

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа инженерная школа информационных технологий и робототехники
Направление подготовки управление в технических системах
Отделение школы (НОЦ) отделение автоматизации и робототехники

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Параметрический синтез ПИ регуляторов для систем с аффинной неопределенностью УДК 681.515.4:681.51.015.2

Студент

Группа	ФИО	Подпись	Дата
8АМ61	Гурьев Андрей Андреевич		

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Руководитель ВКР	Суходоев М.С.	к.т.н.		
Руководитель ООП	Пушкарев М.И.	к.т.н.		

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН	Баннова К.А.	к.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Авдеева И.И.			

ДОПУСТИТЬ К ЗАЩИТЕ:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Руководитель ОАР	Леонов С.В.	к.т.н.		

Томск – 2018 г.

Планируемые результаты обучения по направлению
27.04.04 «Управление в технических системах»

	Результаты
P1	Применять глубокие естественнонаучные и математические знания для решения научных и инженерных задач в области анализа, синтеза, проектирования, производства и эксплуатации средств автоматизации и систем управления техническими объектами.
P2	Уметь обрабатывать, анализировать и обобщать научно-техническую информацию, передовой отечественный и зарубежный опыт в области теории, проектирования, производства и эксплуатации средств автоматизации и систем управления техническими объектами
P3	Ставить и решать инновационные задачи инженерного анализа, связанные с разработкой технических систем управления с использованием аналитических методов и сложных моделей.
P4	Выполнять инновационные инженерные проекты по разработке программно-аппаратных средств автоматизированных систем различного назначения с использованием современных методов проектирования, систем автоматизированного проектирования, передового опыта разработки конкурентно способных изделий.
P5	Планировать и проводить теоретические и экспериментальные исследования в области проектирования аппаратных и программных средств автоматизированных систем с использованием новейших достижений науки и техники, передового отечественного и зарубежного опыта. Критически оценивать полученные данные и делать выводы.
P6	Осуществлять авторское сопровождение процессов проектирования, внедрения и эксплуатации программно-аппаратных средств автоматизированных систем различного назначения.
P7	Владеть иностранным языком на уровне, позволяющем работать в интернациональной профессиональной среде с пониманием культурных, языковых и социально-экономических различий партнеров
P8	Осуществлять коммуникации в профессиональной среде и в обществе в целом, активно владеть иностранным языком, разрабатывать документацию, презентовать и защищать результаты инновационной инженерной деятельности, в том числе на иностранном языке.
P9	Эффективно работать индивидуально и в качестве члена и руководителя группы, в том числе междисциплинарной и международной, при решении инновационных инженерных задач.
P10	Демонстрировать личную ответственность и ответственность за работу возглавляемого коллектива, приверженность и готовность следовать профессиональной этике и нормам ведения инновационной инженерной деятельности. Демонстрировать глубокие знания правовых, социальных, экологических и культурных аспектов инновационной инженерной деятельности.
P11	Демонстрировать способность к самостоятельному обучению, непрерывному самосовершенствованию в инженерной деятельности, способность к педагогической деятельности.

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа инженерная школа информационных технологий и робототехники
Направление подготовки (специальность) управление в технических системах
Отделение школы (НОЦ) отделение автоматизации и робототехники

УТВЕРЖДАЮ:
Руководитель ООП

(Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

В форме:

Магистерской диссертации

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8АМ61	Гурьев Андрей Андреевич

Тема работы:

Определение робастных корневых показателей качества систем с аффинной неопределенностью	
Утверждена приказом директора (дата, номер)	№ 2181/с от 28.03.2018 г.

Срок сдачи студентом выполненной работы:	
--	--

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	<ol style="list-style-type: none">1. Основные понятия робастной теории управления.2. Методы анализа и синтеза САУ.3. Теория алгоритмов и архитектуры программных приложений..
Перечень подлежащих исследованию, проектированию и разработке вопросов	<ol style="list-style-type: none">1. Отображение параметрического многогранника интервального полинома на корневую плоскость.2. Анализ робастного качества интервальных САУ.3. Целенаправленный анализ ребер и синтез ПИ-регуляторов.4. Программная реализация алгоритмов анализа и синтеза интервальных систем.5. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение;6. Социальная ответственность.

Перечень графического материала <i>(с точным указанием обязательных чертежей)</i>	Презентация
Консультанты по разделам выпускной квалификационной работы <i>(с указанием разделов)</i>	
Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Доцент ОСГН, к.э.н. Баннова К.А.
Социальная ответственность	Ассистент Авдеева И.И.
Названия разделов, которые должны быть написаны на русском и иностранном языках:	
Отображение параметрического многогранника интервального полинома на корневую плоскость.	

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	
---	--

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Суходоев М.С.	к.т.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8АМ61	Гурьев Андрей Андреевич		

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа инженерная школа информационных технологий и робототехники
Направление подготовки (специальность) управление в технических системах
Уровень образования магистратура
Отделение школы (НОЦ) отделение автоматизации и робототехники
Период выполнения _____ (осенний / весенний семестр 2017/2018 учебного года)

Форма представления работы:

Магистерская диссертация

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	14.06.2018
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
	Основная часть	60
	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
	Социальная ответственность	10
	Обязательное приложение на иностранном языке	15

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Суходоев М.С.	к.т.н.		

СОГЛАСОВАНО:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Пушкарев М.И.	к.т.н.		

«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСООБЪЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ»

Студенту:

Группа	ФИО
8АМ61	Гурьеву Андрею Андреевичу

Школа	ИШИТР	Отделение	АиР
Уровень образования	магистратура	Направление/специальность	Управление в технических системах

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	<i>Работа с информацией, представленной в российских и иностранных научных публикациях, аналитических материалах, статистических бюллетенях и изданиях, нормативно-правовых документах; анкетирование; опрос, наблюдение.</i>
2. <i>Нормы и нормативы расходования ресурсов</i>	
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	<i>Проведение предпроектного анализ: планирование комплекса работ.</i>
2. <i>Планирование проведения и формирование бюджета научных исследований</i>	<i>Определение структуры и трудоёмкости работ в рамках НИИ, разработка графика проведения НИИ, планирование бюджета НИИ.</i>
3. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	<i>Расчёт интегрального показателя финансовой эффективности, интегрального финансового показателя, интегрального показателя ресурсоэффективности для всех видов исполнения НИИ.</i>

Перечень графического материала (с точным указанием обязательных чертежей):

1. <i>Карта сегментирования рынка услуг</i>
2. <i>Таблица SWOT-анализа</i>
3. <i>Комплекс работ по разработке проекта</i>
4. <i>Диаграмма Исикавы</i>
5. <i>Линейный график (диаграмма Ганта)</i>

Дата выдачи задания для раздела по линейному графику	
---	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН	Баннова К.А.	к.э.н.		01.03.201

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8АМ61	Гурьев Андрей Андреевич		01.03.201

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8АМ61	Гурьеву Андрею Андреевичу

Инженерная школа информационных технологии и робототехники		Отделение автоматизации и робототехники	
Уровень образования	магистратура	Направление/специальность	27.04.04 Управление в технических системах

Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	<i>В выпускной квалификационной работе разрабатывается лабораторный комплекс на базе промышленного микропроцессорного контроллера «КРОСС». Рабочее место: лаборатория ТПУ для проведения работ по автоматизации. Содержит ЭВМ, стенды, офисную мебель.</i>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p>1. Производственная безопасность</p> <p>1.1. Анализ выявленных вредных факторов при разработке и эксплуатации проектируемого решения в следующей последовательности:</p> <ul style="list-style-type: none"> – физико-химическая природа вредности, её связь с разрабатываемой темой; – действие фактора на организм человека; – приведение допустимых норм с необходимой размерностью (со ссылкой на соответствующий нормативно-технический документ); – предлагаемые средства защиты; – (сначала коллективной защиты, затем – индивидуальные защитные средства). <p>1.2. Анализ выявленных опасных факторов при разработке и эксплуатации проектируемого решения в следующей последовательности:</p> <ul style="list-style-type: none"> – механические опасности (источники, средства защиты); – термические опасности (источники, средства защиты); – электробезопасность (в т.ч. статическое электричество, молниезащита – источники, средства защиты); – пожаровзрывобезопасность (причины, профилактические мероприятия, первичные средства пожаротушения). 	<p>Анализ выявленных вредных факторов:</p> <ul style="list-style-type: none"> - недостаточная освещенность рабочей зоны; -повышенный уровень электромагнитных излучений; - повышенная напряжённость электрического поля; - повышенная или пониженная влажность воздуха; - повышенный уровень шума; - психо–физиологические факторы <p>Анализ выявленных опасных факторов:</p> <ul style="list-style-type: none"> - электрический ток (источником является ПК); - короткое замыкание; - статическое электричество. <p>Законодательные и нормативные документы: СанПиН 2.2.2/2.4.1340 – 03; СанПиН 2.2.4.548 – 96; СП</p>

	52.13330.2011; СН 2.2.4/2.1.8.562 – 96; ГОСТ 12.1.003-83; ГОСТ 12.1.019- 85.
2. Экологическая безопасность: <ul style="list-style-type: none"> – защита селитебной зоны – анализ воздействия объекта на атмосферу (выбросы); – анализ воздействия объекта на гидросферу (сбросы); – анализ воздействия объекта на литосферу (отходы); – разработать решения по обеспечению экологической безопасности со ссылками на НТД по охране окружающей среды. 	При выполнении работы влияние на атмосферу и гидросферу не происходит. Воздействие на литосферу – образование отходов при печати документов; – образование отходов при замене устаревшего или неисправного оборудования, люминесцентные лампы.
3. Безопасность в чрезвычайных ситуациях: <ul style="list-style-type: none"> – перечень возможных ЧС при разработке и эксплуатации проектируемого решения; – выбор наиболее типичной ЧС; – разработка превентивных мер по предупреждению ЧС; – разработка действий в результате возникшей ЧС и мер по ликвидации её последствий. 	В аудиторном помещении возможно ЧС техногенного характера – пожар (возгорание).
4. Правовые и организационные вопросы обеспечения безопасности: <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	Требования к организации оборудования рабочих мест с ПК регулируется СанПиН 2.2.2/2.4.1340-03, ФЗ–197, регулирующий трудовые отношения (ТК РФ).

Дата выдачи задания для раздела по линейному графику	01.03.2018
---	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Авдеева Ирина Ивановна			01.03.2018

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8АМ61	Гурьев Андрей Андреевич		01.03.2018

Реферат

Диссертация состоит из введения, шести глав, заключения и списка литературы, содержит 91 печатную страницу основного текста, 18 рисунков и 10 таблиц.

Ключевые слова: аффинная неопределенность, анализ систем, синтез ПИ–регуляторов, C#, Visual Studio.

Цель работы — исследование параметрического синтеза для систем с аффинной неопределенностью.

Методы проведения работы: методы сбора и анализа информации, методы программирования.

Результат выпускной квалификационной работы — проведен анализ и разработан новый алгоритм анализа и синтеза систем с аффинной неопределенностью. Создано программное приложение, реализующее описанный алгоритм.

Оглавление

Введение.....	13
ГЛАВА 1. Отображение параметрического многогранника интервального полинома на корневую плоскость.....	24
<i>1.1. Основные понятия и обозначения при отображении параметрического многогранника.....</i>	<i>24</i>
ГЛАВА 2. Анализ робастного качества интервальных систем автоматического управления.....	31
<i>2.1. Определение граничных вершин при аффинной неопределенности 31</i>	
<i>2.2. Реберный анализ робастного качества системы при аффинной неопределенности.....</i>	<i>33</i>
<i>2.3. Основные результаты</i>	<i>39</i>
ГЛАВА 3. Целенаправленный анализ ребер и синтез ПИ–регуляторов .	40
<i>3.1. Синтез ПИ–регулятора</i>	<i>40</i>
<i>3.2. Целенаправленный анализ</i>	<i>43</i>
<i>3.3. Основные результаты.....</i>	<i>45</i>
ГЛАВА 4. Программная реализация алгоритмов анализа и синтеза интервальных систем	46
<i>4.1. Анализ программных сред</i>	<i>46</i>
<i>4.2. Архитектура приложения.....</i>	<i>48</i>
<i>4.3. Основные алгоритмы приложения.....</i>	<i>49</i>
ГЛАВА 5. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	51
<i>5.1. Предпроектный анализ.....</i>	<i>51</i>

5.1.1. Потенциальные потребители результатов исследования	51
5.1.2. SWOT–анализ.....	53
5.1.3. Диаграмма Исикавы	54
6.1. Инициализация проекта.....	57
6.1.1. Структура работ в рамках научного исследования.....	57
6.2. Планирование управления научно-техническим проектом.....	59
6.2.1. Определение трудоемкости выполнения работ	59
6.2.2. Составление сметы затрат на разработку проекта.....	63
6.2.3. Расчет сметы затрат на выполнение проекта и на материалы	63
6.2.4. Расчет заработной платы и социального налога.....	64
6.2.5. Расчет затрат на электроэнергию.....	65
6.2.6. Расчет накладных расходов.....	68
6.2.7. Расчет общей себестоимости разработки	68
6.2.8. Расчет прибыли и налога на добавленную стоимость. Цена разработки НИР.....	69
6.3. Оценка экономической эффективности проекта.....	70
6.3.1. Оценка научно-технического уровня НИР.....	70
6.4. Итоговое заключение по разделу.....	73
ГЛАВА 6. Социальная ответственность	74
6.1. Введение	74
6.2. Производственная безопасность.....	75
6.1.1. Параметры микроклимата в помещениях.....	75
6.1.2. Недостаток освещения рабочей зоны	76
6.1.3. Повышенное электромагнитное излучение.....	78
6.1.4. Ионизирующее излучение от дисплея на ЭЛТ компьютера	79
6.1.5. Повышенный уровень шума.....	80
6.1.6. Опасность поражения электрическим током	81
6.1.7. Психофизиологические факторы опасности	82

6.3. Экологическая безопасность.....	84
6.4. Безопасность в чрезвычайных ситуациях.....	86
6.4.1. Пожарная профилактика.....	86
6.4.2. Оценка пожарной безопасности помещения.....	86
6.4.3. Мероприятия по устранению и предупреждению пожаров.....	87
6.5. Заключение.....	89
Заключение.....	91
ЛИТЕРАТУРА.....	92
Приложение А.....	96
Приложение В.....	111

Введение

Интервально–неопределенные параметры имеют крайне широкое распространение среди систем автоматического управления. Данная неопределенность связана с отсутствием достаточных знаний о параметрах системы или законов их изменения в процессе эксплуатации системы. При этом выделяют параметрическую интервальную неопределенность, при которой «известны диапазоны возможных значений постоянных параметров или пределы изменяющихся параметров» [1, 8, 27]. Системы, описываемые такими параметрами, называют интервальными системами автоматического управления [15].

Первостепенной задачей исследования подобных систем являлась проверка на робастную устойчивость, иначе говоря: отвечает ли система критериям устойчивости при любых значениях интервально–неопределенных параметрах. Впрочем, при проектировании систем это является недостаточным условием, так как важным является проанализировать региональную робастную устойчивость, соответствующую определенному робастному качеству системы [11, 15, 20, 22, 23].

Прежде чем обозначить исследование, необходимо выделить определенную классификацию вхождения интервальных параметров в коэффициенты интервального характеристического полинома (ИХП). Это очерчивает тип неопределенности, что влияет на шаги исследования.

Пусть интервальный полином имеет вид $D(s, q) = a_n(q)s^n + a_{n-1}(q)s^{n-1} + \dots + a_1(q)s + a_0(q)$, $q \in P_m$, где параметры q изменяются в допустимом множестве P_m .

Для такого полинома выделяют четыре вида неопределенностей: интервальная, аффинная, полилинейная и полиномиальная [1, 8, 88].

«Интервальная неопределенность — коэффициенты полинома являются интервальными параметрами» [1, 8, 27]:

$$q_2 s^2 + q_1 s + q_0, \quad q_i \in [q_{i \min}, q_{i \max}]$$

«Аффинная неопределенность — коэффициенты полинома образованы суммой или разностью интервальных параметров» [1, 8, 27]:

$$(q_3 + q_2) s^3 + (q_2 + q_1) s^2 + (q_3 - 2q_2 + 5q_1) s + 10q_2 - 7q_1, \quad q_i \in [q_{i \min}, q_{i \max}]$$

«Полилинейная неопределенность — коэффициенты полинома линейно зависят от каждого параметра, если остальные параметры фиксированы» [1, 8, 27]:

$$(q_1 q_2 + q_3) s^3 + (2q_2 q_3 + q_1) s^2 + (9q_1 q_3 - 3q_2) s + q_1 q_3 - 5q_2, \quad q_i \in [q_{i \min}, q_{i \max}]$$

«Полиномиальная неопределенность — коэффициенты полинома зависят полиномиально минимум от одного параметра» [1, 8, 27]:

$$s^2 + (2q_1 + 3q_2^2) s + 10q_1, \quad q_i \in [q_{i \min}, q_{i \max}]$$

Робастно устойчивым интервальный полином $D(s, q)$ является при условии устойчивости при всех $q \in P_m$. Множество P_m может содержать бесконечное количество элементов, что приводит к затруднению непосредственного использования известных критериев устойчивости (необходим перебор бесконечного множества значений). В связи с данным положением были разработаны различными авторами ряд критериев. Одним из таких примеров является теорема В. Л. Харитонова, гласящая: «для робастной устойчивости полинома $D(s, q)$ с интервальной неопределенностью необходимо и достаточно, чтобы четыре специальным образом сформированных полинома Харитонова были устойчивы» [26]. Обозначенные коэффициенты рассматриваемых полиномов имеют предельные значения по заданным интервалам. Описанная теорема имеет

графическое представление, называемое годографом Цыпкина-Поляка [21]. Используя данное представление, достаточно проверить поведение одного годографа.

Однако описанных выше положений недостаточно для анализа более сложных систем, например, обладающих аффинной неопределенностью в ИХП. В таком случае, как правило, применяют реберную теорему. Данная теорема оперирует понятием реберного полинома — ребра параметрического многогранника P_m , соединяющего две соседние вершины, образующие вершинные полиномы. Таким образом, для устойчивости необходима и достаточна устойчивость всех реберных полиномов. Следует заметить, что эффективность падает с увеличением числа интервальных параметров.

Как было упомянуто ранее, важным является определение региональной робастной устойчивости, определение наихудших показателей качества системы при изменении интервальных параметров в заданных диапазонах. До текущего момента велись определенные исследования [6, 7, 21], отличающиеся сравнительной неконсервативностью достаточных условий. В большинстве своем это алгебраические и частотные методы. Так, например, в [24] разработаны «условия попадания корней ИХП в заданный сектор комплексной плоскости, основанные на достаточном алгебраическом критерии устойчивости Липатова-Соколова». Эти условия имеют вид

$$\frac{a_i^2}{a_{i-1} \cdot a_{i+1}} \geq \delta^*, \text{ где } \overline{a}_i \text{ и } \underline{a}_i - \text{границы интервала коэффициента } a_i \text{ ИХП.}$$

Данная проблема рассматривается также в [24], анализируя принадлежность корней ИХП второму и третьему квадранту (левой полуплоскости). Для этого в частотной области «формируется 4 специальных вершинных полинома степени, где n — порядок полинома, и проверяется их устойчивость. При этом количество вершинных полиномов, подлежащих проверке, не зависит от степени ИХП» [24]. Данный подход является

достаточным, а, следовательно, обладает определенной степенью консерватизма.

Консерватизм является не единственным недостатком. Также к ним относится и трудность поиска предельных отклонений параметров систем, обеспечивающих заданные характеристики качества. Указанные методы не дают ответов на следующие вопросы: каковы пределы устойчивости, как изменение параметров повлияет на качества системы, которые и являются главной целью при построении системы. Поэтому актуальной является дальнейшая разработка методов исследования интервальных систем, характеризующихся большей точностью и, при этом, простотой применения.

Описанные выше методы обладают рядом недостатков, избежать которые можно, используя методы, основанные на корневом подходе и миграции корней характеристических полиномов интервальных систем [25]. Корневой подход предполагает наличие связи робастной устойчивости и расположением корней ИХП, что, в свою очередь, связаны с различными сочетаниями интервальных параметров. Таким образом, основной задачей при проектировании интервальных систем является расположение корней в желаемой области, обеспечивающее необходимые параметры качества системы.

Обозначим характеристический полином с интервальной неопределенностью:

$$P(s) = \sum_{i=0}^n a_i s^i, \quad a_{i \min} \leq a_i \leq a_{i \max},$$

где n – максимальная степень интервального характеристического полинома, a_i – интервальные коэффициенты.

Известно, «при интервальной и аффинной неопределенностях характеристических полиномов области отображения параметрического многогранника коэффициентов полинома ограничены образами его вершин

для интервальной и ребер для аффинной неопределенности, по которым можно определить корневые показатели качества интервальных систем» **[Error! Reference source not found.]**. В том числе интерес для исследования представляют совокупность ребер, образующих реберный маршрут, который определяет границы области локализации.

Если система описывается интервальной неопределенностью, то достаточно провести анализ вершин параметрического многогранника [18]. Как, допустим, указывается: «для проверки факта принадлежности корней ИХП заданной области предлагается проверить попадание в нее корней вершинных полиномов, соответствующих всем вершинам многогранника ИХП» [18]. Несмотря на значительное упрощение алгоритма исследования, задача уменьшения проверяемых вершин вызывает значительный интерес.

Упомянутые ранее полилинейная и полиномиальная неопределенности куда больше соответствуют реальным системам управления, но нужно обозначить, что границы области локализации корней могут определяться также и внутренними точками параметрического многогранника, что приводит к необходимости его полного отображения на корневой плоскости, а это сопряжено с трудностями при практической реализации, особенно, если система описывается большим числом интервальных коэффициентов. В дальнейшем в данной работе будут рассматриваться системы с интервальной и, в большей степени, аффинной неопределенностями. Чтобы обеспечить валидность представленных результатов исследования для более сложных систем, можно их привести к упрощению согласно правилам интервальной арифметики, описанной, например, в [18]. Стоит отметить также: «параметрический многогранник в случае интервальной неопределенности полинома образуется его интервальными коэффициентами, а в случае аффинной неопределенности – интервальными параметрами системы, линейно входящими в коэффициенты характеристического полинома» [18].

Корневые оценки качества стационарных систем обычно характеризуются степенью устойчивости η и колебательностью μ , робастные — максимально допустимой колебательностью и минимально допустимой степенью устойчивости. Они определяются путем расположения на комплексной плоскости корней ИХП замкнутой системы. Колебательность определяется путем ограничения области Γ , расположенной между двумя лучами, исходящими из центра осей декартовой плоскости и охватывающей все корни. Угол между вещественной осью и лучом определяет этот параметр: $\varphi = \arctg \mu$ (рисунок 1).

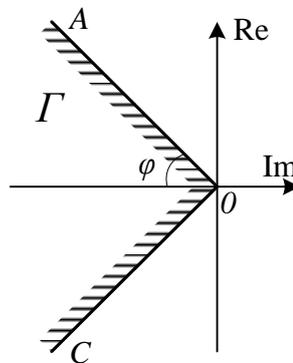


Рисунок 1 – область локализации корней с заданной максимальной колебательностью

Задав вертикальную прямую, параллельную оси ордината (мнимой), отделяется область Γ с корнями, а также фиксируется минимально допустимая степень устойчивости — расстояние области от оси η (рисунок 2).

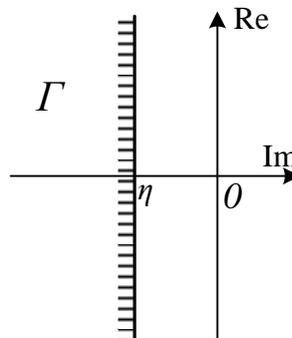


Рисунок 2 – область локализации корней с заданной минимальной степенью устойчивости

Таким образом, располагая корни в секторе с углом 2φ (рисунок 3) на отдалении от оси η , можно добиться обеспечения максимальной допустимой колебательности φ и минимальной допустимой степени устойчивости η соответственно.

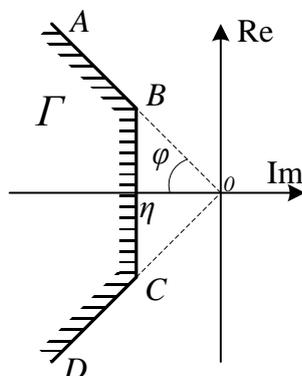


Рисунок 3 – область желаемой локализации корней ИХП

Полученную границу можно аппроксимировать гиперболой. Тогда сектор формируется двумя асимптотами $y = \pm \frac{b}{a}x$, угловой коэффициент асимптот $\mu = \pm \operatorname{tg} \varphi = \pm \frac{b}{a}$, где a и b – элементы гиперболы, описываемые уравнением $\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1$. При $a = \eta$, граница области выглядит следующим образом (рисунок 4):

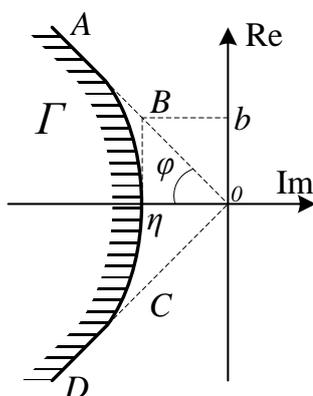


Рисунок 4 – аппроксимированная область желаемой локализации корней

Также возможна постановка задачи, требующая обеспечить отсутствия множественных колебаний переходного процесса, иначе говоря, обеспечить апериодичность. Такая задача обозначается, а также анализируется в [14, 15], основываясь на частотных критериях робастной устойчивости. «Апериодичным называется процесс, степень затухания которого находится в пределах от 0,55 до 1 (совершается менее одного колебания)» [10]. Отношение амплитуд двух перерегулирований (последовательных колебаний одного знака) определяет степень затухания переходного процесса. Числитель — это амплитуда первого колебания. Если степень затухания меньше 0,55, данный процесс называется колебательным. Если больше 1 — монотонным.

