — 832 с.

92. Экспертные системы. Принципы работы и примеры / А. Бруклич [и др.]; пер. с англ. С.И. Рудаковой; под ред. Р. Форсайта. – М.: Радио и связь, 1987.

93. Юдин Д.Б. Алгоритмы обучения нейронной сети (алгоритмы пополнения знаний) // Автоматика и телемеханика, 1996. № 11. С. 148 - 158.

94. Яковлев С. А. Эволюционные имитационные модели процессов и систем как методологическая основа интеллектуальных технологий обучения // Тез. докл. Междунар. конф. «Современные технологии обучения». — СПб., 1996.
95. Altiok T., Melamed B. Simulation modeling and analysis with ARENA. Burlington, USA : Elsevier Inc., 2007. - 440 с.
96. Baker J. Adaptive selection methods for genetic algorithms. Proc. International Conf. on Genetic Algorithms and Their Applications. J. Grefenstette, ed. Lawrence Erlbaum, 1985.
97. Bentley P.J., Wakefield J.P. Finding Acceptable Solutions in the Pareto-Optimal Range using Multiobjective Genetic Algorithms. In Proceedings of the 2nd On-Line World Conference on Soft Computing in Engineering Design and Manufacturing, 1997.
98. Cakaj Sh., Zeliko Debeljuh. Modeling, Simulation and Optimization. Focus on Applications. Vikovar, Croatia : In-Teh, 2010. - 312 с.
99. Goldberg D.E. Genetic Algorithms in Search, Optimization, and Machine Learning. Reading, Massachusetts: Addison-Wesley, 1989.
100. Kalyan Deb. A fast elitist multi-objective genetic algorithm: NSGA-II / K. Deb et al. // IEEE Transaction on Evolutionary Computation. 2002. Vol. 6, no. 2. P. 182-197.
101. Michalewicz Z. Genetic algorithms, numerical optimization and constraints // Proc. of the Sixth Int. Conf. on Genetic Algorithms and their Applications, Pittsburgh, PA, 1995.
102. Perros H. Computer Simulation Techniques: The definitive introduction! USA, 2009.- 177 с.
103. Sokolowski J.A., Banks C.M. Modeling and simulation fundamentals. Theoretical Underpinnings and Practical Domains. New Jersey, USA : A John Wiley & Sons, 2010.- 437 с.
104. Wainer G.A., Mosterman P.J. Discrete-event modeling and simulation. New York, USA : CRC Press, 2011. - 493 с.


ПРИЛОЖЕНИЕ 1

Акт внедрения результатов диссертационного исследования в производство

СОГЛАСОВАНО

Проректор по научной работе
ФГБОУ ВО «ВГТУ» И.Г. Прозлов
«20» _____ 2018 г.

УТВЕРЖДАЮ

Руководитель Департамента
строительной политики Воронежской
области О.Ю. Гречишников
«30» _____ 2018 г.

А К Т

о внедрении результатов диссертационной работы
в производствоМы, представители Департамента строительной политики Воронежской области
(наименование предприятия)настоящим актом подтверждаем, что результаты диссертационной
работы Тишукова Бориса Николаевича «Структурное моделирование
(Ф.И.О.) (наименование работы)сложных объектов на основе интегрированной программной среды численной
оптимизации», выполненной в Воронежском государственном техническом
университете, в рамках основного научного направления «Интеллектуальные
информационные системы» по г/б (х/д) НИР 16.19 АААА-А16-116031710013-4
(№ темы, № госрегистрации)в период с 01.10.2014г. по 11.12.2018г.приняты к внедрению _____ 20.12.2018 г.

(приказ руководителя предприятия, распоряжение, поз. плана внедрения и др., дата, номер)

и внедрены в автоматизированную систему проектирования дорожной сети
(цех, отдел и т. д., дата внедрения)города Воронежа1. Вид внедрения результатов _____ методика
(комплекс, машина, система, прибор, инструмент, технология,

методика, зарегистрированные программы для ЭВМ, базы данных, и т.д.)

2. Область и форма внедрения _____ единичное производство
(производственный процесс – серийное, уникальное или единичное

производство; проектные разработки; научные исследования и т.п.)

3. Технический уровень НИР свид. о рег. программы для ЭВМ № 2018618331 от
(поданы заявки на объекты промышленной собственности, получены11.07.2018г., результаты разработки отмечены дипломами научных конкурсов,
положительные решения, патенты России, дипломы, медали и др. их № и дата)выставок и конференций, где были отмечены дипломами победителя и
призера.

4. Публикации по материалам НИР по результатам диссертационного исследования опубликовано 17 печатных работ, 4 из которых в журналах перечня ВАК РФ; имеется свидетельство о регистрации программы для ЭВМ.
(количество, где опубликованы)


5. Эффект от внедрения (ожидаемый, фактический)

а) социальный выражен в повышении эффективности функционирования дорожно-транспортной сети города Воронежа; формирование эффективных проектных решений при организации дорожной сети города и при необходимости реорганизации проблемных участков городской дорожной сети.
(улучшение условий труда, оздоровление окружающей среды и др.)

