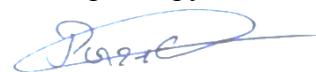


ФГБОУ ВО «Сибирский государственный аэрокосмический университет
имени академика М.Ф. Решетнёва»,

На правах рукописи



РЫЖИКОВ ИВАН СЕРГЕЕВИЧ

ЭВОЛЮЦИОННЫЕ АЛГОРИТМЫ РЕШЕНИЯ ЗАДАЧ ИДЕНТИФИКАЦИИ И УПРАВЛЕНИЯ ДЛЯ ДИНАМИЧЕСКИХ СИСТЕМ

Специальность 05.13.01

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

Диссертация на соискания ученой степени кандидата технических наук

Научный руководитель:
доктор технических наук, профессор
Е.С. Семенкин

Красноярск – 2016

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
ГЛАВА I. ЭВОЛЮЦИОННЫЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧ ПОИСКА ЭКСТРЕМУМА.....	11
1. 1. Алгоритм эволюционных стратегий.....	12
1. 2. Модификация схемы и операторов поиска алгоритма эволюционных стратегий.....	18
1. 3. Обобщение алгоритма эволюционных стратегий. Мета-эвристика управления перезапусками алгоритма	29
1. 4. Другие эвристические алгоритмы стохастического поиска экстремума.	35
Выводы	40
ГЛАВА II. РЕШЕНИЕ ЗАДАЧИ ИДЕНТИФИКАЦИИ ЛИНЕЙНЫХ ДИНАМИЧЕСКИХ СИСТЕМ С ПОМОЩЬЮ МОДИФИЦИРОВАННОГО АЛГОРИТМА ЭВОЛЮЦИОННЫХ СТРАТЕГИЙ.....	44
2. 1. Задача идентификации линейных динамических систем: актуальность и подходы.	45
2. 2. Постановка задачи идентификации и сведение ее к экстремальной.	56
2. 3. Повышение эффективности алгоритма эволюционных стратегий для решения задачи идентификации ЛДС.	60
2. 4. Моделирование изменения концентрации продуктов распада гексадекана.	70
Выводы	76
ГЛАВА III. РЕШЕНИЕ ЗАДАЧИ ТЕРМИНАЛЬНОГО УПРАВЛЕНИЯ ДЛЯ ДИНАМИЧЕСКИХ СИСТЕМ С ИСПОЛНИТЕЛЬНЫМ МЕХАНИЗМОМ В ВИДЕ МНОГОУРОВНЕВОГО РЕЛЕ.80	
3. 1. Постановка задачи терминального управления в виде многоуровневого реле и сведение ее к экстремальной задаче.	84
3. 2. Разработка обобщенного алгоритма ЭС для прямого решения задачи терминального управления.....	89
3. 3. Определение эффективных настроек и решение задачи управления космическим аппаратом.	100
Выводы	105
ГЛАВА IV. РЕШЕНИЕ ЗАДАЧИ ОПТИМАЛЬНОГО УПРАВЛЕНИЯ ЧИСЛЕННО- АНАЛИТИЧЕСКИМ НЕПРЯМЫМ МЕТОДОМ	109
4. 1. Задача оптимального управления и непрямой метод численно-аналитического поиска решения.	110
4. 2. Экстремальная задача численно-аналитического метода	114
4. 3. Методы стохастической оптимизации в решении экстремальных задач численно- аналитического непрямого метода нахождения оптимального управления.	119
Выводы	127
ЗАКЛЮЧЕНИЕ	130
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	134

ВВЕДЕНИЕ

Актуальность. Современные подходы к решению задач оптимизации становятся все более эффективными в нахождении решений, но при этом количество новых задач и их сложность задач тоже возрастает. Появляются новые классы задач, которые требуют совершенствования алгоритмов оптимизации. Известные методы активно развиваются, совершенствуются и сравниваются между собой, разрабатываются и исследуются алгоритмы, основанные на вновь появляющихся парадигмах. Особый интерес продолжают вызывать эволюционные методы оптимизации и иные методы, суть которых заимствована из природы. Надежность подобных подходов и их модификаций в нахождении желаемого решения сложных задач поиска экстремума демонстрируется в различных работах, в которых решаются, приведенные к экстремальным задачам, задачи различной природы. До сих пор одним из наиболее известных и эффективных методов является метод эволюционных стратегий. Как показано в работах по исследованию методов оптимизации, разные алгоритмы могут по-разному проявлять себя при решении задач: определенный класс алгоритмов быстрее и точнее находит решение некоторых задач, но для других задач данный класс менее эффективен. На настоящий момент существует большое множество модификаций алгоритма эволюционных стратегий, как для решения задач вещественной оптимизации, так и для комбинаторных задач. Тестирование эвристических алгоритмов осуществляется, как правило, на различных наборах целевых функций, которые признаны в тех или иных научных сообществах, и заключается в некоторой оценке эффективности нахождения решения. Более того, в этом направлении происходит развитие методов генерирования специальных целевых функций для проверки эвристических алгоритмов оптимизации. В свою очередь, разрабатываются модификации поисковых алгоритмов или их операторов для повышения эффективности в решении оптимизационных задач определенного класса. Все вышеуказанное свидетельствует о широкой распространенности эвристических алгоритмов поиска экстремума и растущему интересу к ним.

Множество задач математического моделирования и управления сводятся к экстремальным задачам, которые ввиду определенных особенностей формируют собственный класс целевых функций. Задачи математического моделирования, идентификации, управления часто возникают и исследуются во многих областях науки: биология, инженерные дисциплины, химические дисциплины, медицина, физика, эконометрия. Появляющиеся прикладные задачи требуют совершенствования подходов к их решению.

Известно, что существуют различные постановки задач управления и идентификации, которые остаются малоизученными, для которых еще не разработано общего, универсального подхода к нахождению решения. Приведение их к экстремальным задачам и применение эффективных оптимизационных алгоритмов часто позволяет находить решения. Тем не менее, повышение эффективности алгоритмов нахождения решения невозможно без модификаций, совершенствующих поисковые алгоритмы, разработанных с учетом особенностей решаемых задач. Таким образом, новые алгоритмы должны быть эффективнее по некоторым критериям в нахождении решений задач определенного класса. Для их разработки должны быть выявлены особенности задач, предложены подходы к компенсации их влияния и соответствующие модификации структуры алгоритмов или их операторов.

Учитывая актуальность задач идентификации и управления и, в связи с этим, важность повышения эффективности алгоритмов для их решения, можно заключить, что работа, направленная на развитие таких алгоритмов является актуальной.

Целью работы является развитие методов идентификации и управления для динамических систем, повышение эффективности метода эволюционных стратегий для решения задач идентификации линейных динамических систем по данным наблюдений, терминального управления динамическими системами с управлением в виде кусочно-постоянных функций и оптимального управления для численно-аналитического непрямого метода.

Поставленная цель определила следующие **задачи**:

1. Выполнить аналитический обзор существующих методов решения задачи идентификации линейных динамических систем, методов решения терминальной задачи управления (прямых методов) для класса кусочно-постоянных функций, непрямых методов решения задач оптимального управления.
2. Предложить вариант приведения задачи идентификации линейных динамических систем, терминального управления и оптимального управления к задачам поиска экстремума на пространстве вещественных векторов или на пространстве векторов с вещественными и целочисленными координатами, которые соответствуют альтернативам, состоящим из количественных и качественных переменных.
3. Предложить повышающие эффективность алгоритмов модификации для решения каждой приведенной задачи, учитывая особенности пространства поиска и постановки экстремальной задачи.
4. Реализовать алгоритмы решения исследуемых задач в виде программных систем, исследовать системы на тестовых задачах.
5. Исследовать эффективность нахождения решения задач модифицированными алгоритмами, оценить прирост эффективности, достигнутый за счет применения разработанных модификаций.

Методы исследования. При выполнении диссертационной работы использовались методы математического анализа, теории автоматического управления, теории оптимизации, теории вероятностей и математической статистики, вычислительной математики, обработки и анализа данных, оптимального управления и вариационного исчисления, эволюционные и биоинформационные методы оптимизации и анализа данных.

Научная новизна результатов диссертационной работы состоит в следующем:

1. Разработан подход к решению задачи идентификации линейных динамических систем, отличающийся от известных возможностью автоматического определения порядка и параметров дифференциального

уравнения по зашумленным измерениям выхода системы, а так же в условиях малой выборки;

2. Разработан модифицированный гибридный алгоритм эволюционных стратегий, отличающийся от стандартного алгоритма измененной мутацией, селекцией, операцией округления решений и гибридизацией со случайным локальным спуском, демонстрирующий высокую эффективность нахождения решения для редуцированной задачи идентификации линейных динамических систем;

3. Разработан подход к решению задач терминального управления в различных постановках для динамических систем с исполнительным механизмом, формирующим управление в виде кусочно-постоянной функции, отличающийся от известных подходов прямого поиска управления возможностью настройки программного управления и универсальностью относительно постановки задачи;

4. Разработан модифицированный гибридный алгоритм эволюционных стратегий, отличающийся от известных алгоритмов его применимостью к решению задач с количественными и номинальными переменными, с измененными операторами поиска для вещественных переменных и предложенными операторами поиска для номинальных переменных, подходящих для алгоритма эволюционных стратегий;

5. Разработан гибридный модифицированный алгоритм эволюционных стратегий для решения экстремальных задач в численно-аналитическом методе оптимального управления, отличающийся от известных алгоритмов модификациями операций поиска и введением условия перезапуска алгоритма, высокой надежностью нахождения решения для непрямого метода решения задач оптимального управления.

Теоретическая значимость результатов диссертационной работы состоит в том, что были разработаны, исследованы и апробированы новые эволюционные алгоритмы, каждый из которых был усовершенствован для класса задач оптимизации, порожденного задачами идентификации и управления. В работе учитываются особенности экстремальных задач для каждого из предложенных

подходов: к построению аналитической модели в виде дифференциального уравнения по данным наблюдений; к определению программного управления в виде кусочно-постоянной функции для двухточечной задачи; к нахождению оптимального управления для численно-аналитического непрямого метода. Выявлены повышающие надежность модификации алгоритмов для каждой из задач. Приведение исходных задач и разработка алгоритмов их решения являются существенным вкладом в развитие методов идентификации и управления для динамических систем, а также эволюционных методов оптимизации.

Практическая ценность. В ходе проведения исследований были разработаны четыре программные системы. Первая программа предназначена для решения задачи идентификации и позволяет искать модель в виде дифференциального уравнения с максимальным порядком равным 10 по данным наблюдений. Программная система апробировалась на решении задачи идентификации изменений концентрации гексадекана для реакции его распада. Вторая и третья системы предназначены для поиска программного управления в виде кусочно-постоянных функций, например, реле, что позволяет решать задачи терминального управления с закрепленным или свободным временем, соответственно. Третья система апробирована на решении задачи перевода космического аппарата с одной геостационарной орбиты на другую за конечное время. Четвертая система разработана для параллельной идентификации параметров дифференциальных уравнений различных порядков. Разработано множество программных модулей по решению задач терминального управления в различных постановках, оптимального управления и идентификации для вычислительной среды MATLAB. Все приложения не включают в себя какие-либо встроенные алгоритмы численных методов.

Реализация результатов работы. Разработанный алгоритм решения задач оптимизации применялся для идентификации параметров системы сенсоров по их показаниям, что позволило определить ошибку каждого сенсора по фазе и амплитуде и устранить ее (Hochschule Ulm, Ульм, Германия, 2009). Модифицированный гибридный алгоритм эволюционных стратегий применялся

для настройки параметров системы управления в проекте по разработке системы автоматической парковки транспортного средства с прицепом (Hochschule Ulm, Германия, Ульм, 2009).

Подход к решению задачи идентификации применялся для построения модели изменения концентрации для реакции распада гексадекана, что позволило определить порядок и параметры дифференциального уравнения концентрации гексадекана (Красноярск, СФУ, Институт Нефти и Газа, 2011).

Оптимизационный алгоритм применялся для решения задачи оценки параметров системы дифференциальных уравнений, описывающих изменения концентраций продуктов распада при реакции распада гексадекана (Красноярск, СФУ, Институт Нефти и Газа, 2012).

Модифицированный алгоритм эволюционных стратегий применялся при решении задачи идентификации параметров аппроксимирующей функции для оценки поля скоростей текущей в сосуде жидкости (Hochshule Ulm, Германия, Ульм, 2013).

Реализация алгоритмов и подходов в НИОКР:

1. Метод управления квазилинейными динамическими системами на примере управления движением космических аппаратов, 1200954532, 31.12.10 гг.
2. Развитие методов идентификации и управления динамическими системами на основе эволюционных стратегий. 114030440009, с 03.02.14 по 31.12.16 гг.
3. Разработка алгоритмов программно-аппаратного комплекса контроля процесса изготовления прецизионных антенных рефлекторов из полимерных композиционных материалов. 1201359656, с 11.03.13 по 09.10.13 гг.
4. Распределенные интеллектуальные информационные системы обработки и анализа мультILINGвистической информации в диалоговых информационно-коммуникационных системах. 1201175771, с 18.08.11 по 08.05.13 гг.
5. Эволюционные стратегии в задачах идентификации и управления (на примере движения космического аппарата). 1201154205, с 01.01.11 по 31.12.13 гг.

Основные защищаемые положения.

1. Модифицированный гибридный алгоритм эволюционных стратегий обеспечивает большую эффективность решения экстремальной задачи численно-аналитического метода нахождения оптимального управления, чем стандартный метод эволюционных стратегий, метод дифференциальной эволюции, стайный алгоритм и метод эволюционных стратегий с адаптацией ковариационной матрицы.
2. Разработанный подход к решению задачи идентификации линейных динамических систем позволяет автоматически определять параметры и структуру линейного дифференциального уравнения в условиях малой выборки наблюдений выхода системы.
3. Модифицированный гибридный алгоритм эволюционных стратегий, включающий операцию округления, обеспечивает наиболее высокую надежность нахождения решения задачи моделирования динамического процесса по наблюдаемым данными в виде линейного дифференциального уравнения по сравнению с другими известными алгоритмами.
4. Разработанный подход к решению терминальной задачи управления в виде кусочно-постоянных функций позволяет находить решение для двухточечной задачи управления со свободным или закрепленным временем, настраивать количество и расположение точек переключения для управления в виде реле или многоуровневого реле.
5. Обобщенный модифицированный гибридный алгоритм эволюционных стратегий для задач с вещественными и номинальными переменными успешно справляется с поиском экстремума для задач, порожденных задачами терминального управления в виде кусочно-постоянных функций.

Публикации. По теме диссертационной работы опубликовано 19 работ, из них 6 в изданиях Перечня ВАК и 4 зарегистрированные программные системы.

Апробация работы. Результаты научно-исследовательской работы докладывались и обсуждались на XIV Международной научной конференции «Решетневские чтения» (Красноярск, 2010); IX Всероссийской конференции с

международным участием «Молодежь. Общество. Современная наука, техника и инновации» (Красноярск, 2010); на Международной конференции «The 9th International Conference on Informatics in Control, Automation and Robotics (ICINCO'12)», Италия, Рим, 2012; на Международной конференции «The 2nd International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH)», Италия, Рим, 2012; на 1-й Международной конференции «International conference on Mathematical Modeling and its Applications (IWMMMA)», Иркутск, 2012; на Международной конференции «The 11th International Conference on Adaptive and Natural Computing Algorithms (ICANNGA)», Швейцария, Лозанна, 2013; на Международной конференции «The 10th International Conference on Informatics in Control, Automation and Robotics (ICINCO)», Исландия, Рейкьявик, 2013; на Пятой международной конференции «Системный анализ и информационные технологии, САИТ-2013», Красноярск, 2013; на 2-й Международной конференции «International conference on Mathematical Modeling and its Applications (IWMMMA)», Красноярск, 2013; на Международной конференции «The 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO'14)», Австрия, Вена, 2014; на Международной конференции «The 7th International Conference on Swarm Intelligence (ICSI'16)», Малайзия, Бали, 2016 на Международной конференции «The 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO'16)», Португалия, Лиссабон, 2016; на ряде молодежных научных конференций.

Структура работы. Диссертационная работа состоит из введения, четырех глав, заключения, списка литературы и приложений.

ГЛАВА I. ЭВОЛЮЦИОННЫЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧ ПОИСКА ЭКСТРЕМУМА

Большую практическую ценность имеют методы стохастической оптимизации, которые разработаны для прямого нахождения решения экстремальных задач. Сходимость таких методов, хоть и слабая, не требует от целевой функции гладкости, аналитического вида и, тем более, ограниченности и непрерывности ее частных производных, что позволяет широко их использовать для задач типа “черного ящика” (“black-box”), когда целевая функция не может быть представлена аналитически. В настоящее время, подобные алгоритмы продолжают активно развиваться и широко применяться для задач поиска, для которых другие алгоритмы не применимы, или их использование не приносит желаемого результата.

Тем не менее, несмотря на подтвержденную различными примерами эффективность эвристических алгоритмов случайного поиска [13], было выяснено, что данные методы поиска экстремума решают не все задачи одинаково эффективно. Данное высказывание подтверждается “No free lunch theorem”, [65]. Некоторые сложные задачи поэтапно сводились к экстремальным задачам, постепенно более усложняя поверхность отклика целевой функции и способ представления альтернатив. Это приводит к таким оптимизационным задачам, которые для успешного их решения требуют существенного увеличения вычислительных ресурсов для алгоритмов, или совершенствования алгоритмов, или же поиска иного подхода к решению задачи.

Невысокая эффективность при решении тех или иных задач может быть связана с настройками алгоритма, число которых, в общем случае, велико. Более того, настройки могут быть выражены как номинальными переменными, так и вещественными, что означает бесконечное количество реализаций алгоритма с различными настройками. Для решения этой проблемы существуют и развиваются различные подходы самонастройки или адаптации управляющих параметров алгоритма, например, [6], [17], [39], [113] и [115].

При этом некоторые задачи, например обратные задачи математического моделирования, сводящихся к экстремальным задачам, формируют собственные классы различных целевых функций. В общем случае, поверхность отклика целевой функции может быть такова, что стандартный метод не обеспечивает высокую надежность нахождения решения приведенных задач поиска экстремума. В таких случаях подбор настроек из некоторого множества, не принесет существенного улучшения работы поисковой процедуры. Более того, применение каких-либо взаимодействий алгоритмов, например, островные модели, коэволюции или коллективы, в таких случаях, аналогично не приведет к улучшению результата. Данный факт указывает на то, что необходимо дополнительно исследовать целевую функцию и способ представления решения задачи, чтобы, проанализировав их, предложить качественные изменения алгоритма. Такие изменения или модификации выражаются в изменениях схемы работы алгоритма, операторов или добавлении операторов.

1. 1. Алгоритм эволюционных стратегий.

В данном параграфе рассмотрен алгоритм решения экстремальных задач - эволюционные стратегии (ЭС). Эволюционные стратегии являются стохастическим алгоритмом решения задач оптимизации на пространстве вещественных векторов, в основе которого лежит упрощенная модель принципа эволюции. Данный алгоритм, как и многие другие эволюционные методы поиска, является представителем оптимизационных методов нулевого порядка. Впервые алгоритм был предложен Н.-Р. Schwefel в 60-х годах. В последующие десять лет автор и команда исследователей улучшили алгоритм и представили его адаптивную версию, суть которой излагается в работе [56].

В основе этих методов лежит принцип «выращивания» лучшего решения на пространстве альтернатив за счет процедур, создающих имитацию биологической эволюции – процессы отбора, воспроизводства потомка и мутации, [15], [16], [56]. Иными словами, согласно принципу «выживает наиболее приспособленный» или, точнее, «наиболее приспособленный имеет большую вероятность выжить», в

каждом новом поколении появляются все более и более приспособленные индивиды, что соответствует лучшим альтернативам.

Рассмотрим некоторые общепринятые термины, относящиеся к эволюционным алгоритмам:

- индивид – альтернатива, вариант решения задачи, которому необходимо соответствует некоторое значение целевой функции;
- пригодность – некоторое вещественное число, необходимое для попарного сравнения индивидов или сравнения индивидов в совокупности, так что индивид с большей пригодностью имеет больше шансов выжить и/или оставить потомков, определяется значением функцией пригодности для индивида;
- функция пригодности – некоторое отображение множества значений целевой функции, ставящее в соответствие каждому элементу множества значений целевой функции элемент из множества значений функции пригодности, так что лучшему решению задачи по заданному критерию соответствует большая пригодность представляющего решение индивида;
- поколение – итерация алгоритма, один цикл эволюции;
- популяция – текущее множество альтернатив, множество индивидов соответствующих одному поколению;
- генотип – способ представления приведенного решения;
- фенотип – решение исходной задачи;
- хромосома – набор генов, представленное приведенное решение;
- гены – переменные приведенной задачи;
- скрещивание, мутация, селекция – операторы поиска, олицетворяющие процессы воспроизведения потомка, случайных изменений в генах потомка и отбора индивидов для скрещивания или выживания.

Рассмотрим задачу поиска безусловного экстремума некоторого функционала

$$Q(a) \rightarrow \min_a, a \in A, \quad (1.1)$$

где $Q(\cdot): A \rightarrow R$ - целевая функция, выраженная функционалом, который определен, в общем случае, на всем пространстве поиска A , т.о. A – пространство альтернатив, которое, в рассматриваемых задачах представимо в виде Евклидова пространство размерности q , или прямого произведения такого пространства и пространства векторов с целыми координатами.

Для ее решения используется алгоритм, который представляет собой определенную последовательность эволюционных операторов поиска, и схематично изображен на рисунке 1.1.

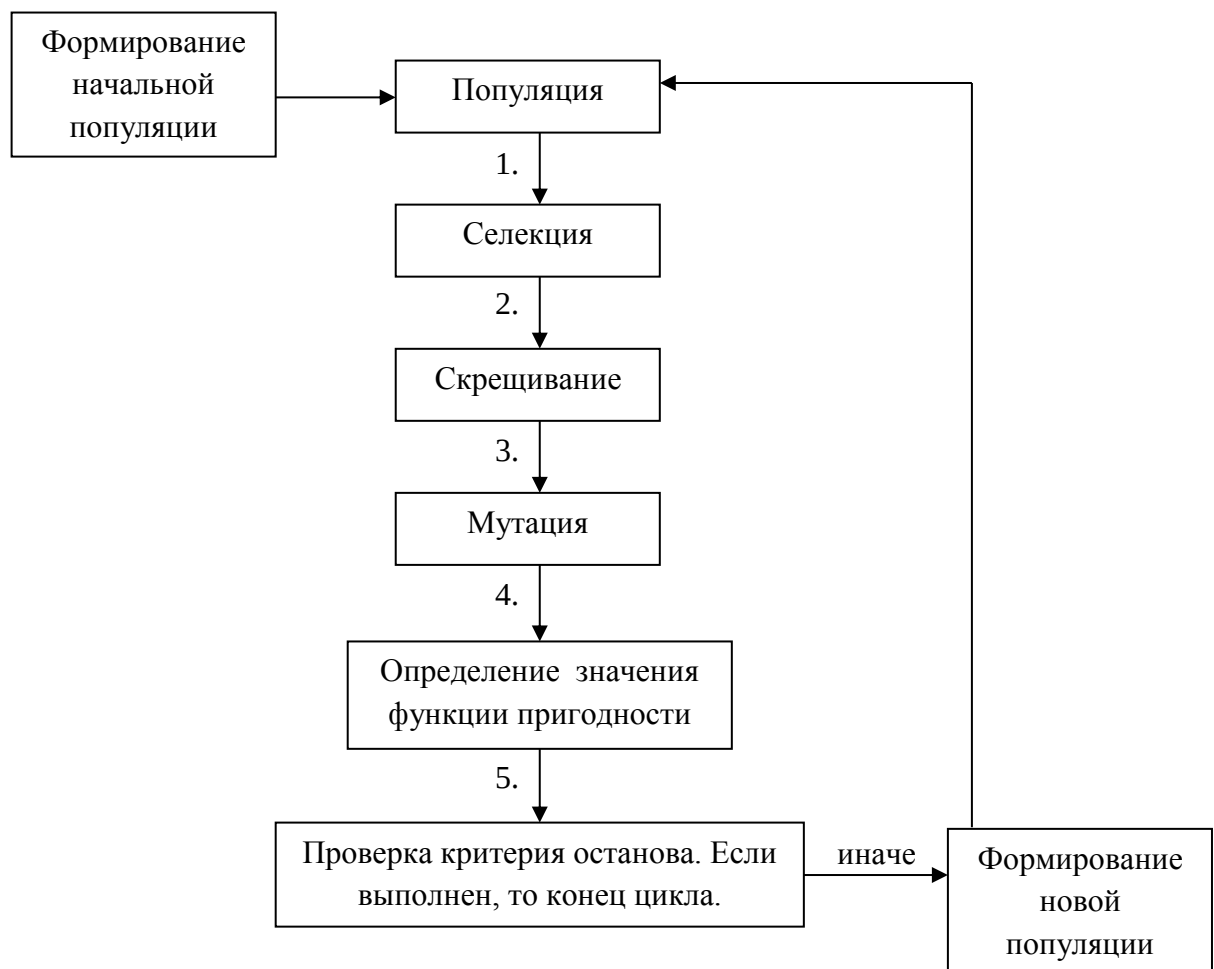


Рисунок 1.1. Схема последовательности действий алгоритма ЭС.

Процесс генерирования стартовой популяции будет рассмотрен ниже. На данном этапе отметим только, что популяция, как правило, генерируется случайным образом.

На этапе 1 происходит отбор нескольких индивидов, согласно выбранной схеме селекции на основе значений пригодности индивидов в текущей популяции. Лучшему решению задачи, как было сказано ранее, соответствует индивид с большей пригодностью. Соответственно, чем больше значение пригодности индивида, тем больше он имеет шансов оставить потомка и/или выжить, т.е. тем выше вероятность этого. Такая схема проведения отбора является зависимой от значения критерия для каждой отдельной альтернативы. Для некоторых типов селекции, вероятность быть выбранным для каждого индивида зависит не от его величины пригодности, а от его порядкового номера в упорядоченном по пригодности наборе индивидов внутри популяции.

На этапе 2, согласно некоторой выбранной схеме скрещивания, хромосомы выбранных особей – родителей, некоторым образом образуют хромосому потомка. Как и для селекции, для скрещивания существуют схемы, в которых учитывается или не учитывается пригодность индивида. Результат скрещивания для каждого отдельного гена – независимое случайное событие от результатов скрещивания для других генов или от порядкового номера текущего гена в хромосоме.

Отметим, что оператор селекции таков, что выбирается несколько индивидов-родителей. Аналогичным образом, оператор скрещивания определен для определенного числа индивидов, в общем случае, любого натурального.

На этапе 3 происходит мутация генов потомка. Мутация предполагает случайное изменение генов полученного индивида, при этом, аналогично, для каждого отдельного гена результат мутации есть случайная величина, определяемая оператором мутации и значением гена, и не зависящая от значений других генов. В классическом варианте алгоритма ЭС случайное возмущение альтернативы осуществляется с помощью нормально распределенной случайной величины с нулевым математическим ожиданием. За счет этого возмущение равновероятно может быть направлено в любую сторону и принимать с некоторой вероятностью любое значение, что соответствует принципу максимума энтропии.

На этапе 4 определяется функция пригодности для вновь полученного потомка. Здесь необходимо указать некоторые особенности перехода от целевой функции к функции пригодности. Согласно определению функции пригодности, которое дано выше:

$$Q(a_1) < Q(a_2) \Rightarrow fit(Q(a_1)) > fit(Q(a_2)), a_1, a_2 \in R^q, \quad (1.2)$$

где $fit(\cdot): F_R \rightarrow R, F_R \subset R$ - функция пригодности для критерия (1.1). В некоторых случаях эта функция может быть непрерывна или кусочно-непрерывна в области определения.

В данной работе рассматриваются критерии (1.1) в виде невязок, область значения которых полностью лежит на неотрицательной полуоси, $Q(\cdot): A \rightarrow [0, +\infty)$. Для удобства сравнения альтернатив выберем такой переход к функции пригодности, чтобы область значений функции была определена закрытым множеством $[0; 1]$. Тогда значению 1 будет соответствовать индивид, для которого значение критерия равно 0. Такой переход осуществим, например, с помощью следующего преобразования:

$$fit(a) = \frac{1}{1 + Q(a)}, \quad (1.3)$$

где $fit(\cdot): A \rightarrow [0, 1]$ - функция пригодности, которая будет использоваться в данной работе.

Легко убедиться в том, что функция пригодности (1.3) не противоречит условиям (1.2).

На этапе 5 проверяется условие останова: достигнута ли желаемая пригодность (что равнозначно определенному значению целевой функции) или достигнуто ли максимальное количество поколений (итераций) алгоритма. Если условие останова не выполнено, то формируется новая популяция из полученных потомков.

Алгоритм эволюционных стратегий, в общем случае, предполагает возможность формирования следующей популяции различными способами. Так, следующая популяция может быть сформирована только из потомков, или может

включать в себя лучшие решения из множеств потомков и родителей. В обоих случаях, выбирается количество индивидов равное числу популяции, которые лучшие по пригодности во множестве потомков, в первом случае, или во множестве родителей и потомков, во втором случае. Выбирая первый вариант, необходимо чтобы число потомков было не меньшим, чем количество индивидов в популяции.

На рисунке 1.2 представлена схема последовательности действий внутри каждого цикла алгоритма, которые определяют конечное решение. Поскольку алгоритм является стохастическим, то, в общем случае, максимальное значение функции пригодности может быть фиксировано не на последней популяции, а, например, в первых популяциях. Прерывистыми линиями обозначены переходы, которые выполняются после сравнения лучшей в найденном множестве решений альтернативы, и только в случае если найденная альтернатива лучше, чем зафиксированная в качестве лучшей до этого. При этом не нарушается схема алгоритма, сохраненное таким образом решение не влияет на популяцию, а, значит, не увеличивается риск преждевременной сходимости алгоритма.

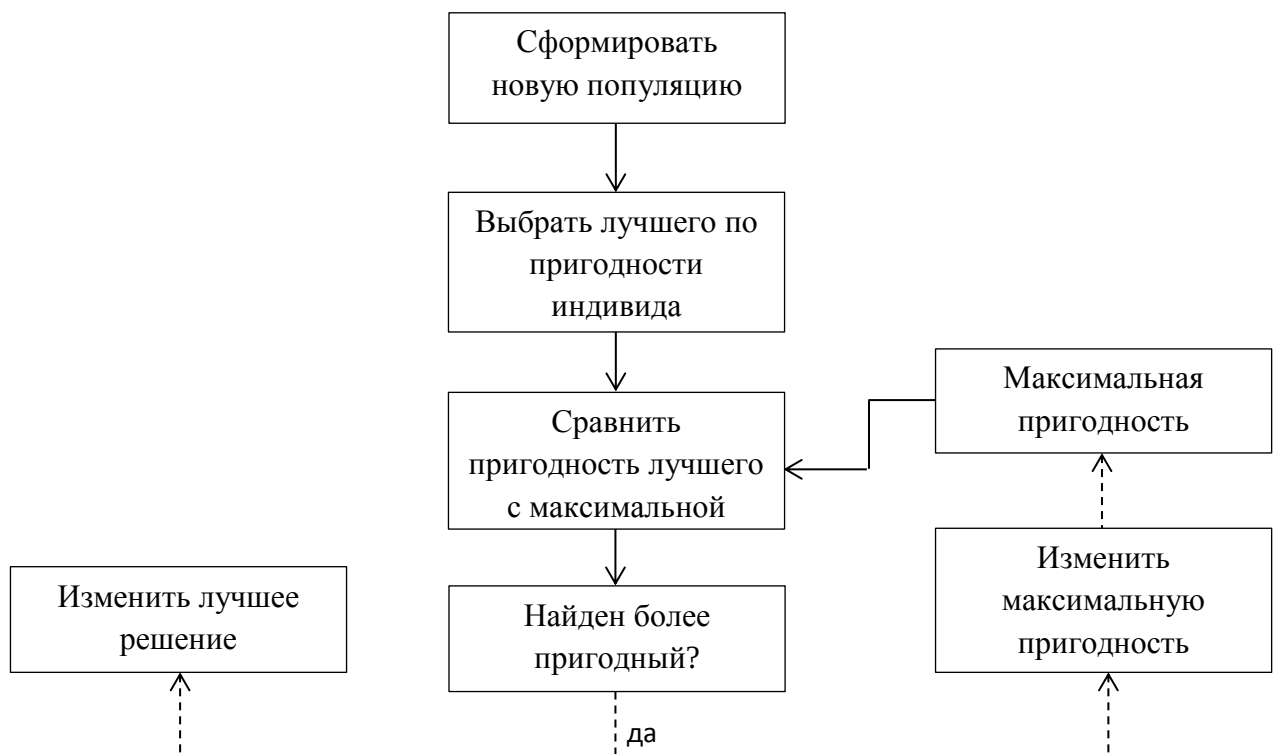


Рисунок 1.2. Определение лучшего решения на каждом цикле алгоритма.

Отметим, что после формирования стартовой популяции и определения пригодности для каждого индивида, решение с максимальной пригодностью автоматически становится лучшим решением. В плане реализации, некоторые схемы селекции требуют, чтобы множество решений было упорядоченным по значению пригодности. Таким образом, процесс нахождения наилучшего решения внутри популяции заметно упрощается, без подключения дополнительных поисковых процедур. Тем не менее, если в сортировки популяции необходимости нет, то эффективнее, в смысле временных затрат, ограничиться сравнением пригодности каждого нового индивида с текущим лучшим найденным решением.

В данной работе предложена схема, в которой число потомков соответствует фиксированному объему популяции, что свойственно и многим другим методам эволюционного поиска, например, генетическому алгоритму, стайному алгоритму, методу дифференциальной эволюции и т.п. Число родителей для некоторых задач выбиралось равным 2, поскольку, в некоторых работах утверждается, что наиболее удачное число родителей для эволюционных алгоритмов - 2, поэтому, предположительно, данная настройка является лучшей.

1. 2. Модификация схемы и операторов поиска алгоритма эволюционных стратегий

Рассмотрим задачу оптимизации на Евклидовом пространстве размерности q . Каждая хромосома $H_i, i = \overline{1, N}$ представлена набором объективных параметров, которые соответствуют переменным задачи, и набором стратегических параметров [10], [56], которые отвечают за мутацию и адаптацию операции мутации в процессе поиска решения, т.е.

$$H_i = \langle op_i, sp_i \rangle;$$

$$op_i = (o_1^i, \dots, o_q^i);$$

$$sp_i = (s_1^i, \dots, s_q^i),$$

где $o_j^i \in R, s_j^i \in R^+, i = \overline{1, N}, j = \overline{1, q}$ - гены, соответствующие объективным и стратегическим параметрам, соответственно; N - объем популяции. Здесь и далее, в соответствии с принятыми определениями для ЭС, объективные параметры – переменные пространства поиска, стратегические параметры – переменные алгоритма поиска.

Таким образом, каждый индивид I определяется своей хромосомой и соответствующим значением функции пригодности:

$$I_i = \langle H_i, fit(op_i) \rangle, i = \overline{1, N}. \quad (1.4)$$

Далее остановимся на том, как формируется стартовая популяция. Изначально формируем стартовую популяцию объема N в некотором шаре $B(c, N) = \left\{ x_i \in R^q : \|x_i - c\| \leq r, i = \overline{1, N} \right\}$ пространства R^q , где q - размерность пространства задачи поиска экстремума, c - центр шара, r - его радиус, а $\|\cdot\|: R^q \rightarrow R^+ \cup \{0\}$ - норма в пространстве R^q . Одним из способов генерирования стартовой популяции является разыгрывание многомерной нормальной или равномерной случайной величины, для каждого $i = \overline{1, N}$:

$$\begin{aligned} o^i &\sim N\left(c_0^o, (\sigma_0^o)^2\right), \\ s^i &\sim \left| N\left(c_0^s, (\sigma_0^s)^2\right) \right|, \end{aligned} \quad (1.5)$$

или

$$\begin{aligned} o^i &\sim U(l^o, r^o), \\ s^i &\sim U(0, r^s), \end{aligned} \quad (1.6)$$

где

$N(m, \sigma^2)$ - нормально распределенная случайная величина с начальным моментом m и вторым центральным моментом σ^2 ;

$U(a, b)$ - равномерно распределенная случайная величина на интервале $[a, b]$;

$c_0^o, \sigma_0^o, c_0^s, \sigma_0^s$ - параметры распределений нормальной случайной величины: математическое ожидание и среднеквадратичное отклонение, для объективных и стратегических параметров, соответственно;

l^o, r^o, r^s - параметры распределения равномерной случайной величины: левая и правая границы объективных параметров и правая граница стратегических параметров, соответственно.