«Апериодический характер переходного процесса можно обеспечить доминантным расположением ближайшего к мнимой оси вещественного корня» [10]. Чтобы достичь описанного условия необходимо определить специальным образом расположение границ областей локализации корней интервального характеристического полинома, как на рисунке 5, где $a_3 < a_2 < a_1$. К тому же на интервале $[a_1, a_2]$ мигрирует один вещественный корень, а остальные корни располагаются в усеченном секторе ABCD, определенный углом 2φ и вертикальной прямой, проходящей через a_3 .

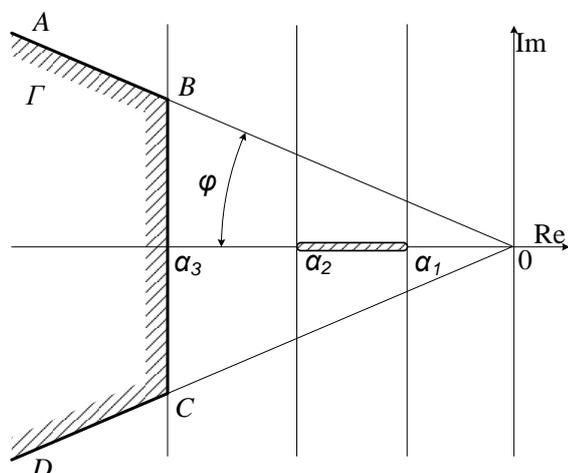


Рисунок 5 – области локализации корней для обеспечения аperiodического переходного процесса

«Наряду с расположением корней ИХП (полюсов замкнутой интервальной системы) необходимо учитывать и расположение ее нулей, так как именно их взаимное расположение влияет на прямые характеристики переходного процесса системы: перерегулирование и время регулирования» [10].

Из задач проведением анализа интервальных систем вытекает не менее актуальная задача синтеза регуляторов [6]. Многие работы, цель которых анализ указанной проблемы, опираются на метод D-разбиения и его робастное расширение. Так, указывается: «для обеспечения робастной устойчивости интервальной системы разработана методика определения ее настраиваемых параметров, основанная на применении прямоугольников Харитонова и метода D-разбиения» [7]. В результате синтеза в плоскости настраиваемых параметров ИС строится граница D-разбиения, состоящая из прямоугольников Харитонова, и из полученной области робастной устойчивости выбирают значения синтезируемых параметров.

Главное требование к проектировщику интервальной системы — обеспечить допустимые робастные показатели качества, недостаточно обеспечить только робастную устойчивость. Корневой подход сводит данную задачу только к определенному расположению корней в области локализации.

Таким образом, для решения поставленных выше задач представляет интерес разработка на основе корневого подхода методов анализа робастного качества интервальных систем, а также методов параметрического синтеза линейных регуляторов, гарантирующих заданное качество управления. Указанные методики предлагается разрабатывать на основе свойств отображения области интервальных параметров (параметрического многогранника) на корневую плоскость. Для этого необходимо провести

анализ отображения ребер и вершин параметрического многогранника и установить связь их образов с корневыми показателями качества (степень устойчивости и колебательность).

Для решения задачи обеспечения требуемого качества интервальных систем предлагается использовать линейный ПИ-закон управления, широко применяемый в промышленных контроллерах реальных систем автоматического управления.

Проектировщик, выполняя разработку систем автоматического управления с интервальными параметрами, нуждается в удобном инструменте решения задач анализа и синтеза, позволяющий автоматизировать трудоемкий процесс исследования систем. Чтобы создать такой инструмент, требуется алгоритмизировать исследуемые методики и воплотить их в программную реализацию на ЭВМ. При этом было принято решение использовать для этого высокоуровневый язык программирования C#, а среду разработки — Visual Studio. Этот язык пользуется большой популярностью в сфере информационных технологий, а спектр его применения крайне широк. Сторонние библиотеки значительно облегчают написание программного кода, позволяя оперировать программными компонентами, не вдаваясь в детали реализации отдельных шагов общего алгоритма. Также имеющиеся средства среды разработки позволяют создать удобный интерфейс для пользователей, а слабая связанность компонентов дает возможность переквалифицировать приложение в любой тип, будь то десктопное, web-приложение или столь популярные на момент написания этой работы мобильные технологии.

Научную новизну работы определяют:

- методики анализа робастного качества ИС с аффинной неопределенностями на основе выбора вершинных характеристических полиномов;

- методики интервально-параметрического синтеза ПИ- регуляторов, обеспечивающих гарантированные корневые показатели качества ИС при любых значениях ее интервальных параметров;

Практическую ценность работы составляют:

- разработанные в среде Visual Studio на основе полученных алгоритмов прикладные программы анализа робастного качества ИС;
- разработанные в среде Visual Studio на основе полученных алгоритмов прикладные программы параметрического синтеза линейного ПИ- регулятора для ИС, гарантирующий робастное качество управления;
- разработанный пакет прикладных программ, рассчитанный на инженерный уровень использования, что делает его доступным средством для решения практических задач управления в ИС, а также обучения студентов соответствующих специальностей.

ГЛАВА 1. Отображение параметрического многогранника интервального полинома на корневую плоскость

1.1. Основные понятия и обозначения при отображении параметрического многогранника

Фундаментальные результаты, позволяющие исследовать робастную устойчивость системы с интервальной неопределенностью, получены в работах В.Л. Харитонова [27].

Пусть характеристический полином системы с интервальной неопределенностью имеет вид:

$$P(s) = \sum_{i=0}^n a_i s^i, \quad \underline{a}_i \leq a_i \leq \overline{a}_i, \quad (0.1)$$

где a_i — интервальные коэффициенты, \overline{a}_i — максимальное значение a_i , а \underline{a}_i — его минимальное значение.

Согласно теореме, можно выделять полиномы, получившие название полиномы Харитонова. Это четыре полинома, составленные из крайних значений коэффициентов, чередующихся парами (два нижних значения – два верхних)

$$\begin{aligned} P_1(s) &= \underline{a}_0 + \underline{a}_1 s + \overline{a}_2 s^2 + \overline{a}_3 s^3 + \dots; \\ P_2(s) &= \overline{a}_0 + \underline{a}_1 s + \underline{a}_2 s^2 + \overline{a}_3 s^3 + \dots; \\ P_3(s) &= \overline{a}_0 + \overline{a}_1 s + \underline{a}_2 s^2 + \underline{a}_3 s^3 + \dots; \\ P_4(s) &= \underline{a}_0 + \overline{a}_1 s + \overline{a}_2 s^2 + \underline{a}_3 s^3 + \dots \end{aligned} \quad (0.2)$$

Сама теореме звучит следующим образом: «для робастной устойчивости интервального семейства необходимо и достаточно, чтобы все полиномы Харитонова были устойчивы» [27].

Пусть полином имеет m интервальных коэффициентов. Они образуют параметрический многогранник, представляющий собой прямоугольный гиперпараллелепипед, содержащий 2^m вершин и $m \cdot 2^{m-1}$ ребер. Пример параметрического многогранника для полинома системы с тремя интервальными коэффициентами изображен на рисунке 1.6.

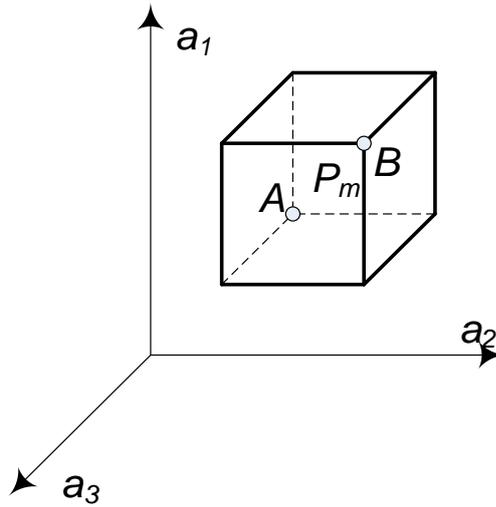


Рисунок 1.6 – Параметрический многогранник P_m

Координаты любой точки P_m относительно вершины V_q , $q = \overline{1, 2^m}$ определяются выражениями:

$$a_i = a_i^q + \Delta a_i, i = \overline{1, n}, \quad (0.3)$$

$$(\underline{a}_i - a_i^q) \leq \Delta a_i \leq (\overline{a}_i - a_i^q), \quad (0.4)$$

где Δa_i – приращение i -го интервального параметра, a_i^q – его значение в вершине V_q . Пусть $M(s)$ и $N(s)$ – два соседних полинома, определяемых двумя соседними вершинами, связанными одним ребром. Введем понятие реберного полинома. Он представляет собой однопараметрическое семейство вида $\gamma M(s) + (1 - \gamma)N(s)$, где $0 \leq \gamma \leq 1$.

Согласно теореме, называемой реберной, «робастная устойчивость семейства полиномов зависит от устойчивости всех ее реберных полиномов» [Error! Reference source not found.]. Эффективность данного подхода падает с увеличением числа коэффициентов (для $m=5$, где m — число неопределенных коэффициентов, нужно проверить $m \cdot 2^{m-1} = 80$ реберных полиномов). Это сопряжено с большим объемом вычислений, а значит, необходимостью найти более эффективное решение.

Из рассмотрения областей отображения параметрического многогранника (рисунок 1.6), следует очевидный вывод, что для анализа

робастной устойчивости достаточно проверить только те его существенные ребра, образы которых составляют границы областей локализации корней полинома. Однако для этого необходимо уметь заранее определять существенные ребра по имеющейся информации о структуре полинома и интервалах неопределенности его коэффициентов.

Обозначим отображения вершин V_q параметрического многогранника как U_q . На рисунке 1.7 представлено отображение вершин P_m на комплексную плоскость корней, при котором комплексно-сопряженные корни локализуются в двух областях, ограниченных образами ребер P_m .

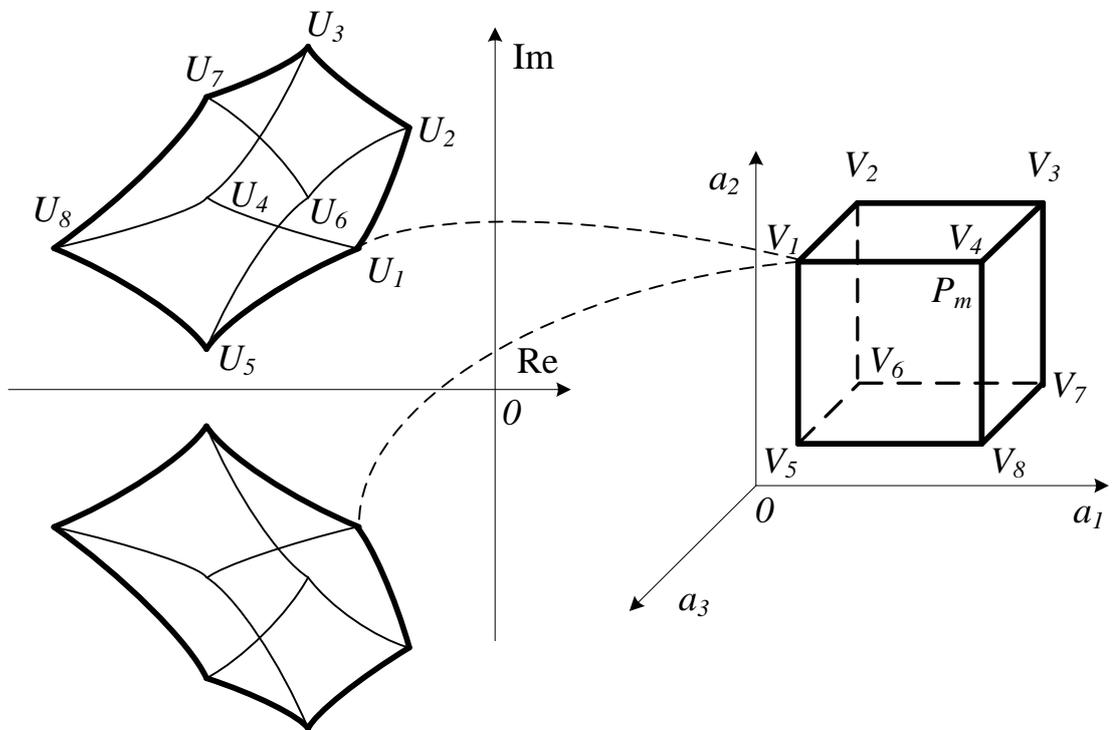


Рисунок 1.7 – Отображение вершин параметрического многогранника

Ребра интервального полинома обозначим как R_i^q , $i = \overline{1, m}$, где m – количество интервальных коэффициентов, q – порядковый номер вершины. Образы ребер обозначим через RS_i^q и назовем их реберными ветвями. Ребра P_m и их образы представлены на рисунке 1.8.

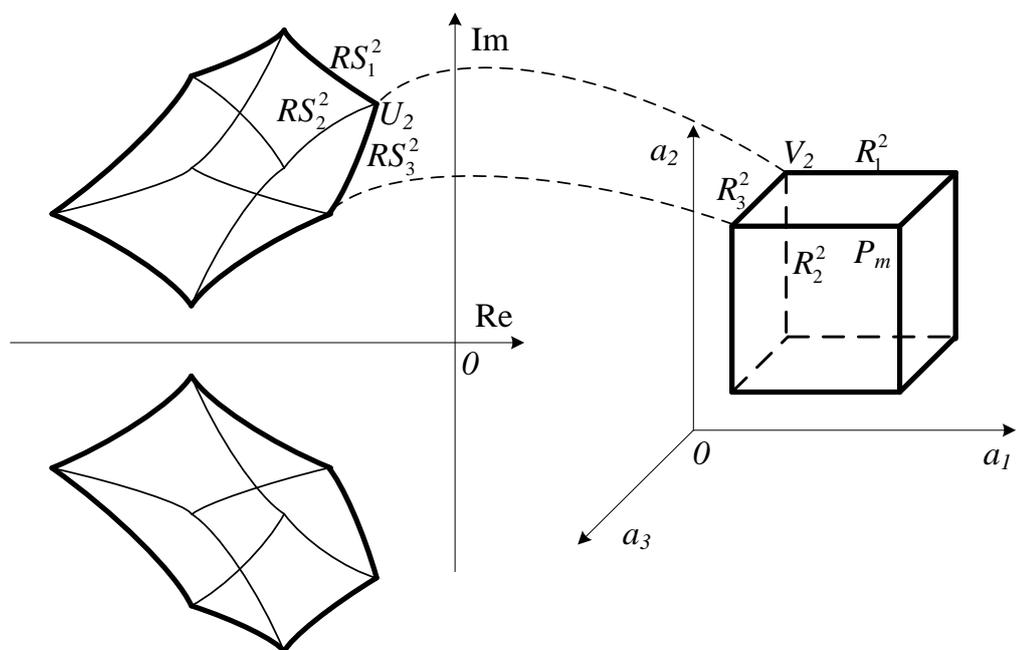


Рисунок 1.8 – Отображение ребер параметрического многогранника

Обозначим угол выхода из корневого узла U_q по j параметру как φ_j (рисунок 1.9). Векторы, определяющие углы выхода реберных ветвей, обозначим через E_j .

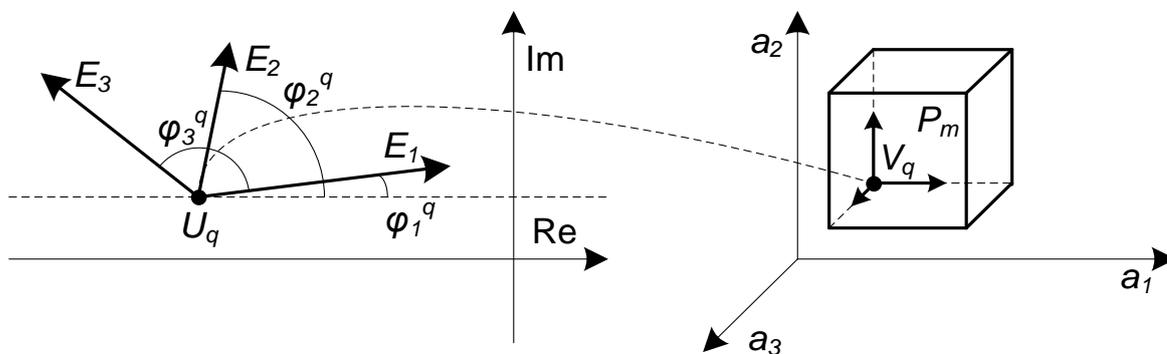


Рисунок 1.9 – Углы выхода реберных ветвей

1.2. Свойства отображения параметрического многогранника при аффинной неопределенности

Для отображения $\phi: P_m \rightarrow S$, где S – множество корней (0.1), необходимо иметь соотношение, связывающее координаты точек P_m с n корнями полинома (0.1). Для систем с аффинной неопределенностью, где параметры системы, линейно входящие в коэффициенты полинома, отношение имеет следующий вид:

$$D(s) = \sum_{i=0}^m (T_i \cdot A_i(s)) + B(s) = 0, \quad \underline{T}_i \leq T_i \leq \overline{T}_i, \quad (1.5)$$

где $A_i(s) = \sum_{j=0}^k (a_{i,j} s^j)$, $B(s) = \sum_{j=0}^k (b_j s^j)$.

Для отображения $\phi: P_m \rightarrow S$ используем следующее выражение:

$$D_q(s) + \sum_{i=0}^k (\Delta T_i \cdot A_i(s)) = 0, \quad (1.6)$$

где $D_q(s)$ – полином со значениями интервальных параметров, соответствующих вершине q параметрического многогранника.

Для рассмотрения ребер P_m будут использоваться следующие обозначения: ребро R_i^q , где i – индекс ΔT_i , $i = \overline{1, m}$, q – индекс V_q , из которой по ребру меняется T_i . Согласно (1.5) можно записать уравнение отображения ребра R_i^q на комплексной плоскости:

$$D_q(s) + \Delta T_i \cdot A_i(s) = 0 \quad (1.7)$$

Если система обладает единичной обратной связью, тогда можно записать передаточную функцию эквивалентной разомкнутой системы следующим образом:

$$W_i^q(\Delta T_i, s) = \frac{\Delta T_i \cdot A_i(s)}{D_q(s)} \quad (1.8)$$

Изменение ΔT_i приводит к образованию однопараметрического интервального корневого годографа благодаря перемещению корней от полюсов функции (1.8) к ее нулям.

Грани множества ребер P_m , будучи рассмотренными в плоскости двух интервальных параметров T_i и T_j для одной вершины V_q , представляют собой прямоугольник. Опираясь на (1.5), можно написать уравнение отображения плоскости грани G_{ij}^q :

$$D_q(s) + \Delta T_i \cdot A_i(s) + \Delta T_j \cdot A_j(s) = 0 \quad (1.9)$$

Пусть $s_r = \alpha + j\beta$, $r \in \overline{1, n}$. Подставляя в (1.9) координаты s_r , получаем систему двух линейных уравнений с двумя переменными T_i и T_j :

$$\begin{cases} T_i \cdot \operatorname{Re}(A_i(\alpha + j\beta)) + T_j \cdot \operatorname{Re}(A_j(\alpha + j\beta)) + \operatorname{Re}(D_q(s)) = 0; \\ T_i \cdot \operatorname{Im}(A_i(\alpha + j\beta)) + T_j \cdot \operatorname{Im}(A_j(\alpha + j\beta)) + \operatorname{Im}(D_q(s)) = 0. \end{cases} \quad (1.10)$$

Уравнение (1.10) отличается двумя случаями:

1. Уравнения независимы и система имеет единственное решение $T_i = T_i^*$, $T_j = T_j^*$. Следовательно, $\phi^{-1}(s_r) = P^*$, $P^* = (T_i^*, T_j^*)$, причем точка P^* принадлежит плоскости G_{ij}^q .

2. Уравнения отличаются постоянным множителем. Из этого следует, что в плоскости G_{ij}^q существует прямая t , описываемая любым из уравнений системы (1.10), причем $\phi^{-1}(s_r) = t$.

Границы, при отображении P_m , могут содержать полную реберную ветвь, так и частично, пересекаясь в U^* . Условия наличия U^* — наличие линейной зависимости уравнения (1.11):

$$a_i s^i + a_j s^j + P(s) = 0. \quad (1.11)$$

Данное равенство выполняется при условии наличия α и β , что выполняются равенства:

$$\frac{\operatorname{Re} A_i(\alpha + j\beta)}{\operatorname{Im} A_i(\alpha + j\beta)} = \frac{\operatorname{Re} A_j(\alpha + j\beta)}{\operatorname{Im} A_j(\alpha + j\beta)} = \frac{\operatorname{Re}(D_q(\alpha + j\beta))}{\operatorname{Im}(D_q(\alpha + j\beta))} \quad (1.12)$$

Следовательно, необходимо решить систему нелинейных уравнений:

$$\begin{cases} \operatorname{Re}(A_i(\alpha + j\beta)) \cdot \operatorname{Im}(A_j(\alpha + j\beta)) - \\ - \operatorname{Re}(A_j(\alpha + j\beta)) \cdot \operatorname{Im}(A_i(\alpha + j\beta)) = 0; \\ \operatorname{Re}(A_j(\alpha + j\beta)) \cdot \operatorname{Im}(D_q(\alpha + j\beta)) - \\ - \operatorname{Im}(A_j(\alpha + j\beta)) \cdot \operatorname{Re} D_q(\alpha + j\beta) = 0. \end{cases} \quad (1.13)$$

Следует отметить, что левые части уравнений системы (1.13) — это полиномы нечетной степени β без свободного члена. Следовательно, если (1.13) не имеет решения с $\beta \neq 0$, то в отображениях рассматриваемой грани и параллельных ей сечений P_m нет U^* . Отсутствие решений (1.13) для всех сочетаний интервальных параметров означает, что границы S_r состоят из непересекающихся реберных ветвей.

1.3. Основные результаты

В текущей главе были введены основные обозначения, которые используются в робастном расширении метода корневого годографа, проанализированы свойства отображения параметрического многогранника ИХП на корневую плоскость при аффинной неопределенности.

ГЛАВА 2. Анализ робастного качества интервальных систем автоматического управления

2.1. Определение граничных вершин при аффинной неопределенности

Допустим, U_q — одно из решений характеристического уравнения на корневой плоскости (рисунок 2.10).

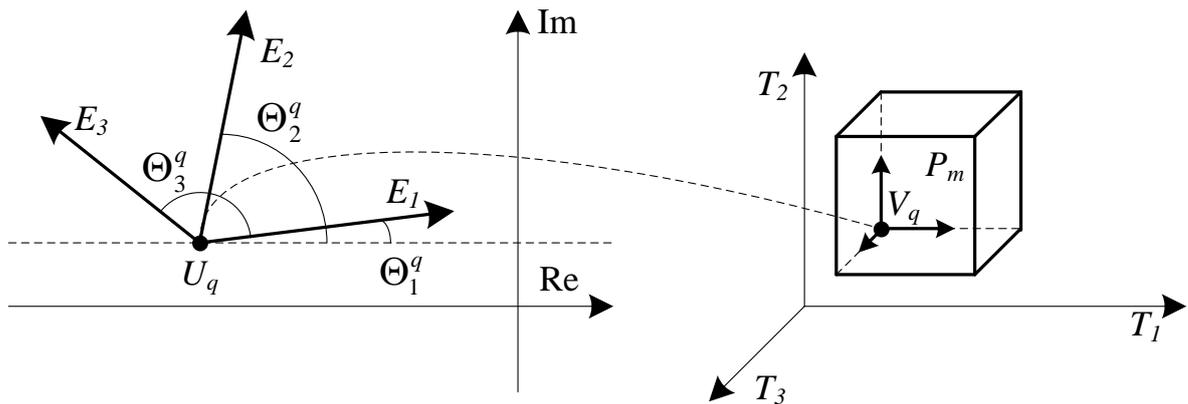


Рисунок 2.10 – Отображение вершин параметрического многогранника

Необходимое для этого определение — угол выхода Θ_i^q из комплексного U_q . Находится он из уравнения фаз [16], записанного для U_q . При $D^q(s)$ со степенью n , $A_i(s)$ со степенью z , тогда, при увеличении T_i , имеется следующая формула:

$$\Theta_i^q = 180^\circ - \sum_{k=1}^n \Theta_k + \sum_{l=1}^z \Theta_l, \quad (0.5)$$

при условии уменьшения T_i аналог:

$$\Theta_i^q = -\sum_{k=1}^n \Theta_k + \sum_{l=1}^z \Theta_l, \quad (0.6)$$

где Θ_k и Θ_l - углы между вещественной осью и векторами, направленными из U_q соответственно к k -ому полюсу и к l -ому нулю функции.

На рисунке 2.1 отображены также возможные направления движения s_r из U_q , учитывая грани P_m с общей вершиной V_q . Как можно заметить,

направление s_r определяется следующим вектором – $\vec{E}_{ij} = \vec{E}_i + \vec{E}_j$, где \vec{E}_i и \vec{E}_j – реберные векторы, задаваемые Θ_i^q и Θ_j^q [2]. Согласно правилам векторной алгебры, \vec{E}_{ij} выходит из U_q и лежит внутри образованного \vec{E}_i и \vec{E}_j угла Ψ_{ij} , причем $0^\circ \leq \Psi_{ij} \leq 180^\circ$.