б) годовой экономической эффект _____ тыс. руб.
(прописью)

От Воронежского государственного
технического университета:

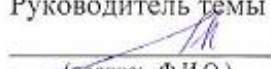
Управление науки и инноваций


_____ Коновалов Д.А.
(подпись, Ф.И.О.)
« 20 » 12 _____ 2018 г.

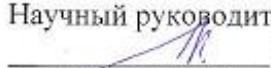
От Департамента строительной
политики Воронежской
области:


_____ (подпись, Ф.И.О.)
« 20 » 12 _____ 2018 г.


Руководитель темы


_____ Львович Я.Е.
(подпись, Ф.И.О.)
« 20 » 12 _____ 2018 г.

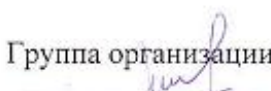
Научный руководитель диссертанта


_____ Львович Я.Е.
(подпись, Ф.И.О.)
« 20 » 12 _____ 2018 г.

Диссертант


_____ Тишуков Б.Н.
(подпись, Ф.И.О.)
« 20 » 12 _____ 2018 г.

Группа организации и планирования НИР


_____ Жилова О.В.
(подпись, Ф.И.О.)
« 20 » 12 _____ 2018 г.

ПРИЛОЖЕНИЕ 2

Акт внедрения результатов диссертационного исследования в учебный процесс
ФГБОУ ВО «Воронежский государственный технический университет»

УТВЕРЖДАЮ

Проректор по учебной работе
ФГБОУ ВО «ВГТУ»

А.И. Колосов

« 29 » 08 2018 г.

АКТ

внедрения результатов кандидатской диссертации
в учебный процесс

ФГБОУ ВО «Воронежский государственный технический университет»

Наименование диссертации: «Структурное моделирование сложных объектов на основе интегрированной программной среды численной оптимизации»

Автор: Тишуков Борис Николаевич

Научный руководитель: Львович Яков Евсеевич

Выполненной в ФГБОУ ВО «Воронежский государственный технический университет» на кафедре систем автоматизированного проектирования и информационных систем в рамках основного научного направления «Интеллектуальные информационные системы» по г/б (х/д) НИР 16.19 АААА-А16-116031710013-4

В период с 01.09.2018г. по н.в. внедрены в учебный процесс кафедры по направлениям подготовки 09.03.01 «Информатика и вычислительная техника» и 09.03.02 «Информационные системы и технологии» на основании решения кафедры САПРИС от «29» 08 2018 г., протокол № 1.

1. Вид результатов, внедренных в учебный процесс: совокупность знаний и представлений по теме диссертационного исследования.

2. Область применения: лабораторный практикум и лекционный курс по дисциплинам «Оптимизация в САПР», «Методы оптимизации в ИС», «Моделирование процессов и систем», «Моделирование систем», а также при выполнении курсовых и дипломных работ.

3. Форма внедрения: описание разработанных в диссертационном исследовании методов и адаптивных алгоритмов оптимизации сложноструктурированных объектов, модели для проведения имитационного моделирования и алгоритмы повышения функционирования объектов на основе результатов оптимизационного и имитационного моделирования, материалы лекционных и лабораторных курсов.

4. Эффект от внедрения: повышение качества образования: применение новых технологий в области моделирования и оптимизации сложноструктурированных объектов, а также методик, позволяющих повысить эффективность их функционирования.

Научный руководитель диссертанта

Я.Е. Львович
(подпись, Ф.И.О.)
« 29 » 08 2018 г.

Диссертант

Б.Н. Тишуков
(подпись, Ф.И.О.)
« 29 » 08 2018 г.

Начальник УМУ ФГБОУ ВО «ВГТУ»

Л.П. Мышовская
(подпись, Ф.И.О.)
« 30 » 08 2018 г.

Декан ФИТКБ

С.М. Пасмурнов
(подпись, Ф.И.О.)
« 30 » 08 2018 г.

Зав. кафедрой САПРИС

Я.Е. Львович
(подпись, Ф.И.О.)
« 29 » 08 2018 г.

ПРИЛОЖЕНИЕ 3

Свидетельство о регистрации программы для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2018618331

**Программная реализация оптимизационных алгоритмов
обезьяньего поиска**

Правообладатель: *Федеральное государственное бюджетное
образовательное учреждение высшего образования «Воронежский
государственный технический университет» (RU)*

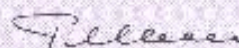
Авторы: *Тишуков Борис Николаевич (RU), Цицилин Денис
Леопидович (RU), Львович Яков Евсеевич (RU)*

Заявка № 2018615517

Дата поступления 30 мая 2018 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 11 июля 2018 г.

Руководитель Федеральной службы
по интеллектуальной собственности

 Г.Н. Ивлёв