Подобно начальному положению для других численных методов оптимизации здесь параметры стартовых распределений определяют не только начальный момент, в окрестности которого будут распределены элементы стартового множества, но и интенсивность случайных изменений, которые определяются значениями стратегических параметров. Для некоторых задач поиска экстремума, где границы области поиска заданы, центры начальных распределений рекомендуется выбирать внутри этих множеств. При отсутствии априорных сведений о характере целевой функции и окрестностях, которые могут содержать решение задачи, нет рекомендаций по поводу выбора начального распределения. В данной работе рассматриваются именно такие задачи оптимизации.

Рассмотрим основные операторы поиска алгоритма ЭС: селекцию, рекомбинацию, мутацию.

В работе число типов селекции было увеличено, за счет добавления уже известных типов из стандартного генетического алгоритма, [15], [16]. Рассматриваются турнирная селекция, ранговая и пропорциональная. Последний тип является стандартным для алгоритма эволюционных стратегий.

В турнирной селекции из текущей популяции случайным образом выбирается n_t различных индивидов, где n_t - размер турнира, затем отбирается лучший по функции пригодности индивид в выбранной группе. Если, как было указано ранее, мы имеем дело с множеством альтернатив, упорядоченным по убыванию значений функции пригодности, то выбор каждого родителя происходит следующим образом:

$$H_{parent} = H_{i_{tournament}},$$

$$i_{tournament} = \max_{i \leq n_t} \left\{ I_{n_t}^i, i = \overline{1, n_t} : P(I_{n_t}^i = 1) = P(I_{n_t}^i = 2) = \dots = P(I_{n_t}^i = N) = 1/N \right\}. \quad (1.7)$$

Далее, пусть $\phi(x): R \rightarrow R, x_1 \geq x_2 \Rightarrow \phi(x_1) \geq \phi(x_2)$. Данная функция может являться некоторым преобразованием, на котором строятся различные вариации пропорциональной и ранговой селекции. В данной работе функция принята следующей: $\phi(x) = x$.

При пропорциональной селекции каждому индивиду присваивается вероятность быть выбранным пропорционально значению его функции пригодности, т.о. вероятность i -го индивида быть выбранным находится по формуле

$$p_i^p = \frac{\phi(\text{fit}(op_i))}{\sum_{j=1}^N \phi(\text{fit}(op_j))},$$

Тогда

$$H_{parent} = H_{i_{proportional}},$$

$$P(i_{proportional} = j) = p_j^p, j = \overline{1, N}. \quad (1.8)$$

Далее рассмотрим схему ранговой селекции. При данном типе селекции каждому индивиду присваивается некоторый ранг, который, в общем случае, является преобразованием от его номера в упорядоченном по пригодности множестве решений. Вероятность для каждого индивида быть выбранным определяется, например, по следующей формуле:

$$p_i^r = \frac{\phi(i)}{\sum_{j=1}^N \phi(j)} \stackrel{\phi(x)=x}{=} \frac{2 \cdot i}{N(N+1)},$$

тогда

$$H_{parent} = H_{i_{rank}},$$

$$P(i_{rank} = j) = p_j^r, j = \overline{1, N}. \quad (1.9)$$

После того, как были выбраны родители, в данном случае – пара, потомок производится согласно операции скрещивания. Для алгоритма эволюционных стратегий существует несколько стандартных вариантов реализации процесса скрещивания. Для удобства, будем считать, что гены потомка определяются кортежем

$$H_{crossover}(i, j) = \langle op_{crossover}^{i,j}, sp_{crossover}^{i,j} \rangle,$$

где

i, j - индексы родителей в упорядоченном множестве решений;

$op_{crossover}^{i,j}$ - объективные параметры потомка;

$sp_{crossover}^{i,j}$ - стратегические параметры потомка.

Перечислим различные схемы скрещивания, определенные для n_p родителей. Первая основана на взвешенной сумме параметров, $j = \overline{1, q}$:

$$(op_{crossover})_j = \sum_{i=1}^{n_p} w_i \cdot (op_{parent})_i; \quad (1.10)$$

$$(sp_{crossover})_j = \sum_{i=1}^{n_p} w_i \cdot (sp_{parent})_i \quad (1.11)$$

В основе второй схемы лежит случайный выбор объективного или стратегического параметра потомка от одного из родителей, таким образом, случайная величина i имеет следующие вероятности распределения:

$P(i = k) = w_k, \quad k = \overline{1, n_p}$, тогда для $k = \overline{1, n_p}, \quad j = \overline{1, q}$:

$$P(op_{crossover})_j = (op_{parent})_i; \quad (1.12)$$

$$P(sp_{crossover})_j = (sp_{parent})_i. \quad (1.13)$$

Таким образом, изменяя способ выбора параметров, мы получаем различные типы скрещивания, при этом для каждого типа скрещивания

необходимо выполняется условие нормированности параметров, $\sum_{i=1}^{n_p} w_i = 1$, далее

для $j = \overline{1, q}$ получаем следующие типы формирования весовых коэффициентов:

- среднее:

$$w_j = \frac{1}{n_p}; \quad (1.14)$$

- средневзвешенное по пригодности:

$$w_j = \frac{\text{fit}\left((op_{parent})_j\right)}{\sum_{i=1}^{n_p} \text{fit}\left((op_{parent})_i\right)}; \quad (1.15)$$

- средневзвешенное со случайными параметрами и случайными по пригодности, соответственно:

$$w_j = \frac{r_j}{\sum_{i=1}^{n_p} r_i}, r_j \sim U(0,1), \quad (1.16)$$

$$w_j = \frac{r_j}{\sum_{i=1}^{n_p} r_i}, r_j \sim U\left(0, \text{fit}\left((op_{parent})_j\right)\right). \quad (1.17)$$

Далее, для предотвращения вычислительных трудностей, введем специальную величину, которая будет являться нижним порогом значения весового коэффициента. Параметр $b^w \in (0,1]$, если $\exists j: w_j < b^w \Rightarrow w_j = b^w$ и после проверки всех коэффициентов, для $j = \overline{1, n_p}$, выполним $w_j = \frac{w_j}{\sum_{i=1}^{n_p} w_j}$.

Используя вышеприведенные обозначения, получаем, что при (1.10)-(1.11) и (1.14) – средневзвешенное скрещивание, (1.10)-(1.11) и (1.15) – средневзвешенное по пригодности, (1.12)-(1.13) и (1.14) – дискретное скрещивание.

Различные схемы скрещивания могут быть более или менее эффективны для различных задач. Ниже представлен оператор, который, по мнению авторов метода эволюционных стратегий, является основным оператором поиска.

Мутация в эволюционных алгоритмах есть поисковый оператор, который случайным образом изменяет стратегические и объективные параметры индивида. Посредством оператора мутации определяется новая точка поискового

пространства, которая есть результат случайного изменения координат порождающей ее точки. Для удобства обозначения, потомок:

$$\tilde{H} = \langle o\tilde{p}, s\tilde{p} \rangle = \langle op_{crossover}, sp_{crossover} \rangle,$$

а особь после мутации:

$$H_{mutation} = \langle op_{mutation}, sp_{mutation} \rangle.$$

Гены индивида-мутанта задаются следующими операциями:

$$op_{mutation} = (o_1 + Nr_p^1, \dots, o_q + Nr_p^q), \quad (1.18)$$

$$sp_{mutation} = (s_1 \cdot e^{\tau \cdot Nr_s^1}, \dots, s_q \cdot e^{\tau \cdot Nr_s^q}), \quad (1.19)$$

где

$Nr_p^i \sim N(0, s_i^2), i = \overline{1, q}$ - нормально распределенная одномерная случайная величина, центрированная, среднеквадратичное отклонение - s_i ;

$Nr_s^i \sim N(0, 1), i = \overline{1, q}$ - нормально распределенная одномерная случайная величина, центрированная и с единичным среднеквадратичным отклонением;

$\tau \in R^+$ - параметр адаптации оператора мутации. Если не указано иного, принимается равным 1.

Как будет показано ниже, для некоторых сложных задач поиска экстремума будет более эффективной иная схема реализации мутации для стратегических параметров:

$$sp_{mutation} = (|s_1 + \tau \cdot Nr_s^1|, \dots, |s_q + \tau \cdot Nr_s^q|). \quad (1.20)$$

Часто используется адаптивная мутация, где параметр адаптации тоже претерпевает равновероятные случайные изменения. Для удобства, обозначим в таком случае параметр мутации как $A \in R^q$, где A - случайный вектор, причем:

$$P\left(A_i = \frac{A_i}{2}\right) = P(A_i = 2 \cdot A_i) = 0.5, i = \overline{1, q}.$$

Тогда мутация для стратегических параметров алгоритма может происходить следующим образом:

$$sp_{mutation} = (A_1 \cdot s_1 \cdot e^{Nr_s^1}, \dots, A_q \cdot s_q \cdot e^{Nr_s^q}), \quad (1.21)$$

или

$$sp_{mutation} = (|s_1 + A_1 \cdot Nr_s^1|, \dots, |s_q + A_q \cdot Nr_s^q|). \quad (1.22)$$

Был предложен метод модификации оператора мутации так, что каждая пара параметров: объективный и, соответствующий ему, стратегический параметр мутируют с определенной, заранее заданной, вероятностью. Пусть z - случайный вектор, каждая координата которого принимает одно из двух значений:

$$\forall i \leq q: P(z_i = 0) = p_m, P(z_i = 1) = 1 - p_m,$$

где p_m - вероятность для каждой пары генов мутировать.

Таким образом, мутация будет происходить согласно ниже представленной схеме (для упрощения представления, используем умножение по Адамару):

$$op_{mutation} = op_{crossover} + z \circ Nr_p, \quad (1.23)$$

$$sp_{mutation} = abs(sp_{crossover} + \tau \cdot z \circ Nr_s), \quad (1.24)$$

где $y = abs(x): y_i = |x_i|, x_i \in R, i = \overline{1, q}$.

В случае применения адаптивной мутации:

$$sp_{mutation} = abs(sp_{crossover} + z \circ (A \circ Nr_s)). \quad (1.25)$$

Аналогично, для стандартной схемы мутации:

$$sp_{mutation} = (A_1 \cdot s_1 \cdot e^{\tau \cdot z_1 \cdot Nr_s^1}, \dots, A_q \cdot s_q \cdot e^{\tau \cdot z_q \cdot Nr_s^q}). \quad (1.26)$$

Таким образом, оператор мутации был изменен так, что изменением величины контролирующего вероятность мутации параметра можно управлять интенсивностью мутации и, например, снижать ее, в случае, когда незначительные возмущения объективных параметров приводят к существенному ухудшению значения целевой функции.

Скрещивание для предлагаемого модифицированного алгоритма тоже имеет вероятностный характер: для каждого набора родителей существует вероятность получить потомка через рекомбинацию, равная p_c . В используемых алгоритмах вероятность скрещивания бралась близкой к 1 и не меньшей 0.8. В случае если

для конкретного набора родителей было разыграно, что потомок получается не путем скрещивания, то случайным образом выбирался родитель, который и становился потомком. Вероятность выбора для каждого из родителей была равной:

$$P(i_c = 1) = \dots = P(i_c = n_{op}) = \frac{1}{n_{op}},$$

$$H_{crossover} = H_{i_c}.$$

В данной работе исследуется гибридизация алгоритма ЭС, в которой реализована схема улучшения решений, основанная на вероятностном алгоритме покоординатного спуска. Для каждой популяции N_1 раз случайным образом выбирается некоторое решение, в котором N_2 раз выбирается координата, по которой производится N_3 шагов величиной h_l в одном из случайно выбранных направлений, пока и если это приводит к улучшению целевой функции для данного решения. Нужно отметить, что в одной из модификаций гибридного алгоритма величина шага для каждого отдельного цикла бралась равной $U(0, h_l)$.

Таким образом, локальное улучшение решения можно представить в следующем виде:

$$(I): i_1 : P(i_1 = 1) = \dots = P(i_1 = N) = \frac{1}{N},$$

$$(II): i_2 : P(i_2 = 1) = \dots = P(i_2 = q) = \frac{1}{q},$$

$$(III): s_l : P(s_l = 1) = P(s_l = -1) = 0.5,$$

$$(IV): op_j^{loc} = \begin{cases} op_j^{i_1}, \forall j \in \overline{1, q}, j \neq i_2, \\ op_j^{i_1} + s_l \cdot h_l, j = i_2, \end{cases}$$

$$(V): fit(op^{i_1}) \geq fit(op^{loc}) \Rightarrow op^{i_1} \leftarrow op^{loc}.$$

В предложенной выше схеме, на шаге (IV) выполняется N_3 итераций, пока выполняется условие (V). Шаги (I) и (II) представляют собой циклы, выполняющиеся N_1 и N_2 раз, соответственно. При этом если выбор направления

(III) был таковым, что в результате проверки (V), обнаружено, что решение ухудшилось, то выбирается противоположное направление спуска, $s_i = s_i \cdot (-1)$. Далее с новым направлением переходим на шаг (IV).

Часто бывает эффективнее применять условие в виде строгого неравенства, ввиду того, что целевая функция, а, следовательно, функция пригодности может быть локально постоянной: $\exists c > 0: fit(x) = C, \forall x: \|x - x_r\| < c$, где x_r - некоторая альтернатива. В таком случае, без улучшения пригодности решения возможно осуществления большого количества шагов, и, следовательно, вычислений целевой функции, которые, впрочем, не приведут к улучшению решения ввиду величины шага локальной оптимизации. По этой причине, в данной работе использовалось условие в виде строгого неравенства.

Схема алгоритма представлена на рисунке 1.3.

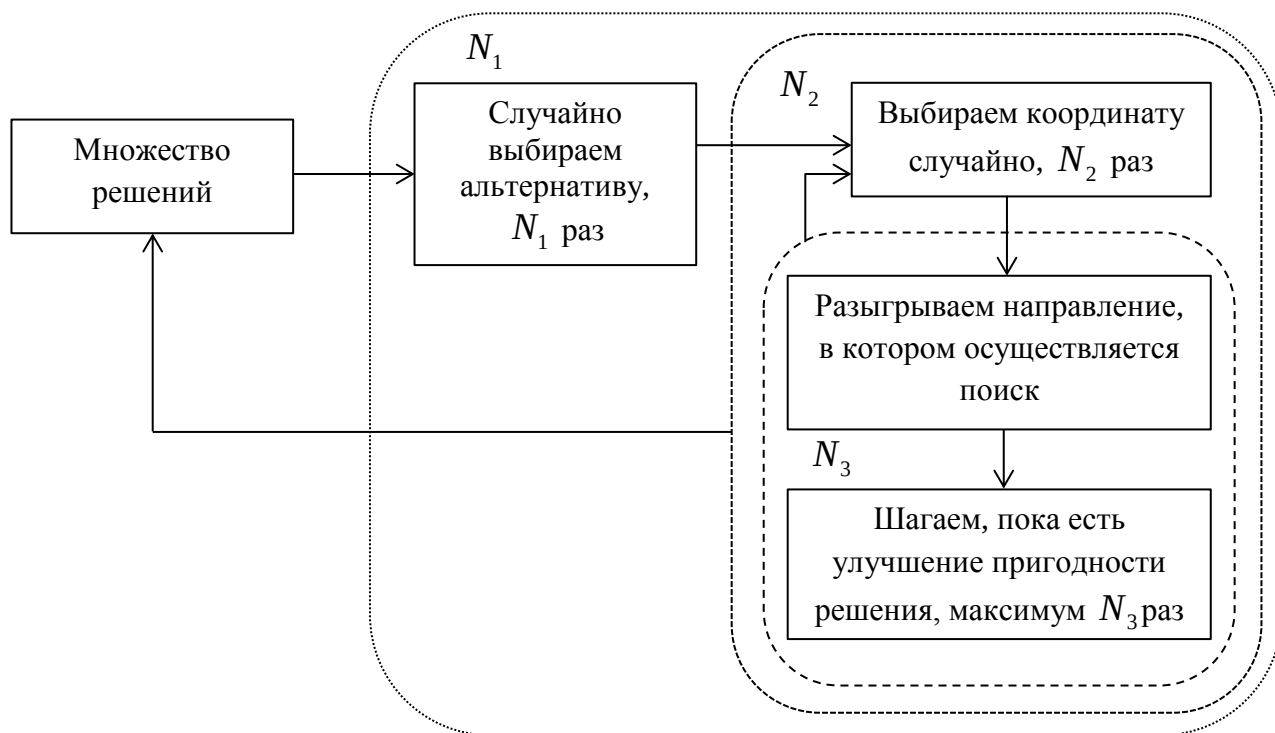


Рисунок 1.3. Схема случайного покоординатного спуска.

При гибридизации, алгоритм локального улучшения добавляется в схему после оператора мутации. Дополняя алгоритм ЭС локальным спуском, мы сталкиваемся с тем, что множество настроек дополняется набором: N_1, N_2, N_3, h_l .

Настройки локального спуска и мутации – наиболее результативные в управлении алгоритмом и оказывают сильное совместное влияние на эффективность алгоритма. Для некоторых задач, надежность нахождения решения эволюционным алгоритмом сильно зависит от того, как в совокупности воздействуют на индивида эти операторы.

На схеме видно, что для после того, как для выбранной случайно альтернативы были произведены все операции по улучшению, она возвращается во множество решений. Таким образом, каждая альтернатива может быть выбрана несколько раз в пределах одного запуска локального улучшения. Аналогично, каждая координата решения может быть выбрана, улучшена, возвращена в альтернативу и, затем, выбрана снова.

Для гибридного алгоритма схема последовательности действий будет выглядеть несколько иначе, чем для модифицированного алгоритма ЭС. На схеме, представленной на рисунке 1.4, видно, что для индивидов последовательно выполняются операции мутации – случайный поиск и локальный спуск, но, тем не менее, предполагается, что оператор случайного поиска включается, только когда набрана новая популяция индивидов. На схеме представлены действия для всех индивидов в совокупности, общие операторы внутри одно цикла работы. Если рассматривать цикл более детально, то, на первом шаге завершается отбор всех потомков, после чего потомки, каждый из которых прошел операцию мутации, формируют множество решений, которое и поступает на вход блока локальной оптимизации.

Формирование новой популяции предполагает подготовку данных для запуска нового цикла алгоритма и включает в себя вычисление функции пригодности для каждого индивида и, поскольку в данной работе используется турнирная и ранговая селекция, сортировку индивидов популяции по убыванию значения пригодности.

Далее рассмотрим обобщение алгоритма эволюционных стратегий для решения задач со смешенными переменными и мета-эвристику, которая позволяет управлять поиском за счет осуществления перезапусков.

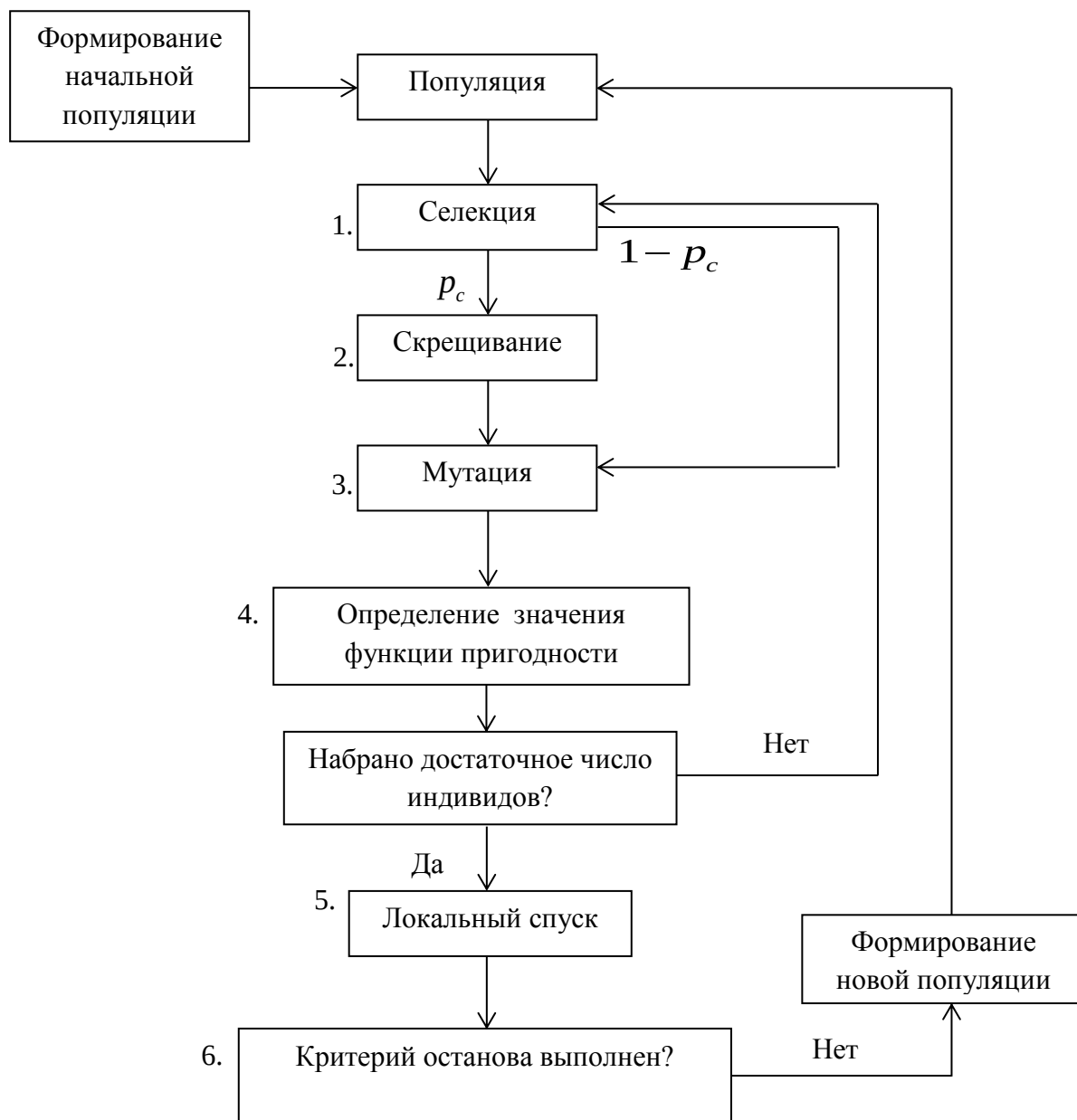


Рисунок 1.4. Схема последовательности действий гибридного алгоритма ЭС.

1. 3. Обобщение алгоритма эволюционных стратегий. Мета-эвристика управления запусками алгоритма.

Данный пункт содержит описание обобщения алгоритма эволюционных стратегий на случай решения задач оптимизации для вещественных и целочисленных переменных. Задачи с переменными смешанных типов возникают, когда некоторые переменные представляют собой ранговую или номинальную величину и можно найти биекцию между множеством возможных значений

каждой из такой переменной и подмножеством целых чисел. Решение таких задач требует переопределения операторов скрещивания и мутации для переменных, представленных целыми числами.

Типы скрещивания были заимствованы из генетического алгоритма: одноточечное, двухточечное и равномерное. Для реализации оператора мутации для целочисленных параметров и, соответствующих им, стратегических параметров, была предложена схема, которая согласуется с концепцией метода эволюционных стратегий: стратегический параметр определяет мутационное воздействие.

Введем функцию вида $f_z(a, b) : R \times R \rightarrow \{0, 1\}$, $f_z(a, b) = \begin{cases} 0, & a < b \\ 1, & a \geq b \end{cases}$. Пусть задана дискретная случайная величина Z_1 , распределенная по следующему закону: $P(z_1 = 1) = p_m^1$, $P(z_1 = 0) = 1 - p_m^1$; задана равномерно распределенная случайная величина $Z_2 \sim U(0, 1)$; дискретная случайная величина R_u , с законом распределения $P(r_u = u_1) = \dots = P(r_u = u_{N_u}) = \frac{1}{N_u}$.

Тогда мутация номинальных переменных определяется через разыгранные случайные величины для каждого гена и проверку условия $f_z(s, z_2) = 1$, так, что если условие выполнено, то выполняются следующие преобразования $\forall j : o_j \in N$:

$$o_j = (1 - z_1) \cdot o_j + z_1 \cdot r_u, \quad (1.27)$$

$$s_j = \left| s_j + z_1 \cdot N(0, 1) \right|, \quad (1.28)$$

для измененной мутации и для стандартной

$$s_j = \left| (1 - z_1) \cdot s_j + z_1 \cdot s_j \cdot e^{N(0, 1)} \right|. \quad (1.28)$$

Таким образом, стратегический параметр управляет вероятностью мутации для каждой пары генов, для которой величина $z_1 = 1$. Отметим, что в этой схеме не отображен еще один параметр – вероятность проведения оператора мутации, который соответствует модифицированному алгоритму. В случае

модифицированной мутации данная схема мультипликативно дополняется еще одним параметром и приобретает вид, обозначенный в выражениях (1.23) и (1.24).

В некоторых случаях, в особенности для ранговых переменных, может быть более эффективным применение иной схемы мутации. Пусть n_r - количество рангов, и множество возможных значений считаем представленным множеством натуральных чисел $\{1, \dots, n_r\}$, так что каждая переменная, являющаяся ранговой, принимает одно из этих значений, $o \in \{1, \dots, n_r\}$. Тогда, мутация для объективных параметров может быть осуществлена по следующей схеме, в которой изменение параметра полностью определяется его текущим значением, при $z_1 \cdot f_z(s, z_2) = 1$:

$$o_j = 1 \Rightarrow o_j = o_j + 1; \quad (1.29)$$

$$1 < o_j < n_r \Rightarrow o_j = o_j + (-1)^{z_4}; \quad (1.30)$$

$$o_j = n_r \Rightarrow o_j = o_j - 1, \quad (1.31)$$

где $P(z_4 = 1) = P(z_4 = 0) = 1/2$ - случайная величина.

Для повышения эффективности работы алгоритмов был также добавлен новый оператор поиска – мета-эвристика, целью которой является перезапуск оптимизационного алгоритма, когда алгоритм не выходит из окрестности некоторого локального оптимума.

Предлагаемый оператор поиска является своего рода аналогом агента-критика [44]. Цель введения этого оператора состоит в том, чтобы по наблюдаемым данным об изменениях положения популяции и значений пригодности лучшего индивида, принимать решение о перезапуске алгоритма.

Суть преобразования заключалась в том, чтобы оценить, находится ли алгоритм в точке локального оптимума и происходит ли дальнейшее улучшение альтернатив, т.е. не претерпевает ли алгоритм стагнации. Для решения подобной задачи предлагается оценивать изменения пригодности лучшего найденного решения алгоритмом. Конечно, может быть рассмотрен и вариант оценки динамики изменения пригодности и по другим признакам, например, среднему положению популяции, что осуществляется в работах [4], [11] и [14]. Для

решения рассматриваемых в работе задач, данный подход не применим, о чем более подробно будет сказано ниже. Прирост эффективности при применении мета-эвристики рассматриваемого типа представлен в работе [27].

Таким образом, значение функции пригодности для лучшего найденного алгоритмом решения сохраняется для нескольких последних предшествующих популяций. Получается, что некоторый вектор заданной длины с координатами равными пригодностям на каждой популяции, отображает историю изменений пригодности лучшего индивида. На основании данной информации вводится критерий, по которому и происходит оценка – притянулся ли алгоритм к окрестности локального оптимума.

Дополнительно предлагается использовать накопленную в ходе работы алгоритма информацию о пространстве поиска. Так, если лучшее, найденное в текущем запуске алгоритма, решение содержится в окрестности, радиус которого является настройкой метода, ранее найденной точки поискового пространства, из которой был осуществлен перезапуск, то алгоритм автоматически перезапускается без проверки условия отсутствия прироста пригодности. Для реализации этого варианта, необходимо, чтобы при принятии решения о перезапуске при отсутствии изменений значений целевой функции для лучшего решения или если данное изменение является ниже некоторого порога, лучшее решение сохранялось в массив памяти. Таким образом, предполагаемый массив содержит в себе координаты лучших решений, при которых был осуществлен перезапуск. Получается, что при каждом изменении координат лучшей найденной альтернативы в данном запуске алгоритма, альтернатива сравнивается со всеми решениями из массива памяти и в случае их близости, т.е. если существует окрестность установленного радиуса, которой они принадлежат, то осуществляем перезапуск. Особенности работы предложенных операторов изложены ниже.

Учитывая оба предложенных варианта, схема оператора могут быть представлены следующим образом:

1. Пусть вектор $tail, |tail| = n_{tail} : tail_i = best_{N_I-i+1}, i = \overline{1, n_{tail}}$, где $best_{N_I-i+1}$ - лучшее решение, найденное алгоритмом за все популяции до $N_I - i$ популяции, где N_I - номер текущей популяции. Тогда, если пригодность последних n_{tail} лучших решений не меняются, т.е. $\max(tail) - \min(tail) < boarder_{tail}$, то перезапустить алгоритм. Величина $boarder_{tail} > 0$ является одной из настроек оператора, которой определяется, порог нечувствительности к изменению величины пригодности для перезапуска алгоритма.

2. Если условие пункта 1 было выполнено, то алгоритм перезапускается. При этом база данных лучших решений $database$ пополняется еще одной альтернативой $best_solution$, которая является лучшей альтернативой найденной в текущем запуске, и, таким образом, для которой и осуществился перезапуск. Значит, $database = database \cup best_solution$.

3. Если условие перезапуска, изложенное в пункте 1, не выполнилось, и при этом база решений пуста, т.е. $database \neq \emptyset$, то проверяем условия $\|best_solution - database_i\| < boarder_{best}, i = \overline{1, |database|}$, где $boarder_{best} > 0$ - еще одна настройка перезапуска. Т.е. если текущее лучшее решение попадает в окрестность одного из решений, при котором был осуществлен перезапуск, то алгоритм перезапускается. Здесь параметром $boarder_{best}$ определяется радиус окрестности, определяющий близость решений.

4. Опционально, может быть потребовано, чтобы при перезапуске менялись параметры закона распределения для стартовой популяции. Учитывая то, как разыгрывается начальная популяция в данной задаче, будем менять центр распределения нормального закона, т.е. математическое ожидание, $c_0^o = c_0^o + \xi_{restart}, \xi_{restart} \sim N(0, \sigma_{restart}^2)$. Здесь среднеквадратичное отклонение $\sigma_{restart}$ - параметр нового оператора, отвечающего за величину случайного шага для математического ожидания нормального закона распределения, по которому генерируются положения индивидов в стартовой популяции.

На рисунке 1.5 наглядно представлено действие одного из операторов перезапуска алгоритма. Здесь ограничимся рассмотрением задачи поиска на двумерном евклидовом пространстве. На некоторой ограниченной области пространства, точками обозначим уже найденные ранее решения задачи, которые привели к перезапуску. Теперь предположим, что лучшее решение найденное алгоритмом находится в одной из указанных окружностей. Окружность слева – окрестность, которая содержит уже найденные решения, справа – окрестность, которая не содержит таких решений. В данном случае, перезапуск осуществится для алгоритма слева, ввиду близости его лучшего решения к точке – т.е. ранее найденное решение попадает в окрестность текущего решения. В случае же алгоритма, лучшее решение которого находится справа, осуществляется переход к следующей популяции.

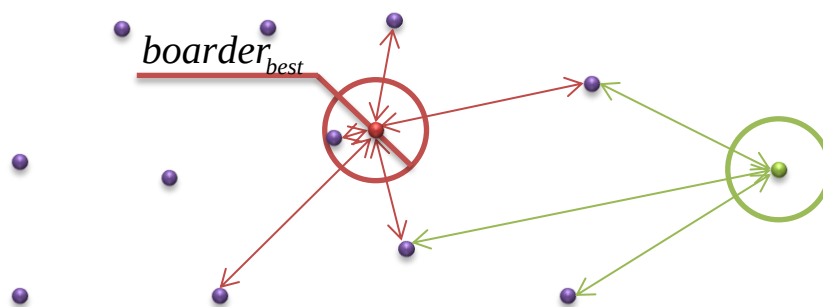


Рисунок 1.5. Пример перезапуска алгоритма, лучшее решение которого находится близко к ранее найденным. Если лучшее решение находится справа – продолжается работа алгоритма, слева – алгоритм перезапускается.

На рисунке 1.6 аналогичным образом представлена операция изменения математического ожидания для закона распределения индивидов в стартовой популяции. Здесь, штрих линией обозначен переход к точке, соответствующей новому математическому ожиданию генерирования стартовой популяции. Как видно, в текущей реализации оператора, центр распределения стартовой популяции является блуждающим. В данной работе было решено, через разыгрывание нормированной нормальной случайной величины, тогда $\sigma_{restart} = 1$.

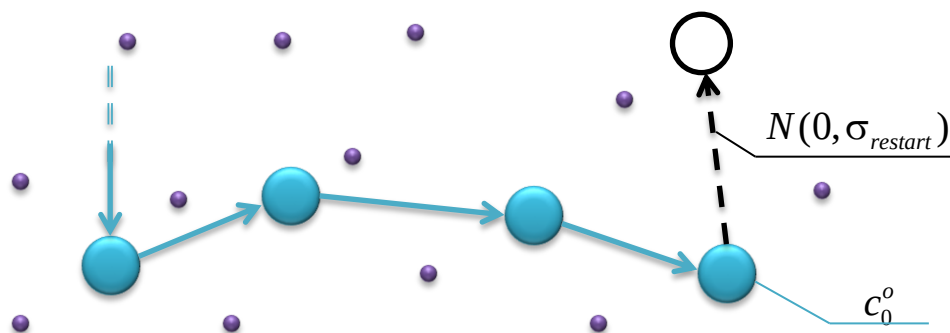


Рисунок 1.6. Наглядное представление изменения параметров распределения начальной популяции при каждом запуске алгоритма.

В следующем разделе рассмотрим некоторые другие стандартные алгоритмы оптимизации на Евклидовых пространствах. Рассмотренные алгоритмы будут сравнены с предложенным гибридным модифицированным алгоритмом. Ввиду наличия модификаций исследуемого алгоритма, которые ориентированы на повышение его эффективности в решении оптимизационных задач определенного класса, некорректным будет сравнение модифицированного алгоритма со стандартными алгоритмами, хоть и разработанными для решения задач вещественной оптимизации.

1. 4. Другие эвристические алгоритмы стохастического поиска экстремума.

В данном параграфе рассмотрим некоторые распространенные эвристические алгоритмы поиска экстремума функционала (1.1), эффективность которых подтверждена в различных работах по решению сложных задач глобальной оптимизации, [13] и [96]. Большинство современных эвристических алгоритмов берут в основе упрощенные модели каких-либо биологических или физических процессов. Таким образом, поиск основан на какой-то определенной идее, заимствованной из природы.

Алгоритмы, которые основаны на теории эволюции, называют эволюционными методами оптимизации. Данный класс методов включает в себя такие известные алгоритмы, как алгоритм эволюционных стратегий, [56], генетический алгоритм [15], [115], дифференциальную эволюцию, [62]. Каждый

из приведенных алгоритмов имеет множество модификаций для решения различных классов задач. Во многих работах встречаются различные изменения схемы поиска или операторов поиска. Существенно отличается от стандартного, например, метод эволюционных стратегий с адаптацией ковариационной матрицы (Evolutionary Strategies algorithm with Covariance Matrix Adaptation, CMA-ES), [18], который лежит в основе множества модификаций и новых эвристик, мета-эвристик, решения задач вещественной оптимизации.

Также развиваются, исследуются и широко применяются алгоритмы, которые основаны на моделях поведения групп живых организмов, [40], например, стайный алгоритм [24], алгоритм светлячков [67], пчелиный алгоритм, поисковый алгоритм кукушки, [23], алгоритм стаи волков и др.