Из вершины GU_q выходят две граничные реберные ветви, которые образуют граничный угол $G\Psi_{ij}$ в диапазон $[0^\circ, 180^\circ]$. Из чего следует, что прочие углы Ψ_{ij} и соответствующие векторы принадлежат $G\Psi_{ij}$. Используя ранее обозначенные углы выхода, можно выразить неравенство, описывающее обозначенное условие:

$$|\Theta_m^{Vq} - \Theta_1^{Vq}| < 180^\circ \quad (0.7)$$

Благодаря (2.3) можно установить принадлежность вершины P_m границе области S_r локализации комплексного корня ИХП.

Рисунок 2.2 демонстрирует возможное расположение векторов выхода RS_i^q из граничного корневого узла (рис. 2.2а) и расположенного внутри границы (рис. 2.2б).

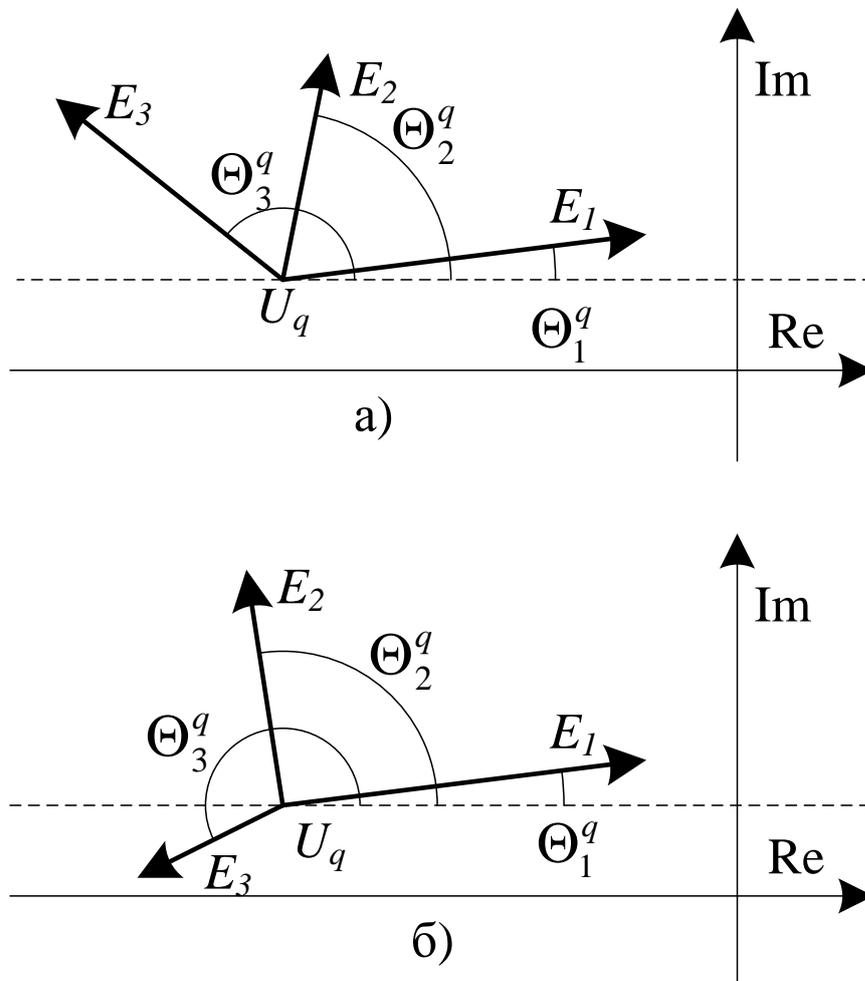


Рисунок 2.11 – Расположение векторов выхода RS_i^q из корневого узла

2.2. Реберный анализ робастного качества системы при аффинной неопределенности

Для анализа рассматриваются реберные ветви (RS_i и RS_j) из соседних граничных корневых узлов и соотношения углов выхода между ними. Так как они связаны одной граничной реберной ветвью GRS_k , рассматриваемые ветви также являются прообразами ребер одной грани P_m .

Пусть $U^* \notin S_r$. При изменении T_k по ребру GR_k корень s_r движется по GRS_k . При этом могут изменяться углы выхода из s_r ветвей RS_i^s и RS_j^s . Однако, так как $U^* \notin S_r$, то RS_i^s и RS_j^s не пересекаются. Поэтому при переходе

s_r , от одного GU_q к другому будет сохраняться последовательность величин Θ_i^s и Θ_j^s (рисунок 2.12а). Обобщая этот случай на большее число рассматриваемых реберных ветвей, можно заключить, что в соседних граничных корневых узлах значения их углов выхода будут располагаться в одинаковой последовательности.

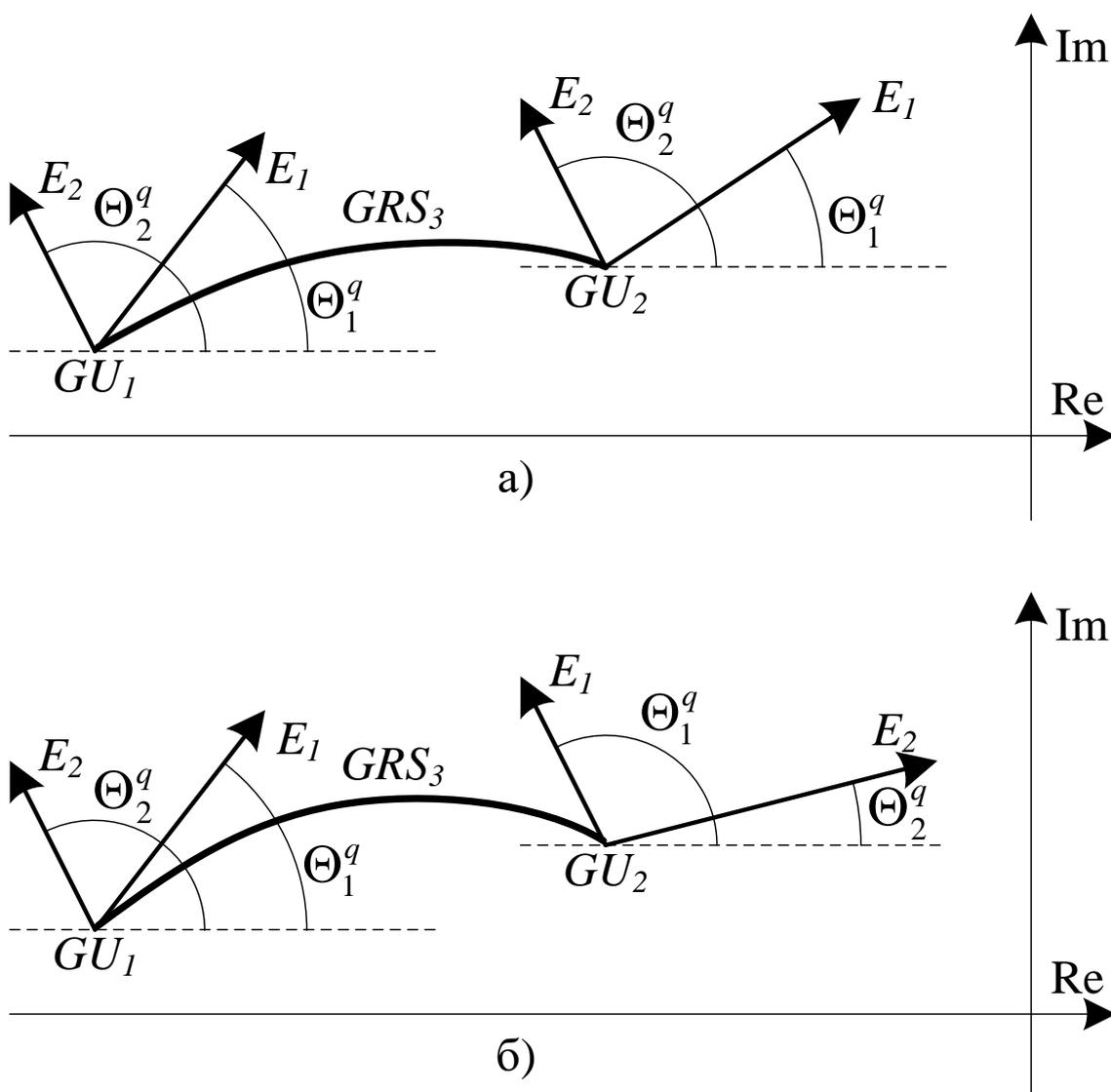


Рисунок 2.12 – Расположение векторов выхода RS_i^q из корневого узла при варьировании одного параметра

Пусть $U^* \in S_r$, тогда RS_i^s и RS_j^s могут пересекаться в U^* . Если при изменении T_k возникнет ситуация $s_r = U^*$ и далее U^* выйдет из S_r , то на оставшемся участке GRS_k изменится соотношение Θ_i^s и Θ_j^s . В этом случае в соседних граничных корневых узлах не будет сохраняться последовательность углов выхода реберных ветвей (рисунок 2.12б).

На основе выше сказанного можно установить шаги, формирующие сам алгоритм поиска граничной реберной маршрутизации P_m .

Допускается предположение, что система (1.10) с аффинной неопределенностью не имеет решения ни для одной из граней P_m с выбранной общей вершиной ($U^* \notin S_r$). В таком случае, при известных GU_q и выходящая из него GRS_i^q , последующая граничная реберная ветвь определяется модулем разности между собственным углом выхода и аналогичным для GRS_i^q , а именно наименьшим (наибольшим) его значением. Таким образом, располагая информацией о величинах углов Θ_i^q , $i = \overline{1, m}$ для любого граничного корневого узла, а затем расположив их в порядке возрастания (убывания), можно определить последовательность ветвей RS_i^q , $q = \overline{1, 2m}$, $i = \overline{1, m}$ ограничивающих область локализации комплексного корня.

Данная последовательность и является упорядоченным набором ребер P_m , то есть замкнутым граничным реберным маршрутом, состоящим из $2m$ ребер, связывающих m пар вершин P_m , где координаты пар имеют противоположные граничные значения интервальных параметров.

Для системы (1.10) достаточно единственного решения, чтобы можно было свидетельствовать о наличии U^* в области S_r . Если осуществляется описанный сценарий, значит, необходимо найти узел GU_q , для которого определить последовательность Θ_i^q и соответствующую ей последовательность ребер P_m . Таким образом получается реберный маршрут, который должен также

включать в себя все ребра грани G_{ij} , при условии, что очередное ребро и следующее за ним образуются параметрами T_i и T_j , при которых рассматриваемая система имеет решение.

Подытожив, можно сформировать следующий алгоритм анализа системы с интервальными параметрами:

1. Необходимо привести характеристический полином к виду (1.5).
2. Определить координаты вершин реберного многогранника P_m , образованные граничными значениями интервальных параметров.
3. Определение граничного узла U^* из (1.10), путем нахождения углов выходов из проекции вершины параметрического многогранника на корневой плоскости.
4. Построение реберного маршрута.
5. Анализ расположения области локализации корней, определение показателей робастного качества системы.

Ниже представлен более детальный разбор построения реберного маршрута.

Для исследования взят аффинный характеристический полином с интервальными параметрами, максимальная степень полинома равна l .

$$P(s) = B(s) + \sum_{i=0}^m (T_i A_i^*(s)), \quad (2.3)$$

где $B(s) = \sum_{j=0}^k (b_j s^j)$, $A_i^*(s) = \sum_{j=0}^l (a_{i,j} s^j)$, $\underline{T}_i < T_i < \bar{T}_i$.

Так как полином $B(s)$ может быть любой степени, то нет такой пары интервальных параметров (2.3), чтобы имела решения, следовательно, в области S_r невозможно наличие U^* .

Нули характеристического полинома отображаются на вещественной оси (рисунок 2.4).

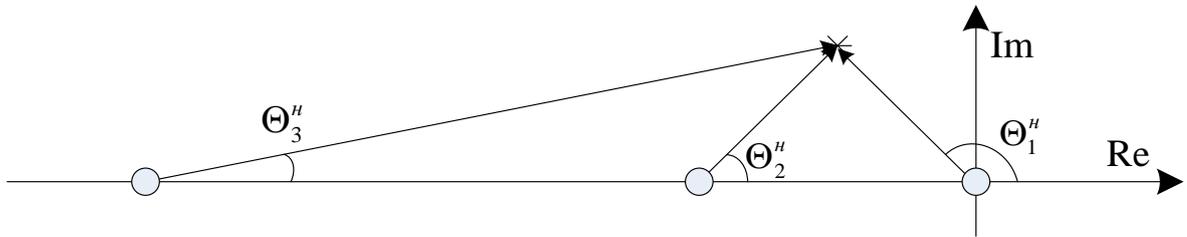


Рисунок 2.13 – Пример расположения нулей характеристического полинома с аффинной неопределенностью с максимальным значением степени полинома при интервальных параметрах, равным l

Это приводит к тому, что существует последовательность углов выхода $\Theta_1^n > \Theta_2^n > \dots > \Theta_n^n$, когда нули передаточных функций образуют соответствующую последовательность $s_1 > s_2 > \dots > s_m$. Рассматриваемый на рисунке 2.4 узел является граничным, так как разность крайних углов для представленной последовательности всегда меньше 180° .

Ранее были описаны уравнения фазовых соотношений (2.1) и (2.2). Согласно им, разность крайних углов выхода из вершины, образованной граничными значениями интервальных коэффициентов (всеми минимальными или всеми максимальными), будет меньше 180° , т.к. минимальный угол от нуля до полюса равен 0° (в случае, если нуль стремится к минус ∞), а максимальный равен 180° (в случае, если нуль стремится к плюс ∞).

Далее следует рассмотреть векторы углов выхода из граничной вершины (рисунок 2.5).

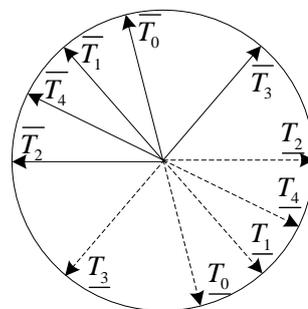


Рисунок 2.14 – Пример изображения векторов углов выхода из граничной вершины

Если последовательно изменять интервальные параметры на противоположные в порядке их увеличения (уменьшения), можно получить все граничные вершины (рисунок 2.6), из которых и состоит искомый минимальный реберный маршрут.

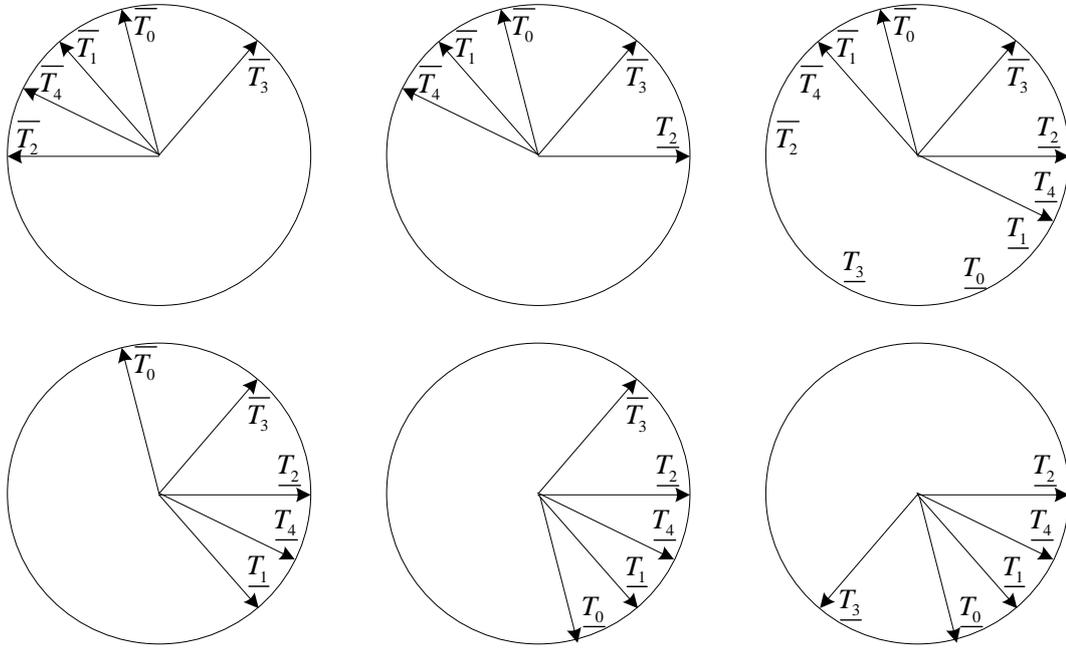


Рисунок 2.15 – Вектора углов выхода из граничных вершин

Алгоритм определения границ областей локализации корней полинома с интервальными параметрами предусматривает выполнение следующих этапов:

1. Приведение характеристического полинома системы к виду (2.3).
2. Определение координат нулей характеристического полинома.
3. Формирование последовательности координат нулей характеристического полинома в порядке возрастания (убывания).
4. Формирование вершины с максимальными значениями интервалов параметров, записанных в порядке возрастания (убывания) нулей.
5. Формирование реберного маршрута последовательным изменением граничных значений интервалов параметров в порядке, записанном в п.4.
6. Отображение реберного маршрута на корневую плоскость.

7. Анализ расположения областей локализации корней интервального полинома и определение робастного качества системы.

2.3. Основные результаты

В данной главе рассмотрен алгоритм анализа, который стал основой для текущего исследования. Основные концепции данного подхода, как минимизация анализируемых данных, корневой подход, будут использованы при рассмотрении нового метода, рассмотренного в следующей главе.

ГЛАВА 3. Целенаправленный анализ ребер и синтез ПИ–регуляторов

3.1. Синтез ПИ–регулятора

На рисунке 3.1 представлена типизированная САУ с обратной связью с ПИ-регулятором.

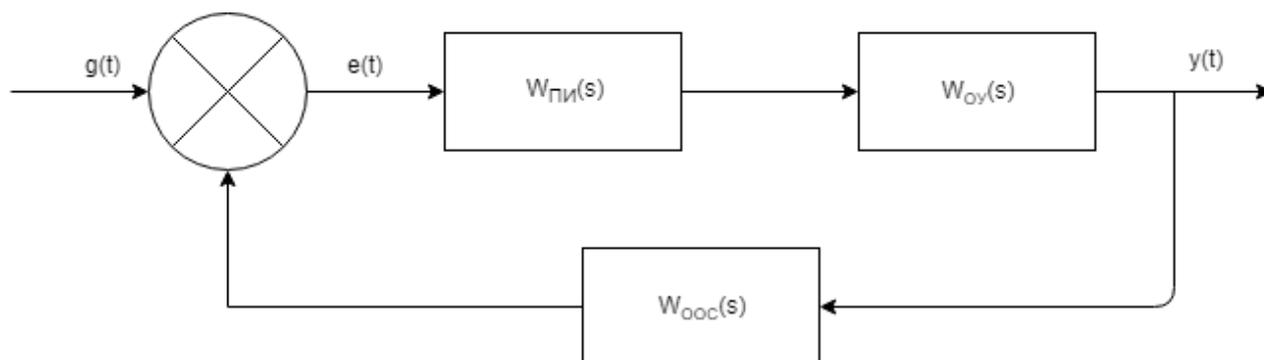


Рисунок 3.1 — Типовая схема САУ с обратной связью

Представленная САУ состоит из трех компонентов: регулятор, объект управления, отрицательная обратная связь. Эти блоки могут быть описаны следующими передаточными функциями:

$$W_{\text{ПИ}}(s) = K + \frac{T}{s} \quad (4.1)$$

$$W_{\text{ОУ}}(s) = \frac{A(s)}{B(s)}, \quad (4.2)$$

$$W_{\text{ООС}}(s) = -1, \quad (4.3)$$

где K – коэффициент пропорциональности, T – время интегрирования.

Для такой САУ передаточная функция будет следующей:

$$W_{\text{ЗС}}(s) = \frac{K + \frac{T \cdot A(s)}{s B(s)}}{1 + K + \frac{T \cdot A(s)}{s B(s)}}. \quad (4.4)$$

Теперь, согласно (4.4), можно выделить характеристическое уравнение, которое в дальнейшем будет использоваться для синтеза:

$$D(s) = s \cdot B(s) + (K \cdot s + T) \cdot A(s) \quad (4.5)$$

Приравняв (4.5) к нулю, получится уравнение, из которого можно найти коэффициенты регулятора. Для этого один из коэффициентов необходимо приравнять к нулю. Затем провести корректировку, зафиксировав ранее найденный коэффициент.

Подставив в (4.5) вершины корневого годографа, можно, изменяя исследуемый коэффициент с заданным шагом, наблюдать за изменением расположения области корней. Так как, несмотря на большое разнообразие методов синтеза параметров регулятора для определенных систем (системы, где коэффициенты полинома заданы явно), наибольшую популярность снискал подход эмпирического перебора значений. Текущий подход можно перенести и на неопределенные системы благодаря (4.5). Более того, перебор значений благодаря вычислительным мощностям ЭВМ приобретает куда большую ценность. Проблема возникает из-за того, что неопределенные системы предполагают работу с большим объемом данных. Во второй главе рассмотрен метод минимизации реберного маршрута, позволяющий сократить количество исследуемых данных. В таблице 4.1 представлены результаты такого подхода.

Таблица 4.1 — Сравнительный анализ методов анализа

Интервальный полином с аффинной неопределенностью					
Количество интервальных коэффициентов	Количество точек на ребре	Время, с		$t_{\text{реб}}/t_{\text{min}}$	Средне-арифметическое $t_{\text{реб}}/t_{\text{min}}$
		Реберная теорема ($t_{\text{реб}}$)	Прохождение по минимальному маршруту (t_{min})		
4	5	0,8	0,18	4,4444	4,1111
4	20	2,2	0,55	4	

4	50	5,2	1,3	4	
4	100	10	2,5	4	
5	5	1,7	0,2	8,5	8,4010
5	20	5,5	0,6	9,1667	
5	50	12,7	1,6	7,9375	
5	100	24,8	3,1	8	
6	5	3,8	0,25	15,2	
6	20	14,8	0,9	16,4444	
6	50	34,5	2,2	15,6818	
6	100	66,9	4,1	16,3171	

Однако и данный подход вынуждает тратить определенное время на получение результатов, что, в случае с итерационным подходом для синтеза, вынуждает пользователя затрачивать большое количество времени. Поэтому необходим иной способ анализа данных.

3.2. Целенаправленный анализ

Допустим, имеется следующий корневой годограф (см. рисунок 3.2):

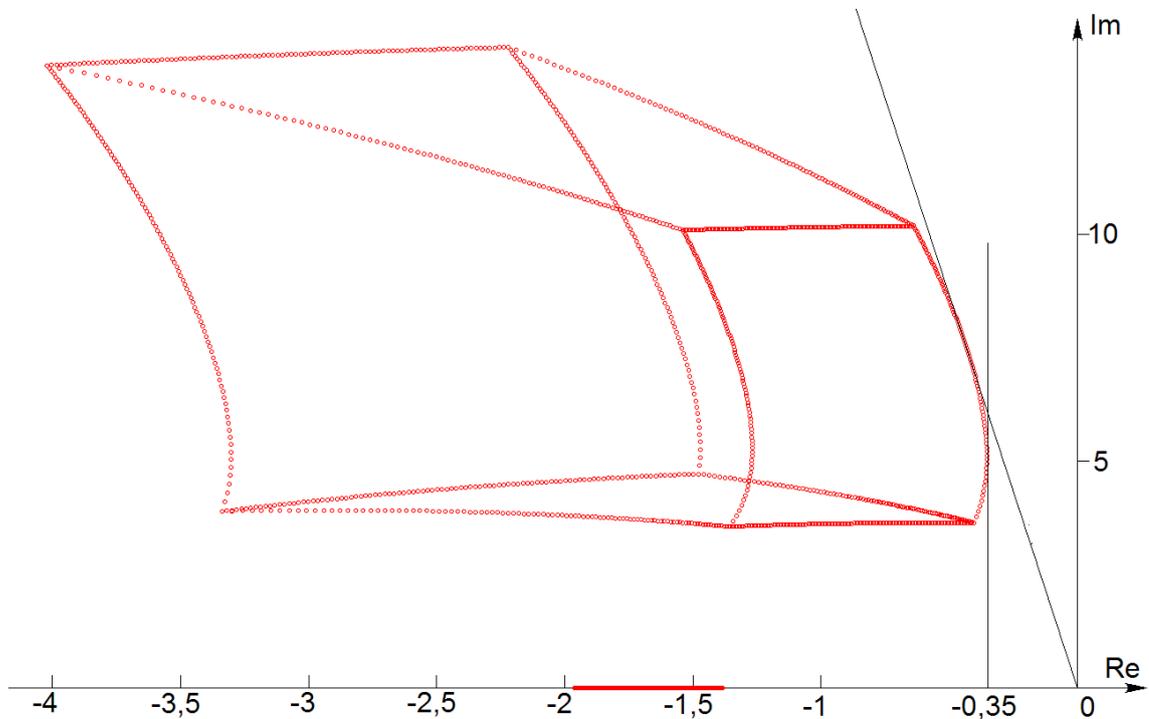


Рисунок 3.2 — Пример корневого годографа

На рисунке 3.2 видно, что определяющие корни расположены на одном из ребер. В общем случае, два корня, описывающие показатели системы, могут располагаться на двух ребрах. Это значит, что большая часть ребер являются излишни при проведении анализа. Первой задачей является определение этих ребер. Будет очевидным предположение, что ребра, образованные вершинами, которые максимально близко к искомым корням, будут содержать эти самые вершины. Это означает, что из всего множества вершин требуется определить те, которые были бы максимально близки к искомым (расположены на наименьшем удалении от мнимой оси и образующие максимальный угол с вещественной осью). Затем от найденных вершин начать построение ребер, изменяя параметры системы. Таким образом будут образовываться ребра, максимально близко расположенные от искомым точек. Это приведет к тому,

что количество ребер будет определяться количеством параметров системы, умноженным на две (две вершины). Следующая проблема состоит в количестве точек, которыми будет определяться ребро. Чем больше дискретизация ребра, тем точнее будет результат, однако это приведет к большему времени анализа, что требуется избежать. Для этого было принято учитывать направление изменения ребра. Это означает, что обход ребра необходимо осуществлять до момента, пока ребро действительно определяет возможную искомую точку. Если ребро стало, например, отдаляться от мнимой оси, когда как задача была найти минимальную степень устойчивости, необходимость строить ребро дальше отпадает.

Для данного алгоритма также возможна модификация, основанная на дихотомическом методе поиска экстремума. Она состоит в следующем. Из диапазона изменения некоторого параметра системы достаточно взять центральное значения и соседние по обе стороны с некоторым шагом, определяющим точность алгоритма. Это позволит значительно сократить количество рассматриваемых точек и, таким образом, ускорить алгоритм анализа системы. Результаты представлены в таблице 4.2

Таблица 4.2 — Численные результаты работы алгоритма

Интервальный полином с аффинной неопределенностью			
Количество интервальных коэффициентов	Количество точек на ребре	Время, с	Порядок уравнения
4	10^5	6,65	3
4	10^5	6,55	3
4	10^5	16,7	4
5	10^5	16,34	4
5	10^5	26,01	5
5	10^5	26,55	5

3.3. Основные результаты

В данной главе рассмотрен подход, позволяющий производить синтез ПИ–регулятора. Он основан на улучшении анализа системы и итерационном поиске желаемых показателей. Для анализа был предложен целенаправленный поиск ребер, основанный на двух модификациях: проверка направления роста ребра и дихотомический поиск экстремума.

ГЛАВА 4. Программная реализация алгоритмов анализа и синтеза интервальных систем

4.1. Анализ программных сред

Процесс проектирования система автоматического управления всегда сопряжен с необходимостью проводить численные расчеты, анализировать экспериментальные данные, выполнять моделирование и прочее. В связи с этим существуют различные специализированные инструменты, решающие те или иные задачи, которыми они ограничены. Именно это условие вынуждает разработчика в сторону инструмента с большей степенью абстрактности, для которого не имело бы значение частные детали.

Наибольшей популярностью среди программистов в сфере автоматизированных систем управления в данный момент пользуются общематематические пакеты Mathcad и MatLab. Приведенные инструменты решают различные численные и аналитические задачи, в том числе выполняя графическое представление результатов. Каждый из инструментов обладает своим языком программирования, представляющие собой пример специализированных высокоуровневых языков программирования. С одной стороны, это позволяет с относительной простотой решать множество стандартных задач, благодаря встроенным библиотекам и готовым компонентам, но с другой стороны имеющиеся языки вносят определенные ограничения. Будучи крайне высокоуровневыми, они ограничивают разработчика в свободе выбора алгоритмов, архитектурных решений ПО, сужая поле деятельности. Это позволяет низкоквалифицированным специалистам в области программирования решать определенный спектр задач, однако данные языки из-за этих ограничений не соответствуют стремительно развивающейся области программирования. Отсутствие различных явлений, таких как ООП, компонентный подход приводят к созданию тяжело поддерживаемых и сложных программ, не соответствующих современным реалиям.

Данная проблема может быть решена благодаря использованию языков программирования более широкого профиля. Таким примером служит язык C#. Данный язык позволяет решать множество задач, обладает возможностью интеграции с вышеупомянутыми инструментами, а также имеет в распоряжении множество библиотек, упрощающих разработку конкретно математических приложений.

C# — это язык, относящийся к так называемой семье C-подобных языков, где ближе всего по синтаксису к нему C++ и Java. Отличается статической типизацией (в противовес языкам MatLab и Mathcad), поддерживает такие возможности, как полиморфизм, перегрузку операторов, свойства, делегаты, события, атрибуты, обобщенные типы и методы, анонимные функции с поддержкой замыканий, LINQ, исключения, итераторы, комментарии в формате XML.

Проекты, созданные с помощью Mathcad и MatLab, неразрывно связаны со средой разработки. Это приводит к невозможности получения результатов без наличия соответствующего ПО. Эта задача легко решается с помощью интегрированной среды разработки Visual Studio. Проект, написанный на C#, данная среда может скомпилировать на исполнимый файл расширения .exe, который будет воспроизводиться на любом компьютере.

4.2. Архитектура приложения

Архитектура приложения представлена на рисунке 4.1.

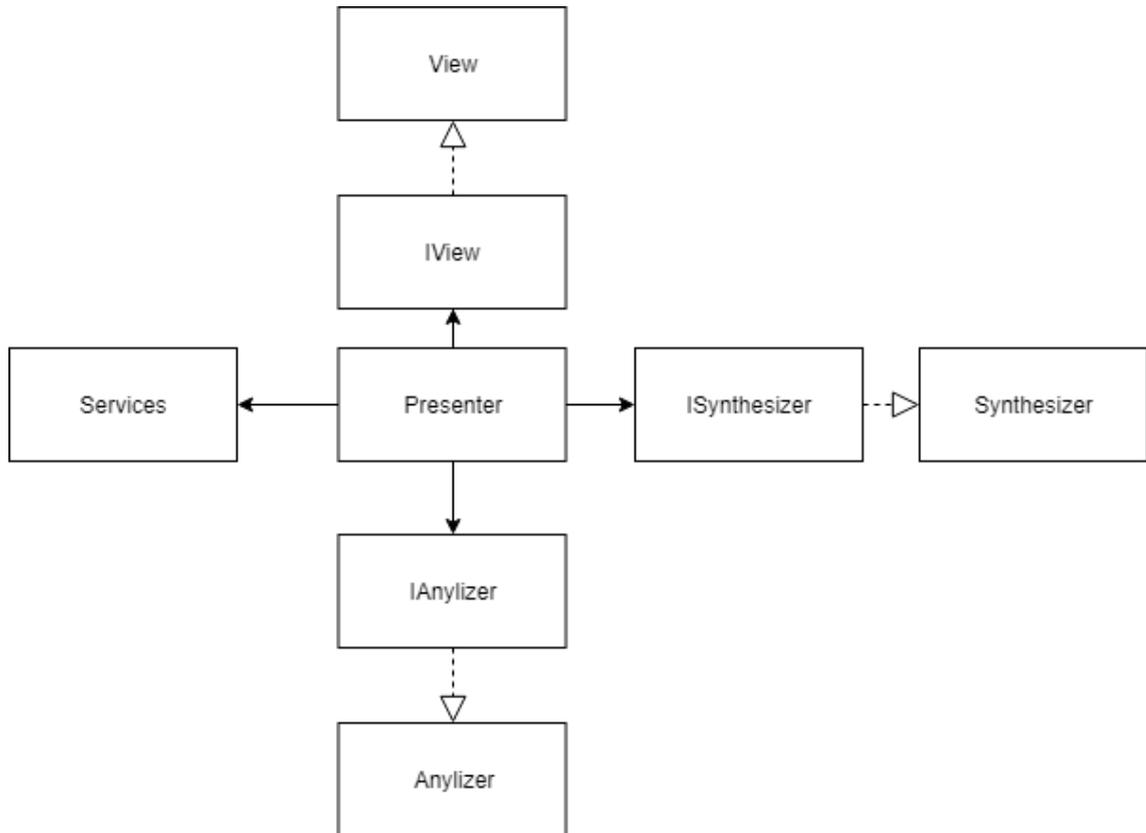


Рисунок 4.1 — Архитектура приложения

Данная архитектура основана на распространенном для десктопных приложений паттерне MPV (model–presenter–view). Его главной особенностью является разделение бизнес-логики приложения и представления результатов, а также обеспечение слабой связанности между компонентами. В данном приложении используются следующие модули:

- Presenter (представитель) — содержит бизнес-логику, оформленную по паттерну «Стратегия» — набор инструкций, каждая из которых подменяется при необходимости.
- View — представление, содержащее логику отображения данных и сами графические элементы. Не имеет связи с другими компонентами, реализую паттерн «Подписчик». В данном случае представитель подписывается на изменения представления.

- *Services* — службы, содержащие различные алгоритмы для общего пользования, например, поиск корней полинома.
- Отдельно расположены модули анализа и синтеза, содержащие в себе всю необходимую логику.

Большая часть модулей используются совместно с интерфейсами. Это позволяет реализовать *Dependency Injection* подход, когда формируется слабая связь между компонентами. Это приводит к их взаимозаменяемости без переписывания всего приложения.

4.3. Основные алгоритмы приложения

Сбор данных — состоит из обхода необходимых контроллов формы после инициации пользователем. Все данные собираются в необходимые конструкции и отдаются подписчикам.

Чтение данных — представитель осуществляет обработку данных. Он формирует интервалы, коэффициенты к параметрам и осуществляет поиск вершин корней.

Анализатор — проводит анализ в зависимости от требуемого метода: по ребру или дихотомически. Это *statefull* объект, при создании принимающий определенные параметры.

Синтезатор — проводит синтез параметров и тесно взаимодействует с анализатором.

Службы — в приложении используются три службы: поиск корней, работа с массивами, работа с интервалами. В них осуществляется поиск корней, обновление массивов коэффициентов, формирование интервалов и прочее.

Листинг кода представлен в приложении А. Пример программной реализации представлен на рисунке 4.2 (основная форма приложения), 4.3 (графическая форма с частичными ребрами).

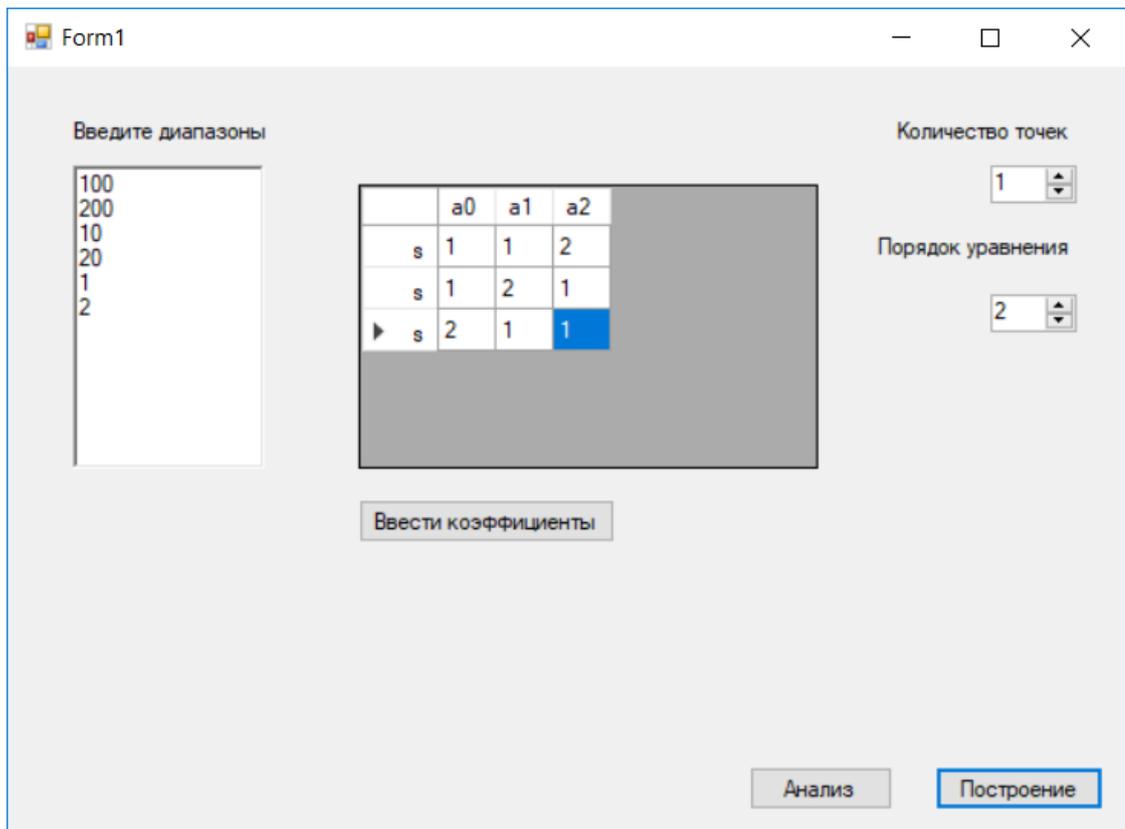


Рисунок 4.2 — Изображение главной формы

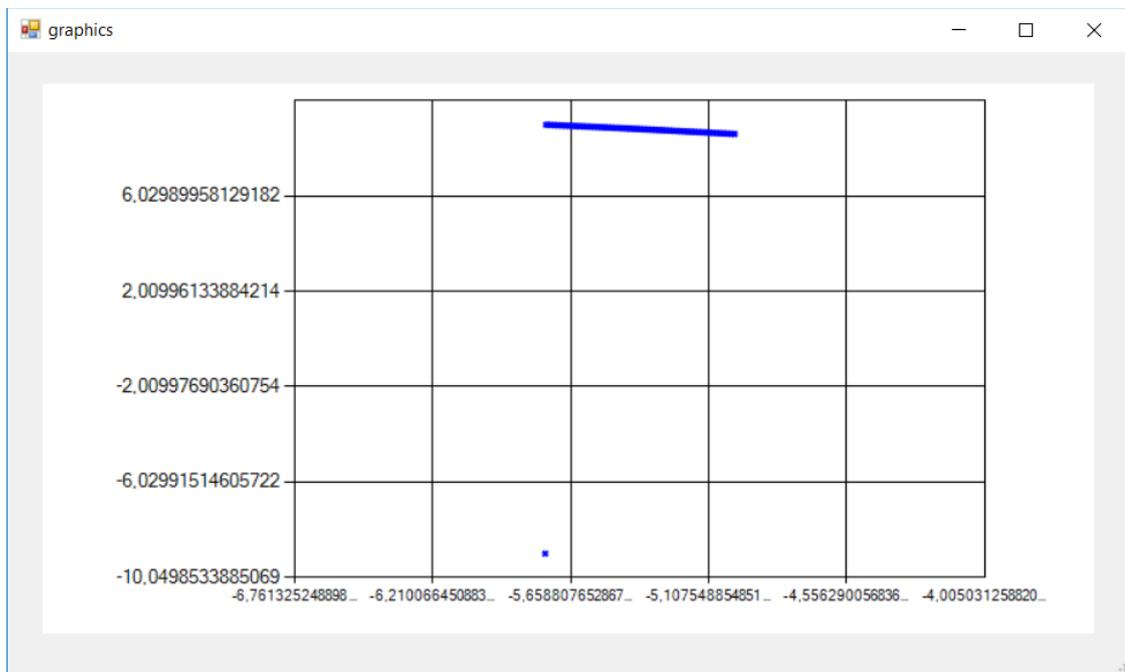


Рисунок 4.3 — Пример частичного построения ребра

ГЛАВА 5. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Диссертационная работа содержит в себе научное исследование, выходным продуктом которого является создание программного обеспечения, реализующего автоматизацию математических расчетов. Таким образом целью раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» является анализ конкурентоспособности и эффективности полученного ПО на современном рынке IT-технологий.

Задачами данного исследования являются:

- Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения;
- Определение возможных альтернатив проведения научных исследований;
- Планирование научно-исследовательских работ;
- Определение ресурсной, финансовой, бюджетной, социальной и экономической эффективности исследования.

5.1. Предпроектный анализ

5.1.1. Потенциальные потребители результатов исследования

Основная целенаправленность текущей разработки преследует цель удовлетворить потребность предприятий малого звена, в частности достичь большей эффективности за счет добавление автоматизации в процесс проектирования. Описание приведено в таблице 1.

Таблица 1.1.1.1 — Карта сегментирования рынка услуг

		Услуги по автоматизации проектирования		
		Документооборот	Плагины для ПО	Научное проектирование
Размер компании	Большая	Томск НИПИ нефть	Томск НИПИ нефть	Томск НИПИ нефть
	Средняя			ЭлеСи
	Малая	ТомскАСУпроект	ТомскАСУпроект	

Таким образом, в данной работе были выявлены возможные конкуренты, отобранные по отрасли и региону (г. Томск). Представленные конкуренты работают в сфере автоматизации производства, что является целевой областью применения продукта текущей диссертации.

ТомскАСУпроект — компания, основная направленность которой это создание ПО в сфере добычи нефти и газа. Они фокусируют свое внимание на обеспечение автоматизации документооборота, проектирование и работы с геоданными.

ЭлеСи — компания, производящая продукцию в сфере автоматизации, выполняющая проектную и пуско-наладочную работы. Дополнительно они выпускают ПО для своего оборудования и работы с автоматизированными системами управления.

Томск НИПИ нефть — компания, обеспечивающая полный цикл по добыче нефти и газа, от проектирования добывающих установок до веб-приложений.

Основные потребители продукта данной диссертации — малые и в меньшей степени средние предприятия, занимающиеся автоматизацией сложных объектов, где необходимо проводить научные расчеты, а требования к разрабатываемым системам повышены.

Согласно приведенной карте сегментирования, можно сделать вывод, что самая слабая конкурентная среда — это наукоемкое проектирование среди малых компаний. Именно на эту область и нацелен текущий проект.

5.1.2. SWOT–анализ

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – представляет собой комплексный анализ научно-исследовательского проекта. SWOT-анализ применяют для исследования внешней и внутренней среды проекта.

Согласно описанному выше приводит SWOT–анализ в табличной форме (см. Таблица 2).

Таблица 1.1.2.1 — SWOT–анализ

	Сильные стороны научно-исследовательского проекта:	Слабые стороны научно-исследовательского проекта:
	<p>C1. Удобство в эксплуатации (соответствует требованиям потребителей).</p> <p>C2. Функциональная мощность (предоставляемые возможности).</p> <p>C3. Конкурентоспособность продукта.</p> <p>C4. Срок выхода на рынок.</p> <p>C5. Высоко квалифицированный научный труд.</p>	<p>Сл1. Отсутствие прототипа научной разработки.</p> <p>Сл2. Отсутствие у потенциальных потребителей квалифицированных кадров.</p> <p>Сл3. Отсутствие необходимого оборудования для проведения испытания ПО.</p> <p>Сл4. Уровень проникновения на рынок.</p> <p>Сл5. Большой срок поставок плат, используемых для проведения научного исследования.</p>
Возможности:	Использование инновационной структуры ТПУ позволит повысить конкурентоспособность ПО и ускорить выход на рынок. Так же использование развитой международной инфраструктуры поможет ускорить выход ПО на рынок. Возможно появление дополнительного спроса на новый продукт благодаря использованию высоко квалифицированного научного труда. Благодаря снижению таможенных пошлин на платы возможно повышение конкурентоспособности ПО.	Появление дополнительного спроса на новый продукт может привести к отсутствию у потенциальных потребителей квалифицированных кадров. Снижение таможенных пошлин на сырье и материалы, используемые при научных исследованиях может привести к увеличению срока поставки плат, используемых для проведения научного исследования.
<p>V1. Использование инновационной инфраструктуры ТПУ для быстрого внедрения ПО на рынок.</p> <p>V2. Использование развитой международной инфраструктуры для более быстрой доставки плат.</p> <p>V3. Появление дополнительного спроса на новый продукт.</p> <p>V4. Снижение таможенных пошлин на сырье и материалы,</p>		

используемые при научных исследованиях. В5. Повышение стоимости конкурентных разработок.		
Угрозы: У1. Отсутствие спроса на новые технологии производства. У2. Развитая конкуренция производителей ПО. У3. Ограничения памяти платы Arduino. У4. Выпуск более новых чипов для Arduino. У5. Высокая стоимость оборудования и плат.	Отсутствие спроса на новые технологии производства может замедлить срок выхода ПО на рынок и понизить квалификацию научного труда. Развитая конкуренция производителей ПО может привести к снижению конкурентоспособности продукта. Ограничения памяти платы Arduino и высокая стоимость оборудования и плат требует более высоко квалифицированный научный труд и затягивает срок выхода на рынок.	Отсутствие спроса на новые технологии производства и высокая стоимость оборудования и плат может привести к отсутствию прототипа научной разработки, отсутствию потенциальных потребителей, необходимого оборудования для проведения испытания ПО, ухудшить уровень проникновения на рынок и увеличить сроки поставки плат. Выпуск более новых чипов для Arduino может способствовать к отсутствию необходимого оборудования для проведения испытания ПО.

Благодаря анализу выделены такие особенности, как роль ТПУ в распространении продукта, а также возможные последствия снижения спроса и усиление конкуренции.

5.1.3. Диаграмма Исикавы

Диаграмма причины-следствия Исикавы (Cause-and-Effect-Diagram) – это графический метод анализа и формирования причинно-следственных связей, инструментальное средство в форме рыбьей кости для систематического определения причин проблемы и последующего графического представления (см. Рис 1.1.3.1).



Рисунок 1.1.3.1 — Диаграмма Исикавы

Основные этапы при создании конечного продукта: методология, разработка, требования, контроль качества.

Методология включает в себя создания научной базы для будущего приложения, а также необходимые знания непосредственно в программировании. Сюда относятся указанные проблемы: методология описана недостаточно, нет понимания методологии, отступление от методологии. Недостаточное описание предполагает наличие пробелом в теоретической базе, доступной во время разработки. Важным также является ее полное понимание, что содержит в себе риск, учитывая значительную наукоемкость материала. Отступление от методологии — это внесение погрешностей, связанных с трудностями в реализации.

Требования определяют общий каркас продукта. К проблемам относятся: ведение нескольких проектов одновременно, переделки из-за непонимания требований, нечеткие требования, требования меняются, недоступен заказчик. В данном контексте основные требования исходят от научного руководителя, которые сопряжены с указанными рисками. Проблема

ведения нескольких проектов сказывается на погруженности и заинтересованности. Недоступен заказчик — серьезная проблема, связанная из-за специфики работы — научная диссертация.

Разработка — это основной процесс в создании ПО со стороны исполнителя. Выделенные проблемы: недостаток обучающих мероприятий, нехватка квалификации разработчика, нет важности понимания TDD, нет модульных тестов, нет времени на тесты, нет аудита кода и парного программирования, слабо используются специальные инструменты. Мероприятия и квалификация сказываются на скорости разработки и ее непосредственном качестве. Создания тестов программистом является важной компонентой в разработке по очень часто не на enterprise-уровне игнорируется. Отсутствие аудита кода (review) приводит к накоплению ошибок, а неиспользование специальных инструментов — к ослаблению конкурентоспособности.

Контроль качества — важный этап, определяющий выход продукта в свет. Выделены следующие риски: отсутствие автотестов, контроль качества осуществляется постфактум, не проводится тестирование в бою, тестируются большие куски функционала. Текущие риски сопряжены с отсутствием в процессе разработки компетентного тестировщика, а возможность провести тестирование в бою — отсутствие конкретного заказчика.

На выходе сбрасывания рисков возможны дефекты, такие как некорректное поведение ПО согласно сценарию возможного показа, не эргономичный интерфейс, наличие ошибок и неточностей в продукте самого ПО.

Согласно представленному анализу, можно выделить слабые места в алгоритме создания конечного продукта и примерное (эмпирически выведенное) влияние каждого из них. Несмотря на отсутствие конкретного

заказчика и тестировщика, главным риском служит процесс разработки и методологии, в первую очередь из-за продолжительности

6.1. Инициализация проекта

6.1.1. Структура работ в рамках научного исследования

В данном разделе необходимо составить перечень этапов и работ в рамках проведения научного исследования, провести распределение исполнителей по видам работ. Примерный порядок составления этапов и работ, распределение исполнителей по данным видам работ приведен в табл. 1.2.1.1.

Таблица 1.2.1.1 – Перечень этапов, работ и распределение исполнителей

№	Перечень работ	Исполнители
<i>Подготовительный</i>		
1	Составление задания ,подбор литературы	Инженер Научный руководитель
2	Ознакомление с литературой и элементной базой	Инженер Научный руководитель
3	Разработка технических предложений	Инженер Научный руководитель
<i>Разработка алгоритмической базы приложения</i>		
4	Исследование технологического стека	Инженер
5	Тестирование сторонних программных библиотек	Инженер
6	Составление архитектуры приложения	Инженер
<i>Разработка программного обеспечения</i>		
7	Экспериментальная работа с C#	Инженер
8	Экспериментальная работа Visual Studio 2017	Инженер
9	Разработка программно-методического обеспечения	Инженер
10	Проведение исследований	Инженер Научный руководитель
11	Редактирование и отладка программы	Инженер
12	Подготовка методического обеспечения	Инженер
<i>Заключительный</i>		
13	Разработка презентации	Инженер

14	Мероприятия по безопасности и экологичности работы	Инженер
15	Технико-экономическое обоснование ВКР	Инженер
16	Составление и оформление пояснительной записки	Инженер
17	Защита ВКР	Инженер

В основном, порядок работ распределен таким образом, чтобы задействовать стороннего человека (научного руководителя) только на определенный неразрывный промежуток времени, тем самым обеспечив большую независимость и свободу в выполнении работ.

6.2. Планирование управления научно-техническим проектом

6.2.1. Определение трудоемкости выполнения работ

При составлении планов комплекса работ используются в основном сетевые и линейные методы планирования.

Сетевые методы планирования используются при большой сложности научно-исследовательских и опытно-конструкторских работ, одновременном участии многих исполнителей, необходимости параллельного выполнения работ и т.д.

Если разработка не велика по объему и не требует большого состава исполнителей, то целесообразно использовать линейный метод планирования.

Система планирования основана на графическом представлении определенного комплекса работ, отражающих их последовательность, взаимосвязь и длительность. Разработка имеет малый штат исполнителей: руководитель (Р) и инженер (И). Так же работа проводится с малыми затратами, поэтому целесообразно применить систему линейного планирования с построением линейного графика.