Более подробно остановимся лишь на некоторых эвристических алгоритмах поиска экстремума. Прежде всего, рассмотрим метод дифференциальной эволюции (Differential Evolution, DE). Данный алгоритм относится к эволюционным алгоритмам и, таким образом, алгоритм принадлежит к семейству методов случайного поиска 0-ого порядка. В основе подхода лежит идея о том, чтобы возмущения генома или мутация для каждого индивида порождалась некоторым взаимодействием других индивидов. В данном случае, - линейной комбинацией случайно выбранных векторов - альтернатив, внутри одной популяции, [35].

Пусть, для простоты представления, каждый индивид в популяции алгоритма будет представлен кортежем, как для ЭС – (1.4),

$$id_i^2 = \langle op^i, fit(op^i) \rangle, i = \overline{1, N_p},$$

где

$op_j^i \in R, j = \overline{1, q}$ - альтернатива, представленная вектором вещественных параметров;

N_p - объем популяции;

$fit(\cdot)$ - функция пригодности, как и для алгоритма ЭС, (1.3).

Функция пригодности для эволюционных методов всегда интерпретируется как некоторая мера приспособленности особи. Основные свойства, которыми должна обладать функция пригодности, представлены выше. Здесь отметим, что для рассматриваемых алгоритмов и задач, целевая функция всегда есть отображение функции пригодности, которое было одинаковым для всех исследуемых алгоритмов. При ограниченном сверху и снизу множестве значений целевой функции, его всегда можно отобразить на множество $(0,1)$ или, если границы множества достижимы – то $[0,1)$, $(0,1]$ или $[0,1]$, сохраняя порядок.

Алгоритм дифференциальной эволюции управляется двумя параметрами: $C_r \in [0,1]$ и $F \in [0,2]$. Множества значений параметров указаны, согласно рекомендациям автора. На каждой итерации алгоритма, для каждой выбранной альтернативы случайным образом выбираются 3 различные альтернативы, которые отличаются от текущей альтернативы. В общем случае, число выбранных решений может быть увеличено, что рассматривается уже в более поздних работах, посвященных исследованиям этого алгоритма. Случайная величина равновероятно принимает значения из множества $N_r \in \{1, \dots, q\}$. Разыграем случайный вектор, размерностью равной размерности задачи, координаты которого распределены равномерно, $rand_i \sim U(0,1)$, $i = \overline{1, q}$. Далее, сформируем пробный вектор, для которого каждая координата определяется следующим соотношением:

$$op_i^{trial} = \begin{cases} op_i, & f_{de}(i) = 1, \\ op_i^a + F \cdot (op_i^b - op_i^c), & f_{de}(i) = 0, \end{cases} \quad (1.29)$$

где

$op_i^a, op_i^b, op_i^c, i = \overline{1, q}$ - переменные случайно выбранных из популяции альтернатив;

$$f_{de}(i) = \begin{cases} 1, & \text{if } rand_i < C_r \text{ or } i = N_r, \\ 0, & \text{if } rand_i > C_r \text{ and } i \neq N_r, \end{cases} \quad \text{- индикаторная функция.}$$

Если для пробного вектора выполняется неравенство $fit(op^{trial}) > fit(op^i)$ то в популяции текущий индивид заменяется на пробный, т.е. $op^i \leftarrow op^{trial}$.

Условие останова для рассмотренного алгоритма задается, как и для алгоритма ЭС, максимальным количеством вычислений целевой функции или приемлемым для лица, принимающего решения, значением исходного функционала. Представленный метод получил широкое распространение не только потому, что прост в реализации и управляется двумя параметрами, но и ввиду его высокой эффективности нахождения решений.

Следующим рассмотрим стайный оптимизационный алгоритм (Particle Swarm Optimization, PSO). Изначально предлагаемый метод был представлен в работе [24]. Суть стайных методов, в общем случае, состоит в представлении множества решений стайей, роем или другой взаимодействующей системой особей или групп особей, в смысле организованной группы живых организмов, преследующих определенную цель. Для PSO целью стаи является найти наиболее пригодную для обитания местность. При этом любые две точки пространства, на котором осуществляется поиск, могут быть сравнены между собой, т.е. координаты каждой точки местности отображаются на подмножество вещественных чисел. В данном случае, на множество значений целевой функции, а, значит, и функции пригодности.

Пусть каждая особь представлена кортежем

$$Id_i^3 = \langle op^i, v^i, fitness(op^i) \rangle, i = \overline{1, N_p},$$

где

$op_j^i \in R, j = \overline{1, q}$ - переменные экстремальной задачи, координаты места нахождения индивида;

$v_j^i \in R, j = \overline{1, q}$ - координаты вектора скорости движения индивида в пространстве;

N_p - объем популяции;

$fit(\cdot)$ - функция пригодности, как и для алгоритмов ЭС и DE, так же, см. (1.3).

В процессе работы алгоритма, каждым i -ым индивидом запоминаются координаты лучшего, по значению целевой функции, положения op_i^* , в котором он находился, начиная со стартовой популяции. Иными словами, каждому

индивиду соответствует точка в пространстве поиска, в которой пригодность индивида была максимальной. После каждого цикла алгоритма, который представлен ниже, каждый индивид сравнивает свое текущее положение с лучшим и, если текущее положение соответствует большему значению пригодности, то оно запоминается как лучшее.

Аналогично, всей текущей популяции соответствует лучшее положение в пространстве поиска, найденное за все время работы алгоритма среди всех индивидов op^{best} .

Внутри каждого цикла работы алгоритма для каждого индивида генерируются две случайные величины, распределенные равномерно, $r_1, r_2 \sim U(0,1)$, после чего рассчитывается скорость движения индивида и координаты нового местоположения $j = \overline{1, k}$:

$$v_j = \omega \cdot v_j + \varphi_1 \cdot r_1 \cdot (\hat{op}_j - op_j) + \varphi_2 \cdot r_2 \cdot (op_j^{best} - op_j), \quad (1.30)$$

$$op_j = op_j + v_j. \quad (1.31)$$

Условия останова алгоритма, как и генерирование стартовой популяции аналогичны для всех представленных здесь методов. Вектор скорости для каждого индивида в PSO является аналогом стратегического параметра, за тем исключением, что может принимать отрицательные значения.

Алгоритм CMA-ES является усовершенствованным алгоритмом ЭС, который был разработан для численного решения задач оптимизации для непрерывных нелинейных невыпуклых целевых функций. В данном алгоритме, процесс поиска решения связан с оценкой ковариационной матрицы, которая отражает некоторые свойства целевой функции. Оценка ковариационной матрицы находит свое применение для стохастических алгоритмов оптимизации и применяется по аналогии с оценкой матрицы Гессе для квазиньютоновских методов [102], [117].

Основные операции алгоритма, способ оценки ковариационной матрицы и теоретические обоснования эффективности CMA-ES представлены во множестве печатных работ, например, [18], [19], [20] и ввиду их объема не будут приводиться в работе.

Помимо адаптации стратегических параметров посредством ковариационной матрицы, алгоритм отличается еще тем, что каждая новая популяция генерируется не потомками, каждый из которых получается в результате скрещивания родителей и мутации, а генерируется вокруг некоторого вектора в пространстве решений, является нормальной случайной величиной с математическим ожиданием равным этому вектору и соответствующей ковариационной матрицей. Метод основан на оценке направлений эволюции: анизотропного и не анизотропного. Первое полностью основано на матрице ковариаций и существует для оценки наиболее удачных направлений, чтобы направить популяцию в эту сторону. Другая необходима для контроля шага движения популяции и разыгрывания дополнительных измерений для адекватной оценки матрицы.

Выводы

В данной главе были рассмотрены эвристические алгоритмы глобальной оптимизации, суть которых основывается на моделях различных природных явлений. Была показана возможность перехода от критериев задач оптимизации к функции пригодности. Показано, что для задач минимизации невязки, данный переход всегда осуществим. Выбранный способ представления функции пригодности легко интегрируется в общую схему решения и всегда является ограниченной сверху значением 1 для задач минимизации с целевой функцией ограниченной снизу значением 0, что характерно для множества обратных задач математического моделирования и управления.

Особое внимание уделяется алгоритму эволюционных стратегий, в принципе которого лежит модель естественной эволюции и на основе которого разрабатывались алгоритмы оптимизации для каждой из представленных в других главах задач. Именно по этой причине, было уделено особое внимание основным поисковым операторам алгоритма, особенностям каждого из предложенных преобразований и действий на пути к получению решения.

Оператор селекции был дополнен схемами выбора, заимствованными из теории стандартного генетического алгоритма и ставшими уже классическими. В качестве скрещивания рассматривались две обобщенные схемы, которые включают в себя и стандартные операторы скрещивания, и вновь предлагаемые.

Каждое оператор рассматриваемого алгоритма связан со случайными событиями. Это очевидно для основных операторов поиска, выполнения операторов скрещивания и мутации и характерно для локального спуска.

Алгоритм случайного покоординатного спуска, на каждом этапе которого, процесс выбора осуществляется случайным образом. Предложен также вариант в котором и шаг спуска является случайной величиной. Локальное улучшение изначально рассматривается как дополнение к стохастическому алгоритму, таким образом, в качестве входного воздействия на блок локального улучшения подается не решение, которое выступает в качестве начального приближения, а все множество решений – популяция. Предполагается именно такая реализация, потому что при ней сохраняется возможность настройки баланса между стохастическим поиском и уточнением решений.

Рассматривается модификация оператора мутации, который является основным оператором поиска алгоритма. Уход от стандартного представления мутации есть необходимая мера для повышения эффективности алгоритма в решении сложных, многомерных задач оптимизации, и в особенности, при высокой чувствительности функции к незначительным изменениям объективных параметров. Также предложен иной способ адаптации стратегических параметров.

Таким образом, был предложен гибридный модифицированный алгоритм эволюционных стратегий для решения сложных задач оптимизации. Исследуемый алгоритм представляет собой стохастический численный метод оптимизации, 0-ого порядка. Алгоритм относится к классу прямых методов поиска экстремума, следовательно, не требует аналитического вида целевой функции, ее дифференцируемости.

Далее, метод ЭС был обобщен на случай поиска экстремума функционалов смешанного типа, когда переменные поиска являются не только вещественными

числами, но и могут быть выражены числами натуральными, что свойственно задачам с ранговыми и номинальными переменными. Предложены операторы скрещивания и мутации, которые сохраняют парадигму эволюционных стратегий, т.е. каждому объективному параметру соответствует стратегический параметр, которым управляется интенсивность мутации.

Отметим, что большинство методов стохастического поиска, популяционных алгоритмов обладают большим количеством настроек, от которых зависит их эффективность. Конечно, с одной стороны, развитие алгоритмов и их совершенствование напрямую связано с проблемой разработки механизма выбора настроек, создания мета-эвристик. Но, тем не менее, для многих задач, которые успешно сводятся к экстремальным задачам, ввиду определенного способа представления альтернативы в виде набора разнородных переменных, даже наборы лучших настроек существующих методов не способны обеспечить приемлемую надежность нахождения решения задачи, по сравнению с алгоритмами поиска, которые учитывают особенности решаемой задачи. Поэтому для некоторых обратных задач математического моделирования и задач управления, актуальной является проблема разработки и совершенствования оптимизационных алгоритмов, которые бы учитывали особенности непосредственно определенного класса задач.

Отсюда важно исследование особенностей семейства функций, которые порождены сведением изначальной задачи к задаче оптимизационной. В таком случае, изучение этих особенностей, равно как и особенностей представления альтернативы может открыть новые пути совершенствования алгоритмов. Как правило, в таких случаях, даже если добавляются дополнительные настройки, то недостаточная изученность влияния совокупности параметров алгоритма не является препятствием для интегрирования совершенствований, ввиду значительного прироста эффективности алгоритма уже при некоторых выбранных настройках.

В работе рассматривается и мета-эвристика, в основе которой лежит идея о оценке возможности нахождения в стагнации или приближения к окрестности

стагнации и перезапуск алгоритма в каждом из этих случаев. Далее будет показано, что применение такой ме-та эвристики способно значительно повысить эффективность алгоритмов, даже в случае их ориентированности на конкретный класс задач.

В главе представлены и другие общеизвестные и признанные эффективными алгоритмы вещественной оптимизации. Для дальнейшего сравнения были выбраны наиболее популярные из таковых. Для каждого алгоритма представлена основная схема работы и показано, что в основе каждого из изложенных алгоритмов лежит своя парадигма поиска.

Далее перейдем к прикладным задачам глобальной оптимизации: к задаче идентификации линейных динамических систем, к прямому методу решения задачи терминального управления при кусочно-постоянном управлении, к непрямому методу решения задач оптимального управления. Будет показано, что решение каждой из представленных задач формируют свой класс оптимизационных задач, для каждого из которых были сделаны соответствующие модификации алгоритмов, которые значительно повышают их эффективность при нахождении решения приведенной задачи оптимизации.

ГЛАВА II. РЕШЕНИЕ ЗАДАЧИ ИДЕНТИФИКАЦИИ ЛИНЕЙНЫХ ДИНАМИЧЕСКИХ СИСТЕМ С ПОМОЩЬЮ МОДИФИЦИРОВАННОГО АЛГОРИТМА ЭВОЛЮЦИОННЫХ СТРАТЕГИЙ

В данной главе рассматривается задача идентификации динамического объекта по данным наблюдений его выхода, так называемая обратная задача математического моделирования, когда по исходным данным о входном и выходном воздействии строится математическая модель, описывающее взаимосвязь между ними. Под динамической системой понимается система, обладающая состояниями, для которых существует некоторый переход от одного состояния к другому, который и принято называть переходным процессом. Иными словами, динамической называется система, состояние которой в некоторый момент времени зависит от состояний системы в предыдущие моменты времени. Такие системы называют также системами с памятью. В общем случае, понятие динамической системы является лишь математической абстракцией, и такие системы представляются дифференциальными или разностными уравнениями.

Задача построения математической модели для динамических систем существует в различных постановках, которые полностью обусловлены целями моделирования. В данной работе рассматривается идентификация динамической системы линейным дифференциальным уравнением, при этом необходимо построить модель именно в символьном виде, т.е. определить линейное дифференциальное уравнение с известной правой частью, которое бы хорошо описывало выборочные данные. Математическое представление динамического процесса в аналитическом виде, а именно, в виде дифференциального уравнения обладает множеством преимуществ перед другими подходами к идентификации линейных динамических систем, поскольку такие модели широко применимы в дальнейшем исследовании изучаемой системы и решения различных задач имитационного моделирования, анализа свойств и управления.

2. 1. Задача идентификации линейных динамических систем: актуальность и подходы.

Задача идентификации динамических процессов была и остается актуальной по сегодняшний день, поскольку для любых задач принятия решений, достижения каких-либо целей невозможно без адекватной математической модели, на основании которой исследуются или оцениваются свойства системы, строятся прогнозы или ищется управление.

Множество реальных процессов являются динамическими: движение объекта в пространстве, финансовая математика, биология, медицина, изменение численности популяции бактерий, изменение концентрации продуктов химической реакции [45], [52], сигналы электрических и электро-магнитных систем, и т.п. Динамическим процессом можно считать и изменение биржевых показателей или измерения каких-либо характеристик процессов совершенно различной природы. Рост интереса к методам идентификации динамических систему говорит об актуальности задачи.

Во многом актуализация методов решения обратной задачи математического моделирования происходит ввиду развития инновационных технологий, робототехники, систем искусственного интеллекта, в частности систем распознавания мимики и жестов [43], производства современных, управляемых протезов, характеристик солнечных батарей, и многого другого [57], [94], [95], и [114]. Как уже было отмечено, реальные или разрабатываемые технологические процессы, равно, как и процессы иной природы, являются динамическими, и многие из них исследуются, но не имеют математических моделей. Так же, в некоторых случаях, динамические процессы представляют собой важные части некоторых сложных систем и определяют некоторые их свойства. Например, некоторыми классами дифференциальных уравнений описываются переходные процессы внутри скрытых Марковских цепей, которые, в свою очередь, используются уже для решения задач автоматической классификации. Подобное встречается в современных разработках диалоговых систем типа «человек - цифровой интерфейс» [66], [68]. С развитием методов интеллектуального анализа

данных, наборы линейных динамических систем используются для описания нелинейных динамических процессов, а именно, при построении нечетких аппроксимаций нелинейных динамических систем или регуляторов для таких систем [34]. Множество сложных задач классификации или идентификации, так или иначе, связаны с решением задачи моделирования или подбора дифференциальных уравнений, в том числе и линейных, например, задачи распознавания эмоций, речи [63], [66]. Более того, для некоторых систем, используют линеаризации для решения задач управления или анализа свойств систем в окрестности рабочего режима [71]. Все это указывает на актуальность задачи развития методов идентификации динамических систем, в особенности, восстановления математических моделей в символьном виде.

Современные работы часто связаны с идентификацией конкретных динамических процессов, при этом для решения возникающей экстремальной задачи используются различные математические представления. Например, в виде дифференциальных уравнений с дробными производными, в виде стохастических дифференциальных уравнений, [68]. Линейные дифференциальные уравнения используются в построении фильтров, идентификации артикуляции [43] и [68] – для стохастических дифференциальных уравнений, которые могут быть определены аналогичным способом. В некоторых работах рассматриваются алгоритмы идентификации параметров уравнения Бесселя или других отдельных дифференциальных уравнений с известной структурой. Идентификация стохастических дифференциальных уравнений применима для построения Марковских процессов, одна из таких задач рассмотрена в работе [29].

На многих зарубежных международных конференциях по теории управления, информатике, оптимизации и математическому моделированию (например, ICINCO'12, SIMULTECH'12, ICANGAA'13, ICINCO'13, ICINCO'14, ICINCO'16) встречаются работы, связанные с генерированием или идентификацией систем с переходными процессами между состояниями. Авторы различных работ указывали на то, что качество их методов и показатели разрабатываемых ими решений полностью или во многом зависят от

адекватности моделей динамики, на которых и основывается дальнейшее решение.

Большинство вновь разрабатываемых, инновационных решений, зачастую, не имеют математической модели, которая бы была выведена исходя из физической, химической или иной природы, ввиду большой сложности разрабатываемых решений или отсутствия теории, описывающих такие процессы. В таком случае, отсутствие знаний о процессе не позволяет решить прямую задачу математического моделирования. Некоторые существующие системы описываются идеализированными математическими моделями, которые не всегда успешно верифицируются. Причин может быть множество, укажем только некоторые из них: невозможность математически описать некоторые факторы; потеря адекватности после введения некоторых упрощений, использование идеализаций; построение модели при исследовании объекта в окрестностях некоторого рабочего режима и др. Сложность поставленной задачи, невозможность математически описать зависимости так, чтобы модель не противоречила данным эксперимента, так же свидетельствует о том, что задача идентификации систем является важной и актуальной.

При разработке методов решения обратных задач математического моделирования необходимо учитывать тот факт, что измерения выходной характеристики для некоторых реальных процессов являются крайне дорогостоящими, и, учитывая ограниченность бюджета, это делает невозможным набор достаточного количества данных для применения множества существующих методов. Задача построения модели динамических процессов при отсутствии прочей априорной информации представляет собой аккумулялирование знаний эксперта и использование их для подбора определенной структуры и оценки параметров модели. Здесь и далее, в случае линейных систем с одним выходным и одним выходным воздействием, под структурой нами будет пониматься порядок дифференциального уравнения, а под параметрами – его коэффициенты.

Уникальность каждой отдельной динамической системы требует для построения адекватной математической модели привлечения эксперта в данной области: автоматизации, биологии, медицины, информационных наук, кибернетики, физики или других наук. Таким образом, развитие автоматических методов идентификации линейных динамических систем, в общем случае, которому соответствуют условия малой и/или разреженной выборки, может значительно поспособствовать упрощению процесса построения математической модели и свести к минимуму человеческий фактор.

Заранее отметим, что использование алгоритмов полного перебора не является эффективным решением задачи только потому, что для каждой отдельной структуры, коэффициенты являются вещественными числами, что приводит к неоправданно большому количеству решений на некоторой дискретной сетке для каждого из возможных порядков дифференциального уравнения.

В данной работе рассматривается определенный класс динамических процессов – класс линейных динамических процессов. Линейными называются такие динамические процессы, для которых выполняется принцип суперпозиции: если X_1 есть реакция на u_1 , а X_2 есть реакция системы на u_2 , то реакция системы на воздействие $\lambda \cdot A \cdot u_1 + \mu \cdot B \cdot u_2$ есть $\lambda \cdot A \cdot X_1 + \mu \cdot B \cdot X_2$.

Далее будем рассматривать непрерывные системы, которые могут быть описаны дифференциальными уравнениями:

$$L(x(t), u(t), t) = 0 \text{ или } \frac{dx}{dt} = f(x, u, t), \quad (2.1)$$

где

$x(t) \in R^n$ - вектор состояния динамической системы, выход системы;

$u(t) \in R^m$ - вектор управлений, входное воздействие;

$t \in R / R^-$ - параметр времени, независимый аргумент;

$f(\cdot): R^n \times R^m \times R^+ \rightarrow R^n$ - непрерывная вектор-функция своих аргументов.

Рассматривается задача моделирования линейного динамического процесса, который является стационарным, т.е. может быть представлен как решение задачи Коши для автономной системы уравнений:

$$x'(t) = A \cdot x(t) + B \cdot u(t), \quad (2.2)$$

где

$A = (a_{ij})_{i=1, j=1}^{n,n}$, $a_{ij} \in R$, $i, j = \overline{1, n}$ - матрица системы;

$B = (b_{ij})_{i=1, j=1}^{n,m}$, $b_{ij} \in R$, $i = \overline{1, n}$, $j = \overline{1, m}$ - матрица правых частей системы;

при известных начальных условиях $x(t_0) = x^0$.

Необходимо отметить, что, в общем случае, дифференциальное уравнение или разностная схема есть только способы представления динамического процесса.

Таким образом, об объекте наблюдений известно, что он является линейным и динамическим (для некоторых задач мне предполагаем, что он объект может быть описан линейным дифференциальным уравнением), известен вектор начальных условий, функция управления, а в остальном, он представляет собой, так называемый, «черный ящик». При этом вектор начальных условий в данном случае предполагается известным, поскольку система наблюдается, начиная с момента времени, когда система находилась в установившемся состоянии, т.е. выходная характеристика была стабилизирована около некоторой величины. В таком случае, для некоторых реальных задач необходимо только оценить координату выходной характеристики системы, приняв остальные координаты равными 0.

Рассматриваемая схема функционирования объекта изображена на рисунке 2.1. Представленный объект является непрерывной системой. Обратим внимание, что предлагаемый подход применим и для дискретной по времени системы. Состояние системы, и управление, которое подается на объект, будем называть, соответственно, выходом и входом динамического объекта или объекта управления.

Конечно, в общем случае, представленном на рисунке 2.1, на систему воздействуют и наблюдаемые, но не контролируемые процессы $\mu(t)$, и некоторые неизвестные процессы $\zeta(t)$. В данной работе мы ограничимся задачей идентификации динамического объекта, на который воздействует только функция управления $u(t)$. Тем не менее, подход может быть обобщен и на задачи идентификации при наличии наблюдаемых, но не контролируемых входов.

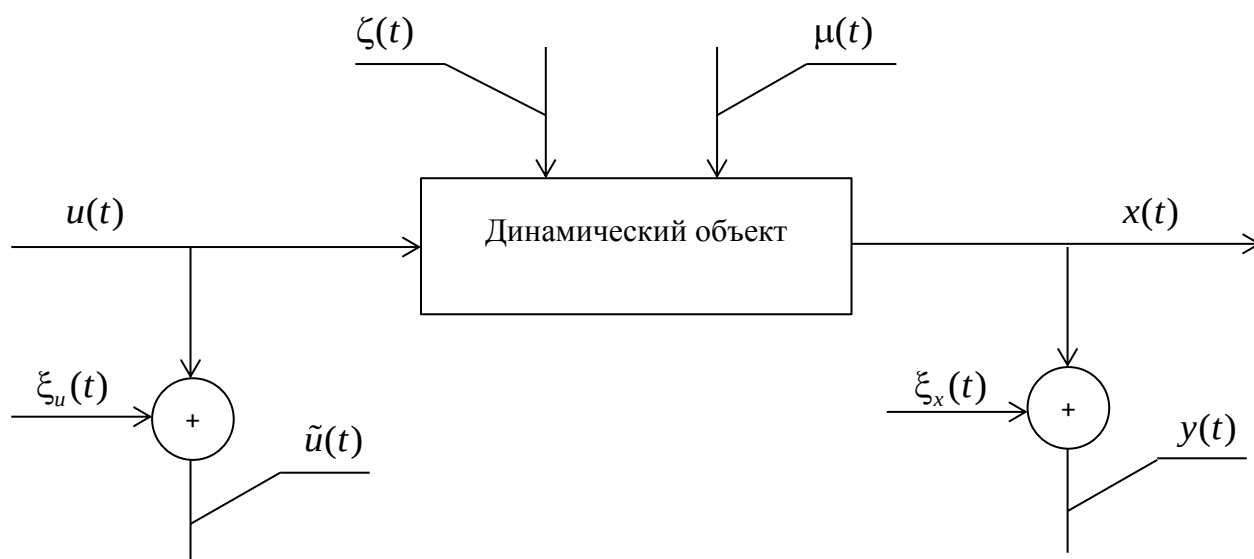


Рисунок 2.1. Динамический объект. Задача идентификации.

В общем случае, в каналах измерения входного воздействия и выхода системы действует помеха. Таким образом, измерения выхода $y(t)$ и входа $\tilde{u}(t)$ системы являются зашумленными, при этом $M(\xi_u) = 0, D(\xi_u) < \infty$ и $M(\xi_x) = 0, D(\xi_x) < \infty$. Здесь $M(\cdot)$ - первый начальный момент или математическое ожидание, $D(\cdot)$ - второй центральный момент или дисперсия. В данной работе рассматривается случай, когда функция управления полностью известна или вычисляема в любой точке интервала.

Конечно, подобные задачи исследовались и исследуются на протяжении длительного времени, выработано множество подходов к решению задач идентификации систем линейной динамики [76], [78], [94]. Далее рассмотрим некоторые из подходов, которые несколько отличаются в постановках задач для

подчеркивания различий в постановках и неприменимости известных методов для нахождения решения изложенной задачи.

Прежде всего, можно выделить методы определения переходной функции системы, по которой, в силу ее свойств, можно рассчитать реакцию на заданное управление или найти управление, необходимое для достижения определенной цели [28]. Если $h(t)$ - реакция линейной динамической системы на единичное

ступенчатое воздействие, функцию Хэвисайда $\eta(t) = \begin{cases} 0, & t \leq 0 \\ 1, & t > 0 \end{cases}$, тогда решение

системы можно определить через интегральную свертку:

$$x(t) = x^0 \cdot h(t) + \int_0^t x'_t(\tau) \cdot h(t - \tau) d\tau. \quad (2.3)$$

Множество методов основано именно на оценке переходной функции динамической системы. Конечно, на вход, как правило, невозможно подать функцию Хэвисайда, но учитывая принцип суперпозиции и используя выражение (2.3), можно численно оценить переходную функцию. Как видно, в такой постановке задачи, необходимым является знание функции управления, равно, как и вектор начальных координат.

Получается, что решение задачи идентификации будет полностью зависеть от адекватности модели выхода системы при известном управлении. Необходимо описать выход системы как некоторую функцию от времени. Большинство методов идентификации, основанные на моделировании переходной функции, базируются на решении задачи

$$\int_0^T (h(t) - h_m(\alpha, t))^2 dt \rightarrow \min_{h_m, \alpha}, \quad (2.4)$$

где

T - время наблюдения переходного процесса, начало наблюдения принято считать при $t = 0$;

$h_m(\cdot): R^p \times R \rightarrow R$ - некоторая функция, определенная с точностью до параметров;

$\alpha \in R^p$ - параметры или коэффициенты функции.

Критерий (2.4) для задачи идентификации по выборочным данным представлен ниже:

$$\sum_{i=1}^s (h_i - h_m(\alpha, t_i))^2 \rightarrow \min_{h_m, \alpha}, \quad (2.5)$$

где s - объем выборки, а $h_i = h(t_i), i = \overline{1, s}$ - измерения переходной функции.

Решение представленных задач, а именно (2.4), (2.5), может быть достигнуто с помощью методов непараметрической статистики или идентификации параметров [75]. Прибегая к оценке Надарая-Ватсона [101], можно оценить переходную функцию, после чего построить модель процесса. Подробнее алгоритм непараметрической идентификации представлен в работе [33].

Тем не менее, методы непараметрической оценки зависимостей, ввиду отсутствия явной структуры, не рассчитаны на работу с малыми объемами данных, поскольку адекватность модели во многом зависит от свойств выборки, в частности от степени разреженности и от объема. Для таких выборок, применение непараметрических подходов возможно после предобработки данных.

Для восстановления функции, описывающей реакцию системы на известное управляющее воздействие применимы другие интеллектуальные методы восстановления зависимостей и аппроксимации, например, искусственные нейронные сети [22], [23], [84], [120], или системы на нечеткой логике, [34], [86]. Использование подобных подходов предполагает либо наличие определенных знаний о построении искусственных нейронных сетей, систем нечеткого вывода, либо применение некоторого алгоритма автоматического подбора подобных системы для известных выборочных данных. Подобные алгоритмы рассматриваются в [7], [113] или более обще в [111], [112]. Как правило, рассматриваемые подходы требуют применения эффективных оптимизационных алгоритмов глобальной оптимизации или алгоритмов перебора и не всегда гарантируют результат, ввиду сложности подбора структуры. В настоящее время активно развиваются методы построения ансамблей интеллектуальных технологий, которые представляют собой совокупность различных моделей исследуемого объекта или совокупности различных интеллектуальных агентов,

которые скооперированы и взаимодействуют согласно некоторой парадигме для эффективного решения задачи идентификации. Подобные подходы представлены в работах [94], [95].

Восстановление функции перехода возможно так же и посредством применения эволюционных методов, таких как генетическое программирование [8], [98], [113]. В работах решение дифференциального уравнения, т.е. реакция динамической системы на известное воздействие, аппроксимируется функцией. Применение генетического программирования позволяет находить решение в аналитическом виде, но решение, часто, является громоздким. Модели функции выхода динамической системы, полученные таким образом, часто, являются неудобными для дальнейшей работы. То же относится и к моделям, полученным посредством нейросетевого моделирования или аппроксимацией системами нечеткого вывода.

Все рассмотренные выше подходы позволяют получить модель не динамического процесса, но только его реакции, а сама модель определяется уже из соотношения (2.4). Понятно, что переходный процесс ограничен временем наблюдения, таким образом, полученная модель является адекватной только на том временном отрезке, на котором наблюдался объект. Что также ограничивает рассмотренные методы идентификации.

Модель, полученная в виде функции или аппроксимации функции, неприменима, в смысле классических теорий, для решения задач оптимального управления, регулирования, изучения динамических свойств системы в некоторых случаях и прогноза реакции системы на другое управляющее воздействие. Дальнейшее использование таких моделей требует применения численных методов интегрирования к найденной аппроксимации. Если начальное положение системы является ненулевым, что характерно для реальных задач математического моделирования, то усложняется процесс идентификации переходной функции.

Для случая, когда входное воздействие тоже представлено зашумленной выборкой, применимы и другие методы идентификации, которые, правда,

требуют, чтобы реакция наблюдалась на различные входные воздействия. В таких случаях возможно применение соотношения Винера-Хопфа, [94]. Конечно, статистический метод сильно чувствителен к объему измерений и количеству запусков измерений, но он позволяет находить модель и в случае, когда входное воздействие является наблюдаемым. В данном исследовании такая постановка задачи не рассматривалась.

Нахождение моделей в аналитическом виде, в виде дифференциальных уравнений второго порядка и в том случае, если есть математическая модель более высокого порядка, представлена в работах [36] и [37]. Более современные методы решения задачи идентификации позволяют находить более точное решение и для этой задачи.

Стоит отметить, что во многих работах осуществлялся поиск модели динамического процесса в виде дифференциального уравнения, в том числе и линейного, но для случаев, когда известна структура дифференциального уравнения. Для решения таких задач идентификации широко применяются методы глобального поиска минимума ошибки между наблюдаемыми данными и моделью по значениям параметров модели. Впервые генетический алгоритм был использован для решения задачи идентификации параметров динамической модели с известной структурой в работе [58]. Так, в работе [33] рассматривается задача идентификации линейных динамических систем, обратная задача математического моделирования, где по данным наблюдений, но в отсутствии помех, строится модель в виде дифференциального уравнения второго порядка. Коэффициенты уравнения подбираются посредством решения приведенной задачи оптимизации генетическим алгоритмом. Авторы демонстрируют высокую эффективность предложенного подхода для решения реальных задач моделирования. Аналогичная задача решается в работах [12], [32] и [42], где в первой работе используется так же генетический алгоритм, а во второй и третьей применяются другие современные эвристические алгоритмы оптимизации, основанные на биологических идеях поведения стай или эволюционных идеях.

Конечно, существует множество различных методов решения задачи оценивания параметров линейных дифференциальных уравнений, например, работы [2], [3], [78], [79] и [80]. Видно, что даже при известной структуре дифференциального уравнения, такие задачи приводятся к многоэкстремальным и сложным целевым функциям.

В данной работе предложен подход к оцениванию коэффициентов дифференциального уравнения с помощью авторского алгоритма глобальной оптимизации, но при одновременной настройке порядка дифференциального уравнения, подбора его структуры, для системы один вход – один выход (single input – single output, SISO).

Таким образом, разработан подход, в котором решается обратная задача математического моделирования в символьном виде, т.е. мы получаем решение в виде дифференциального уравнения определенного порядка и определенными коэффициентами, которое максимально хорошо описывает облако точек – измерения выхода реального объекта.

Здесь подробнее остановимся на вопросе численного решения задачи Коши для дифференциальных уравнений при известном управлении.

Определим n -мерную вектор-функцию $\tilde{f}(x, t)$, представляющую собой функцию системы (2.1) или, в данном случае (2.2), где управление представим как функцию от времени, т.е.

$$\tilde{f}(x, t) = f(x, u, t) \Rightarrow x' = \tilde{f}(x, t).$$

Функция $\tilde{f}(x, t)$ определена в цилиндре $Z = D_x \times I_t$, $I_t = [t_0, \infty)$, а D_x - открытая область в R^n . Будем предполагать, что вектор-функция $\tilde{f}(x, t)$ непрерывна по t и имеет ограниченные первые производные по остальным аргументам. При этих условиях справедлива теорема Коши [74], [77]: каждой системе значений $(t_0, x^0) \in Z$ соответствует единственное решение $x = x(t, t_0, x^0)$, определенное в некотором интервале I_t , т.е. $\exists a \in R^+ : (t_0, t_0 + a) \in I_t$, и удовлетворяющее начальному условию $x(t_0, t_0, x^0) = x^0$.

Для дальнейшей работы с динамическими системами необходимо определиться с методами решения задачи Коши. Учитывая наличие функции управления в правой части уравнений и то, что уравнения рассматриваются различного порядка, гораздо удобнее использовать численные методы, метод Рунге-Кутты 4-го порядка точности [74], [77]:

$$x_{k+1} = x_k + \frac{1}{6} \cdot (k_1 + 2 \cdot k_2 + 2 \cdot k_3 + k_4); \quad (2.6)$$

$$k_1 = h_{ode} \cdot \tilde{f}(x_k, t_k); \quad (2.7)$$

$$k_2 = h_{ode} \cdot \tilde{f}\left(x_k + \frac{k_1}{2}, t_k + \frac{h_{ode}}{2}\right); \quad (2.8)$$

$$k_3 = h_{ode} \cdot \tilde{f}\left(x_k + \frac{k_2}{2}, t_k + \frac{h_{ode}}{2}\right); \quad (2.9)$$

$$k_4 = h_{ode} \cdot \tilde{f}(x_k + k_3, t_k + h_{ode}). \quad (2.10)$$

Здесь решение задается на равномерной сетке с шагом h_{ode} , $k = \overline{1, N_{ode}}$, где $N_{ode} = \frac{T}{h_{ode}}$, а T - конечное время интегрирования, $x_k = x(k \cdot h_{ode})$.

2. 2. Постановка задачи идентификации и сведение ее к экстремальной задаче.