Для построения линейного графика, разбиваем всю работу на этапы, количество и содержание которых определяется спецификой темы. Объективный экономический расчет позволяет равномерно распределить время работы и нагрузку на исполнителей, а также увеличить эффективность работ.

Процесс разработки разделим на 4 этапа:

- 1) подготовительный;
- 2) разработка алгоритмической базы приложения;
- 3) разработка программного обеспечения;
- 4) заключительный.

В таблице 1.3.1.1 приведен перечень, длительность, исполнители и загрузка основных этапов и работ, имеющих место при разработке лабораторного комплекса.

Таблица 1.3.1.1 - Комплекс работ по разработке проекта

№	Перечень работ	Исполнители	Продолжительность работ, дней	Загрузка	
				дни	%
<i>Подготовительный</i>					
1	Составление задания ,подбор литературы	И Р	1	1 1	100 100
2	Ознакомление с литературой и элементной базой	И Р	1	1 1	100 50
3	Разработка технических предложений	И Р	3	3 1	100 25
<i>Разработка алгоритмической базы приложения</i>					
4	Исследование технологического стека	И Р	2	2 1	100 50
5	Тестирование сторонних программных библиотек	И Р	3	3 2	100 50
6	Составление архитектуры приложения	И	4	4	100
<i>Разработка программного обеспечения</i>					
7	Экспериментальная работа с С#	И Р	2	2 1	100 50
8	Экспериментальная работа Visual Studio 2017	И	3	3	100
9	Разработка программно-методического обеспечения	И	2	2	100
10	Проведение исследований	И Р	19	20 6	100 50
11	Редактирование и отладка программы	И Р	10	12 4	100 50
12	Подготовка методического обеспечения	И Р	3	3 2	100 50
<i>Заключительный</i>					
13	Разработка презентации	И	3	3	100
14	Мероприятия по безопасности и экологичности работы	И	3	3	100
15	Технико-экономическое обоснование ВКР	И	15	15	100
16	Составление и оформление пояснительной записки	И	4	4	100
17	Защита ВКР	И	1	1	100
ИТОГО		И Р	79 19		

Рассчитаем трудоемкость работ, которая определяется по сумме трудоемкости этапов и видов работ, оцениваемых экспериментальным путем

в человеко-днях, и носит вероятностный характер, так как зависит от множества трудно учитываемых факторов, поэтому ожидаемое значение трудоемкости $T_{ож}$ рассчитывается по формуле:

$$T_{ож} = \frac{A_i + 4 \cdot M_i + D_i}{6},$$

где A_i - кратчайшая продолжительность заданной работы (оптимальная оценка), дни;

D_i - самая большая продолжительность работы (пессимистическая оценка), дни;

M_i - наиболее вероятная, по мнению экспертов, продолжительность работы (реалистическая оценка), дни.

Полученные результаты представим в таблице 1.3.1.2.

Таблица 1.3.1.2 — Перечень работ и оценка трудоёмкости

№	Перечень работ	Продолжительность работ, дни			
		A_i	D_i	M_i	$T_{ож}$
1	Составление задания, подбор литературы	1	2	1	1
2	Ознакомление с литературой и элементной базой	1	2	1	1
3	Разработка технических предложений	2	4	3	3
4	Исследование технологического стека	1	3	2	2
5	Тестирование сторонних программных библиотек	2	4	3	3
6	Составление архитектуры приложения	3	5	4	4
7	Экспериментальная работа с C#	1	3	2	2
8	Экспериментальная работа Visual Studio 2017	2	4	3	3
9	Разработка программно-методического обеспечения	1	3	2	2
10	Проведение экспериментальных исследований	16	22	19	19
11	Редактирование и отладка программы	8	12	10	10
12	Подготовка методического обеспечения	2	4	3	3
13	Разработка презентации	2	4	3	3
14	Мероприятия по безопасности и экологичности работы	2	4	3	3
15	Технико-экономического обоснование ВКР	14	17	15	15
16	Составление и оформление пояснительной записки	3	5	4	4
17	Защита ВКР	1	1	1	1
ИТОГО					79

При построении линейного графика (таблица 1.3.1.3) был охвачен весь спектр работ по теме.

Учтено, что дипломник выступает в качестве основного исполнителя, занятого выполнением темы на протяжении всего периода ее проведения, начиная с утверждения ТЗ и вплоть до ее оформления. Поэтому на линейном графике было учтено требование, что не должно быть перерывов в работе дипломника. Для проведения отдельных видов работ был привлечен руководитель проекта. Он осуществлял научный ликбез в целевой отрасли, осуществлял консультацию, проводил анализ работ и помогал сформулировать общий план работ. На графике это будет показано в виде параллельных линий, характеризующих одновременное проведение нескольких видов работ разными исполнителями.

6.2.2. Составление сметы затрат на разработку проекта

В рамках данного раздела было проведено экономическое обоснование затрат на выполнение проекта.

Так как в рамках данного проекта создается одна разработка, определение затрат производится путем составления сметы затрат, т.е. группировка проводится по элементам.

Рассмотрены следующие затраты, образующие себестоимость разработки:

- материальные затраты;
- затраты на заработную плату и социальный налог;
- затраты на электроэнергию;
- амортизация основных фондов;
- накладные расходы.

6.2.3. Расчет сметы затрат на выполнение проекта и на материалы

Основные материальные расходы состоят из канцелярских принадлежностей и картриджей для принтера, так как все необходимые для

работы над проектом материалы имелись в распоряжении кафедры, на которой велась разработка.

Материалы необходимые для выполнения данной работы и расчет материальных затрат представлены в табл. 1.3.3.1.

Таблица 1.3.3.1 – Материальные затраты

Наименование материалов	Цена за ед., руб.	Кол-во	Сумма, руб.
Бумага для принтера формата А4	210	1 уп.	210
Ручка шариковая	30	2 шт.	60
Картридж	1600	1 шт.	1600
Блокнот	45	2 шт.	90
Итого:			1960

Таким образом материальные затраты не превышают двух тысяч рублей и составляют незначительный вклад в общие расходы.

6.2.4. Расчет заработной платы и социального налога

Расчет основной заработной платы выполняется на основе трудоемкости выполнения каждого этапа и величины месячного оклада исполнителя. Месячный оклад (МО) НР, занимающего должность доцента и имеющего степень кандидата технических наук, составляет 33664 руб./мес. МО исполнителя, являющегося инженером первого уровня, составляет 15000 руб./мес.

Исходя из того, что в месяце в среднем 24,83 рабочих дня при шестидневной рабочей неделе среднедневная тарифная заработная плата ($ЗП_{дн-т}$) рассчитывается по формуле:

$$ЗП_{дн-т} = МО/24,83 \quad (1.3.4.1)$$

Расчеты затрат на полную заработную плату приведены в таблице 1.3.4.1. Также в расчет используется районный коэффициент $K_p = 1,3$.

Таблица 1.3.4.1 – Затраты на заработную плату

Исполнитель	Оклад, руб./мес.	Среднедневная ставка, руб./раб.день	Затраты времени, раб.дни	Коэффи циент	Фонд з/платы, руб.
НР	33664	1762,51	38	1,3	66975,38
И	15000	785,34	108	1,3	84816,72
Итого:					151792,1

Затраты на единый социальный налог (ЕСН) включают в себя отчисления в пенсионный фонд, на социальное и медицинское страхование и составляют 30,2% от полной заработной платы по проекту.

То есть ЕСН ($C_{\text{соц.}}$) определяется следующим образом:

$$C_{\text{соц.}} = C_{\text{зп}} * 0,302 = 151792,1 * 0,302 = 45841,21 \text{ руб.}$$

6.2.5. Расчет затрат на электроэнергию

Данный вид расходов включает в себя затраты на электроэнергию при работе оборудования, а именно компьютера и принтера. Затраты на электроэнергию при работе оборудования $C_{\text{эл.об.}}$ рассчитываются по формуле:

$$C_{\text{эл.об.}} = P_{\text{об}} \cdot \text{Ц}_{\text{э}} \cdot t_{\text{об}}, \quad (1.3.5.1)$$

где $P_{\text{об}}$ – мощность, потребляемая оборудованием, кВт;

$\text{Ц}_{\text{э}}$ – тарифная цена за 1 кВт·час;

$t_{\text{об}}$ – время работы оборудования, час.

Мощность $P_{\text{об}}$, потребляемая оборудованием, определяется по формуле:

$$P_{\text{об}} = P_{\text{ном.}} \cdot K_{\text{с}}, \quad (1.3.5.2)$$

где $P_{\text{ном.}}$ – номинальная мощность оборудования, кВт;

$K_{\text{с}}$ – коэффициент загрузки (для технологического оборудования малой мощности $K_{\text{с}} = 1$).

Номинальная мощность персонального компьютера составляет 0,3 кВт, принтера – 0,1 кВт.

Для ТПУ с учетом налога на добавленную стоимость (НДС)

$$Ц_3 = 5,257 \text{ руб./кВт}\cdot\text{час}$$

Время работы оборудования $t_{\text{ОБ}}$ для исполнителя вычисляется на основе данных таблицы трудозатрат Приложения В:

$$t_{\text{ОБ}} = T_{\text{РД}} \cdot K_t, \quad (1.3.5.3)$$

где $K_t \leq 1$ – коэффициент использования оборудования по времени, равный отношению времени его работы в процессе выполнения проекта к $T_{\text{РД}}$.

Из расчета, что продолжительность рабочего дня равна 8 часов, а работа выполнялась 108 рабочих дней, получим, что общее время выполнения проекта составляет 864 часа.

Так как работа на компьютере проводилась по 7 часов в день из 8, то $K_t = 0,88$. Тогда из 864 часов, потраченных исполнителем на осуществление проекта, 760 часов были проведены за компьютером. Принтер использовался примерно в течении 15 часов.

Затраты на электроэнергию при работе оборудования сведены в табл. 1.3.5.1.

Таблица 1.3.5.1 – Затраты на электроэнергию для технологических целей

Наименование оборудования	Время работы оборудования $t_{\text{ОБ}}$, час	Потребляемая мощность $P_{\text{ОБ}}$, кВт	Затраты $\text{Э}_{\text{ОБ}}$, руб.
Персональный компьютер	760	0,3	1198,60
Лазерный Принтер	15	0,1	7,89
Итого:			1206,48

Таким образом расходы на электроэнергию не превышают полторы тысячи рублей и составляют незначительную часть.

1.1.1. Расчет амортизационных расходов

В статье «Амортизационные отчисления» рассчитывается амортизация используемого оборудования за время выполнения работы.

Амортизационные отчисления рассчитываются по времени использования компьютера по формуле:

$$C_{AM} = \frac{N_A \cdot C_{OB}}{F_D} \cdot t_{рф} \cdot n, \quad (1.3.6.1)$$

где N_A – годовая норма амортизации;

C_{OB} – цена оборудования;

F_D – действительный годовой фонд рабочего времени;

$t_{рф}$ – время работы вычислительной техники;

n – число задействованных единиц оборудования, $n = 1$.

Расчет затрат на амортизационные отчисления представлен в таблице 1.3.6.1.

Таблица 1.3.6.1–Затраты на амортизационные отчисления

Наименование оборудования	Норма амортизации оборудования N_A	Стоимость оборудования C_{OB} , руб.	Фактич. время работы оборудования $t_{рф}$, ч	Действ. годовой фонд раб.времени F_D , ч	Амортизац. отчисления C_{AM} , руб.
Персональный компьютер	0,33	45000	760	2384	4734,06
Лазерный принтер	0,5	15000	15	500	225
Итого:					4959,06

Годовая амортизация N_A определяется как величина, обратная сроку амортизации оборудования C_A , который определяется согласно постановлению правительства РФ «О классификации основных средств, включенных в

амортизационные группы». Для компьютера примем $C_A = 3$ года, тогда $H_A = 0,33$. Для принтера примем $C_A = 2$ года, тогда $H_A = 0,5$.

6.2.6. Расчет накладных расходов

В данном разделе производится оценка расходов на выполнение проекта, которые не были учтены в предыдущих статьях, оплата услуг связи, копирование материалов и др. Величина прочих расходов составляет 16% от суммы всех предыдущих затрат и вычисляется по следующей формуле:

$$C_{\text{проч}} = 0,16 \cdot (C_{\text{мат}} + C_{\text{зп}} + C_{\text{соц}} + C_{\text{эл.об.}} + C_{\text{ам}}). \quad (1.3.7.1)$$

$$C_{\text{проч}} = 0,16 \cdot (1960 + 151792,1 + 45841,21 + 1206,48 + 4959,06) = 32921,42 \text{руб.}$$

Таким образом, накладные расходы составили 32921,416 руб.

6.2.7. Расчет общей себестоимости разработки

Общая стоимость разработки по созданию математического и программного обеспечения сегментации и визуализации легкого человека определяется путем суммирования затрат по всем статьям и представлена в табл. 1.3.8.1.

Таблица 1.3.7.1– Смета затрат на разработку проекта

Статья затрат	Условное обозначение	Сумма, руб.
Материалы и покупные изделия	$C_{\text{мат}}$	1960
Основная заработная плата	$C_{\text{зп}}$	151792,1
Отчисления в социальные фонды	$C_{\text{соц}}$	45841,21
Расходы на электроэнергию	$C_{\text{эл.об.}}$	1206,48
Амортизационные отчисления	$\Delta_{\text{ам}}$	4959,06

Прочие расходы	$C_{\text{проч}}$	32921,42
Итого:		238680,27

Наибольший расход был в заработной плате и составил 151792,1.
Наименьший — расход на электроэнергию — 1206,48.

6.2.8. Расчет прибыли и налога на добавленную стоимость. Цена разработки НИР

Прибыль от реализации данного проекта предполагается равной 15% от расходов на разработку и составляет 35802 руб.

НДС составляет 18% от суммы затрат на разработку и прибыли:

$$(238680,27 + 1206,48) \cdot 0,18 = 43179,62 \text{ руб.}$$

Данный параметр представляет сумму полной себестоимости, прибыли и НДС. Таким образом, получаем:

$$C_{\text{НИР}} = 238680,27 + 35802 + 43179,62 = 317661,89 \text{ руб.}$$

6.3. Оценка экономической эффективности проекта

6.3.1. Оценка научно-технического уровня НИР

Научно-технический уровень характеризует, в какой мере выполнены работы и обеспечивается научно-технический прогресс в данной области.

Сущность метода заключается в том, что на основе оценок признаков работы определяется коэффициент ее научно-технического уровня по формуле:

$$I_{\text{НТУ}} = \sum_{i=1}^3 R_i \cdot n_i, \quad (1.4.1.1)$$

где $I_{\text{НТУ}}$ – интегральный индекс научно-технического уровня;

R_i – весовой коэффициент i -го признака научно-технического эффекта;

n_i – количественная оценка i -го признака научно-технического эффекта, в баллах.

Результаты оценок признаков научно-технического уровня для данного проекта приведены в таблице 1.4.1.1.

Таблица 1.4.1.1 — Весовые коэффициенты признаков НТУ

Признак научно-технического эффекта НИР	Характеристика признака НИОКР	Балл	R_i
Уровень новизны	По-новому объясняются те же факты, закономерности, новые понятия дополняют ранее полученные результаты	6	0,3
Теоретический уровень	Разработка способа (алгоритм, программа мероприятий, устройство, вещество и т.п.)	6	0,2
Возможность реализации	Время реализации в течение первых лет	10	0,5

Исходя из оценки признаков НИОКР, показатель научно-технического уровня для данного проекта составил:

$$I_{\text{НТУ}} = 0,3 \cdot 6 + 0,2 \cdot 6 + 0,5 \cdot 10 = 1,8 + 1,2 + 5 = 8$$

Таким образом, исходя из полученной оценки признаков НИОКР, проект имеет высокий уровень научно-технического эффекта благодаря новому подходу к анализу медицинских изображений и визуализации их результатов.

Оценка научно-технического уровня представлена в таблице 1.4.1.1.

Таблица 1.4.1.1 — Оценка научно технического уровня НИР

Значимость	Фактор НТУ	Уровень фактора	Выбранный балл	Обоснование выбранного балла
0,3	Уровень новизны	Новая	6	По-новому объясняются те же факты, закономерности, новые понятия дополняют ранее полученные результаты
0,2	Теоретический уровень	Разработка способа	6	Разработка нового способа сегментации и визуализации легких
0,5	Возможность реализации	В течение первых лет	10	Реализуется на основе хорошо известных технологий

Таким образом, проект имеет высокий уровень научно-технического эффекта (т.к. $I_{НТУ} = 8$) и достаточно быструю возможность реализации в течение первых лет.

Разработка по созданию программного обеспечения сегментации и визуализации легкого человека с точки зрения ресурсоэффективности является достаточно трудоемкой и затратной, но имеет высокий уровень научно-технического эффекта. Этот факт, а также актуальность решаемой проблемы повышают конкурентоспособность проекта в заданной предметной области. Кроме того, данная разработка может найти широкое применение, так как имеет невысокие требования к ресурсам ЭВМ. По сравнению с существующими

коммерческими аналогами по обработке медицинских изображений, использование данной разработки не является столь затратным, что позволит медицинским учреждениям сэкономить по данной статье расходов. Также данная разработка повлияет на сферу медицинского обслуживания и будет иметь экономический социальный эффект, так как, в результате улучшения методов диагностики и повышения ее качества, позволит сократить затраты в этой области медицины, а также снизить уровень заболеваемости и смертности от заболеваний легких среди населения.

6.4. Итоговое заключение по разделу

Диссертационная работа содержит в себе научное исследование, выходным продуктом которого является создание программного обеспечения, реализующего автоматизацию математических расчетов. Данный раздел содержит анализ конкурентоспособности и эффективности полученного ПО на современном рынке IT-технологий.

Были обозначены возможные конкуренты в сфере разработки продукта, их направленность, а также выделена целевая аудитория. На основе полученной информации можно заключить, что основные потребители — малые предприятия в области автоматизации, где конкуренция наименьшая.

SWOT-анализ позволил взглянуть на картину в целом и спрогнозировать возможные последствия срабатывания рисков, но также обозначил преимущества, такие как инновационная структура ТПУ.

Возможные проблемы, такие как нечеткие требования, отсутствие должного тестирования были рассмотрены в диаграмме Исикавы. Самое важное звено в создании продукта оказался период непосредственной разработки.

Был составлен план работ, учитывающий специфику занятости научного руководителя. Были проведены расчеты по загруженности научного руководителя и исполнителя, представленный в табличном и графическом виде.

Выделены все расходы и проведены соответствующие расчеты. Наибольший расход был в заработной плате и составил 151792,1. Наименьший — расход на электроэнергию — 1206,48. Прибыль составляет 35802 руб. Сумма полной себестоимости, прибыли и НДС — 317661,89 руб.

Так как работа содержит значительную наукоемкую составляющую, был оценен научно-технический уровень НИР.

ГЛАВА 6. Социальная ответственность

6.1. Введение

Определим условия труда для группы разработчиков, которая будет состоять из трех человек: руководителя и двух инженеров-программистов.

В данной выпускной квалификационной работе разрабатывается лабораторный комплекс на базе промышленного микропроцессорного контроллера «КРОСС».

При выполнении работы используется следующее оборудование:

- три диспетчерских станции в виде ПЭВМ;
- стенд «Гидравлический объект»;
- программируемый контроллер «КРОСС».

– Каждая из диспетчерских станций представляет из себя IBM-совместимый персональный компьютер на базе системного блока, который включает в себя процессор Intel Pentium 4 2ГГц, оперативную память емкостью 1 Гб, накопитель на жестких дисках емкостью 250 Гб и другие устройства. Производительности такого системного блока достаточно для работы всех приложений, необходимых для реализации проекта. В качестве мониторов будут использованы ЭЛТ мониторы фирмы SAMSUNG SyncMaster 765 DFX.

– Работа состоит в создании программного обеспечения диспетчерской станции и прикладного программного обеспечения промышленного микропроцессорного контроллера КРОСС, разработке принципиальной и функциональной схемы комплекса. Создание программы визуализации диспетчерской станции, разработке методического указания по выполнению лабораторных работ для учебного процесса.

6.2. Производственная безопасность

Для данного проекта опасными и вредными производственными факторами являются следующие факторы, согласно [13]:

Источник фактора, наименование видов работ	Факторы (по ГОСТ 12.0.003-74)		Нормативные документы
	Вредные	Опасные	
Работа с компьютером и орг. техникой	<ol style="list-style-type: none">1. Недостаточная освещенность рабочей зоны;2. Повышенный уровень электромагнитных излучений;3. Повышенная напряжённость электрического поля;4. Повышенная или пониженная влажность воздуха;5. Повышенный уровень шума.6. Психо–физиологические факторы	<ol style="list-style-type: none">1. Электрический ток (источником является ПК).2. Короткое замыкание.3. Статическое электричество.	<ol style="list-style-type: none">1. СанПиН 2.2.2/2.4.1340 – 03;2. СанПиН 2.2.4.548 – 96; СП 52.13330.2011;3. СН 2.2.4/2.1.8.562 – 96;4. ГОСТ 12.1.003-83; ГОСТ 12.1.019-85.

6.1.1. Параметры микроклимата в помещениях

Воздух рабочей зоны (микроклимат) производственных помещений определяют следующие параметры: температура, относительная влажность, скорость движения воздуха. Эти параметры по отдельности и в комплексе влияют на организм человека, определяя его самочувствие. По степени физической тяжести работа инженера-программиста относится к категории лёгких работ [10]. Оптимальные и допустимые значения характеристик микроклимата устанавливаются в соответствии с [9] и приведены в таблице 16.

Таблица 16 – Оптимальные и допустимые значения характеристик микроклимата по СанПиН 2.2.4-548-96

Период года	Температура, С	Относительная	Скорость движения
Холодный и переходный	24	40-60	0,1
Тёплый	24	40	0,1

В помещении ВЦ должны подаваться следующие объёмы наружного воздуха: при объёме помещения до 20 м³ на человека – не менее 30 м³ в час на человека; при объёме помещения более 40 м³ на человека и отсутствии выделения вредных веществ допускается естественная вентиляция.

В аудитории отсутствует принудительная вентиляция. Имеется лишь естественная, т.е. воздух поступает и удаляется через щели, окна, двери. Основной недостаток такой вентиляции в том, что приточный воздух поступает в помещение без предварительной очистки и нагревания. Естественная вентиляция допускается при условии, что на одного работающего приходится более 40 м³ объема воздуха в помещении, согласно [12].

В помещениях с повышенными требованиями к чистоте воздуха должно использоваться водяное отопление. В рассматриваемом помещении используется водяное отопление со встроенными нагревательными элементами и стояками.

Итого помещение соответствует рассмотренным нормам и пригодно для работы.

6.1.2. Недостаток освещения рабочей зоны

Естественное освещение положительно влияет не только на зрение, но также тонизирует организм человека в целом и оказывает благоприятное психологическое воздействие. В связи с этим по санитарным нормам и правилам [12] все помещения должны иметь естественное освещение. В рассматриваемом помещении реализовано одностороннее боковое естественное освещение через два световых проема. С помощью естественного бокового освещения помещение освещается крайне неравномерно и только в светлое время суток [12].

Для производственных помещений допускается нормированные значения КЕО принимать в соответствии с таблицей 17.

В помещениях с боковым освещением, глубина которых по условиям технологии или выбору рациональных объемно-планировочных решений, не позволяет обеспечить нормированное значение КЕО, указанное в таблице 17 – для совмещенного освещения. Разряд зрительных работ в данном помещении равен III согласно СанПиН 2.1.1.1278-03.

Таблица 17 - Разряд зрительных работ СанПиН 2.1.1.1278-03

Разряд зрительных работ	Наименьшее нормированное значение КЕО, ен , % при совмещенном освещении	
	при верхнем или комбинированном освещении	при боковом освещении
I	3	1,2
II	2,5	1
III	2	0,7
IV	1,5	0,5
V и VII	1	0,3
VI	0,7	0,2

Нормами, [15] для данных работ установлена необходимая освещенность рабочего места $E_n=300$ лк.

Согласно [12]:

– Минимальная освещенность при использовании ПЭВМ составляет $E_T=300\div 500$ лк.

– Освещенность рабочего стола должна быть не менее $300\div 500$ лк, что может достигаться установкой местного освещения.

– Местное освещение не должно создавать бликов на экране. Следует ограничивать отраженную блескость на рабочих поверхностях (экран, стол, клавиатура) за счет правильного выбора и расположения светильников, яркость бликов на экране не должна превышать 40 кд/м².

– Светильники местного освещения должны иметь не просвечивающий отражатель.

Приведем проверочный расчет искусственного освещения в помещении.

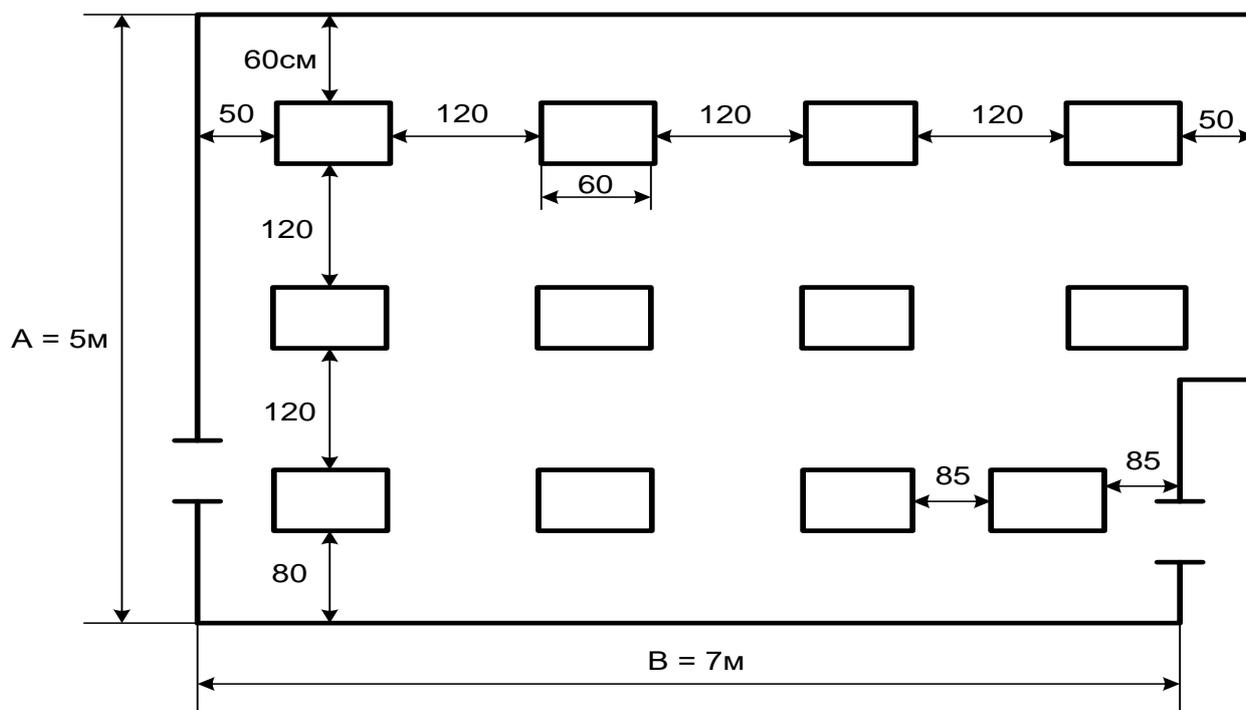


Рисунок 29 - Схема размещения источников света

Определим нормированное значение минимальной искусственной освещенности в помещении. Известно, что минимальная освещенность при использовании ПЭВМ, согласно [15], составляет $E_T=300$ лк.

Скорректируем величину освещенности E_T , т.к. с течением времени освещенность за счет уменьшения светового потока ламп и загрязнения светильника уменьшается.

Таким образом, с учетом минимальной освещенности в аудитории 113а должно быть:

$$E_T=300*1.5=450 \text{ лк.}$$

Минимальная освещенность при использовании ПЭВМ составляет $E_T=300\div 500$ лк, следовательно, фактическая освещенностью аудитории соответствует нормам [15].

6.1.3. Повышенное электромагнитное излучение

Большая часть ЭМИ происходит не от экрана монитора, а от видеокабеля и системного блока. В портативных компьютерах практически все электромагнитное излучение идет от системного блока, располагающегося под клавиатурой. Современные машины выпускаются заводом-изготовителем со

специальной металлической защитой внутри системного блока для уменьшения фона электромагнитного излучения [10].

Согласно [9] напряженность электромагнитного поля на расстоянии 50 см вокруг ВДТ по электрической составляющей должна быть не более:

В диапазоне частот 5 Гц ÷ 2 кГц – 25 В/м;

В диапазоне частот 2 кГц ÷ 400кГц – 2,5 В/м.

Плотность магнитного потока должна быть не более:

В диапазоне частот 5 Гц ÷ 2 кГц – 250 нТл;

В диапазоне частот 2 кГц ÷ 400кГц – 25 нТл.

Возможные способы защиты от ЭМП:

Основной способ – увеличение расстояния от источника, во избежание последствий экран видеомонитора должен находиться на расстоянии не менее 50 см от пользователя [10], что выполняется на используемом рабочем месте.

Применение приэкранных фильтров, специальных экранов и других средств индивидуальной защиты, прошедших испытание в аккредитованных лабораториях и имеющих соответствующий гигиенический сертификат, что также выполняется.

6.1.4. Ионизирующее излучение от дисплея на ЭЛТ компьютера

ЭЛТ мониторы SyncMaster 765 DFX, которые будут использованы в работе, являются источником ионизирующего излучения. Под влиянием ионизирующего излучения в организме может происходить торможение функций кроветворных органов, нарушение нормальной свертываемости крови и увеличение хрупкости кровеносных сосудов, снижение сопротивляемости организма инфекционным заболеваниям и др. [10]

Доза облучения при расстоянии до дисплея 20 см составляет 50 мкбэр/час. По нормам, согласно [9] конструкция ВДТ и ПЭВМ должна обеспечивать мощность экспозиционной дозы рентгеновского излучения в любой точке на расстоянии 0,05 м от экрана и корпуса ВДТ не более 1 мк зв/ч. Способы защиты: применение приэкранных фильтров, специальных экранов и других средств

индивидуальной защиты, прошедших испытание в аккредитованных лабораториях и имеющих соответствующий гигиенический сертификат.

При выполнении дипломной работы использовался монитор с низким уровнем излучения (стандарт MPR II) – соответствующий требованиям степени защиты TCO99.

6.1.5. Повышенный уровень шума

Шум ухудшает условия труда, оказывает вредное воздействие на организм человека. Действие шума различно: он затрудняет разборчивость речи, вызывает снижение работоспособности, повышает утомляемость, вызывает необратимые изменения в органах слуха человека. Ослабляется внимание, ухудшается память, снижается реакция, увеличивается число ошибок при работе [10].

Производственные помещения, в которых для работы используются ПЭВМ, не должны граничить с помещениями, в которых уровень шума и вибрации превышают нормируемые значения. Допустимые уровни звукового давления и уровней звука, создаваемого ПЭВМ, согласно [9] не должны превышать значений, представленных в таблице 18.

Таблица 18 – Допустимые значения уровней звукового давления в октавных полосах частот и уровня звука, создаваемого ПЭВМ по СанПиН 2.2.2/2.4.1340-03

Уровни звукового давления в октавных полосах со среднегеометрическими частотами									Уровни звука в дБА
31,5 Гц	63 Гц	125 Гц	250 Гц	500 Гц	1000 Гц	2000 Гц	4000 Гц	8000 Гц	
86 дБ	71 дБ	61 дБ	54 дБ	49 дБ	45 дБ	42 дБ	40 дБ	38 дБ	50

Источником шума в лаборатории будут вентиляторы компьютеров. Исходя из вышесказанного и опираясь на субъективные ощущения можно сделать заключение, что шумовая обстановка на рабочем месте соответствует норме [9].

6.1.6. Опасность поражения электрическим током

В зависимости от условий в помещении опасность поражения человека электрическим током запрещается работать с компьютером, стендом «Гидравлический объект», с контроллером КРОСС в условиях повышенной влажности (относительная влажность воздуха длительно превышает 75%), высокой температуры (более 35°C), согласно [12], наличии токопроводящей пыли, токопроводящих полов и возможности одновременного соприкосновения к имеющим соединение с землёй металлическим элементам и металлическим корпусом электрооборудования. Таким образом, работа с компьютером, стендом и контроллером КРОСС может проводиться только в помещениях без повышенной опасности, и возможность поражения током может быть только при прикосновении непосредственно с элементами этих конструкций.

Инженер–программист работает с электроприборами: компьютером (дисплей, системный блок, манипулятор «мышь» и клавиатура), стендом «Гидравлический объект», контроллером КРОСС. В данном случае существует опасность поражения электрическим током при прикосновении к нетоковедущим частям, оказавшимся под напряжением (в случае нарушения изоляции токоведущих частей ПЭВМ, стенда «Гидравлический объект», контроллера КРОСС).

Наше помещение по опасности поражения электрическим током будет относиться к помещениям без повышенной опасности, то есть отсутствуют условия, создающие повышенную опасность [12].

В лаборатории используются приборы, потребляющие напряжение 220В переменного тока с частотой 50 Гц. Это напряжение опасно для жизни, поэтому необходимо применять надлежащие меры предосторожности.

Для защиты от поражения электрическим током все токоведущие части должны быть защищены от случайных соприкосновений с кожухами, корпус устройства должен быть заземлен. Согласно [14], заземление выполняется изолированным медным проводом сечением 1,5 мм², который присоединяется к общей шине заземления с общим сечением 54 мм² при помощи сварки. Общая

шина присоединяется к заземлению, сопротивление которой не должно превышать 4 Ом. Питание устройств должно осуществляться от силового щита через автоматический выключатель, который срабатывает при коротком замыкании нагрузки. Данное условия выполнено.

При работе с компьютером прикосновения к его элементам могут возникнуть токи статического электричества, которые в свою очередь имеют свойство притягивать пыль и мелкие частицы к экрану. Пыль на экране ухудшает видимость, а при подвижности воздуха может попасть на кожу лица и в легкие, что вызывает заболевание кожи и дыхательных путей.

Типичные неисправности как скрытой, так и открытой электропроводки сводятся в основном к короткому замыканию междуфазовым и нулевым проводами, замыканию фазового провода на «землю», плохим контактам в соединениях и обрыву проводов. Обычно короткое замыкание происходит в момент подключения какого-либо электрического прибора к сети. Наиболее распространенные причины, по которым может произойти короткое замыкание — это перетирание изоляции в местах, где провода перегибаются. В имеющемся помещении подобные факторы отсутствуют, используются автоматические предохранители многократного действия. Для установки на щитке предохранитель снабжен цоколем и резьбой; включается и выключается он нажатием кнопок.

6.1.7. Психофизиологические факторы опасности

На трудовую деятельность человека и его организм влияют объем восприятия и переработки информации, физическое, нервно-психологическое, умственное, эмоциональное перенапряжение, ритм и темп работы, монотонность труда. их оценки позволяет определить степень и характер нагрузки во время работы, соответствие рабочего места и средств труда анатомо-физиологическим особенностям человека, установить рациональные режимы труда и отдыха, обустроить рабочие места, наладить профессиональный отбор и профориентацию и др.

Работа за ЭВМ приводит к гиподинамии. Физическая активность в течение получаса в день может существенно снизить риск развития многих распространенных неинфекционных заболеваний: гипертонии, сахарного диабета и др., поскольку они часто обусловлены именно гиподинамией.

Монотонность труда также является сопутствующим фактором, так как возникает необходимость длительной однообразной работы за ЭВМ. Предотвратить утомлению и повысить работоспособность позволяет физиологическая рационализация, в частности оптимальная организация режима труда и отдыха, рациональное построение трудового процесса, эффективное обучение с целью быстрого овладения трудовыми навыками. Критериями эффективности режима деятельности считают повышение производительности труда, наличие устойчивого динамического рабочего стереотипа, приемлемых производственных и физиологических показателей.

Эмоциональный фактор был исключен из работы.

Таким образом психо–физиологические факторы были исключены самим исполнителем благодаря приведенным выше методикам, а имеющееся рабочее место не обладало внешним влиянием.

6.3. Экологическая безопасность

При выполнении данной работы отсутствуют воздействия на атмосферу и гидросферу. Присутствует воздействие на литосферу в виде отходов, возникающих при замене устаревшего или неисправного оборудования, образование отходов при печати документов, люминесцентные лампы.

Если используемое оборудование по каким-либо причинам выйдет из строя, и не будет поддаваться ремонту, то его необходимо будет утилизировать. Пластмассовые, железные детали, можно пустить на переработку. Переработке так же поддается и использованная и ненужная бумага, ее можно сдать в прием макулатуры.

Для сохранения окружающей среды организациям необходимо обращаться в утилизирующие компании.

Процедура утилизации для организации заключается в выполнении следующих этапов:

1. Выявление неисправного оборудования, которое подлежит утилизации.
2. Списание оборудования.
3. Формирование списка оборудования, передаваемого на утилизацию.
4. Подписание договора с утилизирующей компанией.

При переработке утилизирующая компания производит разбор техники. Определяет содержащиеся в ней компоненты и материалы, которые подлежат переработке в сырье для изготовления новой техники.

Утилизация люминесцентных ламп представляет собой определенную проблему. Опасность этой проблемы вызвана тем, что лампы дневного света всех типов и моделей содержат в малом количестве ртуть. Сам принцип работы устройств не позволяет отказаться от данного металла. Любой источник света подобного типа работает следующим образом: при протекании тока через пары ртути начинают излучаться лучи ультрафиолетового спектра, которые, проходя через слой люминофора, преобразуются в видимый спектр лучей, то есть ртуть

содержится в любых видах изделий, различие может быть только в ее количестве.

Приблизительное содержание Hg в разных типах источников света:

- обычная трубчатая лампа – примерно 45–50 мг ртути;
- люминесцентная лампа компактного типа – 5–6 мг ртути;
- продукция высокого давления – от 75 до 600 мг ртути.

Метод утилизации люминесцентных источников света начинается с их сбора и последующей транспортировки на предприятия, которые занимаются демеркуризацией. На практике использованные лампы всех типов должны сдаваться в пункты приема подобных отходов. Производственные организации и коммерческие структуры обязаны заключать договоры с переработчиками дневных источников освещения, к которым относится и ТПУ, обслуживающий лабораторию (рабочее место).

Были рассмотрены воздействия на литосферу в виде выбросов. Все обнаруженные отходы имеют определенный алгоритм утилизации, который реализуется для данного рабочего места, что нивелирует вредное воздействие.

6.4. Безопасность в чрезвычайных ситуациях

6.4.1. Пожарная профилактика

Организационные мероприятия предусматривают правильную эксплуатацию оборудования, правильное содержание зданий и территорий, противопожарный инструктаж рабочих и служащих, обучение производственного персонала правилам противопожарной безопасности, издание инструкций, плакатов, наличие плана эвакуации.

К техническим мероприятиям относятся: соблюдение противопожарных правил, норм при проектировании зданий, при устройстве электропроводов и оборудования, отопления, вентиляции, освещения, правильное размещение оборудования.

К режимным относятся установление правил организации работ и соблюдение противопожарных мер.

6.4.2. Оценка пожарной безопасности помещения

Согласно СанПиН 2.2.2.542-96 [8], в зависимости от характеристики используемых в производстве веществ и их количества, по пожарной и взрывной опасности помещения подразделяются на категории А, Б, В, Г, Д.

В лабораторной аудитории, которую мы проектируем, будет множество деревянных изделий (столы, шкафы), электропроводов напряжением 220В, а также применение электронагревательных приборов с открытыми нагревательными элементами – паяльниками дает право отнести помещение по степени пожаро- и взрывобезопасности к категории В. Для категории В характерно наличие в помещении горючих веществ и материалов в холодном состоянии.

При неправильной эксплуатации оборудования и коротком замыкании электрической сети может произойти возгорание, которое грозит уничтожением ЭВМ, документов и другого имеющегося оборудования. Система вентиляции может стать источником распространения возгорания.

Так как помещение оператора по степени пожаро-взрывобезопасности относится к категории В (согласно СанПиН 2.2.2.542-96 [8]), т.е. к помещениям, где характерно наличие негорючих веществ и материалов в холодном состоянии, необходимо предусмотреть ряд профилактических мероприятий технического, эксплуатационного, организационного плана.

В качестве возможных причин пожара можно указать следующие:

- короткое замыкание;
- перегрузка сетей, которая ведет за собой сильный нагрев токоведущих частей и возгорание изоляции;
- возможность возгорания при пуске оборудования после ремонта.

Для предупреждения пожаров от коротких замыканий и перегрузок необходимы правильный выбор, монтаж и соблюдение установленного режима эксплуатации электрических сетей, дисплеев и других электрических средств автоматизации.

Причиной возгорания может быть:

- неисправность токоведущих частей установок;
- работа с открытой электроаппаратурой;
- короткие замыкания в блоке питания;
- несоблюдение правил пожарной безопасности;
- наличие горючих компонентов: документы, двери, столы, изоляция кабелей и т.п.

6.4.3. Мероприятия по устранению и предупреждению пожаров

Для предупреждения возникновения пожара необходимо соблюдать следующие правила пожарной безопасности:

- исключение образования горючей среды (герметизация оборудования, контроль воздушной среды, рабочая и аварийная вентиляция);
- применение при строительстве и отделке зданий негорючих или трудно сгораемых материалов.

Необходимо в аудитории проводить следующие пожарно-профилактические мероприятия:

- организационные мероприятия, касающиеся технического процесса с учетом пожарной безопасности объекта;
- эксплуатационные мероприятия, рассматривающие эксплуатацию имеющегося оборудования;
- технические и конструктивные, связанные с правильным размещением и монтажом электрооборудования и отопительных приборов.

Организационные мероприятия:

- противопожарный инструктаж обслуживающего персонала;
- обучение персонала правилам техники безопасности;
- издание инструкций, плакатов, планов эвакуации.
- эксплуатационные мероприятия:
 - соблюдение эксплуатационных норм оборудования;
 - обеспечение свободного подхода к оборудованию;
 - содержание в исправности изоляции токоведущих проводников.

Технические мероприятия:

– соблюдение противопожарных мероприятий при устройстве электропроводок, оборудования, систем отопления, вентиляции и освещения. В помещении ВЦ установлен рубильник, обесточивающий весь кабинет, на двери кабинета приведен план эвакуации (см. рис. 30) в случае пожара, и на достигаемом расстоянии находится пожарный щит. Если возгорание произошло в электроустановке, для его устранения должны использоваться углекислотные огнетушители типа ОУ–2, которые есть в наличии.

- профилактический осмотр, ремонт и испытание оборудования.

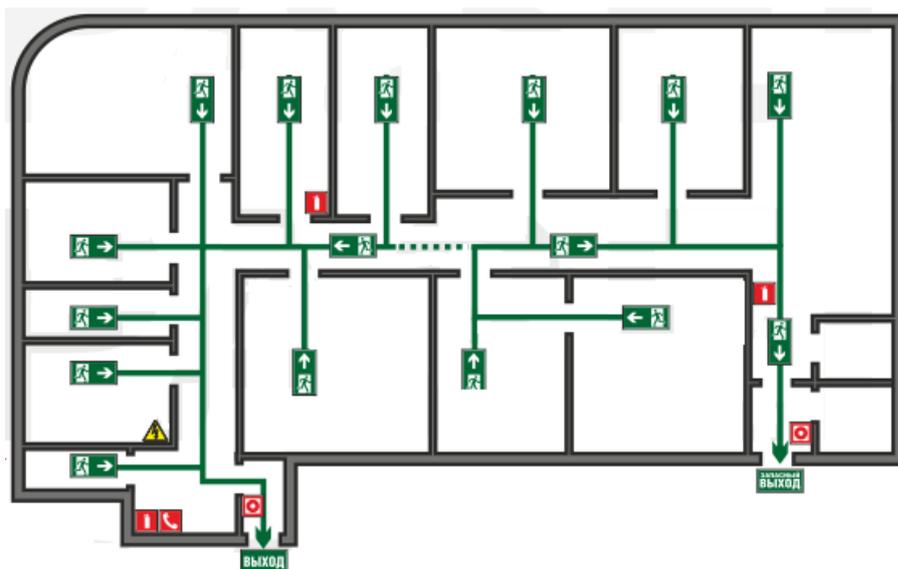


Рисунок 30 — План эвакуации

Кроме устранения самого очага пожара, нужно своевременно организовать эвакуацию людей [9].

6.5. Заключение

Данный раздел преследовал цель определить условия труда для разработчиков. Были рассмотрены производственная безопасность, а именно проанализированы вредные и опасные факторы, экологическая безопасность, безопасность в чрезвычайных ситуациях, как пожар, и правовые и организационные вопросы обеспечения безопасности.

Согласно проанализированным пунктам рабочее место — лаборатория 113а 10 корпуса ТПУ — соответствует стандартам и является пригодной для проведения деятельности. Рассмотренный микроклимат соответствует норме — более 40 м³ объема воздуха в помещении на одного работающего. Освещенность помещения равна 450 лк, что входит в допустимый диапазон. Также есть в наличие необходимые средства защиты от электромагнитного излучения, а оборудование обладает необходимыми требованиями защиты от ионизирующего излучения. Уровень шума по субъективным ощущениям соответствует норме и не мешает рабочему процессу. В помещении

присутствуют меры защиты от поражения электрическим током. Психологические вредные факторы были исключены.

Было рассмотрено влияние на экологию, а именно выброс отходов в литосферу. Согласно проведенного анализа, в рабочем помещении приняты меры по уменьшению вредного влияния.

В качестве чрезвычайной ситуации было выделено и проанализировано возгорание. В помещении ВЦ установлен рубильник, обесточивающий весь кабинет, на двери кабинета приведен план эвакуации в случае пожара, и на достигаемом расстоянии находится пожарный щит, углекислотные огнетушители типа ОУ-2 есть в наличии. То есть, помещение подготовлено в возможной чрезвычайной ситуации.

Заключение

Данная работа нацелена на исследование неопределенных систем с аффинной неопределенностью и синтез ПИ–регуляторов для таковых.

Чем сложнее математическая модель, тем более точно она отражает реальный объект. Однако использование таковых моделей сопряжено со значительными трудозатратами. Требуются сложные алгоритмы вычислений. Данная работа содержит в себе исследования, направленные на облегчение проведения работ со сложными математическими моделями.

Был проведен обширный анализ текущих подходов, указаны их недостатки. Особое место занимает алгоритм поиска минимального реберного маршрута, который и лег в основу подхода, представленного в данной работе.

В ходе исследований был придуман алгоритм работы с системами с аффинной неопределенностью, который показал себя как более быстрый.

Полученный алгоритм был реализован в виде программного модуля с использованием современных техник программирования.

ЛИТЕРАТУРА

1. Гайворонский, С.А. Анализ локализации корней интервального полинома в заданном секторе / С.А. Гайворонский, С.В. Замятин // Изв. Томского политех. ун-та. –2004. Т. 307. № 4. – С. 14-18.
2. Гайворонский, С.А. Анализ робастной относительной устойчивости на основе многопараметрического интервального корневого годографа / С.А. Гайворонский, С.В. Новокшенов // Тез.докл. X междунард. Научн.-технич. Конф. Состояние и перспективы развития электротехнологии. – Иваново: Издат. ИГЭУ, 2001.
3. Гайворонский, С.А. Параметрический синтез линейного регулятора электромеханической системы при интервальной неопределенности объекта управления / С.А. Гайворонский // Изв. ВУЗов. Электромеханика, 1990. №5. –С. 69-72.
4. Гусев, Ю.М. Анализ и синтез линейных интервальных динамических систем (состояние проблемы). Анализ с использованием интервальных характеристических полиномов / Ю.М. Гусев, В.Н. Ефанов, В.Г. Крымский // Техн. кибернетика. 1991. №1. – С. 3-30.
5. Замятин С.В. Размещение областей локализации доминирующих полюсов интервальной системы с обеспечением заданных показателей качества / С.В. Замятин // Изв. Томского политех. ун-та, №7, Том 309 2006. – С. 10-14.
6. Захаров, А.В. Синтез систем управления при интервальной неопределенности параметров их математических моделей / А.В. Захаров, Ю.И. Шокин // ДАН СССР. 1988. Т. 299, №2. – С. 292-295.
7. Киселев, О.Н. Синтез регуляторов низкого порядка по критерию H_∞ и по критерию максимальной робастности / О.Н. Киселев, Б.Т. Поляк // Автоматика и телемеханика, 1999. N 3, – С. 119-130.
8. Ключев А.С. Наладка систем автоматического регулирования котлоагрегатов. / А.С. Ключев, А.Г. Товарнов //– М.: Энергия, 1970. – 280 с.

9. Римский Г.В. Корневые методы исследования интервальных систем / Г.В. Римский. – Минск: Институт технической кибернетики НАН Беларуси, 1999. – 186 с.
10. Кузьменко Д.Я. Регулирование и автоматизация паровых котлов. / Д.Я. Кузьменко. Изд. 2-е, перераб. и доп. – М.: Энергия, 1978. – 160 с.
11. Неймарк Ю.И. Динамические системы и управляемые процессы / Ю.И. Неймарк – М.: Наука, 1978. – 336с.
12. Поляк, Б.Т. Робастная устойчивость и управление / Б.Т. Поляк, П.С. Щербаков – М.: Наука, 2002. – 303 с.
13. Поляк, Б.Т. Робастный критерий Найквиста / Б.Т. Поляк, Я.З. Цыпкин // АиТ. 1992. №7. – С.25-31.
14. Поляк, Б.Т. Частотные критерии робастной устойчивости и апериодичности линейных систем / Б.Т. Поляк, Я.З. Цыпкин // АиТ. 1990. №9. –С. 45-54.
15. Римский, Г. В. Корневые методы исследования интервальных систем / Г.В. Римский – Минск: Ин-т техн. кибернетики НАН Беларуси. 1999. – 186с.
16. Скворцов, Л.М. Интерполяционный метод решения задачи назначения доминирующих полюсов при синтезе одномерных регуляторов / Л.М. Скворцов // Изв. АН. ТиСУ. 1994. №4. – С. 10-13.
17. Скворцов, Л.М. Интерполяционный метод решения задачи назначения доминирующих полюсов при синтезе многомерных регуляторов / Л.М. Скворцов // Изв. АН.ТиСУ. 1997. № 1. – С. 31-34.
18. Скворцов, Л.М. Синтез закона управления по заданным полюсам и нулям передаточной функции / Л.М. Скворцов // Изв. АН СССР. Техн. кибернетика. 1987. № 6. – С. 149-153.
19. Скворцов, Л.М. Синтез линейных систем методом полиномиальных уравнений / Л.М. Скворцов // Изв. АН СССР. Техн. кибернетика. 1991. № 6. – С. 54-59.
20. Суходоев, М.С. Анализ и синтез робастных систем автоматического управления в среде Matlab // М.С. Суходоев, С.А. Гайворонский, С.В.