Пусть выход динамического объекта представлен измерениями, формирующими выборку объема s , то есть $\{y_i, t_i\}, i = \overline{1, s}$, где $y_i \in R$ - измерение выхода динамической системы в момент времени $t_i \in [0, +\infty)$, $u = u(t)$ - известное управляющее воздействие, которое является входом динамической системы. Известно также, что система является линейной и описывается линейным дифференциальным уравнением вида и считаем известным начальное условие уравнения:

$$a_k \cdot x^{(k)} + a_{k-1} \cdot x^{(k-1)} + \dots + a_0 \cdot x = b \cdot u(t), \quad (2.11)$$

$$x(0) = x_0.$$

Необходимо по данным выборки определить параметры системы и порядок n дифференциального уравнения, который мы будем считать ограниченным, т.е. $n-1 \leq M, M \in N$. Предполагается, что в канале измерения выхода системы действует симметрично распределенная аддитивная помеха $\xi: M(\xi) = 0, D(\xi) < \infty$, т.е.

$$y_i = x(t_i) + \xi_i.$$

В общем случае, распределение случайной величины неизвестно, в данной работе необходимые условия, накладываемые на случайную величину, указаны выше: величина центрирована и ограничена. В работе рассматривалась равномерно распределенная аддитивная помеха.

Получается, что при неизвестном порядке системы мы решаем задачу структурно-параметрической идентификации, причем задача будет частично параметризованной, поскольку максимальную степень входящей в уравнение производной мы определяем заранее, ограничивая ею размерность поискового пространства. Впервые такая постановка рассматривалась в работе [90], и была развита дальше в работе [54].

Будем считать, что для системы любого порядка ее коэффициент при старшей степени равен 1, таким образом

$$x^{(k)} + \frac{a_{k-1}}{a_k} \cdot x^{(k-1)} + \dots + \frac{a_0}{a_k} \cdot x = \frac{b}{a_k} \cdot u(t),$$

или

$$x^{(k)} + \tilde{a}_k \cdot x^{(k-1)} + \dots + \tilde{a}_1 \cdot x = \tilde{b} \cdot u(t).$$

Тогда, решение задачи идентификации мы будем искать как дифференциальное уравнение порядка $m \leq M, M \in N$, такое, что решение задачи Коши, при заданных начальных условиях,

$$\begin{aligned} \hat{x}^{(m)} + \hat{a}_m \cdot \hat{x}^{(m-1)} + \dots + \hat{a}_1 \cdot \hat{x} &= \hat{a}_0 \cdot u(t), \\ \hat{x}(0) &= x_0, \end{aligned} \tag{2.12}$$

с параметрами $\hat{a} = (0 \ \dots \ 0 \ \hat{a}_m \ \dots \ \hat{a}_1 \ \hat{a}_0)^T \in R^n$, т.е. $n = M + 1$, доставляют экстремум выбранной функции,

$$I_1(a) = \sum_{i=1}^N |y_i - \hat{x}(t_i)| \Big|_{\hat{a}=a} \rightarrow \min_{a \in R^n}, \quad (2.13)$$

$$I_2(a) = \max_i |y_i - \hat{x}(t_i)| \Big|_{\hat{a}=a} \rightarrow \min_{a \in R^n}. \quad (2.14)$$

Для оценки модели процесса (2.1) в виде (2.12) как решение одной из задачи на поиск экстремума: (2.13) или (2.14), необходимо обладать информацией о начальном положении системы так, чтобы могла быть решена задача Коши. В общем случае, вектор начального положения системы, если он не является известным изначально, может быть численно оценен, что, конечно, не всегда возможно и во многом зависит от свойств выборки. Другим вариантом определения начального положения системы является включение вектора в задачу оптимизации. Как правило, для многих задач процесс наблюдения начинается с установившегося режима, следовательно, все координаты начального положения для динамической системы равны 0, кроме положения выхода системы.

Задача идентификации параметров линейной динамической системы известного порядка и начальных условий задачи Коши, представлена в работе [50]. Обобщение этой задачи на случай неизвестного порядка дифференциального уравнения представлено в работе [51].

Функция управления для многих систем является известной, а в том случае, когда управление является некоторым процессом, аналитический вид которого неизвестен, необходимо аппроксимировать функцию по данным наблюдений. Рассмотренный в работе подход может быть обобщен на решение задачи идентификации при неизвестном управлении, что приведет, соответственно, к усложнению задачи.

Подобные критерии, основанные на минимизации невязок между выборочными данными и выходами системы, могут быть использованы для идентификации элементов матриц системы дифференциальных уравнений, которая подбирается как модель процесса с несколькими зависимыми выходами.

Итак, пусть необходимо построить математическую модель динамического процесса, которую для удобства представим в матричном виде - (1.2),

$$\tilde{x}' = \hat{A} \cdot \tilde{x}(t) + \hat{B} \cdot u(t), \quad x(0) = x_0, \quad (2.15)$$

где

$\hat{A} = (\hat{a}_{ij})_{i=1, j=1}^{n,n}$ - матрица системы линейных дифференциальных уравнений;

$\hat{B} = (\hat{b}_{ij})_{i=1, j=1}^{n,m}$ - матрица правых частей, коэффициенты управления;

$\tilde{x}(t) \in R^n$ - модель состояния системы;

$u(t) \in R^m$ - управляющие воздействия, представленные в виде вектор-функции.

Ввиду того, что наблюдается несколько различных выходов системы, которые могут отличаться по амплитуде отклика, необходимо нормировать каждый отдельно взятый критерий. Для этого определим диаметр множества измерений для каждого наблюдаемого выхода при включении в это множество начального положения выхода. Тогда, критерий приобретает следующий вид:

$$I(a) = \sum_{j=1}^{N_o} \frac{\sum_{i=1}^{s_j} |y_i^j - \hat{x}^j(t_i^j)|}{\sup(|a - b| : a, b \in Y^j \bigcup x_0^j)} \bigg|_{\hat{A}=A, \hat{B}=B} \rightarrow \min_{A, B}, \quad (2.16)$$

где

N_o - число выходов динамической системы;

$s_j, j = \overline{1, N_o}$ - объем выборки для каждого выхода динамической системы;

$y_i^j, i = \overline{1, s_j}, j = \overline{1, N_o}$ - измерения выходов, образующие выборки;

$t_i^j, i = \overline{1, s_j}, j = \overline{1, N_o}$ - времена измерений для каждого j -ого выхода;

$\sup(|a - b| : a, b \in Y^j \bigcup x_0^j)$ - диаметр множества данных измерений для каждого выхода;

$\hat{x}^j(t) \big|_{\hat{A}=A, \hat{B}=B}$ - j -ый выход модели при матрицах A, B .

Предложенный подход представлен в работах [45], [46].

Критерий (2.16) является аналогичным критерием для задачи с одним входом и одним выходом, как (2.13). Не трудно видеть, что критерий (2.14) может быть так же обобщен на случай систем с несколькими входами и выходами.

Таким образом, задача идентификации динамического объекта была приведена к задаче поиска экстремума на пространстве векторов с вещественными координатами. При этом особенность представления структуры объекта приводит к сложному поведению целевой функции в окрестности некоторых точек пространства, для которых первые координаты вектора обращаются в 0.

2. 3. Повышение эффективности алгоритма эволюционных стратегий для решения задачи идентификации ЛДС.

Итак, в дополнение к тому, что было сказано выше, каждая целевая функция: (2.13), (2.14) или (2.16), является многоэкстремальной, сложной и, в общем случае, не задана аналитически, не дифференцируема по своим переменным. Переменные, которые соответствуют коэффициентам дифференциального уравнения, представлены вещественными числами и заранее неизвестно какому интервалу принадлежит каждый коэффициент. Более того, представление решения для задач (2.13) и (2.14) так же значительно повышает сложность решаемой задачи. Ввиду всего вышеизложенного был выбран алгоритм решения задачи глобальной оптимизации для вещественных переменных – алгоритм ЭС. Ниже будут представлены и обоснованы модификации алгоритма, которые обеспечивают значительное повышение эффективности нахождения решения. Модификации алгоритма представлены в работах [48], [55], [109].

Ввиду отсутствия априорной информации о коэффициентах и порядке дифференциального уравнения использование бинарного представления переменных оптимизации становится затруднительным и неэффективным в смысле нахождения решения.

Согласно принятому переходу от вектора, т.е. индивида, к дифференциальному уравнению, вектор, ввиду особенности выбранного представления решения содержит в себе информацию о порядке, структуре и коэффициентах дифференциального уравнения, что необходимо учитывать для совершенствования работы алгоритма.

Коротко рассмотрим процесс определения порядка дифференциального уравнения по виду вектора решений. Пусть $\hat{a}$ - вектор, содержащий решение задачи, тогда $i_{order} \in N, i_{order} \leq M, i_{order} : \hat{a}_{i_{order}} \neq 0, \hat{a}_i = 0, i > i_{order}$. Если $\hat{a}_M \neq 0 \Rightarrow i_{order} = M$. Тогда порядок уравнения будет определяться индексом i_{order} . Учитывая предложенный здесь подход к определению порядка дифференциального уравнения, заметим, что важно, чтобы у алгоритма решения задачи была возможность сохранять некоторые координаты равными 0.

В связи с этим, одной из специальных модификаций алгоритма было введение операции округления координат векторов:

$$op_j^i = round(op_j^i), j = \overline{1, n}, i = \overline{1, N_I}, \quad (2.17)$$

где $round(\cdot): R \rightarrow Z$ - функция, округляющая число до его ближайшего целого. Такой оператор, воздействующий на объективные параметры алгоритма, решает задачу приведения координат вектора к целым числам. Поскольку для представления структуры системы важно, чтобы в некоторых случаях определенное количество координат решения подряд обращаются в нуль, а стохастический поисковый алгоритм возмущает переменные ввиду природы операции мутации, нужен оператор, который бы сохранял найденный порядок.

Оператор округления применяется непосредственно после оператора мутации и после округления происходит локальное улучшение вновь полученной популяции.

По этой же причине был модифицирован оператор мутации. О модификации оператора мутации говорилось в Главе 1, таким образом, вероятность мутации для каждой пары типа объективный - стратегический параметр, $p_m = \frac{1}{q}$. Тогда,

случайные возмущения не будут приводить к сильному разбросу индивидов последующей популяции вокруг некоторого найденного решения, который не устраняется при последующем локальном улучшении альтернативы после округления.

Уделим особое внимание вопросу генерирования начальной популяции. Поскольку, случайное разыгрывание коэффициентов для стартовой популяции не приведет к возможности появления в популяции различных решений, которые бы соответствовали уравнениям различного порядка. Учитывая подобное представление решений, будем генерировать популяцию следующим образом:

1. Для каждого индивида с вероятностью $\frac{1}{M}$ выбирается порядок дифференциального уравнения.
2. Для выбранного порядка i_{order} каждая ненулевая координата решения разыгрывается равномерно на интервале $[-5, 5]$.
3. Все стратегические параметры индивида разыгрываются равномерно на интервале $[0, 1]$.

Предложенная схема была выбрана, как лучшая, путем перебора различных вариантов начального генерирования решений. Среди вариантов генерирования начальной популяции рассматривались следующие: каждый коэффициент есть значение разыгранной случайной величины, распределенной нормально с нулевым математическим ожиданием; популяция заполнялась альтернативами, содержащими решения с порядком $M/2$, коэффициенты разыгрываются случайно, согласно равномерному закону распределения; популяция заполнялась уравнениями порядка 1.

Необходимо учитывать некоторые особенности локального улучшения альтернатив предложенным алгоритмом случайного покоординатного спуска. Оператор округления (2.17) каждой координаты приводит к тому, что теряется точность решения, ввиду отсечения мантиссы. Иными словами, огрубляются альтернативы. Для того чтобы компенсировать потери точности и повысить

эффективность алгоритма в целом, потребуем, чтобы покоординатный спуск осуществлял такое количество шагов, что при выбранной длине шага, округленный коэффициент мог быть уточнен так, что возвращалось значение, из которое предшествовало целому.

Поскольку рассматривается несколько различных схем поиска решений экстремальной задачи, необходимо исследовать каждую из них в отдельности для определения наиболее эффективной. При этом, каждая схема имеет общие настройки: выбор оператора поиска. Здесь, и далее, проверяются все типы селекции: (1.7), (1.8) и (1.9); представленные в первой главе типы рекомбинации: (1.10)-(1.11), (1.12)-(1.13), при (1.14), (1.15), (1.16) и (1.14), (1.17), соответственно. Рассматриваются различные схемы мутации: (1.18)-(1.19), (1.18) и (1.20), (1.18) и (1.21), (1.18) и (1.22), модифицированная мутация (1.23) и (1.24), (1.23) и (1.25), (1.23) и (1.26). Алгоритмы с классической мутацией и модифицированной будут рассматриваться как алгоритмы с различными схемам.

Для оценки эффективных настроек алгоритма 100 систем были случайно сгенерированы: по 10 систем на каждый порядок дифференциального уравнения, с 1 по 10. Параметры каждой системы разыгрывались следующим образом: $\hat{a}_k^i \sim U(-5,5)$, $\hat{b}_k \sim U(-5,5)$, $i = \overline{1,10}$, $k = \overline{1,i}$. Время функционирования системы было выбрано равным 5. Функция управления для всех рассматриваемых задач была выбрана единичной функцией, т.е. $u(t)=1$. Выборочные данные отбираются из численного решения дифференциального уравнения. Пусть $\{x_i, t_i\}$, $i = \overline{1, T/h_{ode}}$ есть численное решение системы. Тогда для заданного объема выборки $s < T/h_{ode}$, $s=100$, выберем s различных точек случайно из численного решения дифференциального уравнения.

Для того чтобы оценить эффективность параметров оптимизационного алгоритма была рассмотрена идентификация без возмущений в каналах измерений, чтобы данный фактор не вносил дополнительной сложности в задачу и можно было оценить какие находятся решения. По этой же причине объем выборки был выбран достаточно большим, чтобы можно было говорить о ее

репрезентативности, т.е. что выборочными данными отображаются все особенности переходного процесса.

Для каждой отдельной системы осуществлялось по 20 запусков алгоритма с определенными настройками. Все начальные условия для данных задач были приняты равными нулю.

Выше приведены ссылки на различные типы операторов поиска для метода ЭС. Обозначим, какие параметры, если они предполагаются, для каждого типа операторов рассматривались. Алгоритм был тестирован на решении задач идентификации для сгенерированной выборки решений при различных типах селекции: пропорциональная, ранговая, турнирная (объема 2, объема 10%, объема 20%); рекомбинации: средняя, средневзвешенная, дискретная, случайная; при различных вероятностях мутации $p_m \in \left\{ \frac{1}{11}, \frac{5}{11}, \frac{1}{5}, 1 \right\}$, и ее типах: адаптивная, стандартная. Чтобы определить наиболее эффективную комбинацию настроек для каждой отдельной рассматриваемой схемы необходимо все прочие параметры поиска оставить одинаковыми. Объем популяции был выбран равным 50, число популяций – 50, параметры для локального спуска $N_1 = 50$, $N_2 = 50$ и $N_3 = 1$ при $h_l = 0.05$.

Конечно, при других параметрах, например, локального спуска, иная схема нахождения решения могла быть более эффективной. Тем не менее, невозможно найти лучшую в среднем настройку для решения всех возможных задач идентификации ЛДС, можно только ценить эффективность каждой настройки для определенного набора задач и сделать выводы о работоспособности подхода в целом.

Как уже было замечено выше, сравнивались несколько различных оптимизационных алгоритмов решения поставленной экстремальной задачи (2.13). Среди них: алгоритм эволюционных стратегий (ES); гибридный алгоритм эволюционных стратегий (HES); гибридный алгоритм эволюционных стратегий с модифицированными операциями селекции, мутации и рекомбинации (HES+m.mutation); гибридный алгоритм эволюционных стратегий с операцией

округления (HES+rounding); гибридный модифицированный алгоритм эволюционных стратегий с округлением (HMES). При этом количество вычислений целевой функции для ES было выбрано равным числу измерений целевой функции для других алгоритмов, в циклах которых использовалось локального улучшения.

Для решения задачи идентификации различными алгоритмами по данным наблюдений была создана программа, код которой написан на языке C++, [108]. Программа позволяет решать задачу идентификации для загруженных из файла данных. Данные в файле должны иметь определенную структуру и представлять измерениями выход объекта в заданные моменты времени, при этом данные должны быть предварительно обработаны так, чтобы начальные условия были нулевыми. Впоследствии, в ходе дальнейших исследований была разработана клиент-серверная система параллельных алгоритмов идентификации для уравнений различного порядка, [104].

Таким образом, как уже отмечалось выше, для каждой схемы и для каждой задачи алгоритм запускался 20 раз с фиксированным набором настроек для усреднения результатов. На рисунке 2.2 представлен график, на котором обозначены средние значения пригодности лучших решений, найденных внутри каждой схемы с лучшими в среднем настройками. Видно, что алгоритм, включающий все предложенные схемы модификаций, значительно превосходит по эффективности все остальные алгоритмы. Более того, каждый из предложенных алгоритмов значительно превосходит стандартный ЭС.

Значение критерия (2.13) соответствует степени близости модели к наблюдаемым данным и чем критерий ближе к 0, а, следовательно, функция пригодности – к 1, тем лучше модель описывает функцию перехода на данном промежутке. Тем не менее, значение критерия не является единственной мерой определения эффективности алгоритма, потому что определить систему, которая хорошо описывает облако точек, еще не значит найти лучшую из возможных структур и истинные параметры.

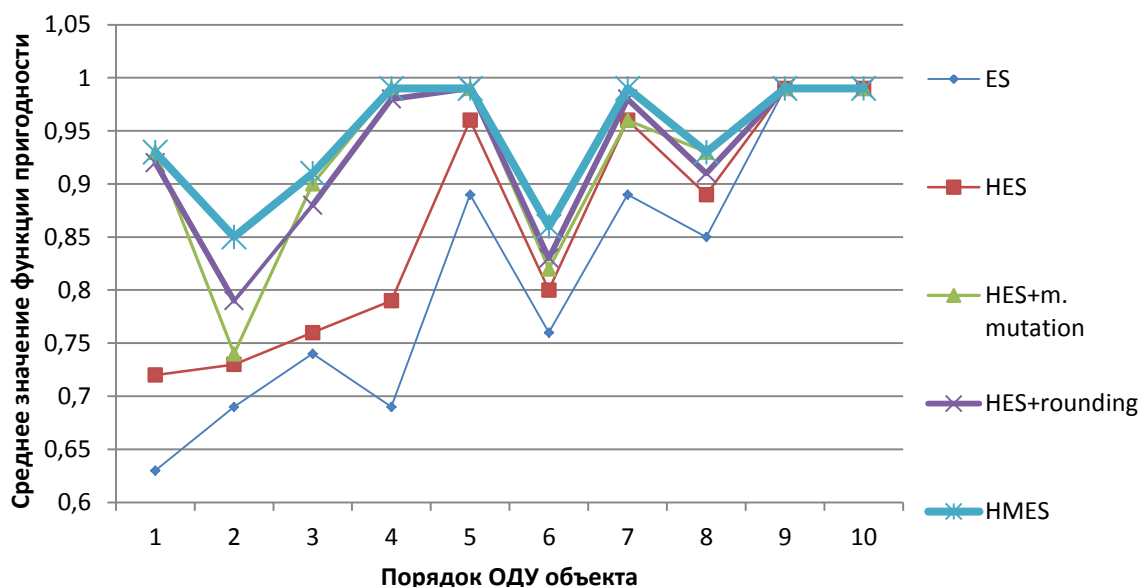


Рисунок 2.2. График сравнения средней пригодности для различных алгоритмов решения задачи идентификации ЛДС.

При вычислении значения критерия (2.13) для каждой альтернативы, по значениям координат вектора-решения определялся порядок дифференциального уравнения, а остальные координаты вектора, отличные от нуля, определяли коэффициенты. Дифференциальное уравнение численно интегрировалось с помощью метода Рунге-Кутты 4-го порядка точности: (2.6)-(2.10), после чего решение сравнивалось в выборочных точках с выборочными значениями.

Исследование эффективности схем показало, что наиболее надежным алгоритмом является гибридный модифицированный алгоритм эволюционных стратегий с операцией округления и модифицированной мутацией. Настройки алгоритма, лучшие в среднем для данной схемы были определены следующими: турнирная селекция с объемом турнира 10%, дискретное скрещивание, модифицированная мутация, реализованная через операцию сложения при вероятностях $p_m = \frac{1}{11}$, $p_c = 0.8$ и $N_1 = 50$, $N_2 = 50$ и $N_3 = 1$ при $h_l = 0.05$.

Как видно на рисунке 2.2, средняя пригодность возрастает с приближением порядка реального объекта к установленному параметру-ограничению максимального порядка модели. Получается, что алгоритмы должны работать

таким образом, чтобы сохранять возможность снижать порядок системы, сохраняя равенство первых координат 0. По этой причине модификация мутации или подключение оператора округления приводят к существенному улучшению. И эти модификации оказываются сравнительно более эффективными, чем только подключение локального улучшения множества решений, что явно указывает на правильную интерпретацию сложности задачи. При этом важно отметить, что рост пригодности связан так же и с тем, что за время наблюдений система более высокого порядка ведет себя так, что проще построить ее модель, чем при аналогичных условиях – модель системы более низкого порядка. Ниже представлен пример, демонстрирующий это.

На рисунках 2.3 и 2.4 представленных ниже приведены два различных примера решения задачи идентификации для динамических систем по данным наблюдений. На рисунке 2.3 изображен случай, когда система хорошо описывает облако точек и модель практически полностью совпадает с истинным объектом. Объект и модель описываются дифференциальными уравнениями, соответственно,

$$x''' + x'' + 2 \cdot x' + x = 1(t),$$

$$\hat{x}''' + \hat{x}'' + 2.1 \cdot \hat{x}' + \hat{x} = 1(t).$$

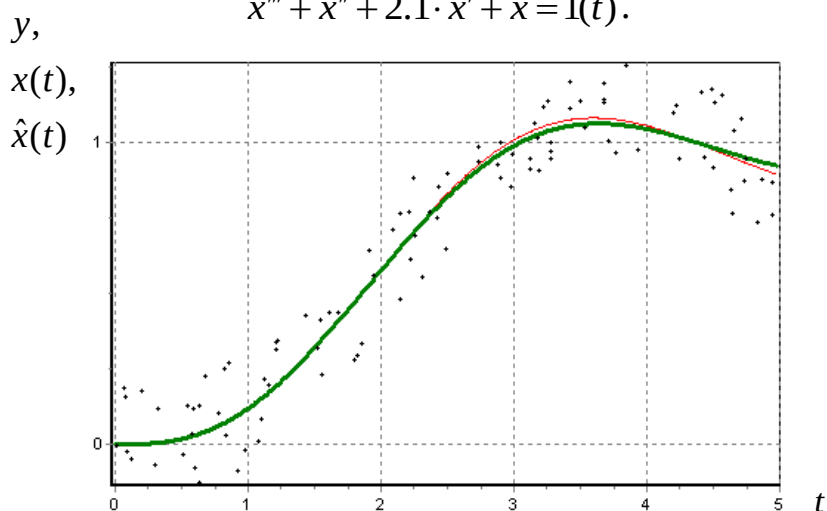


Рисунок 2.3. Решение задачи идентификации, модель хорошо описывает облако точек и практически совпадает с истинной системой.

Здесь мы видим, что облако точек не только хорошо описано кривой – решением дифференциального уравнения модели, но и сама модель практически полностью совпадает с дифференциальным уравнением исходного объекта.

Далее рассмотрим решение задачи идентификации для случая, когда модель хорошо описывает облако точек, переходный процесс близок к переходному процессу истинной системы, но модель отличается от истинной системы, в том числе и порядком. На рисунке 2.4 изображено сравнение выхода модели и истинной системы на фоне облака точек, при этом, система и модель заданы следующими дифференциальными уравнениями

$$x''' + 2 \cdot x'' + 2 \cdot x' + x = 1(t),$$

$$4 \cdot \hat{x}'' + 1.85 \cdot \hat{x} = 1(t),$$

соответственно.

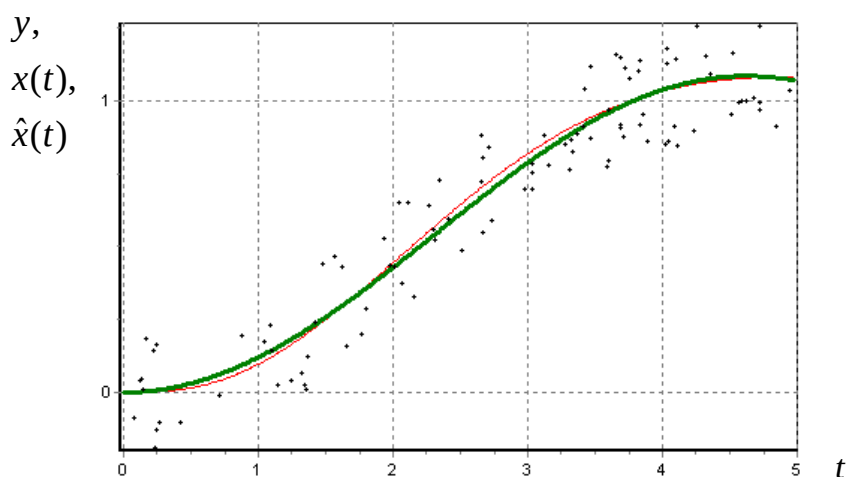


Рисунок 2.4. Решение задачи идентификации, модель хорошо описывает облако точек и практически совпадает с истинной системой, но отличается порядком.

То, что модель хорошо описывает облако точек и проходит в достаточной близости к истинной системе, но их символьные представления отличаются, говорит о том, что выборка недостаточно полно описывает переходный процесс и, как следствие, невозможно оценить по ней истинную систему. Переходные процессы различных систем могут совпадать на определенном интервале, поэтому только увеличение интервала наблюдения за выходом системы и

увеличение частоты снятия измерений способны повысить эффективность нахождения решения. С другой стороны, причиной этого может быть и наличие большого количества локальных оптимумов и достаточно сильная их зона притяжения.

Для алгоритма с настройками, которые были приняты за наиболее эффективные в среднем, в таблице 2.1 представлена оценка вероятности нахождения решения для устойчивых динамических систем, переходный процесс которых, завершается за время наблюдения выхода системы. Для каждого порядка было сгенерировано по 10 систем, причем таких, что каждая из них являлась устойчивой по Гурвицу, [70]. Следующим условием было завершённость переходного процесса в интервале наблюдения системы.

Для определенности, будем считать модель близкой к истинной системе, если выполняется условие $\max_{i \in \{1, \dots, N\}} \{|a_i - \hat{a}_i|\} < 0.05$. Пусть N_s - число таких исходов, при которых $\max_{i \in \{1, \dots, N\}} \{|a_i - \hat{a}_i|\} < 0.05$, а N_n - число исходов, когда разница между коэффициентами дифференциального уравнения модели и коэффициентами истинного объекта больше 0.05. Тогда оценку вероятности нахождения истинного решения будем определять по формуле $\hat{p}_s = \frac{N_s}{N_n + N_s}$. Приведенная статистика наряду со средним значением функции пригодности представлена в таблице 1.

Как видно по данным таблицы, оценка вероятности нахождения истинного решения слабо взаимосвязана со значением средней пригодности лучшего решения, таким образом, они не являются взаимозаменяемыми характеристиками эффективности работы алгоритма идентификации. Если рассмотреть статистику для каждой системы в отдельности и оценить коэффициент корреляции [97], то $\hat{K} = 0.697$, что подтверждает отсутствие линейной зависимости.

Предложенный модифицированный алгоритм эволюционных стратегий по результатам тестирования на решении тестовых задач идентификации является наиболее эффективным алгоритмом автоматического моделирования линейных динамических систем по данным наблюдений.

Таблица 2.1

Оценка вероятности нахождения решения, которое близко к истинному.

Порядок	$\hat{p}_s = \frac{N_s}{N_n + N_s}$	Средняя пригодность
1	0,65	0,959344
2	0,95	0,997950
3	0,9	0,997798
4	0,95	1,000000
5	0,8	0,996173
6	0,7	0,954235
7	0.65	0,992892

Отметим, что оба представленных критерия (2.13) и (2.14) эффективны для поиска решения задачи идентификации линейных динамических систем. В текущих исследованиях использовался критерий (2.13). Оба представленных критерия больше подходят для решения задачи идентификации, поскольку, в отличие от квадратичного критерия они являются более устойчивыми к выбросам. Тем не менее, в одном из исследований было показано, что критерий (2.13) является более эффективным критерия (2.14).

Далее рассмотрим приложение данного алгоритма для решения задачи идентификации параметров системы линейных дифференциальных уравнений, описывающих процесс изменения концентрации продуктов распада в химической реакции.

2. 4. Моделирование изменения концентрации продуктов распада гексадекана.

Рассмотрим изменение концентраций веществ реакции распада гексадекана, которое порождает задачу идентификации. Гексадекан распадается до следующих продуктов: спирты и карбонильные соединения. Начальные значения концентрации всех веществ известны. В теории химической кинетики

предполагается, что изменения концентраций веществ в реакции распада, описываются линейными дифференциальными уравнениями, причем первого порядка, [100]. Предполагается, что нет управляющего воздействия на данный процесс, таким образом, функцию управления можно считать равной 0 или 1, в предположении о таком виде дифференциального уравнения. Далее для решения последующей задачи нами будет рассмотрено оба возможных варианта.

На рисунке 2.5 изображены две возможные схемы протекания реакции распада гексадекана:

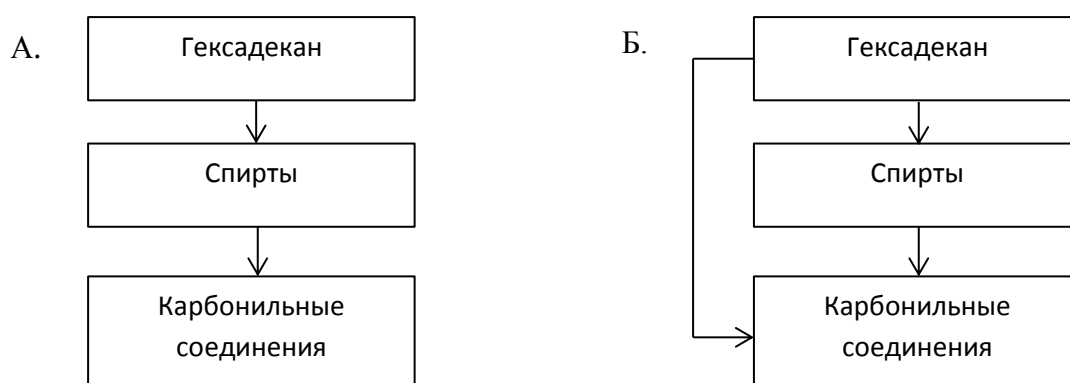


Рисунок 2.5. Схема образования продуктов распада гексадекана. А – последовательная модель, Б – параллельная.

Для дальнейшей работы с системой, включающей несколько выходов, понадобятся матрицы системы, которые описывают переходный процесс. Поскольку порядок дифференциальных уравнений для компонент реакции является неизвестным, оценим его через оценку порядка дифференциального уравнения изменения концентрации гексадекана (2.12). В таком случае необходимо решить задачу обратного математического моделирования, которая рассматривалась выше – определить модель по выборочным данным, которая доставляет минимум критерия (2.13).

Для этого воспользуемся алгоритмом, который был представлен в этой главе – NMES, с лучшими среди исследованных настроек. Максимальный порядок выберем равным 10, т.е. будем подбирать все дифференциальные уравнения, описывающие выход системы, вплоть до 10-го порядка. Измерение концентрации

в начальный момент времени, который обозначим как $t = 0$, нам известно, поэтому включим его в модель. После 50 запусков алгоритма были найдены различные решения в виде дифференциальных уравнений с разными структурами и параметрами, которые представлены в таблице 2.2. Конечно, среди всех запусков находились и совпадающие решения.

Таблица 2.2

Примеры найденных моделей по выборочным данным изменения концентрации гексадекана.

Модели	Величина ошибки (I_1)
$4.05 \cdot x' + 0.9 \cdot x = 1,$	0.3022
$1.05 \cdot x' + 0.4 \cdot x = 1,$	0.2834
$2.1 \cdot x' + 0.55 \cdot x = 1,$	0.1822
$-1.05 \cdot x''' - 0.15 \cdot x'' - 6.85 \cdot x' - 0.9 \cdot x = 1,$	0.2270
$-3.4 \cdot x' - 0.45 \cdot x = 0.$	0.2020

Видно, что лучшее решение соответствует системе первого порядка, что не противоречит гипотезе, утверждающей, что изменение концентраций веществ во всех реакциях распада описываются линейным дифференциальным уравнением первого порядка, [100].

На рисунке 2.6 представлен выход модели и выборочные данные. Еще раз подчеркнем, что среди всех моделей до 10 порядка была выбрана именно модель, соответствующая уравнению первого порядка. В таблице 2.3 есть и модель третьего порядка, которой соответствует колебательное движение выхода системы. Модель хорошо описывает выборочные данные, кривая выхода модели проходит к ним достаточно близко, но, как мы видим, такая модель может не соответствовать действительности, потому что процесс представлен выборкой малого объема. Следовательно, модель нежелательна в оценивании переходного процесса, тем более что модель первого порядка оказалась лучшей в плане значения критерия.

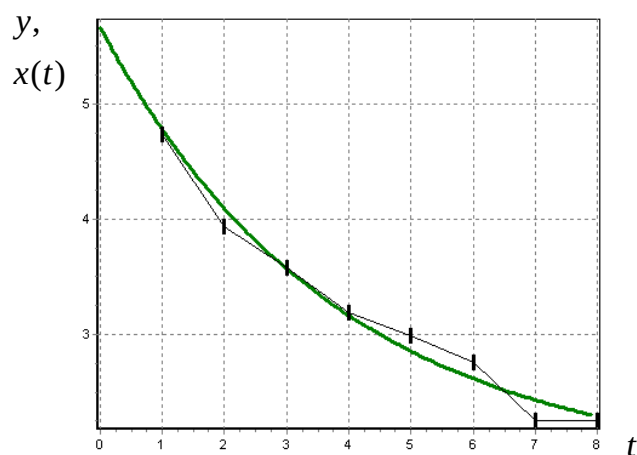


Рисунок 2.6. Модель изменения концентрации гексадекана и данные наблюдений.

Теперь, выдвинем гипотезу о том, что решением является система линейных дифференциальных уравнений первого порядка, определим ее параметры, т.е. оценим матрицу системы. Задача решалась для всех представлений указанных выше: последовательной реакции, параллельной и представления учитывающего все возможные связи.

Отметим, что выбор схемы протекания реакции полностью определяет вид матрицы системы. Для системы, которая описывает последовательную реакцию,

матрица системы имеет следующую структуру: $A = \begin{pmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ 0 & a_{32} & a_{33} \end{pmatrix}$, а система,

описывающая параллельную реакцию, обладает структурой: $A = \begin{pmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$.

В общем случае, можно предположить, что концентрация каждого из компонент реакции оказывает взаимное влияние на процесс изменения концентрации других продуктов реакции, тогда матрица системы будет лишена нулей,

$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$. Как видно, каждому представлению соответствует свое число

переменных задачи оптимизации.

Необходимо найти модель вида (2.15) для каждой из приведенных переходных матриц, которая бы доставляла минимум критерию (2.16). После 20 запусков алгоритма для каждой структуры выберем лучшее решение среди всех структур и запусков.

Выход системы и изначальная выборка представлены на рисунках 2.7, 2.8 и 2.9, соответственно, для гексадекана, спиртов и карбонильных соединений. Как видно по рисункам, измерение в точке $t=7$ выглядит аномальным, но оно не повлияло на модель.

Как показано на рисунках, задача идентификации решалась для выборок малого объема, при наличии незначительного выброса. Поиск модели осуществлялся в виде системы линейных дифференциальных уравнений среди всех схем, таким образом, лучшим из найденных решений задачи идентификации является следующая система:

$$\dot{\hat{x}}' = \begin{pmatrix} -0.2136 & 0.8448 & -0.5420 \\ -0.0899 & -0.4785 & -0.0545 \\ 0.0772 & 0.7317 & -0.4720 \end{pmatrix} \cdot \hat{x} + \begin{pmatrix} 0.5795 \\ 0.7155 \\ -0.4408 \end{pmatrix}.$$

Найденное решение содержит коэффициенты, которые близки к 0, что указывает на слабое влияние одной из характеристик на изменения другой, например, концентрация гексадекана меньше других составляющих влияет на изменение концентрации карбонильных соединений, а те, в свою очередь, на концентрацию спиртов.

Учет всех параметров приводит к незначительному уменьшению величины ошибки моделирования, с 0.1735 для параллельной реакции, до 0.1685, где представленные данные есть значение средней величины невязки.

Конечно, включение всех коэффициентов предполагает наложение дополнительных ограничений, которые обусловлены физическими свойствами процесса, если таковые ограничения имеются. Например, положительная определенность матрицы системы, устойчивость процесса, не отрицательность некоторых коэффициентов уравнения и т.п. Тем не менее, эти требования могут быть учтены при переходе к задаче условной оптимизации. Решение других задач

идентификации изменений концентраций продуктов распада представлено, например, в работах [45] и [46]. Алгоритм идентификации был реализован на языке MATLAB и представляет собой отдельный подключаемый модуль, который решает задачу идентификации для загружаемых данных: измерений выходов, количества выходов и порядки каждого из выходов динамической системы.

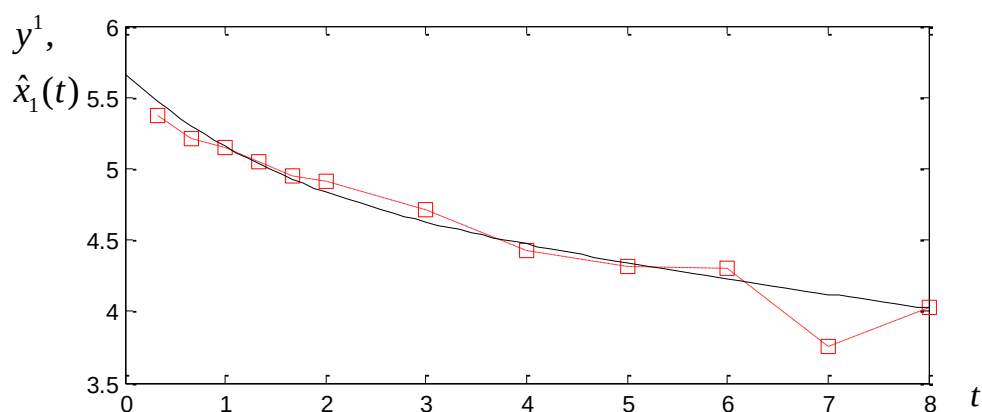


Рисунок 2.7. Модель изменения концентрации гексадекана и данные наблюдений.

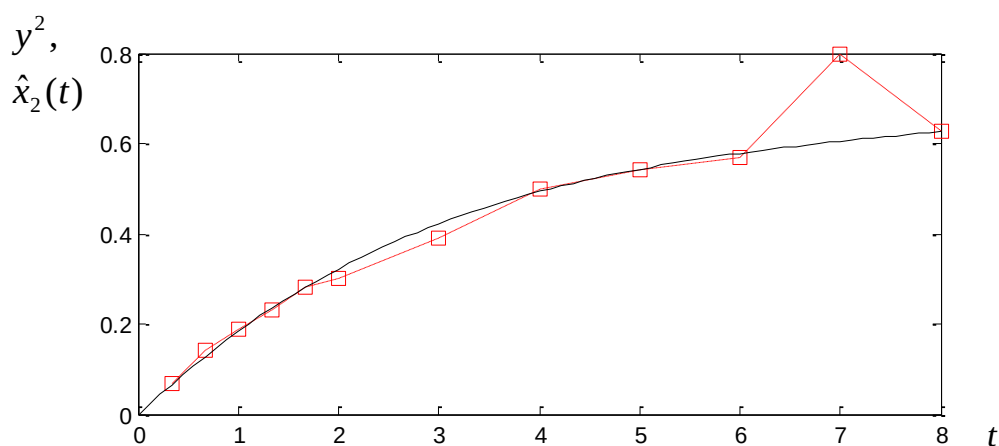


Рисунок 2.8. Модель изменения концентрации спиртов и данные наблюдений.

Таким образом, рассматриваемая задача идентификации эффективно решается при применении модифицированного алгоритма, который был изменен с учетом особенностей решаемой задачи. Была построена модель изменения концентрации гексадекана как независимо от изменения концентраций других продуктов реакции, так и с учетом изменения концентраций других продуктов реакции распада.

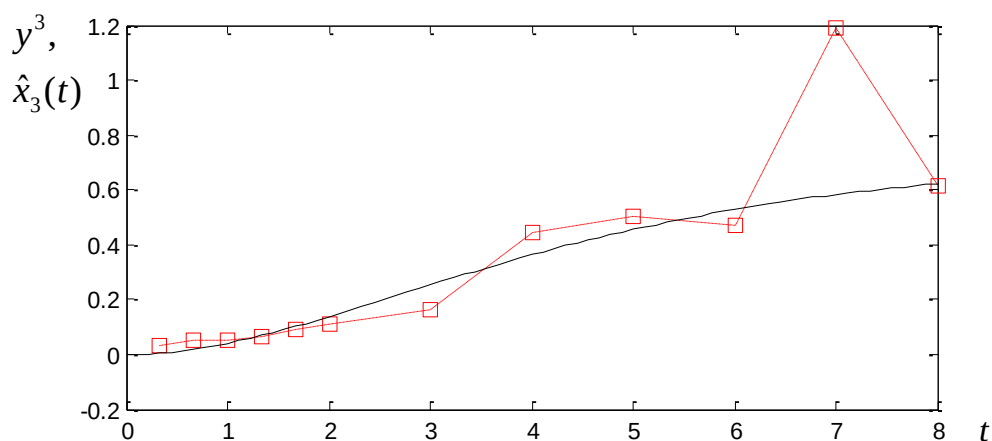


Рисунок 2.9. Модель изменения концентрации карбонильных соединений и данные наблюдений.

Выше говорилось о том, что измерение концентраций в точке 7 является аномальным. Стоит еще раз отметить, что использование абсолютных значений невязок и нормирование каждого аддитивного критерия для отдельного выхода системы в критерии (2.16), позволяет успешно находить решение задачи идентификации динамической системы с системой линейных дифференциальных уравнений (2.15).

Выводы

В данной главе был представлен подход к решению задачи идентификации для линейных динамических систем, который был основан на приведении исходной задачи к задаче поиска экстремума на пространстве вещественных векторов. Одной из особенностей такого подхода является единая схема представления моделей в виде линейных однородных дифференциальных уравнений различного порядка и различными коэффициентами одним вектором настраиваемой длины, что позволяет одновременно решать задачу как поиска наиболее подходящей структуры, так и наиболее подходящих коэффициентов уравнения.

Данное представление является универсальным для математических моделей, но приводит к усложнению целевой функции по сравнению с

идентификацией коэффициентов для системы заданного порядка. Дело в том, что переход от дифференциального уравнения одного порядка к дифференциальному уравнению другого порядка является сложным и происходит скачкообразно для критерия задачи минимизации невязки между выборочными данными и выходом модели. Далее, особенность представления решения такова, что, с одной стороны, важно, чтобы определенные координаты вектора решения сохранялись равными нулю, это условия появляется для систем порядка ниже, чем выбранного в качестве максимального. С другой стороны, коэффициенты уравнения должны быть определены с некоторой точностью, до десятых, сотых или выше. Более того, современные алгоритмы стохастического поиска, эволюционные и популяционные алгоритмы, как и многие другие, основаны на случайном воздействии на переменные альтернативы. Таким образом, получается, что для успешной работы алгоритма необходимо учитывать две противоположные особенности и совместить их. В данной работе были предложены возможные варианты разрешения этого противоречия.

Тем не менее, без привнесения таких изменений, другим вариантом решения задачи является перебор структур, заданных различным порядком. Но это не всегда является удачным ввиду сложности некоторых динамических систем или управляющего воздействия. Так, в качестве решения задачи идентификации процесса изменения концентрации гексадекана были найдены как дифференциальные уравнения первого порядка, так и уравнения третьего порядка. Можно предположить, что для других динамических объектов, которые являются более сложными, поиск порядка и коэффициентов так же может привести к решениям в виде отличающихся друг от друга дифференциальных уравнений. Перебор структур при тех же ограничениях поискового пространства приведет к значительному увеличению необходимых вычислительных ресурсов.

В работе предложен алгоритм решения приведенной задачи оптимизации, который отличается от известных алгоритмов высокой эффективностью нахождения решения для выбранного класса задач. Модифицированы операторы алгоритма, с учетом особенностей решаемой задачи. Изменена мутация, что

позволяет значительно улучшить показатели алгоритма, ввиду того, что мутация в меньшей мере возмущает коэффициенты алгоритма и, следовательно, меньше возмущает структуру уравнения. Некоторые случайно выбранные из популяции решения уточняются с помощью случайного покоординатного спуска. Если алгоритм ЭС дополнен только алгоритмом локального спуска, без включения модифицированной мутации, то эффективность алгоритма хоть и возрастает по сравнению со стандартной схемой ЭС, но в значительно меньшей степени, чем при подключении мутации. При добавлении округления, алгоритм в среднем эффективнее, чем представленные выше схемы, но незначительно эффективнее алгоритма с модифицированной мутацией. Для алгоритма, в котором все обозначенные поисковые процедуры включены, средняя пригодность решений значительно выше стандартного алгоритма ЭС, и выше остальных рассмотренных схем.

При сравнении алгоритмов по частоте определения истинной структуры и параметров объекта, предложенный алгоритм оказался значительно эффективнее других. Оценка вероятности определения истинной структуры и коэффициентов является наиболее важной характеристикой с практической стороны. Ввиду слабой линейной зависимости между вероятностью нахождения истинного решения и пригодностью решения, можно заключить, что эти показатели имеют различную природу. Свойства множества наблюдений переходного процесса системы во многом определяют и адекватность найденного решения: чем дольше наблюдается переходный процесс, чем лучше передается динамика, тем выше вероятность нахождения решения предложенным подходом. Если переходный процесс наблюдается не полностью, т.е. до его стабилизации около положения устойчивости, то возможны решения в виде дифференциальных уравнений, которые идентичны истинному уравнению на данном интервале, но могут отличаться вне этого интервала.

Дальнейшие исследования предполагают развитие алгоритма для решения задач идентификации при неизвестной, но наблюдаемой по измерениям функции управления и при неизвестном векторе начальных условий динамической

системы. Критерий задачи будет включать в себя невязки между выходом модели и наблюдаемыми данными, управлением и измерениями входного воздействия при выбранном для модели начальном положении.

В главе представлено решение задачи моделирования процесса изменения концентраций продуктов распада гексадекана. Модель изменения концентрации гексадекана определялась на пространстве дифференциальных уравнений до 10-ого порядка. Важно отметить, что при многократных запусках алгоритма, решение задачи было найдено в виде дифференциального уравнения первого порядка, что соответствует теории химической кинетики, в которой утверждается, что изменение концентрации продукта, участвующего в реакции распада описывается линейным дифференциальным уравнением первого порядка.

Решенная задача принадлежит классу задач идентификации при малой выборки. Для таких задач количественные методы решения являются неэффективными, а, значит, для построения модели применимы методы, основанные на подборе структуры и параметров функции одновременно. Но такие подходы, как правило, развиты только для идентификации статических систем, следовательно, моделирование линейной динамики будет неразрывно связано с уравнением (2.3).

Подход к приведению задачи идентификации линейных динамических систем и предложенный алгоритм позволяют подбирать математическую модель в виде линейного дифференциального уравнения или систем линейных дифференциальных уравнений автоматически, одновременно определяя структуру и коэффициенты уравнений, в том числе и в условиях малой выборки.

В следующих главах будут рассмотрены задачи управления: двухточечная задача управления, - задача перевода динамической системы из заданного состояния в желаемое и задача оптимального управления при различных критериях оптимальности.

ГЛАВА III. РЕШЕНИЕ ЗАДАЧИ ТЕРМИНАЛЬНОГО УПРАВЛЕНИЯ ДЛЯ ДИНАМИЧЕСКИХ СИСТЕМ С ИСПОЛНИТЕЛЬНЫМ МЕХАНИЗМОМ В ВИДЕ МНОГОУРОВНЕВОГО РЕЛЕ.

В данной главе рассматривается решение задачи управления для динамических систем с исполнительным механизмом особого вида: идеальное или многоуровневое реле. Подобные задачи встречаются, когда исполнительный механизм работает, например, по принципу «включено-выключено» или в нескольких режимах, число которых является счетным и, следовательно, которые, могут быть пронумерованы. В таких случаях на значения, принимаемые управляющим воздействием в каждом из режимов, мы повлиять не можем. При управлении космическими аппаратами (КА) возникают задачи, в которых необходимо найти программное управление по переводу системы из одного положения в другое при том, что двигатель неконтролируемой тяги может работать в одном из трех положений: включено в одну сторону, выключено или включено в другую сторону. Под решением задачи терминального управления понимается поиск такой функции управления, которая бы обеспечила переход объекта из текущего состояния в состояние желаемое [70], [71].

Задача терминального управления является частным случаем задачи оптимального управления [41]: известны левый и правый концы траектории, известно множество допустимых значений управления, динамическая система задана дифференциальным уравнением, но отсутствует интегральный функционал качества. Таким образом, в данной постановке необходимо найти любую функцию управления, которая бы не противоречила ограничениям и, при подстановке ее в систему уравнений, обеспечивала достижение желаемого положения из текущего положения за заданное или конечное время. При этом по причине особого вида функции управления, невозможно использование классических подходов к решению задачи оптимального управления. В общем случае, мы знаем об исполнительном механизме только то, что он может работать в нескольких режимах: 1-ый, 2-ой и т.д., до номера последнего режима. Такого

рода задачи лишены единого универсального подхода к нахождению решения, кроме алгоритмов перебора, или решения приведенной оптимизационной задачи со смешанным типом переменных. В данной работе рассматривается подход к сведению задачи поиска программного терминального управления к экстремальной задаче с вещественными и номинальными переменными и применению модифицированного алгоритма для решения таких задач.

Отметим, что решение подобной задачи рассматривалось в работе [1], где авторы представили алгоритм по коррекции точек переключения режимов работы двигателя КА. Но предложенный подход был приведен для линейных динамических систем и предполагает уже наличие некоторого решения задачи терминального управления так, что коррекция будет направлена на повышение точности текущего решения, а не поиск нового. В работе [99] рассматривается алгоритм настройки точек переключения управления, но только для класса линейных динамических систем и для задачи быстродействия.

Во многих работах исследуются решения задачи терминального управления [85], [87], [88], [116], но в этих работах функция управления либо не имеет ограничений, либо определена внутри некоторой области и является гладкой. Решение двухточечной задачи управления для кусочно-постоянных функций управления рассматривается, например, в работах [60], [83], [118], [119], но функция управления при этом принимает любые значения из некоторого ограниченного множества, и в данных работах подобное представление управляющего воздействия является аппроксимацией некоторой функции. Поскольку задача терминального управления является частным случаем задачи оптимального управления, в некоторых работах рассматриваются методы решения задачи, но требующие гладкости функции управления, например, [64].

Решение задачи нахождения программного управления для класса тригонометрических функций представлено в работе [72]. В работе рассматриваются объекты, управление которыми так же является режимным, но для линейных динамических систем.

В работе [59] представлен подход к решению задачи терминального управления с помощью генетического алгоритма. Функция управления при этом представлена как кусочно-постоянная функция, но при этом, значение управления на каждом интервале не является фиксированным. Конечно, данный подход может быть применен и для случая, когда множество допустимых значений управления является конечным и счетным, но ниже будет показано, почему использование методов оптимизации на пространстве булевых векторов является не желательным, ввиду их существенного ограничения поискового пространства.

Есть работы, в которых развиваются методы управления динамическими системами, решается задача синтеза регулятора, структура которого представлена в виде идеального реле [81], [82].

Количество всех возможных постановок задач, которые представлены выше, говорит о том, что задача управления системами с исполнительным механизмом, который обеспечивает входящее воздействие в виде кусочно-постоянной функции, является актуальной. Также является актуальной задача поиска программного управления особого вида, которое являлось бы решением двухточечной задачи управления.

Эволюционные методы прямого поиска решения задачи оптимального управления рассматриваются в большом количестве работ, например, [21], [60], [66]. Тем не менее, прямые методы поиска остаются плохо изученными для задач со свободным временем. А дискретизация переменных поиска, к которой прибегают при применении некоторых эволюционных алгоритмов, при том, что они не являются ограниченными и в то же время, необходима тонкая настройка их значений, делает такими алгоритмы малоэффективными в решении задач представленного класса.

Интеллектуальные методы идентификации программного управления могут быть использованы для решения задачи терминального управления, но для данной постановки задачи применимы только те, что способны генерировать функцию, которая не является гладкой, а, в общем случае, приводился к кусочно-постоянной функции для нелинейной системы или системы с особой правой

частью. Более того, без некоторых изменений и допущений, данные методы предполагают нахождение приближенного решения вариационной задачи только для случая, когда известен отрезок, на котором функция управления должна быть определена. Последнее не подходит под различные постановки задачи поиска терминального управления для систем с нефиксированным временем.

Для решения задачи было введено специальное представление функции управления. Выделены множества параметров, которые полностью позволяют описать любую кусочно-постоянную функцию с ограниченным по мощности множеством значений, которая эта функция может принимать. Такие множества имеют различную природу, что повлекло за собой последующее совершенствование алгоритма оптимизации. Таким образом, после того, как были выделены параметры, которые полностью описывают программное управление на интервале регулирования, задача была сведена к экстремальной задаче. Критерий в приведенной задаче выражает все цели программного управления.

Итак, предложенный подход является методом прямого поиска управления, а, значит, не использует аналитический вид объекта управления. Отметим, что разработанный подход применим, таким образом, и для поиска управления системами, которые заданы алгоритмически или для дискретных динамических систем. Ниже, на рисунке 3.1 изображено решение задачи терминального управления для многоуровневого реле, которому соответствует кусочно-постоянная функция.

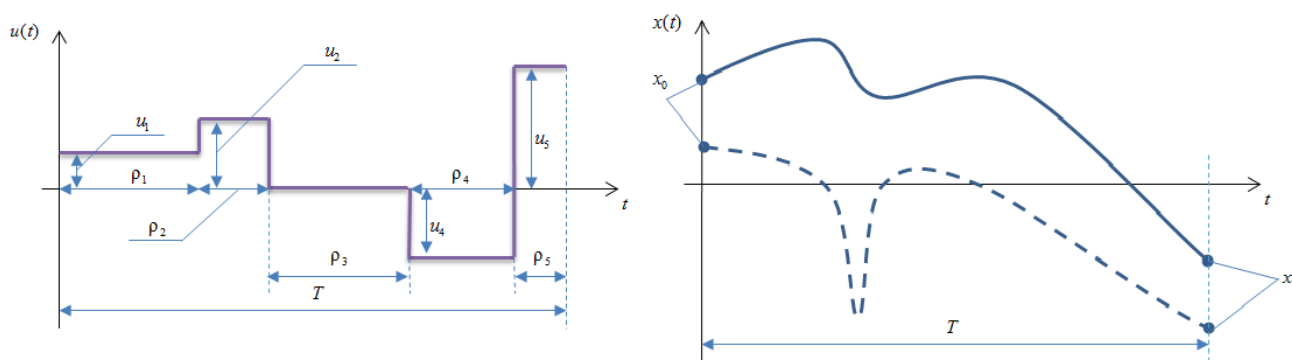


Рисунок 3.1. Решение задачи терминального управления. Слева – функция управления для многоуровневого реле. Справа – график состояния системы.

Здесь T - фиксированное время работы системы, вектора: x^* - правый конец траектории, т.е. конечное положение динамической системы, x^0 - начальное положение. Значения $u_1, u_2, \dots, u_5$ - уровни, которые может принимать функция управления. В другой интерпретации так могут быть обозначены некоторые режимы функционирования исполнительного механизма. Параметры $\rho_1, \rho_2, \dots, \rho_5$ - интервалы, соответствующие каждому из представленных уровней. Интервалы касаются друг друга в точках, которые дальше будем называть точками переключений, т.е. в этих точках меняется режим работы исполнительного механизма.

3. 1. Постановка задачи терминального управления в виде многоуровневого реле и сведение ее к экстремальной задаче.

Для системы, которая описывается дифференциальным уравнением в общем виде

$$\frac{dx}{dt} = f(x, u, t), \quad (3.1)$$

где

$f(\cdot): R^n \times R \times R^+ \rightarrow R^n$ - вектор-функция, непрерывная по своим аргументам;

$x \in R^n$ - вектор состояний системы;

$u(t): R^+ \rightarrow U, U = \{u_i \in R, i = \overline{1, N_u}\}$ - кусочно-постоянная функция управления;

N_u - число уровней управления, число режимов;

n - размерность системы;

необходимо найти функцию управления $u(t)$, которая переводит систему из известного начального состояния $x(0) = x^0$ в заданное конечное состояние $x(T) = x^*$ за конечное время T .

Хотя функция управления такого вида характерна только для некоторых задач управления, алгоритм для решения такого класса задач может быть использован вне приложения к системам с исполнительным механизмом в виде

реле. Дело в том, что иногда решение задачи оптимального управления предполагает поиск управления в виде реле. Так, если Гамильтониан линеен по функции управления, т.е. $H(x, p, u) = \varphi_1(x, p) + \varphi_2(x, p) \cdot u$, а функция управления имеет ограничения по модулю $|u(t)| < A$, или параллелепипедные ограничения $A_{\min} < u(t) < A_{\max}, t \in [0, T]$, то решение задачи сведется к поиску кусочно-постоянной функции с заданными характеристиками [70].

Если исполнительным механизмом является идеальное реле, то функция управления представима в следующем виде:

$$u(t) = \begin{cases} -A, & t \in I_1, \\ A, & t \in I_2, \end{cases} \quad (3.2)$$

где I_1, I_2 представляют собой множества, задаваемые некоторыми точками переключений, в которых функция меняет свой знак, причем $I_1 \cup I_2 = [0, T]$, а A - амплитуда реле.

В случае если реле является многоуровневым, то функция управления может быть представлена в более общем виде:

$$u(t) = \begin{cases} u_1, & t \in I_1, \\ \vdots \\ u_{N_u}, & t \in I_{N_u}, \end{cases} \quad (3.3)$$

где $I_i, i = \overline{1, N_u}$ - интервалы, порожденные точками переключений, причем $\bigcup_{i=1}^{N_u} I_i = [0, T]$, $\bigcap_{i=1}^{N_u} I_i = \emptyset$; $U = \{u_i \in R, i = \overline{1, N_u}\}$ - множество значений, принимаемых функцией управления; N_u - число значений, которое может принимать функция управления.

Пусть $P = \{r_i : r_i < r_{i+1}, r_i \in R^+ \forall i = \overline{1, k}, r_0 = 0\}$ есть множество точек переключений, а k - параметр, определяющий количество точек переключений. Множество $L = \{l_i : l_i \in N \forall i = \overline{1, k}\}$ является набором индексов, который

используется для описания множества интервалов. Каждый интервал $I_i, i = \overline{1, N_u}$ для функции управления (3.3) может быть определен уравнением

$$I_i = \left\{ \bigcup_{j=0}^{N_u} (r_j, r_{j+1}] \cdot f_I(i, j), r_j \in P \forall j \right\}, \text{ в котором } f_I(i, j) = \begin{cases} 1, i = L_j \\ 0, i \neq L_j \end{cases} - \text{ специальная}$$

функция. Для указанного идеального реле (3.2) может быть использована упрощенная схема представления множества интервалов. В этом же случае, мы можем потребовать изменения знака функции управления в каждой точке переключения, тогда все, что нам потребуется для описания функции управления – знак управления в точке $t = 0$, поскольку выполнение данных условий позволяет полностью описать функцию управления на всем интервале работы системы.

Приняв такое описание функции управления, задача терминального управления может включать в себя следующие постановки:

- определение точек переключений управления при фиксированном времени работы системы T ;
- определение точек переключения и времени работы системы T .

Постановка задачи может также включать подбор элементов допустимого множества управления, настраивая, таким образом, сам исполнительный механизм. Получается, что при данном наборе параметров, которыми задается функция управления, могут быть решены различные задачи, сопряженные с задачей нахождения терминального управления.

Вопрос о выборе числа точек переключений остается нерешенным. Данная постановка задачи и предложенный ниже метод решения предполагают неявную настройку количества точек переключений. Прежде всего, если какая-то точка переключений больше конечного времени управления, $r > T$, тогда она не влияет на процесс управления. В том случае, если необходимо, чтобы число точек переключений было фиксированным на интервале, то можно прибегнуть к иному представлению множества точек переключений управления:

$$\tilde{P} = \left\{ r_i : r_i < r_{i+1}, r_i \in R^+, r_i \leq T \forall i = \overline{0, k}, r_0 = 0 \right\}. \quad (3.4)$$

Неявная настройка числа точек переключений также может быть обусловлена применением численной схемы интегрирования системы (3.1). Пусть шаг интегрирования h , тогда при выполнении условия $|r_{i+1} - r_{i-1}| < h$, интервал $(r_i, r_{i+1}]$ не повлияет на поведение системы, поэтому может быть исключен, и фактически не учитывается.

Пусть множество S включает в себя все параметры, определяющие структуру функции управления. Для многоуровневого реле $S = \{P, L, U\}$, а для идеального множество может быть представлено как $S = \{P, u(0)\}$. В случае, если рассматриваемая задача такова, что значения, принимаемые управлением, известны и фиксированы, то множества в случае многоуровневого реле и идеального могут быть заданы, как $S = \{P, L\}$ и $S = \{P\}$, соответственно.

Используя указанные выше обозначения, задача поиска решения двухточечной задачи управления может быть сведена к оптимизационной задаче на пространстве вещественных чисел или на пространстве вещественных и натуральных чисел.

Рассмотрим целевые функции задачи терминального управления. В общем случае, решение терминальной задачи для фиксированного времени управления совпадает с решением экстремальной задачи:

$$F_1(\tilde{S}) = \|x^* - x(T)|_{S=\tilde{S}}\| \rightarrow \min_{\tilde{S}}, \quad (3.5)$$

где $x(T)|_{S=\tilde{S}}$ - вектор координат, решение системы (3.1) в конечный момент времени T , при управлении, определенном множеством $\tilde{S}$. Если время не фиксировано, то критерий приобретает вид

$$F_2(\tilde{S}, T) = \|x^* - x(T)|_{S=\tilde{S}}\| \rightarrow \min_{\tilde{S}, T}, \quad (3.6)$$

. При выбранном представлении функции управления существуют ограничения в виде неравенств

$$r_i < r_{i+1}, r_i \in R^+ \quad \forall i = \overline{1, k},$$

которые обеспечивают выполнение условия нахождения каждой точки переключения внутри интервала $[0, T]$. В случае использования множества $\tilde{P}$ вместо P , появляется еще одно ограничение: $r_k < T$. Впервые такое представление функции управления был предложено в работе [107].

Для решения задачи условной оптимизации введем функцию штрафа $\varphi(x) = \begin{cases} \|x\|, & x > 0 \\ 0, & x \leq 0 \end{cases}$, и весовой коэффициент $\alpha > 0$. Таким образом, задача условной оптимизации может быть представлена как задача безусловной оптимизации с измененной целевой функцией. Прибавляя к обозначенным выше целевым функциям (3.5) и (3.6) функцию штрафа, находим целевые функции для задач управления, в которых все точки переключений лежат внутри интервала регулирования:

$$F_3(\tilde{S}) = F_1(\tilde{S}) + \alpha \cdot \varphi(\tilde{r}_k - T) \rightarrow \min_{\tilde{S}}, \quad (3.7)$$

$$F_4(\tilde{S}, T) = F_2(\tilde{S}, T) + \alpha \cdot \varphi(\tilde{r}_k - T) \rightarrow \min_{\tilde{S}, T}, \quad (3.8)$$

Чтобы ограничения $r_i < r_{i+1}$ были соблюдены, может быть добавлена функция штрафа вида $\beta \cdot \sum_{j=1}^{k-1} \varphi(r_{i+1} - r_i)$ для любого из критериев (3.5)-(3.8), тогда получим целевые функции следующего вида:

$$\tilde{F}_1(\tilde{S}) = \|x^* - x(T)|_{S=\tilde{S}}\| + \beta \cdot \sum_{j=1}^{k-1} \varphi(r_{i+1} - r_i) \rightarrow \min_{\tilde{S}}, \quad (3.9)$$

$$\tilde{F}_2(\tilde{S}, T) = \|x^* - x(T)|_{S=\tilde{S}}\| + \beta \cdot \sum_{j=1}^{k-1} \varphi(r_{i+1} - r_i) \rightarrow \min_{\tilde{S}, T}, \quad (3.10)$$

$$\tilde{F}_3(\tilde{S}) = F_1(\tilde{S}) + \alpha \cdot \varphi(\tilde{r}_k - T) + \beta \cdot \sum_{j=1}^{k-1} \varphi(r_{i+1} - r_i) \rightarrow \min_{\tilde{S}}, \quad (3.11)$$

$$\tilde{F}_4(\tilde{S}, T) = F_2(\tilde{S}, T) + \alpha \cdot \varphi(\tilde{r}_k - T) + \beta \cdot \sum_{j=1}^{k-1} \varphi(r_{i+1} - r_i) \rightarrow \min_{\tilde{S}, T}. \quad (3.12)$$

Таким образом, задача терминального управления была сведена к оптимизации на пространстве вещественных чисел или вещественных и натуральных чисел с целевыми функциями (3.5)-(3.12). Указанные функции многоэкстремальны и не являются аналитическими, могут быть вычислены только численно. Учитывая сложность решаемых задач, размерность поискового пространства и неоднородность переменных, а так же особенности целевых функций, в качестве основного алгоритма решения задачи был выбран обобщенный гибридный модифицированный метод эволюционных стратегий.

3. 2. Разработка обобщенного алгоритма ЭС для прямого решения задачи терминального управления

Учитывая некоторые особенности решаемой задачи, можно модифицировать алгоритм так, чтобы сделать возможным решение задачи поиска экстремума для разнородных переменных, не прибегая при этом к дискретизации вещественных переменных и сохранения единой концепции алгоритма для номинальных переменных, т.е. для переменных, которые принимают натуральные значения.

Важно отметить, что использование алгоритмов оптимизации, которые работают с бинарной строкой недопустимо, поскольку некоторые из изложенных выше задач предполагают, что не существует границы множества допустимых значений для переменных. Так, например, для задачи со свободным временем, одной из переменных может быть время работы системы, которое в общем случае неизвестно и является настраиваемым параметром, которым определяется и число шагов интегрирования уравнения. Выше шла речь о неявной настройке количества точек переключений. Если использовать бинарное представление для

точек переключения, то, с одной стороны, дискретизация должна позволять алгоритму брать значения, принадлежащие длинному отрезку на R , $[0, b_l]$, где b_l - правая граница. С другой стороны, шаг дискретизации должен быть достаточно маленьким, чтобы сравниться с шагом интегрирования дифференциального уравнения, иначе не будет возможна неявная настройка числа точек переключений. Одновременное выполнение таких условий неизбежно приведет к существенному росту числа генов, при этом, множество значений которые могут принимать точки переключений является ограниченным, его определяется частотой дискретизации.

Точки переключений управления могут быть представлены в геноме, не только как они есть, но и как некоторое преобразование от вещественных генов. Подобное решение применимо, чтобы уйти от задачи условной оптимизации с ограничения в виде неравенств $r_i < r_{i+1}, r_i \in R^+ \forall i = \overline{1, k}$. Тогда, опциональным является выбор прямого представления точек переключения и использование критериев (3.9)-(3.12) или использование отображения точек переключения через объективные параметры оптимизационного алгоритма:

$$r_i = \sum_{j=1}^i o_j, i = \overline{1, k}. \quad (3.13)$$

Чтобы сохранить корректную работу алгоритма, которая возможно лишь в случае упорядоченности множества точек переключений, необходимо условие не отрицательности каждого из вещественных объективных параметров: $op_i \geq 0, i = \overline{1, k}$.

Очевидно, что последнее будет достигнуто, если потребовать, чтобы:

- все объективные параметры начальной популяции были сгенерированы на пространстве R^+ , т.е. $P(\exists i, j : op_j^i < 0) = 0$;
- оператор мутации не приводил к попаданию гена на отрицательную полуось.

Для выполнения второго пункта, оператор был дополнительно модифицирован так, что область допустимых значений была неотрицательна:

$$op_j = \left| op_j + z \cdot N(0, sp_j) \right|, j = \overline{1, k}. \quad (3.14)$$

Поскольку функция управления, в общем случае полностью определена множествами вещественных и номинальных величин, было предложено представить номинальные переменные числами натурального ряда, тогда необходимо использовать обобщенный алгоритм для работы с натуральными числами.

Чтобы решать задачу оптимизации одновременно для вещественных и номинальных переменных, были введены операторы мутации и скрещивания для натуральных чисел, которыми эти переменные могут быть представлены. Изложенный алгоритм представлен в параграфе 1.3. Конечно, натуральный ряд есть только способ представления номинальных переменных и при подборе операторов поиска было учтено, что упорядоченность натурального ряда не взаимосвязана со свойствами номинальных переменных.

На рисунке 3.2 показана общая схема расчета функции пригодности при различных постановках задачи терминального управления.

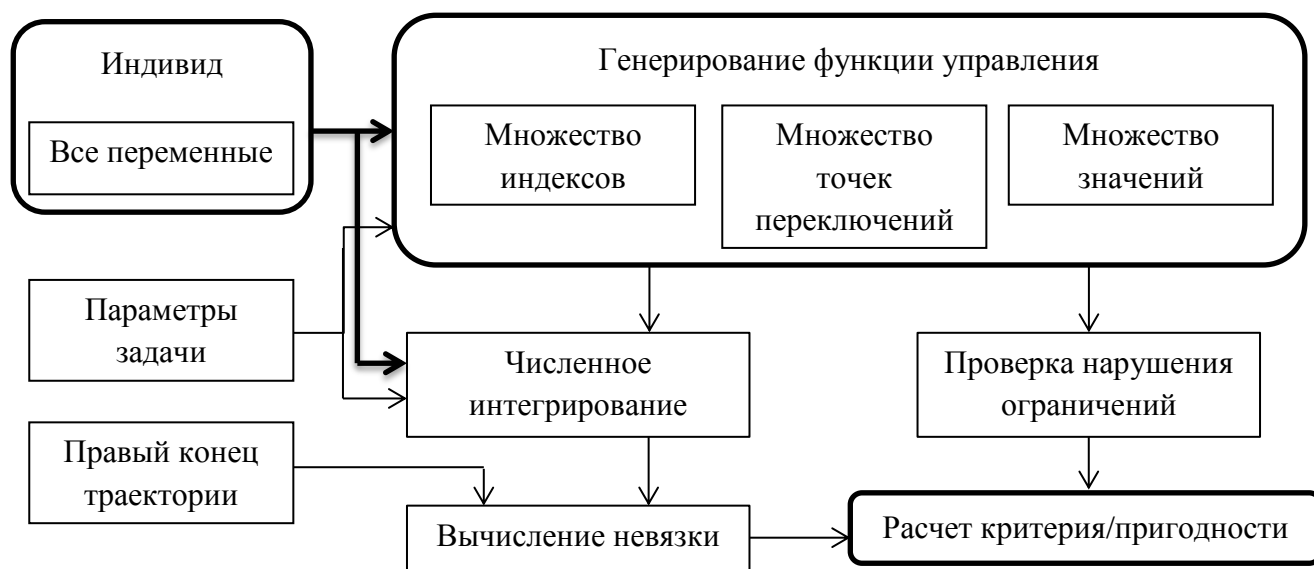


Рисунок 3.2. Схема действий при вычислении функции пригодности.

В данном параграфе были рассмотрены модификации операторов поиска алгоритма ЭС, но за основу был принят уже не стандартный, а модифицированный гибридный обобщенный алгоритм ЭС. В данной главе под

гибридным модифицированным алгоритмом эволюционных стратегий понимается алгоритм, предложенный в параграфе 1.2, т.е. с включением локального поиска, модифицированной мутацией и при операторах поиска, которые были описаны в параграфе 1.3. Таким образом, гибридизация алгоритма используется для локального улучшения альтернатив через оптимизацию вещественных переменных, при фиксированных значениях номинальных переменных для каждой отдельно взятой альтернативы.