- Замятин // Известия Томского политехнического университета, 2008. - т.312 - № 5. – С. 61-65.
21. Суходоев, М.С. Интервально-параметрический синтез робастной системы с гарантированными корневыми показателями качества. / М.С. Суходоев, С.А. Гайворонский // Молодежь и современные информационные технологии: Сборник трудов V Всероссийской научно-практической конференции студентов, аспирантов и молодых ученых. Томск, 2007. – С. 447-449.
 22. Суходоев, М.С. Определение желаемой области расположения доминирующих полюсов замкнутой системы с учетом ее нулей / М.С. Суходоев, С.В. Ефимов, С.В. Замятин, С.А. Гайворонский // Известия Томского политехнического университета, 2008. - т.312 - № 5. – С. 57-61.
 23. Суходоев, М.С. Условия робастной секторной устойчивости интервального полинома. / М.С. Суходоев, С.А. Гайворонский // «Молодежь и современные информационные технологии» III Всероссийская научно-практическая конференция студентов. 2005. С. 216-217.
 24. Суходоев, М.С. Условия робастной устойчивости полинома с аффинной неопределенностью. / М.С. Суходоев, С.А. Гайворонский // Современные техника и технологии: Труды XI Международной научно-практической конференции студентов и молодых учёных. В 2 т. - Т. 2 - г. Томск, ТПУ, 28 марта - 1 апреля 2005 г.
 25. Харитонов В.Л. Об асимптотической устойчивости положения равновесия семейства систем линейных дифференциальных уравнений / В.Л. Харитонов // Дифференц. уравнения, 1978. №11. С. 2086-2088.
 26. Хлебалин, Н.А. Моделирование систем автоматического управления с интервальной неопределенностью параметров / Н.А. Хлебалин, Д.С. Пятых // Интервальная математика и распространение ограничений. 2004 – С. 258-266.

27. Хлебалин, Н.А. Построение интервальных полиномов с заданной областью расположения корней / Н.А. Хлебалин // Аналитические методы синтеза регуляторов. – Саратов: Изд. Саратовского политех. ин-та, 1982. – С. 92-98.

Приложение А

(обязательное)

Листинг приложения

Листинг формы:

```
using System;
using System.Collections.Generic;
using System.Windows.Forms;

namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {
        /// <summary>
        /// Делегат чтения данных пользователя.
        /// </summary>
        /// <param name="list"> Набор диапазонов. </param>
        /// <param name="delta"> Количество точек. </param>
        /// <param name="power"> Порядок уравнения. </param>
        /// <param name="updateCoeff"> Модификаторы коэффициентов. </param>
        public delegate void Read(IList<string> list, int delta, int power, IList<double>
updateCoeff);

        /// <summary>
        /// Делегат на рисование.
        /// </summary>
        public delegate void Draw();

        /// <summary>
        /// Событие прочтения данных.
        /// </summary>
        public event Read Reading;

        /// <summary>
        /// Событие рисования.
        /// </summary>
        public event Draw Drawing;

        /// <summary>
        /// Диапазоны.
        /// </summary>
        private string[] _userData;

        /// <summary>
        /// Количество точек.
        /// </summary>
        private int _delta;

        /// <summary>
        /// Порядок уравнения.
        /// </summary>
        private int _power;

        /// <summary>
        /// Инициализация формы.
        /// </summary>
        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

```

CalculateButton.Click += CalculateButton_Click;
DrawButton.Click += DrawButton_Click;
EnterCoeff.Click += EnterCoeff_Click;

Coeffs.KeyPress += TbCoefA_KeyPress;
}

/// <summary>
/// Нажатие кнопки "Ввести коэффициенты".
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private void EnterCoeff_Click(object sender, EventArgs e)
{
    _userData = Coeffs.Lines;
    _delta = (int)Delta.Value;
    _power = (int)Power.Value;
    SetGrid();
}

/// <summary>
/// Нажатие кнопки "Построение".
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private void DrawButton_Click(object sender, EventArgs e)
{
    Drawing?.Invoke();
}

/// <summary>
/// Нажатие кнопки "Анализ".
/// </summary>
/// <param name="sender"> Объект. </param>
/// <param name="e"> Событие. </param>
private void CalculateButton_Click(object sender, EventArgs e)
{
    if (_userData.Length % 2 != 0)
    {
        MessageBox.Show($"Указаны неполные интервалы", "Ошибка расчета");
        return;
    }

    var dataGrid = GetDataGrid();
    Reading?.Invoke(_userData, _delta, _power, dataGrid);
}

/// <summary>
/// Блокировка ввода букв
/// </summary>
/// <param name="sender"> Объект. </param>
/// <param name="e"> Событие. </param>
private void TbCoefA_KeyPress(object sender, KeyPressEventArgs e)
{
    if ((e.KeyChar <= 47 || e.KeyChar >= 58) && e.KeyChar != 8 && e.KeyChar != 10 &&
e.KeyChar != 13)
        e.Handled = true;
}

/// <summary>
/// Предоставление данных из грида.
/// </summary>
/// <returns></returns>
private IList<double> GetDataGrid()

```

```

    {
        var result = new List<double>();
        foreach (DataGridViewRow row in GridCoeff.Rows)
        {
            var coeff = new List<double>();
            foreach (DataGridViewCell cell in row.Cells)
            {
                var value = cell.Value.ToString();
                coeff.Add(double.Parse(value));
            }

            result.AddRange(coeff);
        }

        return result;
    }

    /// <summary>
    /// Настройка грида.
    /// </summary>
    private void SetGrid()
    {
        GridCoeff.RowCount = _power + 1; // Из-за нулевого порядка в уравнении.
        GridCoeff.ColumnCount = _userData.Length / 2;

        for (int i = 0; i < GridCoeff.RowCount; i++)
        {
            var row = GridCoeff.Rows[i];
            row.HeaderCell.Value = $"s{i}";
        }
        for (int i = 0; i < GridCoeff.ColumnCount; i++)
        {
            var column = GridCoeff.Columns[i];
            column.Width = 30;

            column.HeaderCell.Value = $"a{i}";
        }
    }
}

```

Листинг представителя:

```

using System.Collections.Generic;
using System.Diagnostics;
using System.Linq;
using WindowsFormsApplication1.Services;
using WindowsFormsApplication1.Tools;

namespace WindowsFormsApplication1
{
    /// <summary>
    /// Представитель представлений.
    /// </summary>
    public class Presenter
    {
        /// <summary>
        /// Сервис расчетов.
        /// </summary>
        private readonly Solver _solver;

        /// <summary>
        /// Сервис для работы с интервалами.

```

```

/// </summary>
private readonly IntervalService _intervals;

/// <summary>
/// Сервис для работы с массивами.
/// </summary>
private readonly ArrayService _arrayService;

/// <summary>
/// Главная форма.
/// </summary>
private readonly Form1 _form;

/// <summary>
/// Форма для отображения графиков.
/// </summary>
private readonly graphic _drawForm;

/// <summary>
/// Словарь интервалов.
/// </summary>
private IList<double> DictionaryInterval { get; set; }

/// <summary>
/// Порядок уравнения.
/// </summary>
private int Power { get; set; }

/// <summary>
/// Дискретность.
/// </summary>
private int Delta { get; set; }

/// <summary>
/// Корни.
/// </summary>
private IList<Point> roots;

/// <summary>
/// Конструктор с параметрами.
/// </summary>
/// <param name="form"> Главная форма. </param>
/// <param name="drawForm"> Форма с графиками. </param>
public Presenter(Form1 form, graphic drawForm)
{
    _form = form;
    _drawForm = drawForm;

    _solver = new Solver();
    _intervals = new IntervalService();
    _arrayService = new ArrayService();

    _form.Reading += ReadData;
    _form.Drawing += DrawData;
}

/// <summary>
/// Чтение данных с главной формы.
/// </summary>
/// <param name="list"> Диапазон интервалов. </param>
/// <param name="delta"> Количество точек. </param>
/// <param name="power"> Порядок уравнения. </param>
/// <param name="updateCoeff"> Модификаторы коэффициентов. </param>

```

```

    private void ReadData(IList<string> list, int delta, int power, IList<double>
updateCoeff)
    {
        Power = power;
        Delta = delta;

        var intervals = _intervals.CreateIntervals(list.Select(x =>
double.Parse(x)).ToList());
        var vertexes = _intervals.CreateVertex(intervals);
        var coeff = vertexes.Select(x => _intervals.CreateCoeffs(x, Power)).ToList();

        roots = CalculateRoots(coeff, intervals.Count, updateCoeff);

        var analyzer = new Analyzer(roots, intervals, Power, updateCoeff);

        var q = new Stopwatch();
        q.Start();
        roots = analyzer.Anylize();
        q.Stop();
        var time = q.Elapsed;
    }

    /// <summary>
    /// Адаптер к расчету корней.
    /// </summary>
    /// <param name="coeffs"> Коэффициенты одного уравнения. </param>
    /// <param name="intervalCount"> Количество интервалов. </param>
    /// <param name="updateCoeffs"> Модификаторы коэффициентов. </param>
    /// <returns> Корни. </returns>
    private IList<Point> CalculateRoots(IList<IList<double>> coeffs, int intervalCount,
IList<double> updateCoeffs)
    {
        var result = new List<Point>();

        for (int i = 0; i < coeffs.Count; i++)
        {
            var coeff = coeffs[i];
            var roots = _solver.SolvingWithCoeffs(coeff, intervalCount, updateCoeffs);
            result.AddRange(roots);
        }

        return result;
    }

    /// <summary>
    /// Корректировка коэффициентов для обновления согласно форме коэффициентов
уравнения.
    /// </summary>
    /// <param name="updateCoeffs"></param>
    /// <returns></returns>
    private IList<IList<double>> CorrectUpdateCoeff(IList<IList<double>> updateCoeffs)
    {
        var result = new List<IList<double>>();

        foreach (var coeffs in updateCoeffs)
        {
            var update = _intervals.CreateCoeffs(coeffs, Power);
            result.Add(update);
        }

        return result;
    }

    /// <summary>

```

```

    /// Отрисовка данных.
    /// </summary>
    private void DrawData()
    {
        _drawForm.Roots = roots;
        _drawForm.ShowDialog();
    }
}
}

```

Листинг модели отчета:

```

namespace WindowsFormsApplication1.Models
{
    /// <summary>
    /// Отчет об анализе.
    /// </summary>
    public class AnylizeReport
    {
        /// <summary>
        /// Минимальная степень устойчивости.
        /// </summary>
        public double MinX { get; set; }

        /// <summary>
        /// Максимальная степень колебательности.
        /// </summary>
        public double MaxTan { get; set; }

        /// <summary>
        /// Точка степени устойчивости.
        /// </summary>
        public Point PointX { get; set; }

        /// <summary>
        /// Точка степени колебательности.
        /// </summary>
        public Point PointTan { get; set; }

        public AnylizeReport(double minX, Point pointX, double maxTan, Point pointTan)
        {
            MinX = minX;
            MaxTan = maxTan;
            PointX = pointX;
            PointTan = pointTan;
        }
    }
}

```

Листинг службы массивов:

```

using System.Collections.Generic;
using System.Linq;

namespace WindowsFormsApplication1.Services
{
    /// <summary>
    /// Сервис для работы с массивами.
    /// </summary>
    public class ArrayService
    {
        /// <summary>
        /// Разделитель.
        /// </summary>

```

```

private char Separator = ',';

/// <summary>
/// Уникальные массивы.
/// </summary>
public IList<IList<double>> Distint(IList<IList<double>> data)
{
    var list = data.Select(ListToString);
    var uniq = list.Distinct();

    return uniq.Select(StringToList).ToList();
}

/// <summary>
/// Обновление значений массива.
/// </summary>
/// <param name="current"> Текущий массив. </param>
/// <param name="update"> Данные для обновления. </param>
/// <returns> Новый массив. </returns>
public IList<double> UpdateMassive(IList<double> current, IList<double> update)
{
    var result = new List<double>();

    for (int i = 0; i < current.Count; i++)
    {
        var coeff = current[i];
        var updateCoeff = update[i];
        coeff = coeff * updateCoeff;
        result.Add(coeff);
    }

    return result;
}

private string ListToString(IList<double> data)
{
    return string.Join(Separator.ToString(), data.Select(x => x.ToString()));
}

private IList<double> StringToList(string str)
{
    return str.Split(Separator).Select(x => double.Parse(x)).ToList();
}
}
}

```

Листинг службы интервалов:

```

using System;
using System.Collections.Generic;
using System.Linq;
using WindowsFormsApplication1.Models;

namespace WindowsFormsApplication1.Services
{
    /// <summary>
    /// Служба для работы с интервалами.
    /// </summary>
    public class IntervalService
    {
        /// <summary>
        /// Преобразование диапазонов в интервалы.
        /// </summary>
        /// <param name="list"> Диапазоны. </param>
        /// <returns> Интервалы. </returns>

```

```

public IList<Interval> CreateIntervals(IList<double> numbers)
{
    var result = new List<Interval>();

    for (int i = 0; i < numbers.Count - 1; i += 2)
    {
        var point = new Interval(numbers[i], numbers[i + 1]);
        result.Add(point);
    }

    return result;
}

/// <summary>
/// Получение из пользовательских данных точки на параметрическом многограннике.
/// Несобранные в сами узлы.
/// </summary>
/// <param name="data"> Пользовательские данные. </param>
/// <returns> Точки на многограннике. </returns>
public IList<IList<double>> CreateVertex(IList<Interval> data)
{
    var result = new List<IList<double>>();

    var rankCount = data.Count; // Количество разрядов в маске.
    var allVariants = Math.Pow(2, rankCount);

    for (int i = 0; i < allVariants; i++) // Цикл рассматривает все возможные
варианты 2 в степени количества пар. Например, 00 или 01.
    {
        var maskList = CreateMask(i, rankCount);

        var currentList = new List<double>(); // Набор значений для уравнения (a3,
a2, a1).

        for (int k = 0; k < maskList.Count(); k++) // Перебор разрядов маски.
        {
            var currentMask = maskList[k]; // Значение разряда.
            var currentValue = currentMask == 0 // Число из пары.
                ? data[k].Min
                : data[k].Max;

            currentList.Add(currentValue);
        }

        result.Add(currentList);
    }

    return result;
}

/// <summary>
/// Создание маски для перебора пользовательских значений.
/// </summary>
/// <param name="i"> Номер итерации возможного варианта. </param>
/// <param name="allVariants"> Количество всех возможных вариантов. </param>
/// <returns> Маска. </returns>
private IList<int> CreateMask(int i, int allVariants)
{
    var mask = Convert.ToString(i, 2); // Создание маски числа в двоичной системе.
Она определяет какое число из пары выбирать.
    var insertString = new string('0', allVariants - mask.Length);
    mask = mask.Insert(0, insertString); // Необходимо докинуть нулей в маску для
соблюдения нужного количества разрядов.
}

```

```

        var maskList = mask.Select(x => int.Parse(x.ToString())).ToList(); // Строку в
        массив, чтобы обращаться к каждому разряду.
        maskList.Reverse(); // Инvertировать направление маски, чтобы перебор шел от
        младшего разряда.

        return maskList;
    }

    /// <summary>
    /// Обновление массива. Из-за порядка уравнения.
    /// </summary>
    /// <param name="data"> Вершина. </param>
    /// <param name="power"> Порядок уравнения. </param>
    /// <returns> Обновленные данные. </returns>
    public IList<double> CreateCoeffs(IList<double> data, int power)
    {
        var result = data.ToList();

        var i = 0;
        while (i < power)
        {
            var array = data.ToArray();
            var copy = (double[])array.Clone();

            result.AddRange(copy);

            i++;
        }

        return result;
    }

    /// <summary>
    /// Получить коэффициенты, описывающие одно слагаемое.
    /// </summary>
    /// <param name="data"> Все коэффициенты уравнения. </param>
    /// <param name="count"> Количество интервалов. </param>
    /// <returns></returns>
    public IList<double> GetCoeffs(IList<double> data, int count)
    {
        var result = new List<double>();

        for (int i = 0; i < count; i++)
        {
            result.Add(data[i]);
        }

        return result;
    }
}
}
}

```

Листинг службы решения уравнений:

```

using System.Collections.Generic;
using System.Linq;
using WindowsFormsApplication1.Services;
using static alglib;
using static alglib.polynomialssolver;

namespace WindowsFormsApplication1
{
    /// <summary>

```

```

/// Служба математических расчетов.
/// </summary>
public class Solver
{
    private readonly IntervalService _intervalService;
    private readonly ArrayService _arrayService;

    public Solver()
    {
        _intervalService = new IntervalService();
        _arrayService = new ArrayService();
    }

    /// <summary>
    /// Поиск корней полинома.
    /// </summary>
    /// <param name="coeffs"> Коэффициента полинома. </param>
    /// <param name="count"> Количество интервалов. </param>
    /// <returns> Массив корней. </returns>
    public complex[] Solving(IList<double> coeffs, int count)
    {
        var a = coeffs.ToArray();
        var n = a.Length - 1;

        polynomialsolve(a, n, out complex[] x, out alglib.polynomialsolverreport
report);

        return x;
    }

    /// <summary>
    /// Решение уравнения с обновлением коэффициентов. Адаптер к обычному решению.
    /// </summary>
    /// <param name="coeffs"> Коэффициенты. </param>
    /// <param name="count"> Порядок. </param>
    /// <param name="updateCoeff"> Обновители. </param>
    /// <returns></returns>
    public IList<Point> SolvingWithCoeffs(IList<double> coeffs, int count, IList<double>
updateCoeff)
    {
        var newCoeff = _arrayService.UpdateMassive(coeffs, updateCoeff).ToArray();
        var roots = Solving(newCoeff, count);
        coeffs = _intervalService.GetCoeffs(coeffs, count);
        var result = roots.Select(root => new Point(root.x, root.y, coeffs)).ToList();

        return result;
    }
}
}

```

Листинг анализатора:

```

using System;
using System.Collections.Generic;
using System.Linq;
using WindowsFormsApplication1.Models;
using WindowsFormsApplication1.Services;

namespace WindowsFormsApplication1.Tools
{
    /// <summary>
    /// Анализатор.
    /// </summary>

```

```

public class Analyzer
{
    /// <summary>
    /// Сервис интервалов.
    /// </summary>
    private readonly IntervalService _intervals;

    /// <summary>
    /// Сервис мат.расчетов.
    /// </summary>
    private readonly Solver _solver;

    private readonly ArrayService _arrayService;

    /// <summary>
    /// Конструктор анализатора.
    /// </summary>
    /// <param name="points"> Корни. </param>
    /// <param name="intervals"> Интервалы. </param>
    /// <param name="power"> Порядок уравнения. </param>
    /// <param name="updateCoeffs"> Модификаторы коэффициентов. </param>
    public Analyzer(IList<Point> points, IList<Interval> intervals, int power,
IList<double> updateCoeffs)
    {
        Points = points;
        Intervals = intervals;
        Power = power;
        UpdateCoeffs = updateCoeffs;

        _intervals = new IntervalService();
        _solver = new Solver();
        _arrayService = new ArrayService();
    }

    /// <summary>
    /// Словарь интервалов.
    /// </summary>
    private IList<Interval> Intervals { get; }

    /// <summary>
    /// Точки.
    /// </summary>
    private IList<Point> Points { get; }

    /// <summary>
    /// Коэффициенты для обновления.
    /// </summary>
    private IList<double> UpdateCoeffs { get; }

    /// <summary>
    /// Порядок уравнения.
    /// </summary>
    private int Power { get; }

    /// <summary>
    /// Целенаправленный анализ.
    /// </summary>
    public IList<Point> Anylize()
    {
        var result = new List<Point>();

        var report = CurrentAnylize(Points);

        var minX = report.MinX;

```

```

    var pointX = report.PointX;
    GetDirection(pointX);

    var edges = BiAnalysis(pointX, minX, CheckMin);
    result.AddRange(edges);

    var maxTan = report.MaxTan;
    var pointTan = report.PointTan;
    GetDirection(pointTan);

    edges = BiAnalysis(pointTan, maxTan, CheckMax);
    result.AddRange(edges);

    return result;
}

/// <summary>
/// Промежуточный анализ.
/// </summary>
/// <param name="roots"> Корни. </param>
/// <returns> Отчет об анализе. </returns>
private AnylizeReport CurrentAnylize(IList<Point> roots)
{
    var minX = roots.Select(x => x.X).Min();
    var pointX = roots.ToList().Find(x => Math.Abs(x.X - minX) <
Constants.Tolerance);

    var maxTan = roots.Select(x => Math.Tan(x.Y / x.X)).Max();
    var pointTan = roots.ToList().Find(x => Math.Abs(Math.Tan(x.Y / x.X) - maxTan) <
Constants.Tolerance);

    return new AnylizeReport(minX, pointX, maxTan, pointTan);
}

/// <summary>
/// Двухнаправленный анализ ребер.
/// </summary>
/// <param name="point"> Точка. </param>
/// <param name="prevValue"> Предыдущий показатель. </param>
/// <param name="check"> Метод сравнения. </param>
private IList<Point> BiAnalysis(Point point, double prevValue, CheckChange check)
{
    var directions = point.Directions;
    var coeffs = point.Coeffs;
    var edges = new List<Point> { point };

    for (var i = 0; i < directions.Count; i++)
    {
        var direct = directions[i];
        var steps = direct.Steps.Skip(1).ToList(); // Пропуск первого значения,
равного самому коэффициенту.
        var currentCoeffs = (double[])coeffs.ToArray().Clone(); // Глубокое
клонирование.

        var directEdge = BuildEdge(steps, currentCoeffs, check, ref prevValue, i,
edges).ToList();

        steps.Reverse(); // Обход в обратном направлении.

        var revertEdge = BuildEdge(steps, currentCoeffs, check, ref prevValue, i,
edges).ToList();
    }

    return edges;
}

```

```

}

/// <summary>
/// Построение ребра.
/// </summary>
/// <param name="steps"> Изменения коэффициентов. </param>
/// <param name="currentCoeffs"> Текущие коэффициенты. </param>
/// <param name="check"> Метод анализа. </param>
/// <param name="startPrevValue"> Предыдущий показатель. </param>
/// <returns></returns>
private IList<Point> BuildEdge(IList<double> steps, double[] currentCoeffs,
CheckChange check,
    ref double prevValue, int i, List<Point> points)
{
    var result = new List<Point>();

    var coeffCount = currentCoeffs.Length;
    var correctCoeffs = _intervals.CreateCoeffs(currentCoeffs, Power);

    foreach (var step in steps)
    {
        for (var j = i; j < correctCoeffs.Count; j += coeffCount)
        {
            correctCoeffs[j] = step;
        }

        var roots = _solver.SolvingWithCoeffs(correctCoeffs, Power, UpdateCoeffs);

        var currentReport = CurrentAnylize(roots);
        var checkResult = check(currentReport, prevValue);

        if (checkResult.Item1)
        {
            prevValue = checkResult.Item2;
            result.Add(checkResult.Item3);
        }
        else
        {
            if (result.Any())
            {
                points.Clear();
                points.AddRange(result);
            }
            break;
        }
    }

    if (result.Count == steps.Count) // Все точки подходят.
    {
        points.Clear();
        points.AddRange(result);
    }

    return result;
}

/// <summary>
/// Определение направлений.
/// </summary>
/// <param name="point"> Точка. </param>
private void GetDirection(Point point)
{
    var directions = new List<Direct>();

```

```

    for (var i = 0; i < point.Coeffs.Count; i++)
    {
        var coeff = point.Coeffs[i];
        directions.Add(new Direct(coeff, Intervals[i]));
    }

    point.Directions = directions;
}

/// <summary>
/// Уменьшилось?
/// </summary>
/// <param name="current"> Текущее. </param>
/// <param name="prev"> Прежнее. </param>
/// <returns> Результат. </returns>
private Tuple<bool, double, Point> CheckMin(AnylyzeReport current, double prev)
{
    var value = current.MinX;

    return Math.Abs(value) < Math.Abs(prev)
        ? new Tuple<bool, double, Point>(true, value, current.PointX)
        : new Tuple<bool, double, Point>(false, prev, current.PointX);
}

/// <summary>
/// Увеличилось?
/// </summary>
/// <param name="current"> Текущее. </param>
/// <param name="prev"> Прежнее. </param>
/// <returns> Результат. </returns>
private Tuple<bool, double, Point> CheckMax(AnylyzeReport current, double prev)
{
    var value = current.MaxTan;

    return Math.Abs(value) > Math.Abs(prev)
        ? new Tuple<bool, double, Point>(true, value, current.PointTan)
        : new Tuple<bool, double, Point>(false, prev, current.PointTan);
}

/// <summary>
/// Делегат сравнения.
/// </summary>
/// <param name="current"> Текущее значение. </param>
/// <param name="prev"> Предыдущее значение. </param>
/// <returns></returns>
private delegate Tuple<bool, double, Point> CheckChange(AnylyzeReport current,
double prev);
}
}

```

Листинг графической формы:

```

using System.Collections.Generic;
using System.Drawing;
using System.Linq;
using System.Windows.Forms;
using System.Windows.Forms.DataVisualization.Charting;

namespace WindowsFormsApplication1
{
    /// <summary>
    /// Класс-форма для графиков.
    /// </summary>

```

```

public partial class graphic : Form
{
    /// <summary>
    /// Корни.
    /// </summary>
    public IList<Point> Roots;

    public graphic()
    {
        InitializeComponent();

        Load += Graphic_Load;
    }

    /// <summary>
    /// Обработчик события загрузки.
    /// </summary>
    /// <param name="sender"> Объект. </param>
    /// <param name="e"> Событие. </param>
    private void Graphic_Load(object sender, System.EventArgs e)
    {
        if (Roots?.Count == null)
        {
            MessageBox.Show("Корней нет.");
            return;
        }

        var re = Roots.Select(x => x.X).ToList();
        var im = Roots.Select(y => y.Y).ToList();