Схема алгоритма оптимизации представлена на рисунке 3.3.

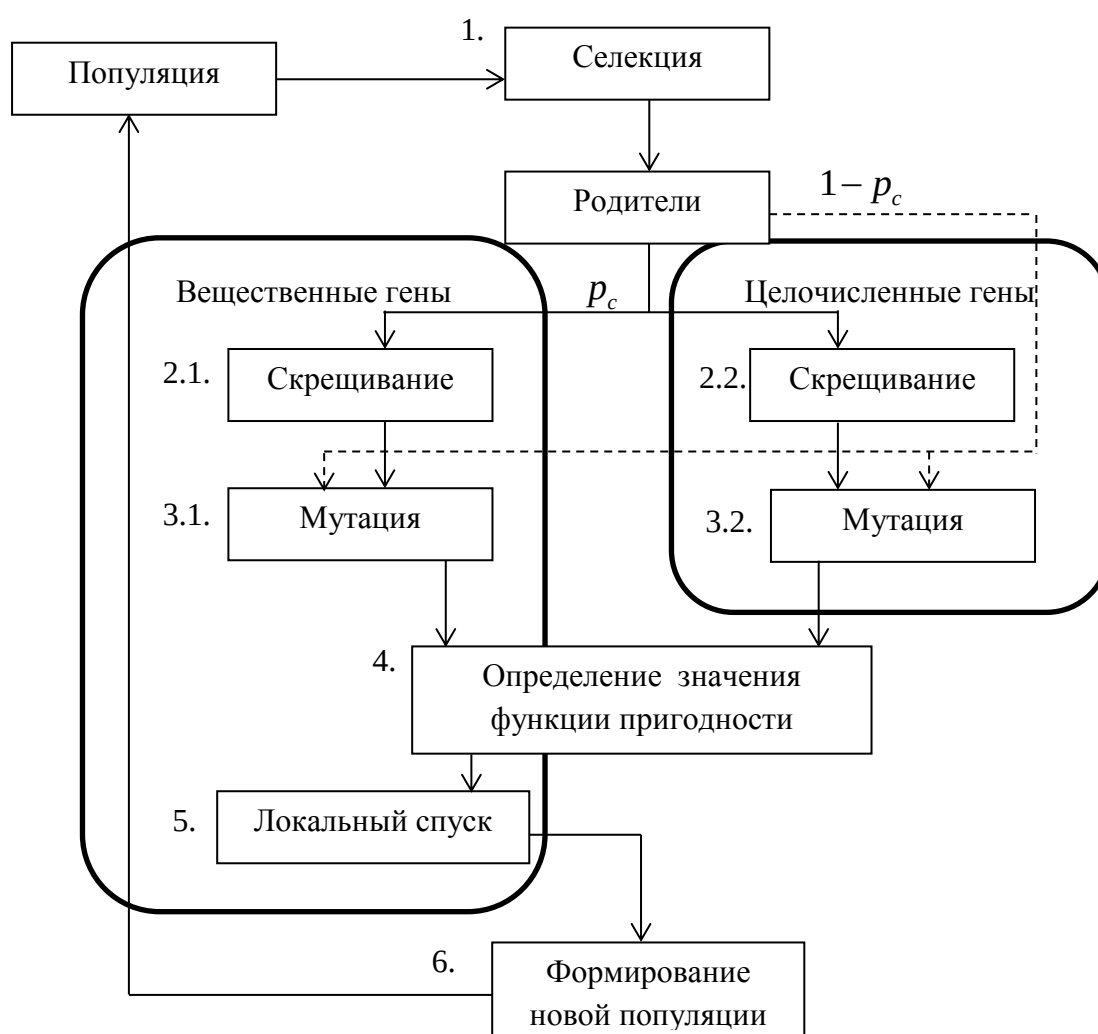


Рисунок 3.3. Схема работы обобщенного алгоритма внутри каждого популяционного цикла.

В каждой отдельной задаче, которая может быть решена с помощью предложенного подхода и предлагаемым алгоритмом, альтернатива содержит в себе все переменные задачи, которые могут быть настроены, в то время как блок

«параметры задачи» содержит все то, что является неизменным. Например, если нужно перевести систему за фиксированное время, при этом известны режимы работы двигателя, то неизменными будут, соответственно, время интегрирования и множество значений управления. Или, в других задачах, значение реле может быть настраиваемым, как, например, и конечное время работы системы.

Обобщенный алгоритм имеет ту же последовательность действий, что и стандартный, и модифицированный алгоритм. На рисунке разделены операторы поиска для вещественных и номинальных переменных для того, чтобы подчеркнуть независимость этих операторов друг от друга и небольшую разницу в общей схеме: по вещественным переменным происходит локальное улучшение альтернативы, но при найденных фиксированных значениях номинальных переменных.

Выбранный способ представления программного управления для режимного исполнительного механизма, многоуровневого или идеального реле, позволяет автоматически находить решение двухточечной задачи управления во всех возможных постановках. Каждая альтернатива суть функция управления или программа управления и параметры задачи, которые соответствуют конкретной постановке, как показано на рисунке 3.2. В следующем параграфе рассматривается тестирование алгоритма с целью определения наиболее эффективных настроек.

Предложенный алгоритм решения задачи представлен в работах [47], [91], [103], [110] на основании которых были реализованы программные системы решения задачи терминального управления при фиксированном времени [105] и при свободном времени [106].

3. 2. Определение эффективных настроек обобщенного алгоритма эволюционных стратегий для решения рассматриваемой задачи терминального управления

Для определения наиболее эффективных настроек алгоритма были выбраны 20 различных задач, для каждой из которых проводилось по 20 запусков алгоритма. Отметим, что для задачи терминального управления в текущей постановке, необходимо прибегать к методам численного интегрирования системы дифференциальных уравнений. Таким образом, каждая альтернатива, которая содержит в себе функцию управления и параметры задачи может быть оценена только после того, как динамическая система будет численно интегрирована при управлении в виде сформированной функции и текущими параметрами задачи. В предыдущем параграфе рисунок 3.3 отображает последовательность действий при вычислении функции пригодности, а рисунок 3.4 отображает последовательность действий внутри каждой популяции. Программное решение реализовано согласно представленным схемам.

Приведем несколько различных постановок задач терминального управления для управляющего механизма в виде многоуровневого реле. Рассмотрим общий случай, тогда каждой постановке задачи будет соответствовать несколько опций представления решения:

1. Задача со свободным временем, множество значений управления известно и фиксировано. Альтернатива представлена вектором длины k или $k+1$. Так же, помимо вектора, альтернатива представлена множеством мощности k - набор индексов, определяющих соответствие интервалов регулирования типу режима.

1.1 Первый вариант соответствует представлению конечного времени работы системы как $T = \sum_{i=1}^k o_i$.

1.2 Второй вариант соответствует схеме, в которой $T = o_{k+1}$.

2. Задача с фиксированным временем, множество значений управления известно и фиксировано. Альтернатива представлена вектором с вещественными

координатами размерности k и множеством натуральных чисел мощности k , - набором индексов.

3. Задача со свободным временем, где множество значений управления не фиксировано. Подобные задачи могут возникать при разработке систем управления, т.е. когда необходимо определить заранее режимы, тягу двигателя или другие характеристики, чтобы исполнительный механизм мог обеспечить выполнение необходимых задач. Альтернатива представлена вектором, размерности N_u , координатам которого соответствуют допустимые значения управления, в данной постановке, представленные вещественными числами, множеством индексов длины k . При этом альтернатива содержит еще одно множество, по аналогии с вариантом с п.1.1. и п.1.2.

3.1 Альтернатива представлена вектором размерности k , которому соответствует представление $T = \sum_{i=1}^k o_i$.

3.2 Альтернатива представлена вектором размерности $k+1$, которому соответствует представление $T = o_{k+1}$.

4. Задача с фиксированным временем, множество значений управления неизвестно. Альтернатива представлена вектором размерности k , множеством индексов мощности k и вектором размерности N_u . Как и для 3 варианта, может быть использована при проектировании систем, но уже для более жесткого условия – фиксированного времени.

При этом для каждой из постановок задачи может быть выбран один из вариантов представления множества точек переключения:

1. Число точек переключения фиксировано на интервале регулирования. Этот вариант соответствует включению в критерий штрафа выход за границу времени управления точки переключения.
2. Алгоритмом автоматически подбирается решение с числом точек переключения не больше, чем k . В данном варианте, некоторые точки переключений могут располагаться за пределами времени управления и, таким

образом, не учитываться. Так осуществляется настройка количества точек переключения.

Более того, точки переключения могут быть представлены двумя способами:

1. Точки переключения представляются в альтернативе, как они есть. Такой вариант соответствует включению штрафа за нарушение упорядоченности.
2. Точки переключения являются преобразованием от значений генов. В данном случае мы переходим от условной задачи оптимизации к безусловной без добавления функции штрафа.

На сегодняшний день, нельзя однозначно определить, какой из предложенных функционалов или способов представления множества точек переключения или их интерпретации является более эффективным для каждой отдельной задачи, поскольку не выявлены формальные признаки, по которым можно было бы судить о большей эффективности одного из способов представления решения и, следовательно, соответствующего ему критерия. Более того, для некоторых задач, возможно, что лучшим будет то или иное представление решения и, как следствие, критерий. Набора тестовых задач, который бы являлся общеизвестным, пока не существует, поэтому в целях исследования и совершенствования методов управления динамическими системами было предложено сформировать набор задач.

При нахождении решения терминальных задач управления со свободным временем, алгоритм решения дифференциального уравнения обновляется для каждой альтернативы, поскольку каждое решение содержит свою оценку конечного времени работы системы, достаточного для решения задачи. Для представления точек переключений как они есть, необходимо учитывать, что за счет штрафов будет происходить первоначальное упорядочивание точек внутри альтернативы. Для задач, в которых одним из условий является нахождение всех переключений внутри интервала регулирования, поиск управления будет малоэффективным до тех пор, пока точки не будут удовлетворять условию допустимости, т.е. условие равенства соответствующей штрафной функции нулю. Для повышения эффективности алгоритма на начальных популяциях, необходимо

стартовую популяцию сгенерировать так, чтобы большинство или все решения принадлежали множеству допустимых решений.

Таким образом, чтобы повысить эффективность поиска, рекомендуется изменить генерирование стартовой популяции. Здесь $o_j \in R, j = \overline{1, k}$ - все вещественные гены стартовой популяции, которым соответствуют точки переключений, а r_e - настраиваемый параметр :

$$1 - \text{для задач (3.5) и (3.6): } o_j \sim U\left(0, \frac{r_e}{k}\right);$$

$$2 - \text{для задач (3.7) и (3.8): } o_j \sim U\left(0, \frac{T}{k}\right);$$

$$3 - \text{для задач (3.9) и (3.10): } o_j \sim U\left(o_{j-1}, \frac{r_e}{k}\right), o_0 = 0, r_e > 0;$$

$$4 - \text{для задач (3.11) и (3.12): } o_j \sim U\left(o_{j-1}, \frac{T}{k}\right), o_0 = 0.$$

В данной работе разыгрывание стартовой популяции осуществлялось по предложенным правилам.

Ниже представлены задачи для тестирования работы алгоритма. По результатам решения задач были оценены наиболее эффективные настройки алгоритма. В таблице 3.1 представлены системы, граничные условия траекторий, заданы количество точек переключений функции управления и время интегрирования, т.е. время управления динамической системой. Рассматриваются задачи с фиксированным временем и системы с функцией управления, соответствующей идеальному реле. Для каждой задачи решение находилось с применением критериев (3.5), (3.7), (3.9), (3.11), и соответствующих им представлений решения.

В таблице 3.2 рассматриваются аналогичные задачи, но при свободном времени работы системы. Критерии для каждой задачи были (3.6), (3.8), (3.10), (3.12). При этом для критериев (3.6) и (3.10) возможно два различных способа представления конечного времени управления системой.

Таблица 3.1

Системы для решения терминальной задачи с фиксированным временем. Идеальное реле.

Система	Число точек переключений	Время интегрирования
$x' + 2 \cdot \sin(x) = u(t), x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} -3 \\ 0 \end{pmatrix}$	10	5
$x' \cdot x^2 + x \cdot u(t) = 0, x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} 3 \\ -2 \end{pmatrix}$	10	5
$x' + 2 \cdot \sin(x) = u(t), x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} -3 \\ 0 \end{pmatrix}$	15	10
$x' \cdot x^2 + x \cdot u(t) = 0, x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} 3 \\ -2 \end{pmatrix}$	15	10

Таблица 3.2

Системы для решения терминальной задачи со свободным временем. Идеальное реле.

Система	Число точек переключений	Амплитуда реле
$x' + 2 \cdot \sin(x) = u(t), x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} -3 \\ 0 \end{pmatrix}$	10	2.5
$x' \cdot x^2 + x \cdot u(t) = 0, x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} 3 \\ -2 \end{pmatrix}$	10	7
$x' + 2 \cdot \sin(x) = u(t), x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} -3 \\ 0 \end{pmatrix}$	10	5
$x' \cdot x^2 + x \cdot u(t) = 0, x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} 3 \\ -2 \end{pmatrix}$	10	10

В таблицах 3.3 и 3.4 представлены задачи терминального управления для тех же систем, но для исполнительного механизма в виде трехуровневого реле. Таким образом, аналогично предыдущим задачам из таблицы 3.1, для таблицы 3.3 каждая из них решалась по критериям (3.5), (3.7), (3.9) и (3.11) с

соответствующими представлениями точек переключения для каждого критерия. Для задач из таблицы 3.4, алгоритм запускался с критериями (3.6), (3.8), (3.10) и (3.12), где для (3.6) и (3.10).

Таблица 3.3

Системы для решения терминальной задачи с фиксированным временем.

Система	Число точек переключений	Время интегрирования
$x' + 2 \cdot \sin(x) = u(t), x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} -3 \\ 0 \end{pmatrix},$ $U = \{-2.5, 0, 2.5\}$	15	5
$x' \cdot x^2 + x \cdot u(t) = 0, x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} 3 \\ -2 \end{pmatrix}, U = \{-7, 0, 7\}$	15	5
$x' + 2 \cdot \sin(x) = u(t), x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} -3 \\ 0 \end{pmatrix}, U = \{-5, 0, 5\}$	20	10
$x' \cdot x^2 + x \cdot u(t) = 0, x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} 3 \\ -2 \end{pmatrix},$ $U = \{-10, 0, 10\}$	20	10

Таблица 3.4

Системы для решения терминальной задачи с свободным временем и системы с фиксированным временем и настраиваемым множеством допустимых управлений.

Система	Число точек переключений	Параметры задачи
$x' + 2 \cdot \sin(x) = u(t), x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} -3 \\ 0 \end{pmatrix}$	10	$U = \{-2.5, 0, 2.5\}$
$x' \cdot x^2 + x \cdot u(t) = 0, x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} 3 \\ -2 \end{pmatrix}$	10	$U = \{-7, 0, 7\}$
$x' + 2 \cdot \sin(x) = u(t), x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} -3 \\ 0 \end{pmatrix}, N_u = 3$	15	$T = 5$
$x' \cdot x^2 + x \cdot u(t) = 0, x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} -3 \\ 0 \end{pmatrix}, N_u = 3$	15	$T = 10$

Для задач из таблиц 3.3 и 3.4 решение может быть найдено только с помощью обобщенного алгоритма ЭС, в то время как для управления в виде идеального реле может быть использована упрощенная схема нахождения решения, которая была изложена в параграфе 3.1. текущей главы. Преимуществом предложенной схемы является то, что благодаря требованиям о перемене знака (или режима) в каждой точке переключения, функция управления может быть задана только набором точек переключения и значением функции управления в начальный момент времени. С другой стороны, подобная схема усиливает взаимное влияние длин интервалов регулирования каждого режима, что затрудняет настройку количества точек переключений.

По результатам тестирования алгоритма при различных настройках, были определены эффективные в среднем настройки, для различных задач, критериев и схем представления функции управления: 100 индивидов на 100 популяций, турнирная селекция с объемом турнира 20% от популяции, дискретное скрещивание при $p_c = 0.9$, модифицированная мутация при $p_m = 1/k$, равномерное скрещивание для номинальных переменных и модифицированная мутация при $p_m^1 = 1/N_u$. Параметры локального спуска: $N_1 = 2 \cdot k$, $N_2 = k$, $N_3 = 5$ при $h_l = 0.05$. Далее используем найденные настройки для решения задачи управления КА.

3. 3. Решение двухточечной задачи управления космическим аппаратом.

Рассмотрим 4 различных задачи терминального управления на основе задачи о переводе спутника с одной геостационарной орбиты на другую.

Координаты спутника описываются нелинейным дифференциальным уравнением вида

$$f(x, t) = \left(x_2, \quad x_1 \cdot x_3^2 - \frac{1}{x_1^2}, \quad \frac{-u(t) - 2 \cdot x_2 \cdot x_3}{x_1}, \quad x_3 \right)^T. \quad (3.17)$$

Необходимо перевести объект из начального положения $x(0) = (1, 0, 1, 0.785)$ в конечное $x(T) = (1, 0, 1, T)$, за конечное время T . Для представленной нелинейной системы дифференциальных уравнений, которая описывает движение КА, и концов траектории, сформулируем еще 4 задачи для оценки наиболее эффективных настроек модифицированного гибридного обобщенного алгоритма ЭС. Дополнительные условия для задачи представлены в таблице 3.5.

Задачи таблицы 3.5 решались для тех же критериев, что и аналогичные сверху задачи.

Таблица 3.5

Условия для задачи перевода КА с одной геостационарной орбиты на другую.

Число точек переключений	Параметры задачи
10	$T = 10$
10	$U = \{-0.005, 0.005\}$
20	$T = 10, U = \{-A, 0, A\}$
20	$U = \{-0.005, 0, 0.005\}$

Важно отметить, что ввиду большой сложности и вычислительных затрат, решение задачи терминального управления занимает длительное время. Популяционные методы оптимизации являются стохастическими и требуют множественного вычисления целевой функции. Варьирование всех настроек для таких задач будет занимать неоправданно много времени. Тестирование алгоритма проводилось только для того, чтобы оценить наиболее эффективные настройки алгоритма и показать его применимость для решения таких задач. Еще одним фактором, указывающим на нецелесообразность проведения большого количества тестов для различных градаций настроек, является то, что нелинейные динамические системы могут и не иметь решение задачи терминального управления, а для различных систем, разные настройки, в общем случае, могут быть эффективными. Нет возможности подобрать лучшие в среднем настройки,

поскольку нет возможности оценить среднюю нелинейную систему, которая бы описывала динамический объект управления.

Хотелось бы отметить тот факт, что любой вероятностный алгоритм оптимизации сходится к решению по вероятности, если это решение существует. Конечно, речь идет о слабой сходимости алгоритма, но, с другой стороны, предложенный подход и алгоритм решения формируют не имеющий аналогов автоматический способ решения задач терминального управления в виде кусочно-постоянных функций (или программы режимов) для различных постановок задачи. Ниже, на некоторых примерах, будет показано, что алгоритм находит решение для поставленных задач поиска программного управления.

Для приведенных в параграфе 3.2 настроек, которые были приняты наиболее эффективными в среднем, рассмотрим решение представленных выше задач терминального управления и для идеального реле, и для многоуровневого реле.

Решим задачу перевода КА с одной геостационарной орбиты на другую. Система (3.17) и концы траекторий для двухточечной задачи управления приведены выше. Пусть требуется найти такое управление в виде идеального реле, чтобы КА был переведен за время $T=10$. Для решения задачи примем число точек переключений $k=10$, а в качестве критерия выберем функционал (3.5). При этом функция управления полностью определяется 11 параметрами: 10 точек переключений и амплитуда исполнительного механизма. Решение поставленной задачи есть ответ на вопрос, какой амплитуды управления хватит для перевода КА за время $T=10$. Точки переключения выражены через объективные параметры алгоритма, значит, решается задача без ограничений в виде неравенств. Последний объективный параметр хромосомы отвечает за амплитуду реле.

Одним из запусков программы было найдено решение, которое представлено ниже. Найденная функция управления изображена на рисунке 3.5 слева, а состояния системы представлены на рисунке 3.5 справа. Координаты системы в конечный момент времени при найденном управлении $x(10) = (1.0007, -0.0064, 1.0069, 10.002)$. Среднее значение критерия (3.6) для

20 запусков составило 0.0028. Число вычислений целевой функции ни в одном из запусков не превысило $4 \cdot 10^4$.

По виду функции управления на рисунке 3.5 слева можно сказать, что количество точек переключений, при отсутствии ограничения для них находиться внутри интервала управления, приводит к некоторой настройке точек. В конкретном примере, число переключений было ограничено 10, но алгоритм выделил решение с 5 переключениями, оставив остальные точки за точкой $T = 10$.

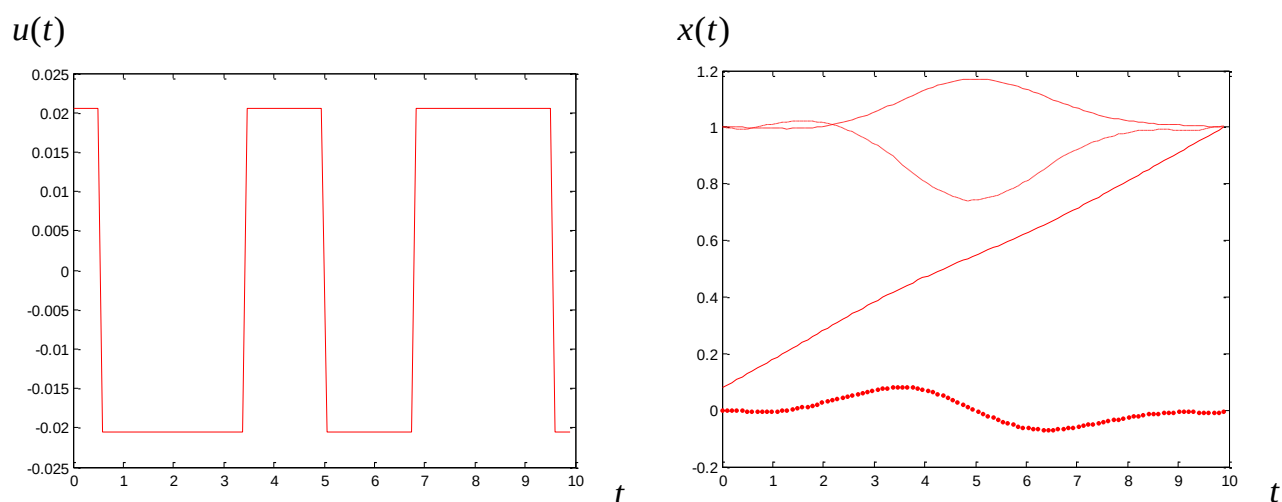


Рисунок 3.5. Функция управления $u(t)$, реле с двумя уровнями (слева). Состояние системы

$$x(t) = x_1(t), x_2(t), x_3(t), \frac{x_4(t)}{T} \text{ (справа).}$$

Далее рассмотрим решение этой же задачи терминального управления для (3.17), но при свободном времени и трехпозиционном исполнительном механизме, характеристики которого известны. Такая задача близка к реальности, поскольку в исходной постановке, двигатель КА может работать в трех режимах, например, $U_u = \{-0.005, 0, 0.005\}$. Решим задачу с критерием (3.6), выбрав $T = o_{k+1}$, что означает, что свободное время было закреплено за одним из вещественных генов. После 20 запусков алгоритма было выбрано лучшее решение: $T = 17.2$, $x(17.2) = (1.009, -0.0065, 1.0048, 17.21)$. При этом среднее значение целевой функции (3.6) по всем запускам составило 0.018. В ходе каждого решения целевая функция вычислялась не более $6.25 \cdot 10^4$ раз.

Функция управления представлена на рисунке 3.6 слева, а состояние системы – на рисунке 3.6 справа. Как видно по рисунку слева, количество переключений получилось меньшим 20. Подобная настройка числа точек происходит и в задачах со свободным временем, если не требуется выполнения условия нахождения всех точек внутри интервала. Схема, в которой время управления задано отдельной переменной, которая не связана со значениями других точек внутри интервала, позволяет осуществлять одновременную настройку параметров управления и конечного времени. При привязке времени управления к другим точкам внутри интервала, например, выражая общее время через сумму найденных переключений, зависимости между переменными и критерием становятся сложнее. Так, например, при коррекции одной точки переключения внутри интервала изменится и конечное время, что, возможно, не всегда приведет к улучшению результата и заметно усложнит локальное уточнение параметров функции управления. С другой стороны, если рассматривать точки переключений, как они есть, то актуальным станет решение задачи с функцией штрафа, которая отвечает за упорядоченность времен переключений внутри альтернативы.

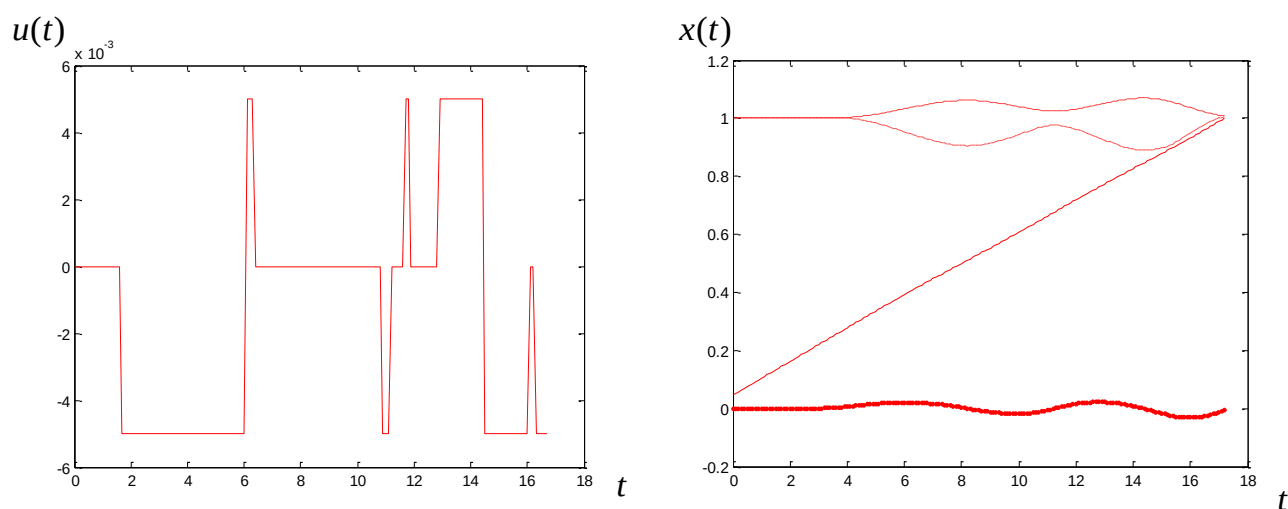


Рисунок 3.6. Функция управления $u(t)$, реле с тремя уровнями (слева). Состояние системы

$$x(t) = x_1(t), x_2(t), x_3(t), \frac{x_4(t)}{T} \text{ (справа).}$$

Решения этой же задачи но для 20 точек переключения и условиях $T = 10, U = \{-A, 0, A\}$ и $U = \{-0.005, 0.005\}$ имеют среднее значение пригодности 0.011 и 0.024, соответственно. Количество вычислений целевой функции не превысило $4 \cdot 10^4$ и $6.25 \cdot 10^4$, соответственно.

Как видно, предложенный алгоритм и разработанный подход эффективны для решения задач терминального управления, если, конечно, оно существует, для сложных объектов, которые заданы, в общем случае, нелинейными дифференциальными уравнениями. Рассмотренные задачи являются предшественниками задачам поиска управления для КА, в которых ограничения, которые накладываются на функцию управления таковы, что для их решения могут быть применены только методы прямого поиска.

Данный подход может быть применен и для решения задач оптимального управления для объектов с исполнительным механизмом, работающим по принципу многоуровневого реле. При этом постановка задачи будет пополнена некоторым интегральным критерием качества управления и/или состояний системы. Подход может быть так же обобщен и на задачи с дополнительными ограничениями. Более того, ввиду независимости подхода от природы критерия, могут рассматриваться задачи и с критериями особого вида. Тем не менее, решение подобных задач может привести к возникновению трудностей, связанных с образованием единого критерия, например, аддитивного или мультипликативного, которые будут обусловлены возможным противоречием функционалов. В общем случае, решение таких задач потребует увеличения вычислительных ресурсов или обобщения алгоритма до алгоритма поиска оптимальных решений для многокритериальных задач.

Выводы

В данной главе был рассмотрен подход к решению задачи терминального управления для динамических систем в виде кусочно-постоянной функции управления, где значения входного воздействия являются определенными

некоторым множество возможных значений, или программы переключения режимов. Решением задачи терминального управления является программное управление, которое реализует перевод системы из текущего состояния в желаемое за конечное время, которое, в свою очередь, может быть заданным или свободным. Двухточечная задача управления была сведена к экстремальной задаче на пространстве вещественных векторов и векторов с целочисленными координатами.

Функция управления полностью представлена набором из трех множеств. Каждое множество состоит из объектов одной природы: множество времен переключения режимов управления, множество значений функции управления, множество индексов соответствий каждому интервалу, ограниченному точками переключений, некоторому значению управляющего воздействия. При подобном представлении искомой функции, задача управления может быть решена через определение элементов неизвестных множеств, как решение задачи поиска экстремума целевой функции, отражающей, насколько точно решена двухточечная задача управления. Таким образом, решение задачи для каждого входящего воздействия оценивается по невязке между вектором конечного состояния системы и выходом модели при данном управлении.

Поскольку функция определяется множествами, одно из которых состоит из натуральных чисел, соответствующих номинальным переменным, то необходим алгоритм оптимизации, работающий на смешанных пространствах, одним из которых является пространство натуральных чисел, а другим - вещественных. Таким образом, для решения задачи оптимизации необходимо провести ее декомпозицию на две различных задачи: поиск экстремума на пространстве вещественных векторов при заданном значении номинальных переменных; поиск экстремума на пространстве натуральных чисел при заданном значении вещественных коэффициентов. Ввиду того, что целевая функция не может быть представлена как сумма функций вещественных переменных и номинальных, то и декомпозиция решения подразумевает решение задачи для одних переменных, при фиксированных значениях других, и наоборот.

Для решения экстремальной задачи с обозначенными типами переменных был модифицирован алгоритм эволюционных стратегий. Согласно общей концепции метода, были добавлены операторы рекомбинации и мутации и интерпретированы стратегические параметры алгоритма для генов, заданных натуральными числами. Все операторы были подобраны с учетом природы переменных. Модифицированный алгоритм с локальным поиском для вещественных переменных позволяет находить решение задачи внутри одного цикла, без деления задачи на два блока оптимизации.

Были рассмотрены различные варианты представления генов решения в виде элементов множеств, определяющих управление, поскольку множество точек переключений, по определению, упорядочено. При этом рассматривались различные варианты подбора свойств элементов множеств. Более того, приведены критерии для задачи терминального управления в различных постановках, для каждой из которых было предложено несколько вариантов представления решения, - функции управления.

Алгоритм прошел проверку при решении множества задач поиска терминального управления в различных постановках и при различных представлениях управляющего воздействия, что свидетельствует о том, что предложенный подход и алгоритм оптимизации действительно применимы для решения двухточечных задач управления. Были найдены наиболее эффективные параметры работы алгоритма для данного набора тестовых задач.

Различные постановки задачи терминального управления порождают различные комбинации переменных экстремальной задачи минимизации невязки между концом траектории при текущем управлении и желаемым положением. Различные способы представления решений, в свою очередь, определяют ограничения, которые будут накладываться на каждую альтернативу.

В общем случае, когда задача на поиск управления может быть приведена к задаче оптимального управления в аналитическом виде, следует применять методы решения, которые бы использовали информацию, содержащуюся в математической модели. Такие методы позволяют находить решение сложных

вариационных задач, учитывая все особенности системы, а так учитывать критерии оптимальности, которые могут быть зависимыми как от функции управления, так и от функций состояний системы, значения конечного времени управления. Поэтому следующая глава посвящена нахождению решения задач оптимального управления, и именно, посредством применения непрямого метода решения таких задач.

ГЛАВА IV. РЕШЕНИЕ ЗАДАЧИ ОПТИМАЛЬНОГО УПРАВЛЕНИЯ ЧИСЛЕННО-АНАЛИТИЧЕСКИМ НЕПРЯМЫМ МЕТОДОМ

В данной главе рассматривается экстремальная задача, которая возникает при нахождении оптимального управления, непрямым численно-аналитическим методом. Задачи оптимального управления, как и задачи вариационного исчисления не новы, и исследуются уже длительное время. Тем не менее, стремительное развитие такие задачи получили лишь в 20-м веке. О значимости, приложениях и методах решения задач оптимального управления говорится в работе [41], [69]. Там же рассматриваются различные методы решения задач, как аналитические, так и численные.

Выше уже говорилось о том, что методы эволюционного поиска применяются для решения задач оптимального управления для прямых методов решения задачи. Ниже будет рассмотрен косвенный метод нахождения оптимального управления и алгоритм решения экстремальной задачи, к которой этот метод приводит исходную задачу.

Несомненным преимуществом рассматриваемого косвенного метода является то, что задача оптимального управления с интегральным функционалом и терминальными ограничениями приводится к оптимизационной задаче на евклидовом пространстве, размерность которого совпадает с размерностью вектора состояний системы. Тем не менее, при этом возникает иная сложность – окрестность экстремума является неизвестной, равно, как и, в общем случае, неизвестна возможность существования решения задачи.

Применение вероятностных популяционных алгоритмов для решения экстремальной задачи внутри косвенных методов рассматривается в работах [21], [26], [32]. Ниже рассмотрены некоторые особенности экстремальной задачи и приведено сравнение различных алгоритмов для поиска решения поставленной задачи, показано, что повышение эффективности оптимизационного алгоритма, как и для задач, которые рассматривались в предыдущих главах, связано с его подстройкой под класс решаемых задач.

В работе [61] рассмотрен подход, основанный на одновременном применении генетического алгоритма и метода стрельбы. Такой метод, как и [5], где описывается численно-аналитический подход, являются универсальным средством решения задачи оптимального управления. Так же существует множество методов решения задачи оптимального управления нелинейными системами [9], [30], [35], [38] и [73], в каждом из которых решение находится для определенного класса задач.

Рассматриваемый косвенный метод решения задачи оптимального управления, который тоже является численно-аналитическим изложен в работе [89]. В более поздней работе - [93], рассматривается оптимизационный алгоритм, который и был усовершенствован для эффективного нахождения решений экстремальных задач данного класса.

4. 1. Задача оптимального управления и непрямой метод численно-аналитического поиска решения.

Основные теоретические сведения о задаче оптимального управления, вариационных задачах, принципе максимума Понтрягина были взяты из источников [25], [70], [71]. В данном пункте будет изложена суть численно-аналитического непрямого метода решения задач оптимального управления для динамических систем.

Здесь рассмотрим двухточечные задачи управления с закрепленным временем и интегральным критерием оптимальности. Известно, что объект задан дифференциальным уравнением

$$\frac{dx}{dt} = f(x, u, t), \quad (4.1)$$

где

$f(\cdot): R^n \times R \times R^+ \rightarrow R^n$ - непрерывная по своим аргументам вектор-функция;

$x \in R^n$ - вектор состояний системы;

$u(t): R^+ \rightarrow R, u(t) \in U$ - кусочно-непрерывная функция управления;

U - множество допустимых значений функции управления или множество, определяющее допустимые управления;

n - размерность системы.

Необходимо найти управление $u(t)$, такое, что будут выполнены граничные условия для вектора состояний при заданном времени T ,

$$x(0) = x^0, x(T) = x^T, \quad (4.2)$$

а реакция системы и функция управления будут доставлять экстремум функционалу:

$$I(x, u) = \int_0^T F(x, u) dt \rightarrow \text{extr}. \quad (4.3)$$

Согласно принципу максимума Понтрягина, составим систему с сопряженными переменными p для задачи (4.1)-(4.3). Для этого определим Гамильтониан:

$$H = -F(x, u) + p \cdot f(x, u, t), \quad (4.4)$$

Теперь, через Гамильтониан определим систему с переменными состояний и сопряженными переменными задачи оптимального управления:

$$\frac{dx}{dt} = f(x, u, t), \quad (4.5)$$

$$\frac{dp}{dt} = -\frac{dH}{dx}.$$

Задача Коши для системы дифференциальных уравнений (4.5) при заданных начальных условиях состояний системы $x(0)$ и сопряженных переменных $p(0)$, при известной функции управления, часто может быть решена численно.

Для того чтобы определить функцию $u(t)$ и замкнуть, таким образом, систему, будем использовать условие стационарности Гамильтониана (4.4) по управлению:

$$\frac{dH}{du} = 0. \quad (4.6)$$

Если задача (4.1)-(4.3) такова, что применение условия стационарности (4.6) позволяет однозначно определить функцию управления $u(t)$, то выраженное через другие переменные задачи управление может быть подставлено в систему (4.5). Теперь, решение задачи Коши будет полностью определяться начальными положениями $x(0)$ и $p(0)$.

По условиям задачи (4.2) начальное положение состояний системы является известным, в то время как начальное положение для сопряженных переменных остается неизвестным.

Определим начальное положение сопряженных переменных $p(0)$ через граничные условия состояний системы, а именно, правый конец траектории, который является известным, (4.2). Таким образом, для решения задачи (4.1)-(4.3) необходимо найти такое начальное условие для сопряженных переменных, что при решении задачи Коши для полной системы (4.5) выполнялось условие для вектора состояния системы $x(T) = x^T$.

Таким образом, задача оптимального управления (4.1)-(4.3) для которой выполняется условие (4.6) сводится к безусловной экстремальной задаче на пространстве R^n , где $n = \dim(x)$ - размерность системы.

Пусть обозначение $\tilde{x}(t), \tilde{p}(t)|_{\tilde{p}(0)=p^0}$ соответствует решению системы (4.5) при начальном положении для сопряженных переменных $\tilde{p}(0) = p^0$. Используя данное обозначение, введем критерий оптимальности, который представляет собой невязку между конечным положением решения системы $\tilde{x}(T)|_{\tilde{p}(0)=p^0}$ при $\tilde{p}(0) = p^0$ и x^T :

$$K(p^0) = \|x^T - \tilde{x}(T)|_{\tilde{p}(0)=p^0}\| \rightarrow \min_{p^0 \in R^n}. \quad (4.7)$$

При значении критерия (4.7) равном 0, будем считать, что функция $u(t)$ и связанный с ней процесс (4.1), удовлетворяют условиям слабого оптимума для задачи оптимального управления.

Переход от критерия (4.7), определяющего невязку между конечными состояниями системы с заданными начальными условиями для сопряженных переменных и желаемыми конечными условиями для вектора состояний системы к функции пригодности, согласно (1.3), приведет к выражению

$$fit(p^0) = \frac{1}{1 + K(p^0)}. \quad (4.8)$$

Предложенный метод предполагает решение задач оптимального управления в том случае, если Гамильтониан (4.4) для задачи (4.1)-(4.3) таков, что выполняется условие (4.6). Конечно, однозначное выражение функции управления $u(t)$ через условие стационарности выполнимо далеко не для всех задач.

В данной работе рассматриваются те задачи, где условие (4.6) выполнимо. Но данное требование необходимо только для замыкания системы (4.5), чтобы впоследствии задача определения вектора начальных состояний $p(0)$ была связана только с решением задачи Коши. Управление выражается через вектор состояний системы и сопряженные переменные:

$$\frac{dH}{du} = 0 \Rightarrow u(t) = F_u(p(t), x(t)), \Rightarrow u(t) = \tilde{F}_u(t),$$

где

$F_u(p(t), x(t))$, $\tilde{F}_u(t)$ - некоторые кусочно-непрерывные функции, ввиду свойства функции управления $u(t)$;

после чего, полученное выражение заменяет управление в системе (4.5).

В общем случае, метод, как и приведенная экстремальная задача, применим и для задач оптимального управления, для которых не выполняется условие стационарности. В подобных случаях вместо стационарности требуется выполнение принципа максимума Понтрягина, и решение задачи Коши будет

неразрывно связано с определением такого управления, чтобы выполнялось условие максимума.

4. 2. Экстремальная задача численно-аналитического метода

Рассмотрим применение численно-аналитического метода на примерах решения задач оптимального управления некоторыми нелинейными системами. Непрямой метод решения задачи опирается на символьные преобразования, которые были указаны выше: составление Гамильтониана (4.4), составление системы с сопряженными переменными (4.5), нахождение частной производной (4.6) и символьные преобразования внутри системы (4.5), результатом которых является запись системы в известных переменных.

Для того, чтобы представить аналитическую часть метода, результатом которой является система дифференциальных уравнений, которая вместе с начальными условиями представляет собой задачу Коши, решаемую численно рассмотрим задачу оптимального управления.

Дана система нелинейных дифференциальных уравнений, которая описывает движение тележки с перевернутым маятником. Уравнение системы с граничными условиями и функционал, соответственно

$$f(x, t, u) = \begin{pmatrix} x_1 \\ -x_1 + \sin(x_0) + u(t) \cdot \cos(x_0) \end{pmatrix}, \quad T = 5, \quad x^0 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad x^T = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad (4.9)$$

$$F(x, u) = u^2 + x_0^2. \quad (4.10)$$

По условию стационарности (4.6) находим $\frac{dH}{du} = 0 \rightarrow u(t) = \frac{p_1 \cdot \cos(x_0)}{2}$, тогда система с сопряженными переменными примет вид

$$F_p(x, p, t) = \begin{pmatrix} x_0 \\ -x_1 + \sin(x_0) + \frac{p_1 \cdot \cos(x_0)}{2} \cdot \cos(x_0) \\ -2 \cdot x_0 + p_1 \cdot \cos(x_0) - \frac{p_1 \cdot \cos(x_0)}{2} \cdot \sin(x_0) \\ -p_1 + p_0 \end{pmatrix}.$$

Теперь, для различных начальных условий сопряженных переменных можно вычислить значение критерия (4.7), а, значит, функцию пригодности (4.8). Поскольку размерность рассматриваемой системы равна 2, то и количество переменных задачи оптимизации равно 2. Отсюда мы можем продемонстрировать наглядно поверхность отклика целевой функции для экстремальной задачи непрямого метода решения задачи оптимального управления (4.9)-(4.10) в виде графика. Для наглядного представления, произведем расчет функции пригодности на сетке с шагом 0.05 в каждом направлении. Функция пригодности представлена на рисунке 4.1.

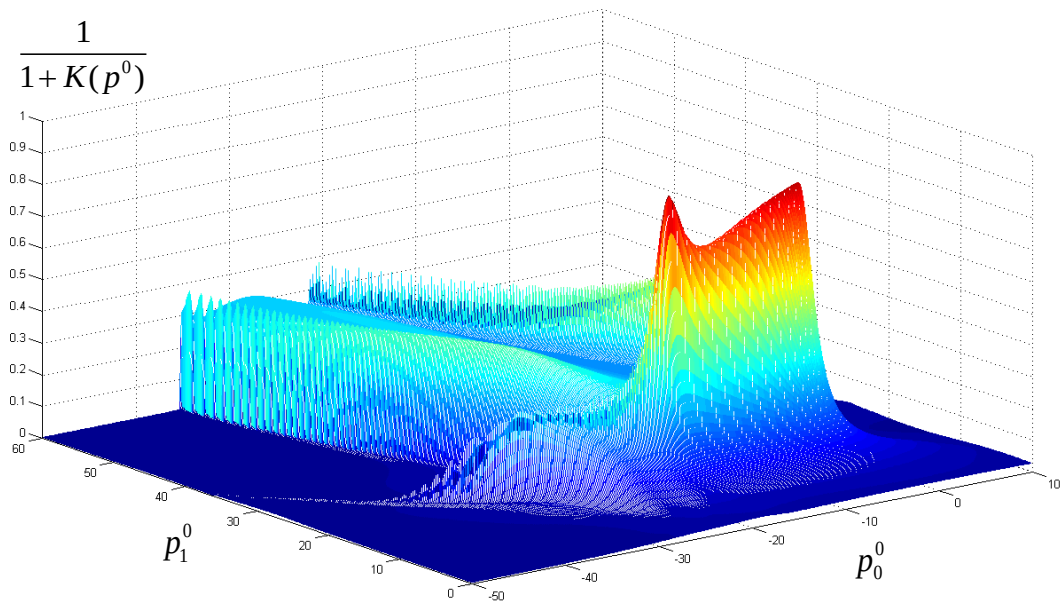


Рисунок 4.1. Функция пригодности для задачи оптимального управления

На рисунке 4.1 видно, что единственным допустимым решением для рассматриваемой задачи, является, округленное до десятых, $p^0 = \begin{pmatrix} 8.9 \\ -22.1 \end{pmatrix}$. При

этом, заметим, что область, содержащая эту точку, имеет гораздо меньшую зону притяжения, чем другие локальные экстремумы.

На рисунках 4.2 и 4.3 представлено полученное решение задачи оптимального управления, на рисунке 4.2 – состояние системы, на рисунке 4.3 – функция управления.

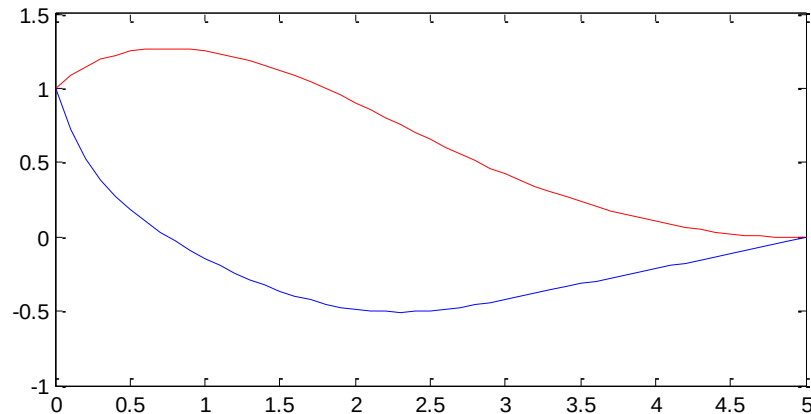


Рисунок 4.2. Решение задачи оптимального правления (4.7)-(4.8). Оптимальный процесс.

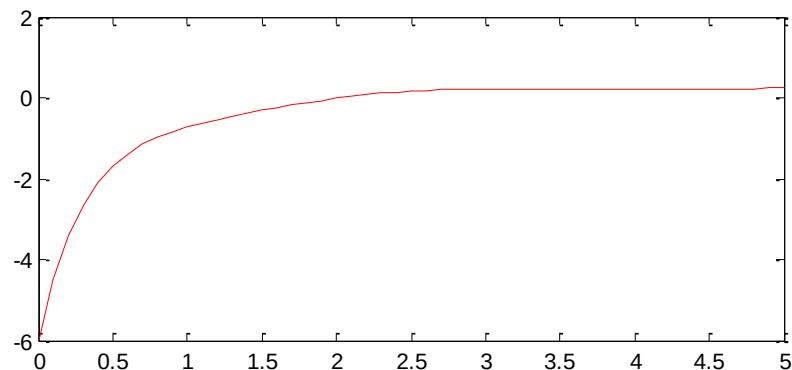


Рисунок 4.3. Решение задачи оптимального правления (4.7)-(4.8). Оптимальное управление.

Далее рассмотрим другую задачу оптимального управления, для того, чтобы продемонстрировать многоэкстремальность и сложность целевой функции. Для системы, заданного времени и концов слева и справа:

$$f(x, t, u) = \begin{pmatrix} \ln(x_1) + \cos(x_0) \\ t \cdot \sin(x_1) + u(t) \end{pmatrix}, \quad T = 1, \quad (4.9)$$

$$x^0 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad x(T) = \begin{pmatrix} 2 \\ 5 \end{pmatrix}, \quad (4.10)$$

необходимо найти функцию управления, удовлетворяющую граничным условиям и доставляющая экстремум функционалу

$$F(x, u) = u^2 \rightarrow \min_u. \quad (4.11)$$

Определим расширенную систему, содержащую сопряженные переменные:

$$F_p(x, p, t) = \begin{pmatrix} \ln(x_1) + \cos(x_0) \\ t \cdot \sin(x_1) + \frac{p_1}{2} \\ \sin(x_0) \cdot p_0 - t \cdot \cos(x_0) \cdot p_1 \\ \frac{-p_0}{x_1} \end{pmatrix},$$

после чего замыкаем систему, определяя структуру управления из условия стационарности гамильтониана по функции управления:

$$\frac{dH}{du} = 0 \rightarrow u(t) = \frac{p_1}{2}.$$

Теперь, как и для предыдущей задачи, приведем функцию пригодности, которая изображена на рисунке 4.4.

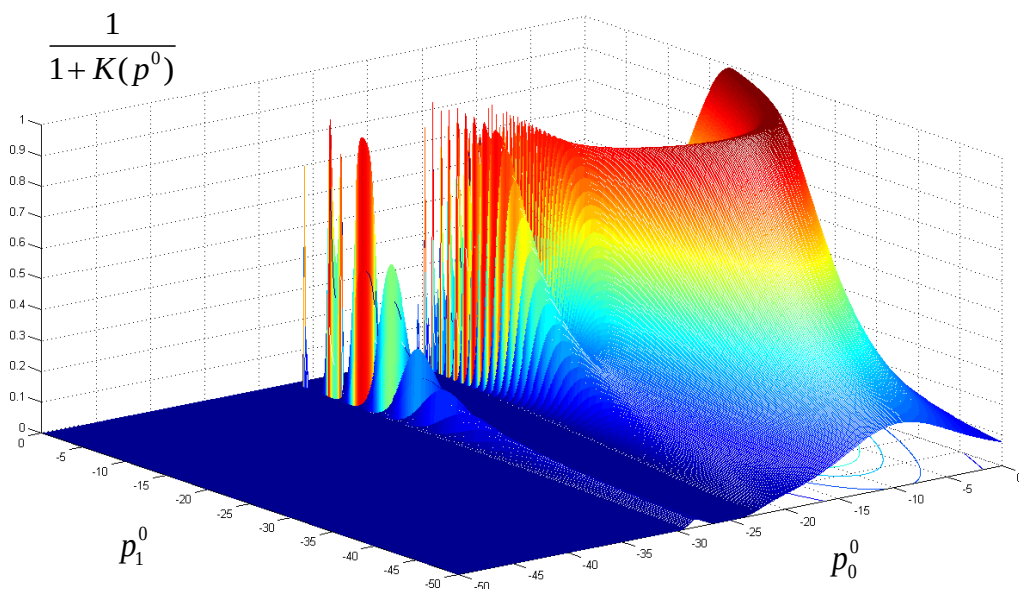


Рисунок 4.4. Функция пригодности для задачи оптимального управления

На рисунках 4.5 и 4.6 представлено полученное решение задачи оптимального управления, на рисунке 4.5 – состояние системы, на рисунке 4.6 – функция управления.

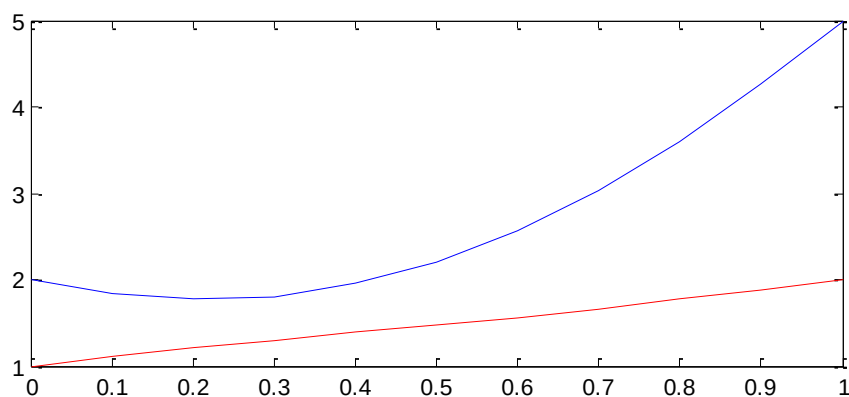


Рисунок 4.5. Решение задачи оптимального правления (4.9)-(4.11). Оптимальный процесс.

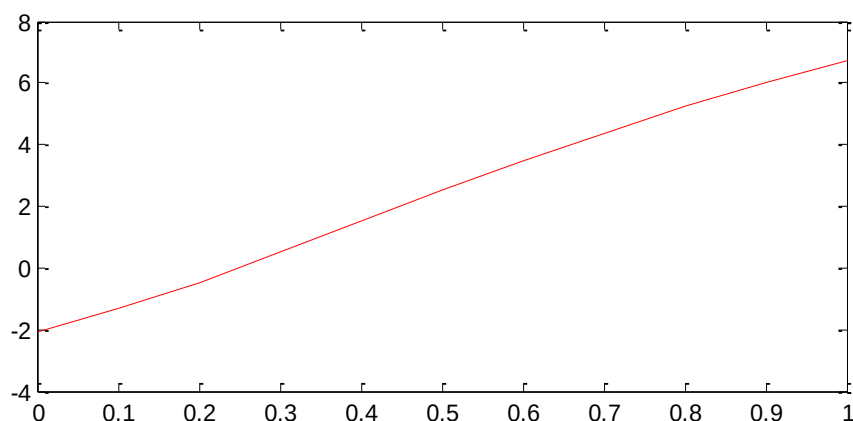


Рисунок 4.6. Решение задачи оптимального правления (4.9)-(4.11). Оптимальный процесс.

Как видно по рисункам 4.1 и 4.4, задачи на поиск экстремума функционала (4.7) по всем возможным значениям вектора начальных координат для сопряженных переменных задачи оптимального управления являются сложными, и многоэкстремальными. При этом неизвестна область нахождения глобального оптимума и, как можно убедиться по представленным рисункам, область притяжения глобального оптимума мала. И эта область мала не только по сравнению с общей областью притяжения других оптимумов, которые не могут являться решениями задачи оптимального управления, но и в некоторых случаях, меньше областей притяжения для отдельных оптимумов.

4. 3. Методы стохастической оптимизации в решении экстремальных задач численно-аналитического непрямого метода нахождения оптимального управления.

Гибридный модифицированный алгоритм эволюционных стратегий (далее HMES) для решения экстремальной задачи в непрямом методе нахождения оптимального управления динамическими системами был предложен в работах [49], [92].

Для решения экстремальной задачи рассматривались некоторые распространенные методы оптимизации, суть которых изложена в параграфе 1.4: стандартный алгоритм эволюционных стратегий (ES), метод дифференциальной эволюции (DE), стайный алгоритм оптимизации (PSO) и алгоритм эволюционных стратегий с адаптацией ковариационной матрицы (CMA-ES). Для представленной выше задачи приведем оценку вероятности нахождения глобального оптимума разными алгоритмами с их наиболее эффективными настройками.

Основные операторы поиска для ЭС представлены выражениями (1.7)-(1.9), (1.10)-(1.13) и (1.14)-(1.17), (1.18)-(1.26), селекция, рекомбинация и мутация соответственно. Гибридный модифицированный алгоритм ЭС, как и для решения вышеприведенных задач, включает в себя локальный поиск, схема которого представлена на рисунке 1.3., модифицированную мутацию и измененные операторы поиска, указанные в главе 1.2. Основные операторы поиска для алгоритма дифференциальной эволюции и стайного алгоритма определяются выражениями (1.26) и (1.27)-(1.28), соответственно.

Для сравнения эффективности алгоритмов глобального поиска экстремума число вычислений целевой функции было выбрано равным 8000. При указанном числе вычислений целевой функции возможны различные комбинации числа индивидов и популяций. Отметим, что для гибридного алгоритма эволюционных стратегий учитывались и вычисления при локальном улучшении решений, поэтому алгоритм останавливался, не доходя до последней популяции. Число индивидов и популяций варьировались и для каждого алгоритма проверялись следующие пары: 1600 популяций и 5 индивидов в каждой, 800 популяций и 10

индивидов, 400 популяций и 20 индивидов, 200 популяций и 40 индивидов, 100 популяций и 80 индивидов.

Поскольку операторы поиска во всех представленных алгоритмах отличаются друг от друга, как отличаются и идеи, на которых они основаны, то для каждого оператора поиска варьировались параметры и рассматривались решения при различных комбинациях значений параметров. Для метода эволюционных стратегий, как и для его гибридного и модифицированного аналога были рассмотрены различные виды селекций: турнирная с объемами 2, 5% и 10%; скрещивания: среднее, средневзвешенное, дискретное, случайное; мутация: стандартная и модифицированная при $p_m = 1/k$. Для метода дифференциальной эволюции варьировался коэффициент $F \in \{a_i = 0.2 \cdot i : i \leq 10, i \in N\}$ при рекомендованном $C_r = 0.5$. В стайном алгоритме: $\omega \in \{0.5 \cdot i : i \leq 4, i \in N\}$ и $\phi_{1,2} \in \{0.4 \cdot i : i \leq 6, i \in N\}$. Для алгоритма с адаптацией ковариационной матрицы настройки выбирались согласно рекомендациям, [18].

Для всех алгоритмов начальная популяция генерировалась одинаковым образом, так, что $op \sim N(0,10)$, $sp \sim N(0,1)$, $v \sim N(0,1)$. Такой разброс стартовой популяции обусловлен тем, что неизвестна окрестность, в которой находится глобальный оптимум задачи, а поверхность целевой функции может быть достаточно сложной, как уже было показано на рисунках 4.1 и 4.4.

Для проверки эффективности алгоритма, рассматривались решения экстремальных задач численно-аналитического метода нахождения оптимального

управления для системы $f(x, t, u) = \begin{pmatrix} \ln(x_1) + \cos(x_0) \\ t \cdot \sin(x_1) + u(t) \end{pmatrix}$, и функционала

$F(x, u) = u^2 \rightarrow \min_u$, остальные условия которых приведены в таблице 4.1.

Таблица 4.1

Задачи для проверки эффективности оптимизационных алгоритмов.

Концы траектории и конечное время	
$T = 1, x^0 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, x(T) = \begin{pmatrix} 2 \\ 5 \end{pmatrix}$	$T = 1, x^0 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$
$T = 0.7, x^0 = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix}, x(T) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$T = 0.5, x^0 = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix}, x(T) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$
$T = 1.5, x^0 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, x(T) = \begin{pmatrix} 2 \\ 4 \end{pmatrix}$	$T = 1, x^0 = \begin{pmatrix} 1 \\ 4 \end{pmatrix}, x(T) = \begin{pmatrix} 0 \\ 4 \end{pmatrix}$

Для задач, которые представлены в таблице 4.1, для каждого алгоритма при всех различных настройках, описанных выше, различных объемах популяций и при ограничении вычислений целевой функции числом 8000, было сделано 20 запусков. Результаты были усреднены.

Проверим эффективность алгоритмов, так же при различных настройках, для задач оптимального управления для динамической системы, определенной следующим уравнением: $f(x, t, u) = \begin{pmatrix} x_1 \\ -x_1 + \sin(x_0) + u(t) \cdot \cos(x_0) \end{pmatrix}$, функционале качества $F(x, u) = u^2 + x_0^2$ и остальных параметрах, представленных в таблице 4.2.

Далее, проведем аналогичные исследования для задач оптимального управления [90], порожденных системой дифференциальных уравнений

$$f(x, t, u) = \begin{pmatrix} x_1 + 1 \\ t \cdot x_0^{\frac{5}{2}} + u(t) \end{pmatrix}, \text{ функционалом } F(x, u) = (2 \cdot x_0)^2 + x_1^2 + u^2 \text{ и другими}$$

параметрами задачи, представленными в таблице 4.3.

Таблица 4.2

Задачи для проверки эффективности оптимизационных алгоритмов.

Концы траектории и конечное время	
$T = 5, x^0 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$T = 3, x^0 = \begin{pmatrix} 3 \\ 3 \end{pmatrix}, x(T) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$
$T = 6, x^0 = \begin{pmatrix} 3 \\ -3 \end{pmatrix}, x(T) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$T = 2, x^0 = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, x(T) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$
$T = 2.5, x^0 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, x(T) = \begin{pmatrix} 5 \\ 0 \end{pmatrix}$	$T = 1, x^0 = \begin{pmatrix} 0.5 \\ -0.5 \end{pmatrix}, x(T) = \begin{pmatrix} -0.5 \\ 0.5 \end{pmatrix}$

Определим для каждого алгоритма настройки, при которых были найдены решения задачи с наименьшими в среднем значениями критерия (4.7). Далее, для каждого лучшего в этом наборе алгоритма приведем среднее значение пригодности решения по всем запускам, по всем представленным задачам. Средние значения функции пригодности по всем задачам и по всем запускам представлены на рисунке 4.7.

Таблица 4.3

Задачи для проверки эффективности оптимизационных алгоритмов.

Концы траектории и конечное время	
$T = 1, x^0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, x(T) = \begin{pmatrix} 2 \\ 3 \end{pmatrix}$	$T = 2, x^0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, x(T) = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$
$T = 1, x^0 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, x(T) = \begin{pmatrix} 1 \\ 4 \end{pmatrix}$	$T = 2, x^0 = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix}, x(T) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$
$T = 0.5, x^0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, x(T) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$	$T = 1.5, x^0 = \begin{pmatrix} 1 \\ 3 \end{pmatrix}, x(T) = \begin{pmatrix} 2 \\ 3 \end{pmatrix}$

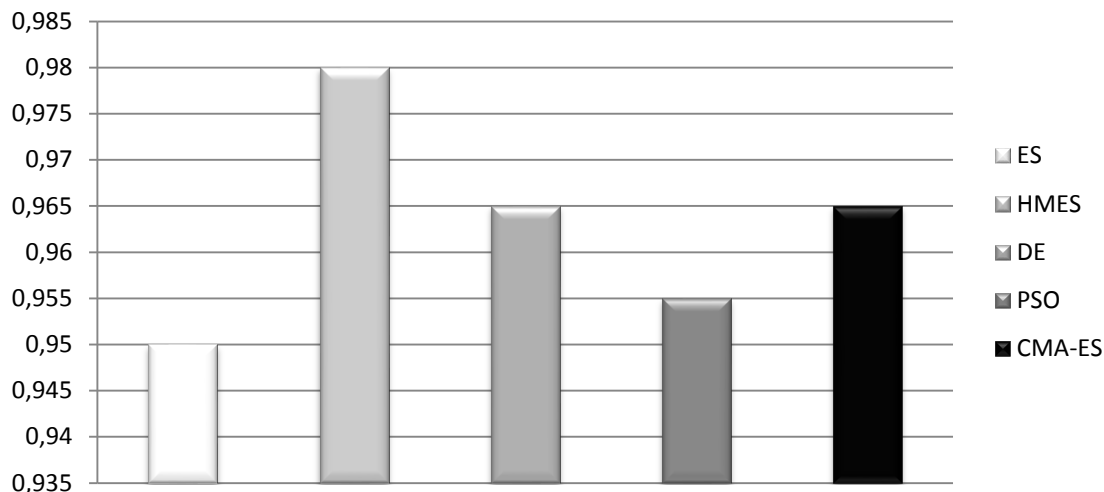


Рисунок 4.7. Средняя пригодность решений по всем задачам и всем запускам алгоритмов с лучшими настройками.

Как и для задачи идентификации линейных динамических систем рассматривалась оценка вероятности нахождения решения близкого к истинному, рассмотрим оценку вероятности алгоритма нахождения глобального оптимума. В параграфе 4.1 дано определение вероятности нахождения глобального оптимума задачи при определенном стартовом наборе значений и параметрах алгоритма.

Оценку вероятности зададим отношением $\hat{p}_{loc}^s = \frac{N_s^+}{N_s^+ + N_s^-}$, где N_s^+ - число найденных решений, близких по норме к глобальному оптимуму задачи, а N_s^- - число решений, которые не попали в окрестность глобального оптимума. Поскольку поверхность отклика целевой функции является сложной для описания, будем считать, что попадание в область глобального оптимума определяется не расстоянием от решения до точки оптимума, а срезом поверхности сечением с заданным уровнем $\alpha_c = 0.96$. Уровень был выбрана исходя из визуальной оценки графиков функций пригодности. Таким образом, ни один локальный оптимум в рассматриваемых задачах не достигает значения функции пригодности $\alpha_c = 0.96$.

Оценки вероятности представлены на рисунке 4.8.

Оценим корреляцию между вероятностью нахождения глобального оптимума и средней пригодностью индивида. Здесь, оценка коэффициента корреляции $\hat{K} = 0.662$. Значит, для решения экстремальной задачи численно-аналитического метода нахождения оптимального управления так же важна такая характеристика алгоритма оптимизации как его эффективность при нахождении точки глобального оптимума. Более того, она в большей мере определяет эффективность алгоритма, чем среднее значение величины критерия найденного решения. Значение коэффициента корреляции указывает на тот факт, что в данном случае один критерий не может быть выражен через другой линейной функцией. А поскольку зависимость между данными критериями остается неизвестной, то пока не представляется возможным выразить одну статистику через другую в принципе. Установление статистической зависимости, в данном случае, не приведет к оценке реальной, ввиду того, что значение данных статистик будет зависеть непосредственно от целевой функции и ее вида.

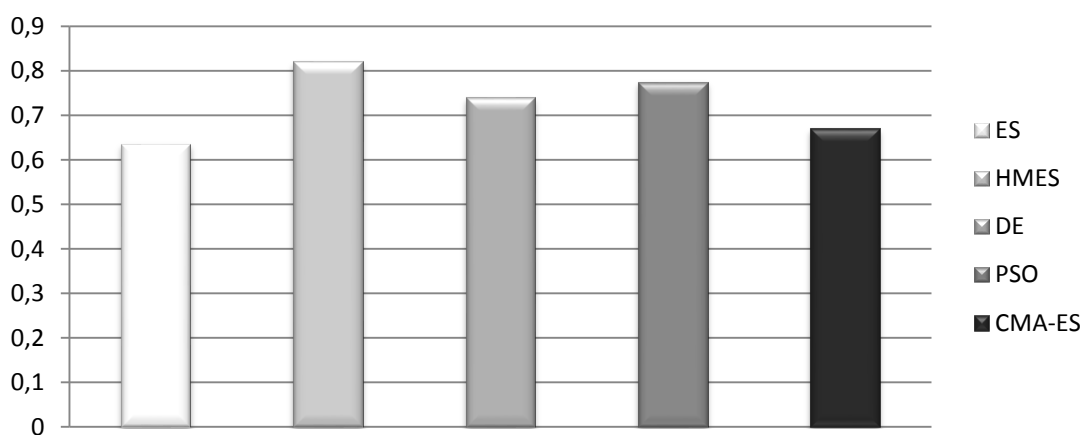


Рисунок 4.8. Оценка вероятности нахождения глобального оптимума для каждого алгоритма, средняя по всем запускам алгоритма и по всем задачам.

Вернемся к задаче (4.7)-(4.8) и для нее отдельно представим аналогичные оценки вероятностей нахождения глобального оптимума для каждого алгоритма. Сравнение оценок представлено на рисунке 4.9.

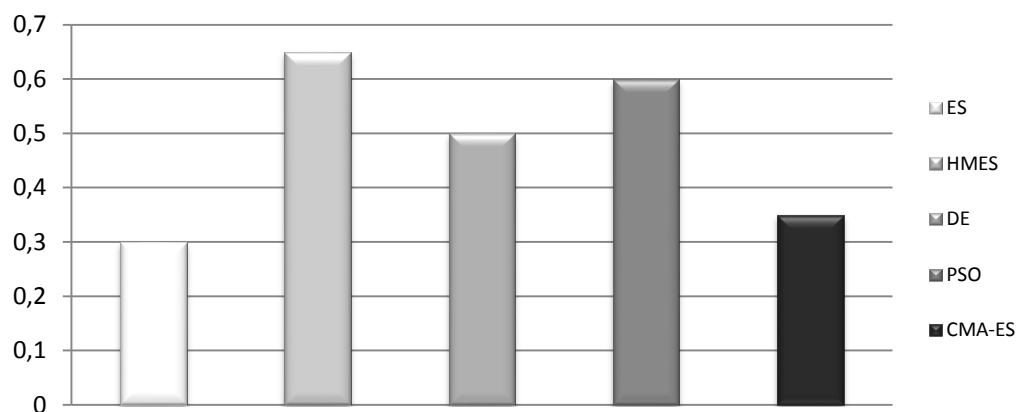


Рисунок 4.9. Оценка вероятности нахождения глобального оптимума для каждого алгоритма, решение экстремальной задачи для (4.7)-(4.8).

Еще раз отметим, что для данного класса задач, предложенный критерий определяет эффективность алгоритма в большей мере, чем среднее значение функции пригодности. Достаточно сравнить диаграммы на рисунках 4.9, 4.8 и 4.7, которые позволяют сделать другой вывод об эффективности каждого из алгоритмов оптимизации.

Судя по рисунку 4.9 можно сказать, что все алгоритмы не достаточно эффективны при данном объеме ресурсов, выделяемых на вычисление целевой функции. Одной из причин малой эффективности является невозможность алгоритма «выбраться» из зоны притяжения локального оптимума. Другой причиной является отсутствие априорных сведений о функции пригодности, и распределение начальной популяции, которое может и не включать точки, принадлежащие множеству приближений. Таким образом, всегда можно подобрать такое распределение, что весь объем популяции будет содержаться в некотором шаре вокруг локального оптимума или, наоборот, в окрестностях из которых при текущих настройках алгоритма глобальный оптимум достигим.

Для того чтобы уменьшить влияние указанных выше факторов, было предложено модифицировать схему работы алгоритма, добавив в нее новый контролирующий элемент – оператор перезапуска, который, применительно к данной задаче впервые был рассмотрен в работе [53].

Текущее исследование включает в себя варьирование всех возможных настроек гибридного модифицированного алгоритма. Согласно проведенным выше тестам алгоритмов, следующие настройки были выбраны как наиболее эффективные: 20 индивидов на 200 популяций, турнирная селекция (10%), дискретное скрещивание, модифицированная мутация при параметре $p_m = 0.75$, параметры локального спуска: $N_1 = N_2 = 10$, $N_3 = 1$, $h_l = 0.1$. Число вычислений целевой функции ограничим значением 8000.

Чтобы оценить степень влияния настроек на значение оценки эффективности работы алгоритма, проведем исследование на решении задачи (4.7)-(4.8), ввиду того, что для этой задачи статистика нахождения глобального оптимума для всех алгоритмов была хуже. Длина вектора памяти, координатам которого соответствуют значения целевой функции, варьировалась от 5 до 20, а параметр, определяющий пороговое значение между пригодностями лучших решений для осуществления перезапуска: 0.005, 0.01, 0.05, 0.1.

На рисунке 4.12 представлена зависимость числа запусков алгоритма от длины вектора памяти, нормированное на 40, для более наглядного внешнего представления.

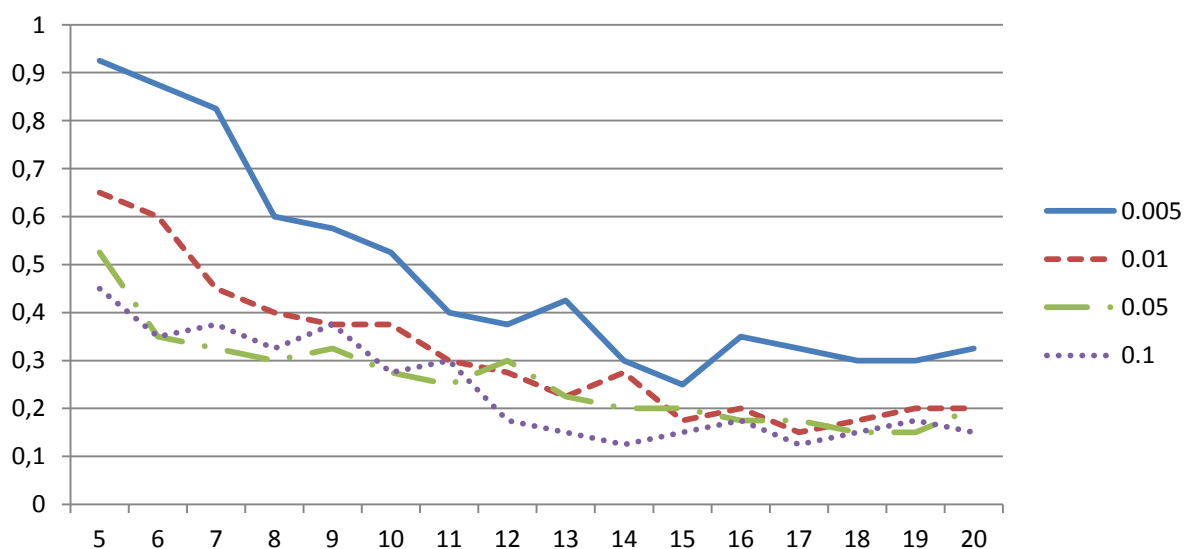


Рисунок 4.12. Зависимость числа запусков алгоритма от длины вектора памяти и значения порога изменения пригодности лучшего решения.

По данным исследования эффективности алгоритмов при различных настройках было определено, что модифицированный алгоритм, который был принят за основу и других алгоритмов решения экстремальных задач в данной работе, в среднем оказался наиболее эффективным по сравнению с другими общеизвестными и распространенными алгоритмами при различных настройках их операторов поиска.

Выводы

В главе был рассмотрен непрямой метод решения задачи оптимального управления, в котором задача оптимального управления сводится к экстремальной задаче, решение которой полностью определяет оптимальный процесс и оптимальное управление. Особенностью непрямого метода является то, что задача управления, включающая терминальные ограничения, интегральный функционал и, в общем случае, связи, которые являются дифференциальными и нелинейными, приводится к поиску экстремума функции на многомерном пространстве вещественных чисел. Любая точка в этом пространстве соответствует некоторому решению системы дифференциальных уравнений. А функцией является невязка между желаемым правым концом оптимальной траектории и конечным положением состоянии системы при заданных начальных условиях. В общем случае, функция рассчитывается численно. По представленным в главе рисункам видно, что целевая функция для приведенной задачи на поиск экстремума является сложной, многоэкстремальной и, как правило, содержит такие элементы как плато и пики. На одном из представленных примеров показано, что для некоторых задачах оптимального управления, область притяжения глобального оптимума значительно меньше области притяжения локальных оптимумов и даже некоторых локальных оптимумов в отдельности, которому соответствует большее значение критерия невязки.

В качестве различных алгоритмов решения экстремальной задачи рассматривались некоторые распространенные и зарекомендовавшие себя

методы. Сравнивались следующие алгоритмы оптимизации: метод эволюционных стратегий, гибридный модифицированный алгоритм эволюционных стратегий, алгоритм дифференциальной эволюции, стайный алгоритм оптимизации, алгоритм эволюционных стратегий с адаптацией ковариационной матрицы.

На примере решений некоторых задач оптимального управления было показано, что предложенный алгоритм является не только эффективнее в плане нахождения решения с наибольшей пригодностью, но и в плане более высокой вероятности найти точку глобального оптимума.

Тем не менее, для некоторых задач оптимального управления, точка глобального оптимума для соответствующей экстремальной задачи находятся с невысокой надежностью. Были выявлены возможные факторы, которые непосредственно влияют на эффективность алгоритма находить точки глобального оптимума. Отметим, что для целевой функции задачи оптимального управления, глобальному оптимуму соответствует значение пригодности равное 1. Была предложена дополнительная модификация алгоритма, которая осуществляет перезапуск популяции из другой стартовой точки, если выполняется одно из условий. Условия основываются на истории изменения пригодности лучшего решения за последние несколько популяций, на истории всех найденных за различные запуски алгоритма решений и координат текущего решения.

Предложенный оператор поиска основывается на оценке того, происходило ли улучшение найденного решения. Если при выбранных параметрах оценка изменений функции пригодности найденного решения меньше некоторого порогового значения, то осуществляется перезапуск популяции. При этом, текущее решение сохраняется в массив памяти. Следующий оператор поиска оценивает решение по его нахождению на пространстве поиска относительно ранее найденных решений, так что если текущее решение находится в окрестности некоторого ранее найденного, то тоже осуществляется перезапуск. Каждый перезапуск может быть осуществлен как с изменением начального распределения, так и без. Под изменением начального распределения, имеется в

виду изменение параметров закона распределения, по которому разыгрываются начальная популяция.

После включения данного оператора поиска в алгоритм поиска экстремума, оценка вероятности нахождения глобального оптимума для задачи (4.7)-(4.8) увеличилась с 0.65 до 1, причем при любых выбранных настройках предложенных операторов поиска, что говорит о существенном повышении эффективности поискового алгоритма.

ЗАКЛЮЧЕНИЕ

Результатами проведенной работы являются предложенные алгоритмы стохастического популяционного поиска, основанные на новом гибридном модифицированном аналоге алгоритма эволюционных стратегий, которые являются более эффективными в решении задач идентификации и управления для динамических систем, чем стандартный алгоритм эволюционных стратегий. Для каждой из приведенных задач алгоритм был изменен и дополнен операторами поиска, благодаря чему значительно повысилась вероятность нахождения решения исходной задачи, а, следовательно, и эффективность алгоритмов.

Исследован гибридный модифицированный алгоритм эволюционных стратегий для решения задач идентификации и управления динамическими системами. Рассматривались различные типы операторов поиска: предложены и добавлены различные схемы операторов селекции, скрещивания, мутации; модифицирован оператор мутации: для каждой пары генов индивида, мутация происходит с заранее установленной вероятностью. Гибридизация представляет собой совмещение двух различных алгоритмов поиска: популяционного алгоритма, основанного на алгоритме эволюционных стратегий, и стохастического поискового алгоритма нулевого порядка, который представлен случайным покоординатным спуском. Таким образом, гибридный модифицированный алгоритм лег в основу решения рассмотренных задач.

Одной из рассматриваемых задач была задача идентификации линейных динамических систем по данным наблюдений выхода объекта. Предложен подход, который позволяет автоматически оценивать порядок и коэффициенты дифференциального уравнения по имеющейся выборке измерений реакции системы на некоторое известное управление. Таким образом, рассматривается подход к решению обратной задачи математического моделирования, причем модель ищется в аналитическом виде, в виде линейного дифференциального уравнения с правой частью в виде известного управляющего воздействия. Данная задача является актуальной, а модели такого типа находят широкое применение и удобны в использовании для решения дальнейших задач. Экстремальная задача, к

которой была сведена задача идентификации линейных динамических систем, наиболее эффективно решается модифицированным гибридным алгоритмом эволюционных стратегий с разработанным оператором округления. Предложенный алгоритм является эффективнее классического алгоритма эволюционных стратегий и алгоритмов с отдельными модификациями, при различных включениях операторов поиска.

Другой рассматриваемой задачей была задача терминального управления для систем с исполнительным механизмом, работающим по принципу многоуровневое реле. Для таких исполнительных механизмов функция управления является кусочно-постоянной, с областью значений, которая совпадает с множеством допустимых значений управления, определяемых характеристиками исполнительного механизма. Рассматривается прямой метод решения двухточечной задачи управления, который подходит для различных систем: линейных или нелинейных, стационарных или нестационарных. Подход является универсальным, поскольку не требует модели системы в аналитическом виде, и применим для решения задач терминального управления в различных постановках, например, для нахождения программного управления при фиксированном или свободном времени, с настройкой характеристик исполнительного механизма или при известных и фиксированных характеристиках. Предложен подход, позволяющий решать задачу на пространстве меньшей размерности для задач с исполнительным механизмом в виде идеального реле. Также предложены различные способы представления решений, причем некоторые позволяют осуществить переход от задач условной оптимизации к задаче безусловной оптимизации, не прибегая при этом к применению метода штрафных функций, а за счет изменений представления решения и модификаций мутации, задания определенного правила разыгрывания стартовой популяции.

Алгоритм эволюционных стратегий был модифицирован и обобщен для решения экстремальных задач с вещественными и номинальными переменными. Номинальные переменные всегда могут быть выражены в виде чисел

натурального ряда. Предложены операции мутации и скрещивания для генов, которые соответствуют номинальным переменным, так что сохраняется парадигма алгоритма эволюционных стратегий. Обобщенный алгоритм применим для поиска решения задач терминального управления, позволяет одновременно решать задачу для переменных разной природы, сохранив преимущества гибридного алгоритма популяционного поиска для вещественных переменных.

Далее был рассмотрен непрямой метод решения задачи оптимального управления. Данный численно-аналитический метод основан на применении принципа максимума Понтрягина, на переходе к расширенной системе с сопряженными переменными и замыкании задачи, через условие стационарности по управлению. После замыкания задача сводится к поиску начальных условий сопряженных переменных, при которых решение задачи Коши для системы с сопряженными переменными является таким, что невязка между вектором состояний в конечный момент времени и правым концом задачи оптимального управления равна нулю. Для решения экстремальной задачи в численно-аналитическом непрямом методе поиска оптимального управления, так же применялся и был исследован гибридный модифицированный алгоритм эволюционных стратегий. Эффективность нахождения решения данным алгоритмом сравнивалась с эффективностью алгоритма дифференциальной эволюции, стайного алгоритма, алгоритма эволюционных стратегий, алгоритма эволюционных стратегий с адаптацией ковариационной матрицы. Поиск экстремума происходит на всем пространстве решений, которое не ограничено, а целевая функция, в общем случае, является сложной и многоэкстремальной. Предложенный алгоритм был сравнен с другими алгоритмами глобальной оптимизации через оценку эффективности нахождения решения. Было показано, что гибридный модифицированный алгоритм ЭС имеет большую частоту нахождения глобального оптимума. Предложен новый оператор перезапуска, применение которого приводит к значительному повышению вероятности нахождения глобального оптимума экстремальной задачи численно-аналитического метода решения задачи оптимального управления.

Разработанные алгоритмы в приложении к предлагаемым подходам были реализованы в виде программ для проведения численных исследований. Программные решения были реализованы как самостоятельные подключаемые модули пакета MATLAB, и как приложения под Windows, написанные на языке программирования C++. Некоторые программы были зарегистрированы.

Предложенный подход к моделированию динамических процессов в аналитическом виде, применялся для решения задачи подбора структуры и параметров дифференциального уравнения, описывающего процесс изменения концентрации продуктов реакции распада гексадекана. Подход к решению терминальной задачи управления является основой для дальнейших разрабатываемых алгоритмов решения задач управления космическими аппаратами при неалгебраических ограничениях, накладываемых на функцию управления.

Учитывая все вышеприведенное, заключим, что предложенные подходы позволяют находить решение поставленных задач, а разработанные алгоритмы поиска экстремума приведенных задач оптимизации являются более эффективными, чем стандартные, или впервые позволяют решать задачу в данной постановке. В этой связи можно утверждать, что поставленная цель диссертации полностью достигнута.

Таким образом, в данной диссертационной работе разработаны и совершенствованы алгоритмы стохастического популяционного поиска, основанные на новом гибридном модифицированном аналоге алгоритма эволюционных стратегий. Предлагаемые алгоритмы являются более эффективными в предложенных подходах решения задач идентификации и управления для динамических систем, чем известные аналоги, что, является существенным вкладом в теорию и практику системного анализа и управления.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Aida-zade K.R., Rahimov A.B. Relay Control of Nonlinear System with Uncertain Values of Parameters // Journal of Automation and Information Sciences. – Vol. 42 (7). – 2010. – P. 54-64.
2. Al-Duwaish H. N. A genetic approach to the identification of linear dynamical systems with static nonlinearities // International Journal of Systems Science. – Vol. 31 (3). – 2000. – P. 307-313.
3. Astreon K.J., Bohlin T. Numerical Identification of Linear Dynamic Systems from Normal Operating Records // Technical paper. IBM Nordic Laboratory, Lidingo, Sweden. – 1967. – 38P.
4. Beligiannis G. N., Tsirogiannis G. A., Pintelas P. E. Restartings: A Technique to Improve Classic Genetic Algorithms' Performance // J. Glob. Optim. 1 – 2004 - P. 112-115.
5. Bertolazzi E., Biral F., Lio M. Symbolic-Numeric Indirect Method for Solving Optimal Control Problems for Large Multibody Systems: The Time-Optimal Racing Vehicle Example // Multibody System Dynamics. – Vol. 13 (2) – 2005. – P. 233–252.
6. Brest J., Greiner S., Boskovic B., Mernik M., Zumer V. Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark functions // IEEE Transactions on Evolutionary Computation. – Vol. 10 (6). – 2006. – P. 646–657.
7. Bukhtoyarov V., Semenkina O. Comprehensive evolutionary approach for neural network ensemble automatic design // Proceedings of the IEEE World Congress on Computational Intelligence, Barcelona, Spain. – 2010. – P. 1640-1645.
8. Cao H., Kang L., Chen Y., Yu J. Evolutionary modeling of systems of ordinary differential equations with genetic programming // Genetic Programming and Evolvable Machines. – Vol. 1 (40). – 2000. – P. 309–337.
9. Chen W. H., Balance D. J., Gawthrop P. J. Optimal control of nonlinear systems: a predictive control approach // Automatica. – Vol. 39 (4). – 2003. – P. 633–641.

10. Cruz P., Torres D. Evolution strategies in optimization problems // Proc. Estonian Acad. Sci. Phys. Math. – Vol. 56 (4). – 2007. – P. 299–309.
11. Dao S. D., Abhary K., Marian R. M. Optimization of partner selection and collaborative transportation scheduling in Virtual Enterprises using GA // Expert Syst. Appl. 41(15) – 2014 - P. 6701-6717.
12. Desai S., Prasad R. A novel order diminution of LTI systems using Big Bang Big Crunch optimization and Routh approximation // Appl. Math. Model. Vol 37 - 2013 - P. 8016-8028.
13. Eiben A. E., Smith J. From evolutionary computation to the evolution of things // Nature, 521(7553) – 2015 - pp. 476–482.
14. Fukunaga A. S. Restart Scheduling for Genetic Algorithm // Lecture Notes of Computer Science, Vol. 1498 – 1998 - P. 357-369.
15. Goldberg D. E., Holland J.H. Genetic Algorithms in Search, Optimization, and Machine Learning // Machine Learning. – Vol. 3 (2). – 1988. – P. 95-99.
16. Goldberg, D.E. Genetic algorithms in machine learning Text. / D.E. Goldberg. Reading, MA: Addison-Wesley. – 1989.
17. Gomez J. Self Adaptation of Operator Rates Evolutionary Algorithms // Genetic and Evolutionary Computation GECCO 2004. - Lecture Notes in Computer Science. – Vol. 3102. – 2004. – P. 1162-1173.
18. Hansen N. Completely derandomized self-adaptation in evolution strategies // Evolutionary Computation. – Vol. 9 (2). – 2001. – P. 159–195.
19. Hansen N. Evaluating the CMA evolution strategy on multimodal test functions // In Xin Yao et al., editors, Parallel Problem Solving from Nature - PPSN VIII. – 2004. – P. 282–291.
20. Hansen N. The CMA evolution strategy: a comparing review. Towards a new evolutionary computation // Advances on estimation of distribution algorithms, Springer. – 2006. – P. 1769–1776.
21. Hashem M.M.A., Watanabe K., Izumi K. A new Evolution Strategy and Its Application to Solving Optimal Control Problems // JSME International Journal, Series C. – Vol. 41 (3). – 1998. – P. 406-412.

22. Jovanovic O. Identification of Dynamic System Using Neural Network // Architecture and Civil Engineering. – Vol. 1 (4). – 1997. – P. 525 – 532.
23. Kaveh A., Bakhshpoori T., Ashoory M. An efficient optimization procedure based on cuckoo search algorithm for practical design of steel structures // International Journal of Optimization in Civil Engineering. – Vol. 2. – 2012. – P. 1-14.
24. Kennedy J. Swarm Intelligence // Kennedy J., Eberhart R. / Morgan Kaufmann Publishers, Inc., San Francisco, CA. – 2001. – 512 P.
25. Kirk D. E. Optimal Control Theory: An Introduction. Englewood Cliffs, NJ: Prentice-Hall. – 1970.
26. Lopez Cruz I.L., Van Willigenburg L.G., Van Straten G. Efficient differential evolution algorithms for multimodal optimal control problems. Applied Soft Computing. – Vol. 3. – 2003. – P. 97–122.
27. Loshchilov I., Schoenauer M., Sebag M. Alternative Restart Strategies for CMA-ES // Lecture Notes in Computer Science, Vol. 7491 – 2012 - P. 296-305.
28. Medvedev A.V. Identification and control for linear dynamic system of unknown order // Optimization Techniques IFIP Technical Conference. Berlin – Heidelberg – New York: Springer-Verlag. – 1975. – P. 48–56.
29. Mineiro P., Movell J. R., Williams R. J. Modelling path distributions using partially observable diffusion networks: a Monte-Carlo approach // Technica lReport CogSci.UCSD-99.01, Department of Cognitive Science, UCSD, SanDiego. – 1999.
30. Modares H, Sistani M.-B. N. Solving nonlinear optimal control problems using a hybrid IPSO–SQP algorithm // Engineering Applications of Artificial Intelligence. – Vol. 24. – 2011. – P. 476–484.
31. Naiborhu J., Firman, Mu'tamar K. Particle Swarm Optimization in the Exact Linearization Technic for Output Tracking of Non-Minimum Phase Nonlinear Systems // Applied Mathematical Sciences, Vol. 7, no. 109. - 2013 - P. 5427-5442.
32. Narwal A., Prasad B. R. A Novel Order Reduction Approach for LTI Systems Using Cuckoo Search Optimization and Stability Equation // IETE Journal of Research, 62:2 - 2016 - P. 154-163/.

33. Parmar G., Prasad R., Mukherjee S. Order reduction of linear dynamic systems using stability equation method and GA // International Journal of computer and Information Engineering. – Vol. 1 (1). – 2007. – P. 26-32.
34. Passino K. M., Yurkovich S. Fuzzy control // Addison Wesley Longman, Inc. – 1998. – 502P.
35. Popescu M., Dumitrache M. On the optimal control of affine nonlinear systems. Mathematical Problems in Engineering. – Vol. 4. – 2005. – P.465–475.
36. Prasad R., Pal J., Pant A. K. Multivariable system approximation using polynomial derivatives // J. Inst. Eng. India Part E Electric. Eng. Div. Vol 76 - 1995 - P. 186-194.
37. Prasad R., Pal J. Use of continued fraction expansion for stable reduction of linear multivariable systems // J. Inst. Eng. India Part E Electric. Eng. Div. Vol 72 - 1991 - P 43-56.
38. Primbs J. A., Nevistic V., Doyle J. C. Nonlinear optimal control: a control Lyapunov function and receding horizon perspective // Asian Journal of Control. – Vol. 1(1). – 1999. – P. 14–24.
39. Qin A.K., Suganthan P.N. Self-adaptive differential evolution algorithm for numerical optimization // Proceedings of the IEEE congress on evolutionary computation (CEC). – 2005. – P. 1785–1791.
40. Rad H. S., Lucas C. A Recommender System based in Invasive Weed Optimization Algorithm // IEEE Congress on Evolutionary Computation (CEC 2007). – 2007. – P. 4297-4304.
41. Rao A. V. Survey of Numerical methods for Optimal Control // AAS/AIAA Astrodynamics Specialist Conference, AAS Paper 09-334, Pittsburg, PA. – 2009. – P.497-528.
42. Ramesh K., Nirmalkumar A., Gurusamy G. Order Reduction of LTIV continuous MIMO system using Stability Preserving approximation method // International Journal of Computer Applications, Vol 36 No 8. - 2011 – P. 1-8.

43. Reimer M., Rudzicz F. Identifying articulatory goals from kinematic data using principal differential analysis // *Proceedings of Interspeech 2010, Makuhari, Japan.* – 2010. – P. 1608—1611.
44. Ryzhikov I. About multiagent system applications for speech recognition problem // *Вестник Сибирского государственного аэрокосмического университета имени академика М.Ф. Решетнева.* – Вып. 4 (44). – 2012. – С. 82-85.
45. Ryzhikov I. Automatic linear differential equation identification in analytical form// *Вестник Сибирского государственного аэрокосмического университета имени академика М.Ф. Решетнева.* – Вып. 4 (44). – 2012. – С. 85-89.
46. Ryzhikov I., Semenkin E. Modified Evolutionary Strategies Algorithm in Linear Dynamic System Identification // *Informatics in Control, Automation and Robotics: Proceedings of the 10th International Conference ICINCO'2013.* – Vol. 1. – P. 618-621.
47. Ryzhikov I., Semenkin E. Modified Hybrid Evolutionary Strategies Method for Termination Control Problem with Relay Actuator // *Informatics in Control, Automation and Robotics: Proceedings of the 9th International Conference ICINCO'2012.* – Vol. 1. – P. 333-337.
48. Ryzhikov I., Semenkin E. The Application of Evolutionary Algorithm for the Linear Dynamic System Modelling // *Proceedings of the International Conference SIMULTECH' 2012.* – P. 234-237.
49. Ryzhikov I., Semenkin E., Okhorzin V. About optimization techniques in application to symbolic-numeric optimal control seeking approach // *Informatics in Control, Automation and Robotics: Proceedings of the 10th International Conference ICINCO'2013.* – Vol. 1. – P. 268-275.
50. Ryzhikov I., Semenkin E., Akhmedova S. Linear ODE coefficients and initial conditions estimation with co-operation of biology related algorithms // *Advances in Swarm Intelligence. Lecture Notes in Computer Science, Volume 7824.* – Springer-Verlag, Berlin, Heidelberg, 2016 - P. 228-235.
51. Ryzhikov I., Semenkin E., Panfilov I. Evolutionary optimization algorithms for differential equation parameters, initial value and order identification // *Proceedings of*

the 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2016) - Volume 1, 2016 - P. 168-176

52. Ryzhikov I., Semenkin E. Evolutionary Strategies Algorithm Based Approaches for the Linear Dynamic System Identification // Adaptive and Natural Computing Algorithms. Lecture Notes in Computer Science, Volume 7824. – Springer-Verlag, Berlin, Heidelberg, 2013. - P. 477-483.

53. Ryzhikov I. About performance improvement of evolutionary strategies technique applied to optimal control problem indirect method // Вестник Сибирского государственного аэрокосмического университета им. академика М.Ф. Решетнева, № 4 (50) – 2013 – P. 123-128

54. Ryzhikov I. Automatic linear differential equation identification in analytical form // Вестник Сибирского государственного аэрокосмического университета им. академика М.Ф. Решетнева, № 1 (53) – 2014 – P. 66-72

55. Saerens M. Viterbi algorithm for acoustic vectors generated by a linear stochastic differential equation // Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Detroit. - 1995. - P. 233-236.

56. Schwefel, H.-P. Evolution and Optimum Seeking. // New York: Wiley & Sons. – 1995. – 456P.

57. Semenkina M., Akhmedova S., Semenkin E., Ryzhikov I. Spacecraft Solar Arrays Degradation Forecasting with Evolutionary Designed ANN-based Predictors. Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2014) – Volume 1, 2014 - P. 421-428

58. Sersic K., Urbiha I. Parameter Identification Problem Solving Using Genetic Algorithm. // Proceedings of the 1. Conference on Applied Mathematics and Computation. – 1999 – P. 253–261.

59. Seywald H., Kumar R.R. Genetic Algorithm Approach for Optimal Control Problems with Linearly Appearing Controls // Journal of Guidance, Control and Dynamics. – Vol. 18 (1). – 1995. – P. 177-182.

60. Shushlyapin E. A. On the equivalency of piecewise-constant control with a known number of switchings and arbitrary amplitude bounded control in a terminal

problem for a linear nonstationary system // Journal of Soviet Mathematics. – Vol. 65 (2). – 1993. – P. 1550-1554.

61. Sim Y.C., Leng S.B., Subramaniam V., A combined genetic algorithms-shooting method approach to solving optimal control problems // International Journal of Systems Science. – Vol. 31 (1). – 2000. – P. 83-89.

62. Storn R., Price K. Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces // Journal of Global Optimization. – Vol. 11. – 1997. – P. 341–359.

63. Tan Y., Dong R., Chien H., He H. Neural networks based identification of hysteresis in human meridian systems // International Journal of Applied Mathematics and Computer Science. – Vol. 22 (3). – 2012. – P. 685-694.

64. Vlassenbroeck J. A Chebyshev polynomial method for optimal control with state constraints // Automatica. – Vol. 24 (4). – 1988. – P. 499–506.

65. Wolpert D. H., Macready W. G. No Free Lunch Theorems for Optimization // IEEE Transactions on Evolutionary Computation 1, - 1997 – P. 67-82.

66. Yamashita Y., Shima M. Numerical computational method using genetic algorithm for the optimal control problem with terminal constraints and free parameters // Nonlinear Analysis, Theory, Methods & Applications. – Vol. 30 (4). – 1997. – P. 2285-2290.

67. Yang X.-S. Firefly Algorithm, Stochastic Test Functions and Design Optimization // International Journal of Bio-Inspired Computation. – № 2. – 2010. – P. 78-84.

68. Zoteev V. Parametrical identification of linear dynamical system on the basis of stochastic difference equations // Matem. Mod. – Vol. 20 (9). – 2008. – P. 120-128.

69. Агошков В.И. Методы оптимального управления и сопряженных уравнений в задачах математической физики / М.: ИВМ РАН, 2003. – 256 С.

70. Александров В.В. Оптимальное управление движением // В.В. Александров, В.Г. Болтянский, С.С. Лемак, Н.А. Парусников, В.М. Тихомиров / М.: ФИЗМАТЛИТ, 2005. - 376с.

71. Алексеев В.М., Тихомиров В.М., Фомин С.В. Оптимальное управление // Алексеев В.М., Тихомиров В.М., Фомин С.В. / Учеб. пособие. – 2-е изд., перераб. и доп. М.: ФИЗМАТЛИТ, 2005. – 384с.
72. Астахова Т.Н., Зуев А.Л. Задача планирования движения для класса нелинейных систем с тригонометрическими функциями управления // Динамические системы. – Том 3 (31), No. 1-2. – 2013. – С.159–167.
73. Баландин Д.В. Синтез законов управления на основе линейных матричных неравенств // Д.В. Баландин, М.М. Коган / М.: ФИЗМАТЛИТ, 2007. - 280с.
74. Бахвалов Н.С. Численные методы // Н.С. Бахвалов, Н.П. Жидков, Г.М. Кобельков / М: Бином, 2007. – 636с.
75. Богуславский И. А. Полиномиальная аппроксимация для нелинейных задач оценивания и управления // И. А. Богуславский / М.: ФИЗМАТЛИТ, 2006. – 208с.
76. Бунич А.Л. Идентификация линейных стационарных объектов с большим отношением сигнал/шум // Труды VIII Международной конференции «Идентификация систем и задачи управления» SICPRO '09. М.:Институт проблем управления им. В.А. Трапезникова РАН. – 2009. – С. 127-129.
77. Вержбицкий В.М. Численные методы. Т.2 // В.М. Вержбицкий / Учеб. пособ. В 2-х Т. Том 2. М: Оникс, 2005. – 400с.
78. Водовозов А.М., Елюков А.С. К вопросу об идентификации линейных динамических систем по результатам экспериментальных исследований // Системы управления и информационные технологии. – Вып. 2.2(32). – 2008. – С. 253-256.
79. Водовозов А.М., Пискунов А.А. О сходимости метода статистической идентификации параметров динамических систем // Информационные технологии моделирования и управления. – Вып. 4 (22). – 2005 – С.530-534.
80. Водовозов А.М., Елюков А.С. Помехозащищенные алгоритмы параметрической идентификации электромеханических систем // Приборостроение. – Вып. №12. – 2009. - С. 40-43.

81. Зубер И. Б. Терминальное управление по выходу для нелинейных нестационарных систем // Дифференциальные уравнения и процессы управления. – № 2. – 2004. – С. 36–42.
82. Зубер И.Е. Синтез терминального управления по выходу нелинейной системы // Электронный журнал «Дифференциальные уравнения и процессы управления». – Вып. 1. – 2004. – С. 36-42.
83. Квитко А. Н., Якушева Д. Б. Алгоритм построения кусочно-постоянного синтезирующего управления при решении граничной задачи для нелинейной стационарной системы // Вестник ВГУ. Серия: Физика. Математика. 2012. №1 – С. 138-145.
84. Клепиков В.Б., Махотило К.В., Сергеев С.А., Обруч И.В. Применение методов нейронных сетей и генетических алгоритмов в решении задач управления электроприводами // Электротехника. – Вып. №5. – 1999. – С. 2-6.
85. Кучеров Д.П., Василенко А.В., Иванов Б.П. Алгоритм адаптивного терминального управления динамической системой с элементами дифференцирования // Автоматика. Автоматизация. Электротехнические комплексы и системы. – Вып. 23. – 2009. – С. 166-171.
86. Леоненков А.В. Нечеткое моделирование в среде MATLAB и fuzzyTECH // А.В. Леоненков / СПб.: «БХВ – Петербург», 2005. – 736с.
87. Оневский П.М., Оневский М.П., Ишин А.А. Реализация алгоритма терминального управления нелинейным динамическим объектом // Вестник ТГТУ. – Том 16 №4. – 2010. – С. 789-796.
88. Оневский, П.М. Итеративные прогнозирующие алгоритмы терминального управления нелинейными объектами // Вестник Тамбовского высшего военного авиационного училища радиотехники. – № 1. – 2008. – С. 23–29.
89. Охорзин В. А. Численно-аналитический метод решения задач оптимального управления // Электронный журнал «Исследовано в России», 12, 2009. – С. 253-260.
90. Охорзин В.А., Рыжиков И.С. Гибридный модифицированный метод эволюционных стратегий для решения задач идентификации динамических

систем // Вестник Сибирского государственного аэрокосмического университета имени академика М.Ф. Решетнева. – Вып. 4 (30). – 2010. – С. 20-23.

91. Охорзин В.А., Рыжиков И.С. Решение терминальной задачи управления в виде многоуровневого реле // Перспективные инновации в науке, образовании, производстве и транспорте '2012": Материалы Международной научно-практической конференции. – Вып. 2. – 2012. – Том 2. – С. 89-93.

92. Охорзин В.А., Рыжиков И.С. Решение экстремальной задачи в численно аналитическом методе решения задач оптимального управления // "Современные направления теоретических и прикладных исследований '2012": Материалы Международной научно-практической конференции. – Вып. 1. – 2012. – Том 10. – С. 36-42.

93. Охорзин В.А., Рыжиков И.С. Решение экстремальной задачи в численно аналитическом методе нахождения оптимального управления нелинейными системами // Электронный журнал «Исследовано в России», 072, 2011. - С. 959-965.

94. Пащенко Ф.Ф. Введение в состоятельные методы моделирования систем. Идентификация нелинейных систем // Ф.Ф. Пащенко / М: «Финансы и статистика», 2006. – 288с.

95. Пащенко Ф.Ф. Введение в состоятельные методы моделирования систем. Математические основы моделирования систем // Ф.Ф. Пащенко / М.: «Финансы и статистика», 2006. – 328 с.

96. Попов Е.А., Семенкина М.Е., Липинский Л.В. Принятие решений коллективом интеллектуальных информационных технологий // Вестник Сибирского государственного аэрокосмического университета им. ак. М.Ф. Решетнева. – Вып. 4 (44). – 2012. – С. 79-82.

97. Пугачев В.С. Теория вероятностей и математическая статистика // В.С. Пугачев / 2-е изд. Испр. и доп. / М.: Физматлит, 2002. – 496с.

98. Бухтояров С.В., Семенкин Е.С. Решение задачи Коши для обыкновенных дифференциальных уравнений методом генетического программирования // Вестник СФУ. Сер. Матем. и физ., 4:1. – 2011. – С. 61–69.

99. Родионова А. Г. О моментах переключения кусочно-постоянного управления в задаче быстродействия для линейной стационарной системы второго порядка с комплексными собственными значениями // Проблемы управления. – 2008. – № 6 – С. 81–83.
100. Романовский Б. Основы химической кинетики. // Б. Романовский / Москва: Экзамен, 1996. – 415с.
101. Рубан А.И. Методы анализа данных // А.И. Рубан / Учеб. Пособие. 2-е изд., исправл. И доп. Красноярск: ИПЦ КГТУ, 2004. – 319 с.
102. Рубан, А.И. Методы оптимизации: учеб. пособие / А.И. Рубан. / Красноярск: НИИ ИЛУ, 2001. – 528 с.
103. Рыжиков И.С. Об одном методе решения терминальной задачи управления в виде реле // Системный анализ и информационные технологии: Труды конференции САИТ-2013. В 2-х т. - Т. 1. – С. 276-284.
104. Рыжиков И.С. Программная система решения задачи идентификации линейных динамических систем по данным наблюдений с помощью параллельного алгоритма эволюционных стратегий. – М.: Роспатент, 2013, № гос. рег. 2013619507.
105. Рыжиков И.С. Расчет программного управления для динамических систем с многопозиционным релейным исполнительным механизмом для двухточечных терминальных задач со свободным временем. – М.: Роспатент, 2013, № гос. рег. 2013616552.
106. Рыжиков И.С. Расчет программного управления для динамических систем с многопозиционным релейным исполнительным механизмом для двухточечных терминальных задач с фиксированным временем. – М.: Роспатент, 2013, № гос. рег. 2013616777.
107. Рыжиков И.С. Решение терминальной задачи управления для нелинейных динамических систем// Вестник Сибирского государственного аэрокосмического университета имени академика М.Ф. Решетнева. – Вып. 5 (38). – 2011. – С. 85-88.

108. Рыжиков И.С. Структурно-параметрическая идентификация линейного динамического объекта по данным наблюдений. – М.: Роспатент, 2011, № гос. рег. 2012611800.
109. Рыжиков И.С., Семенкин Е.С. Об одном методе решения задачи идентификации для линейных динамических систем // Системы управления и информационные технологии. – №2.1(52). – 2013. – С. 173-177.
110. Рыжиков И.С., Семенкин Е.С. Система нахождения релейного программного управления для динамических объектов // Программные продукты и системы. – №1(101). – 2013. – С. 167-171.
111. Семёнкин, Е.С. Адаптивные поисковые методы оптимизации сложных систем // Е.С. Семёнкин, О.Э. Семёнкина, С.П. Коробейников / Красноярск: СИБУП, 1997. – 355 с.
112. Семенкин, Е.С. Методы оптимизации сложных систем // Е.С. Семенкин, О.Э. Семенкина, В.А. Терсков / Красноярск: - Сибирский юридический институт МВД России, 2000. – 254 с.
113. Семенкина М.Е. Самоконфигурируемые эволюционные алгоритмы моделирования и оптимизации // Теория и практика системного анализа. - Рыбинск: ИСА РАН, РГАТА имени П.А. Соловьева, 2012. – С. 24-30.
114. Семенкина М.Е., Семенкин Е.С., Рыжиков И.С., “Прогнозирование динамики электрических характеристик солнечных батарей космических аппаратов методами вычислительного интеллекта”, Вестник СибГАУ №3(55), 2014, с. 139 – 145.
115. Сопов, Е.А. Вероятностный генетический алгоритм с прогнозированием сходимости / Е.А. Сопов // Вестник университетского комплекса. Красноярск: ВСФ РГУИТП, НИИ СУВПТ, 2004. – Вып. 1 (15). – С. 219-227.
116. Сухарев А.Г. Курс методов оптимизации // А.Г. Сухарев, А.В. Тихомиров, В.В. Федоров / Учеб. пособие. – 2-е изд., М.: ФИЗМАТЛИТ, 2008. – 368 с.
117. Трушкова Е.А. Синтез управления в окрестности приближенного решения задачи с частично закрепленным правым концом // Программные системы: теория и приложения. – № 2(6). – 2011. – С. 31–35.

118. Фесько О.В. Алгоритм поиска кусочно-линейного управления с нефиксированными моментами переключений // Вестник Бурятского государственного университета – Вып. 9: Математика и информатика. – 2011. – С. 52-56.
119. Фесько О.В. Параллельный алгоритм оптимизации динамических систем на множестве кусочно-линейных управлений // Вестник Бурятского государственного университета – Вып. 9: Математика и информатика. – 2010. – С. 79-87.
120. Хайкин, С. Нейронные сети: полный курс, 2-е издание.: Пер. с англ. // С. Хайкин / М.: Издательский дом "Вильямс", 2006. - 1104с.