        Godograph.ChartAreas[0].AxisX.Minimum = re.Min() - 1;
        Godograph.ChartAreas[0].AxisY.Minimum = im.Min() - 1;

        Godograph.ChartAreas[0].AxisX.Maximum = re.Max() + 1;
        Godograph.ChartAreas[0].AxisY.Maximum = im.Max() + 1;

        //for (int i = 0; i < re.Count; i++)
        //{
        //    var x = re[i];
        //    var y = im[i];
        //    var name = i.ToString();

        //    Godograph.Series.Add(name);
        //    var series = Godograph.Series[name];

        //    series.MarkerStyle = MarkerStyle.Cross;
        //    series.Color = Color.Black;

        //    series.Points.AddXY(x, y);
        //}
        Godograph.Series.Add("test");
        var series = Godograph.Series["test"];

        series.MarkerStyle = MarkerStyle.Cross;
        series.Color = Color.Black;

        for (int i = 0; i < re.Count; i++)
        {
            var x = re[i];
            var y = im[i];

            series.Points.AddXY(x, y);
        }
    }
}

```

Приложение В
(обязательное)

CHAPTER 1. Displaying a parametric polyhedron of an interval polynomial onto the root plane

Студент:

Группа	ФИО	Подпись	Дата
8АМ61	Гурьев Андрей Андреевич		

Консультант школы отделения (НОЦ): Отделение информационных технологий

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Суходоев М.С.	К.Т.Н.		

Консультант – лингвист отделения (НОЦ) школы: Отделение иностранных языков

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Шепетовский Д.В.			

CHAPTER 1. Displaying a parametric polyhedron of an interval polynomial onto the root plane

1.1. Basic concepts and notation when a parametric polyhedron is displayed.

Fundamental results that allow one to research the robust stability of a system with interval uncertainty were obtained in the works of V.L. Kharitonov [114].

Suppose that the characteristic polynomial of a system with interval uncertainty has the form:

$$P(s) = \sum_{i=0}^n a_i s^i, \quad \underline{a}_i \leq a_i \leq \overline{a}_i, \quad (1.1)$$

where a_i are interval coefficients, \overline{a}_i is a maximum value a_i and \underline{a}_i is its minimum value.

According to the theorem, it is possible to distinguish polynomials that have been called the Kharitonov's polynomials. These are four polynomials composed of the extreme values of the coefficients alternating in pairs (the two lower values - the two upper values)

$$\begin{aligned} P_1(s) &= \underline{a}_0 + \underline{a}_1 s + \overline{a}_2 s^2 + \overline{a}_3 s^3 + \dots; \\ P_2(s) &= \overline{a}_0 + \underline{a}_1 s + \underline{a}_2 s^2 + \overline{a}_3 s^3 + \dots; \\ P_3(s) &= \overline{a}_0 + \overline{a}_1 s + \underline{a}_2 s^2 + \underline{a}_3 s^3 + \dots; \\ P_4(s) &= \underline{a}_0 + \overline{a}_1 s + \overline{a}_2 s^2 + \underline{a}_3 s^3 + \dots \end{aligned} \quad (1.2)$$

The theorem reads as follows: "it is necessary and sufficient that all Kharitonov's polynomials are stable for the robust stability of an interval set" [116].

Suppose that the polynomial has m interval coefficients. They form a parametric polyhedron, which is a rectangular hyperparallelepiped containing 2^m vertices and $m \cdot 2^{m-1}$ edges. An example of a parametric polyhedron for the polynomial of a system with three interval coefficients is shown in Fig. 1.1.

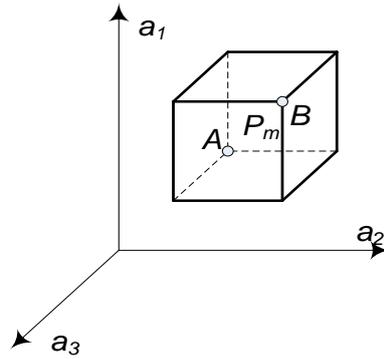


Figure 1.1 - Parametric polyhedron P_m

The coordinates of any point P_m relative to the vertex V_q , $q = \overline{1, 2^m}$ are determined by the expressions:

$$a_i = a_i^q + \Delta a_i, i = \overline{1, n}, \quad (1.3)$$

$$(\underline{a}_i - a_i^q) \leq \Delta a_i \leq (\overline{a}_i - a_i^q), \quad (1.4)$$

where Δa_i is the increment of the «i» interval parameter, a_i^q is its value at the vertex V_q . Let $M(s)$ and $N(s)$ be two neighboring polynomials defined by two neighboring vertices connected by one edge. Let us introduce the concept of an edge polynomial. It is an one-parameter set of the form $\gamma M(s) + (1-\gamma)N(s)$, where $0 \leq \gamma \leq 1$.

According to a theorem called the edge theorem, "the robust stability of the set of polynomials depends on the stability of all its edge polynomials" [5]. The effectiveness of this approach decreases with increasing number of coefficients (for $m = 5$, where m is the number of uncertain coefficients, it is necessary to verify $m \cdot 2^{m-1} = 80$ edge polynomials). This is related to a large amount of computation, and therefore it is necessary to find more efficient solution.

From the consideration of the areas of the displaying of a parametric polyhedron (look at Figure 1.1), there is an obvious conclusion that for the analysis of robust stability it is sufficient to verify only those of its essential edges whose images form the boundaries of the areas of localization of the roots of the polynomial. However, for

this it is necessary to be able to determine in advance the essential edges from the available information about the structure of the polynomial and the intervals of uncertainty of its coefficients.

Let us denote the displays of the vertices V^q of a parametric polyhedron as U^q . Figure 1.2 shows the display of vertices P_m onto the complex plane of the roots, in which the complex conjugate roots are localized in two areas bounded by the images of the edges P_m .

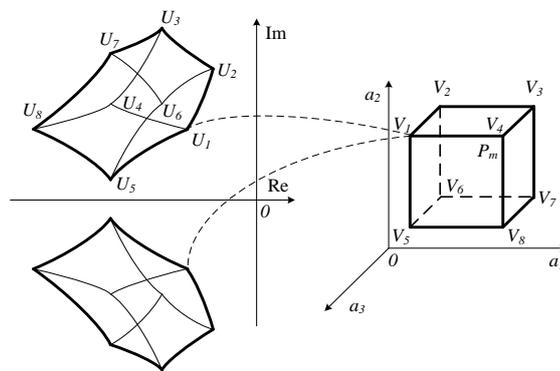


Figure 1.2 - Displaying the vertices of a parametric polyhedron

Suppose that the edges of the interval polynomial are R_i^q , $i = \overline{1, m}$, where m is the number of interval coefficients and q is the numerical order of the vertex. Let's denote the images of the edges by RS_i^q and call it the edge branches. The edges P_m and its images are shown in Figure 1.3.

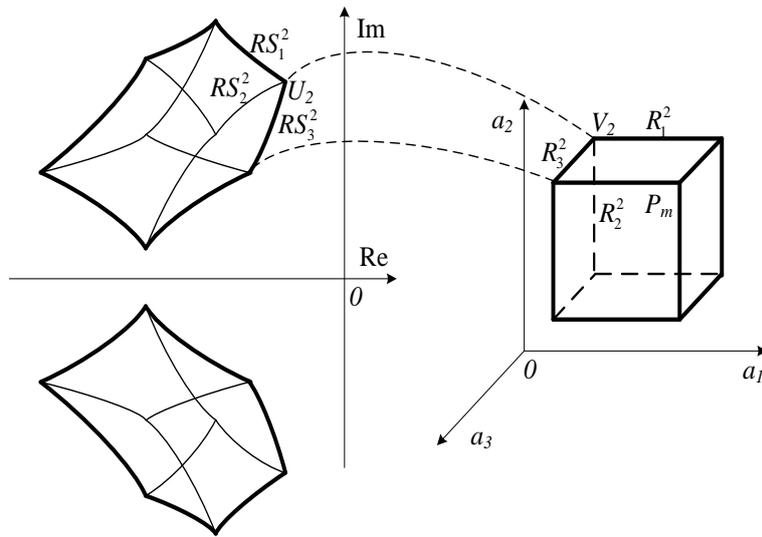


Figure 1.3 - Displaying of edges of a parametric polyhedron

According to the definition of a polyhedron, any of its facets is a rectangle on the plane of variation of two interval coefficients from one vertex.

Denote the facets of the interval polynomial as G_{ij}^q (Figure 1.4), $i = \overline{1, m}$, $j = \overline{1, m}$, $i \neq j$, q is the numerical order of the vertex. The images of the facets are denoted by GS_{ij}^q .

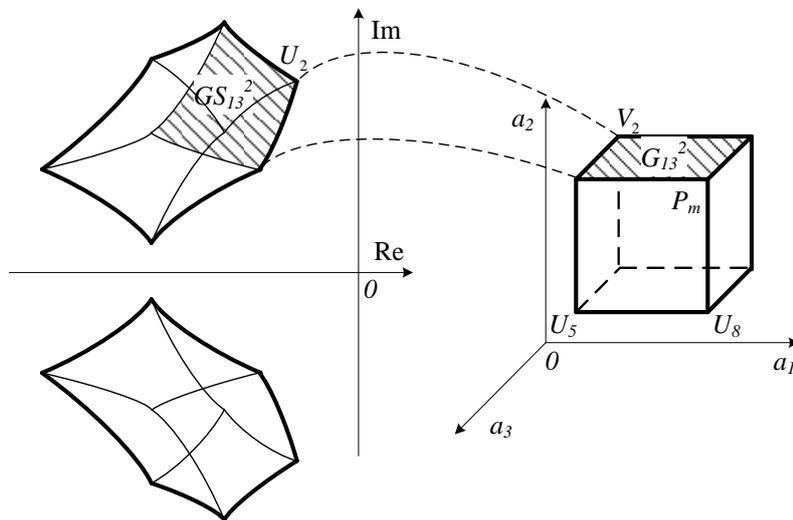


Figure 1.4 - Displaying the facets of a parametric polyhedron

Denote the exit angle from the root node U_q by means j parameter as φ_j (Figure 1.5). The vectors defining the exit angles of the edge branches are denoted by E_j .

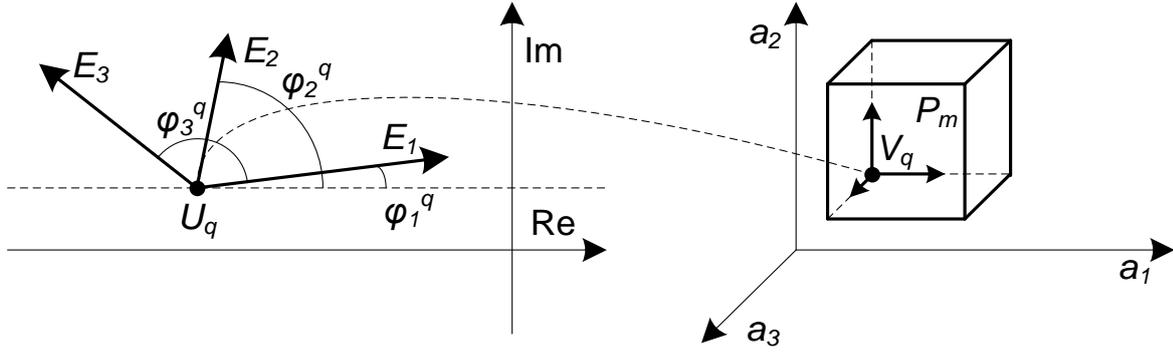


Figure 1.5 - Angles of the exit of the edge branches

1.2. Displaying properties of the parametric polyhedron for systems with affine indeterminacy

It is necessary to have a relation connecting the coordinates of the points P_m with n roots of the polynomial (1.1) for the display $\phi: P_m \rightarrow S$, where S is a set of roots (1.1). For systems with affine uncertainty, the relation has the following form (in this case, system parameters compose the coefficients of the polynomial in the form of linear functions):

$$D(s) = \sum_{i=0}^m (T_i \cdot A_i(s)) + B(s) = 0, \quad \underline{T}_i \leq T_i \leq \bar{T}_i, \quad (1.5)$$

where $A_i(s) = \sum_{j=0}^k (a_{i,j} s^j)$, $B(s) = \sum_{j=0}^k (b_j s^j)$.

For displaying $\phi: P_m \rightarrow S$ it is necessary to use the following expression:

$$D_q(s) + \sum_{i=0}^k (\Delta T_i \cdot A_i(s)) = 0, \quad (1.6)$$

where $D_q(s)$ is polynomial with the values of interval parameters, corresponding to the vertex q of the parametric polyhedron.

The following notation will be used to consider edges P_m : the edge R_i^q , where i is the index ΔT_i , $i = \overline{1, m}$, q is the index V_q from which T_i varies along the edge. According to (1.5), it is possible to write the equation of the displaying of the edge R_i^q on the complex plane:

$$D_q(s) + \Delta T_i \cdot A_i(s) = 0 \quad (1.7)$$

If the system has a single feedback, the transfer function of an equivalent open-loop system can be written as follows:

$$W_i^q(\Delta T_i, s) = \frac{\Delta T_i \cdot A_i(s)}{D_q(s)} \quad (1.8)$$

The change ΔT_i leads to the formation of a one-parameter interval root locus diagram due to the displacement of the roots from the poles of the function (1.8) to its zeros.

The facets of the plurality of edges P_m , being considered on the plane of two interval parameters T_i and T_j for one vertex V_q , are a rectangle. Using (1.5), it is possible to write the equation for the displaying of the plane of the facet G_{ij}^q :

$$D_q(s) + \Delta T_i \cdot A_i(s) + \Delta T_j \cdot A_j(s) = 0 \quad (1.9)$$

Let $s_r = \alpha + j\beta$, $r \in \overline{1, n}$. Substituting in (1.9) the coordinates s_r , we obtain a system of two linear equations with two variables T_i and T_j :

$$\begin{cases} T_i \cdot \operatorname{Re}(A_i(\alpha + j\beta)) + T_j \cdot \operatorname{Re}(A_j(\alpha + j\beta)) + \operatorname{Re}(D_q(s)) = 0; \\ T_i \cdot \operatorname{Im}(A_i(\alpha + j\beta)) + T_j \cdot \operatorname{Im}(A_j(\alpha + j\beta)) + \operatorname{Im}(D_q(s)) = 0. \end{cases} \quad (1.10)$$

Equation (1.10) differs in two cases:

1. *The equations are independent and the system has a unique solution $T_i = T_i^*$, $T_j = T_j^*$. Consequently, $\phi^{-1}(s_r) = P^*$, $P^* = (T_i^*, T_j^*)$ where the point P^* belongs to the plane G_{ij}^q .*

2. *The equations differ by a constant factor.* Consequently, there is a straight line t , described by any of the equations of system (1.10) on the plane G_{ij}^q , where $\varphi^{-1}(s_r) = t$.

When P_m is displayed, the boundaries can contain a full or partial edge branch. These branches can intersect at U^* . Conditions of the presence U^* are the presence of a linear dependence of equation (1.11):

$$a_i s^i + a_j s^j + P(s) = 0. \quad (1.11)$$

This formula will be satisfied if there are α и β , hence the equality is presented below.

$$\frac{\operatorname{Re} A_i(\alpha + j\beta)}{\operatorname{Im} A_i(\alpha + j\beta)} = \frac{\operatorname{Re} A_j(\alpha + j\beta)}{\operatorname{Im} A_j(\alpha + j\beta)} = \frac{\operatorname{Re}(D_q(\alpha + j\beta))}{\operatorname{Im}(D_q(\alpha + j\beta))} \quad (1.12)$$

Therefore, it is necessary to solve a system of nonlinear equations:

$$\begin{cases} \operatorname{Re}(A_i(\alpha + j\beta)) \cdot \operatorname{Im}(A_j(\alpha + j\beta)) - \\ - \operatorname{Re}(A_j(\alpha + j\beta)) \cdot \operatorname{Im}(A_i(\alpha + j\beta)) = 0; \\ \operatorname{Re}(A_j(\alpha + j\beta)) \cdot \operatorname{Im}(D_q(\alpha + j\beta)) - \\ - \operatorname{Im}(A_j(\alpha + j\beta)) \cdot \operatorname{Re} D_q(\alpha + j\beta) = 0. \end{cases} \quad (1.13)$$

It should be noted that the left-hand sides of the equations of system (1.13) are polynomials of uneven degree β without a constant term. Consequently, if (1.13) does not have a solution with $\beta \neq 0$, then there is no point U^* in the displays of the considered facet and parallel sections P_m . The absence of solutions (1.13) for all combinations of interval parameters means that the boundaries S_r consist of disjoint edge branches.

1.3. Main results

In the current chapter, the basic notations were introduced, which are used in the robust extension of the root locus diagram method, the displaying properties of the

parametric polyhedron of ICP onto the root plane with affine indeterminacy are analyzed.

CHAPTER 2. Analysis of robust quality of interval automatic control systems

2.1. Determination of boundary vertices with affine indeterminacy

Suppose that U_q is one of the solutions of the characteristic equation on the root plane. It is necessary to determine its belonging to the boundary (Figure 2.1)?

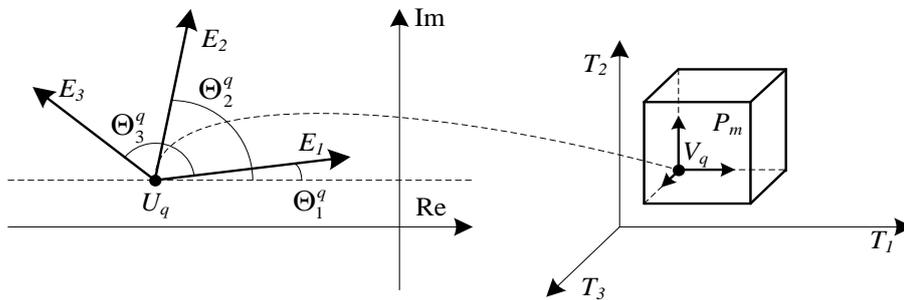


Figure 2.1 - Displaying the vertices of a parametric polyhedron

The necessary definition is the exit angle Θ_i^q from the complex U_q . It is found from the equation of phases [83], written for U_q . If there are $D^q(s)$ with degree n and $A_i(s)$ with degree z , it will be possible to obtain the following formula by changing T_i :

$$\Theta_i^q = 180^\circ - \sum_{k=1}^n \Theta_k + \sum_{l=1}^z \Theta_l, \quad (2.1)$$

under condition of decreasing T_i analog:

$$\Theta_i^q = -\sum_{k=1}^n \Theta_k + \sum_{l=1}^z \Theta_l, \quad (2.2)$$

where Θ_k and Θ_l are the angles between the real axis and vectors directed from U_q to the k pole and to the l zero of the function respectively.

Figure 2.1 also shows the possible directions of motion s_r from U_q , taking into account the facets P_m with a common vertex V_q . As can be seen, the direction s_r is

determined by the following vector — $\vec{E}_{ij} = \vec{E}_i + \vec{E}_j$, where \vec{E}_i and \vec{E}_j are the edge vectors given by Θ_i^q and Θ_j^q [Error! Reference source not found.]. According to the rules of vector algebra, \vec{E}_{ij} leaves U_q and lies inside the angle Ψ_{ij} , formed by \vec{E}_i and \vec{E}_j , where $0^\circ \leq \Psi_{ij} \leq 180^\circ$.

Two boundary edge branches that form a boundary angle $G\Psi_{ij}$ in the range $[0^\circ, 180^\circ]$ emerge from the vertex GU_q . It follows that the other angles Ψ_{ij} and the corresponding vectors belong to $G\Psi_{ij}$. Using the previously indicated exit angles, one can express an inequality describing the indicated condition:

$$|\Theta_m^{Vq} - \Theta_1^{Vq}| < 180^\circ \quad (2.3)$$

Due to (2.3), it is possible to find out the vertex P_m belonging to the boundary of the localization area S_r of the complex root of the ICP.

Figure 2.2 shows the possible location of exit vectors RS_i^q from the boundary root node (Figure 2.2a) and located within the boundary (Figure 2.2b).

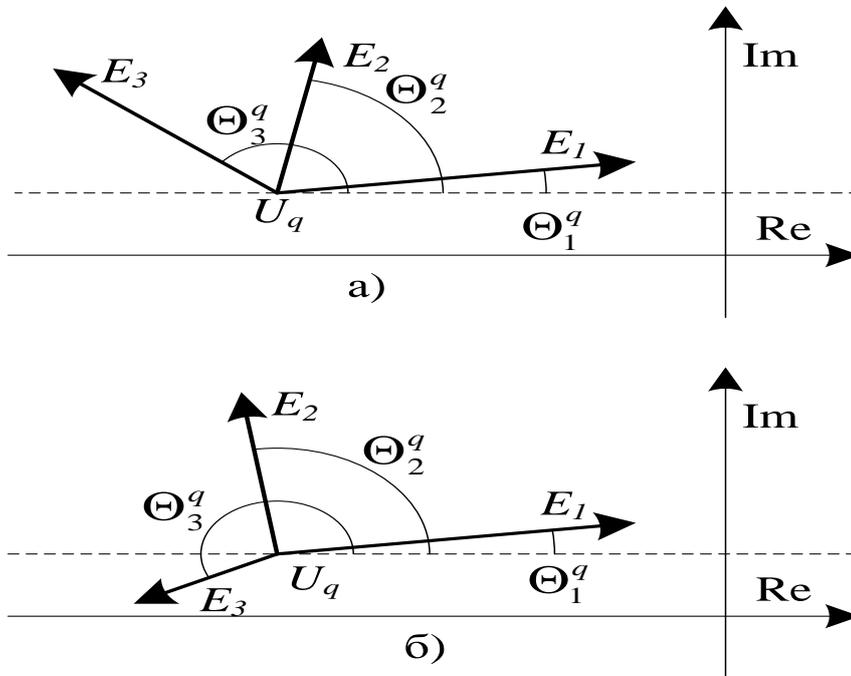


Figure 2.2 - Location of exit vectors RS_i^q from the root node

2.2. Edge analysis of the robust quality of the system with affine indeterminacy

For analysis, the edge branches (RS_i and RS_j) from neighboring boundary root nodes and the relation of exit angles between them are considered. Since they are connected by one boundary edge branch GRS_k , the branches, which are under consideration, are also the inverse images of the edges of one facet P_m .

Let $U^* \notin S_r$. When T_k changes along the edge GR_k , the root moves along GRS_k . In this case, the exit angles from s_r of the branches RS_i^s and RS_j^s can vary. However, since $U^* \notin S_r$, so RS_i^s and RS_j^s don't intersect. Therefore, at the transition of s_r from one GU_q to another, the sequence of quantities Θ_i^s and Θ_j^s (Fig. 2.3a) will be preserved. Generalizing this case to a larger number of considered edge branches, it is possible to conclude that the values of exit angles will be arranged in the same sequence in neighboring boundary root nodes.

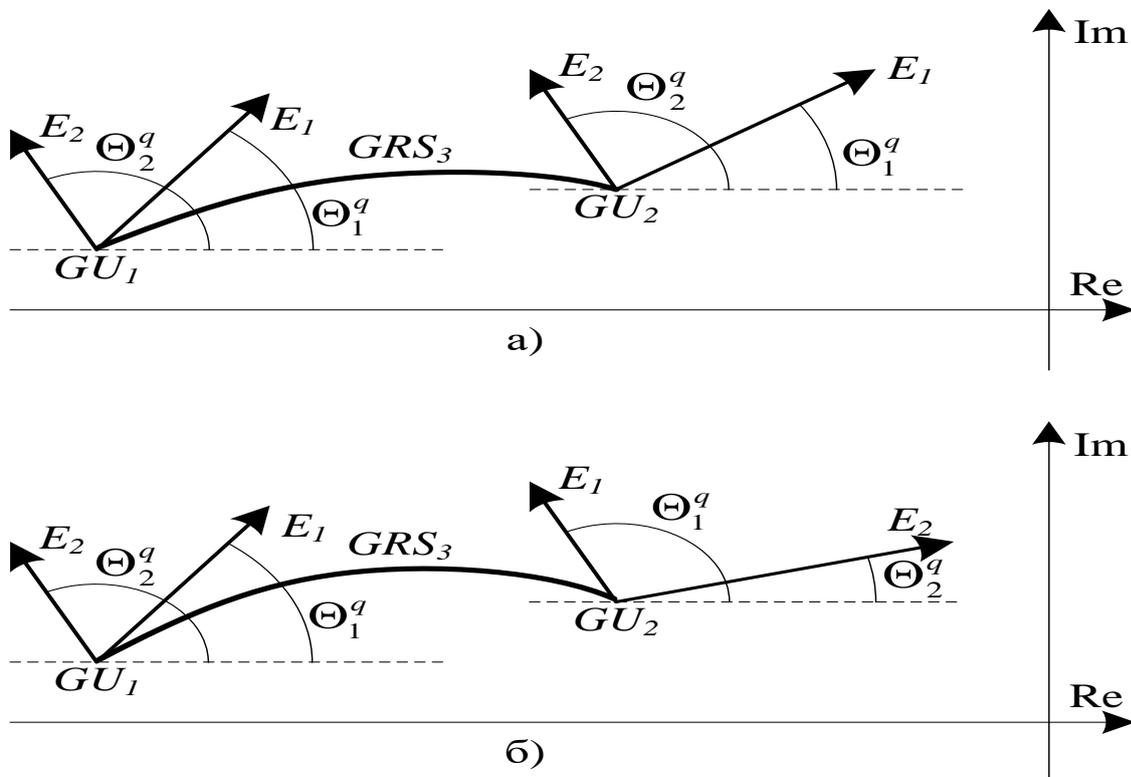


Figure 2.3 - Location of exit vectors RS_i^q from the root node when one parameter is varied

Let $U^* \in S_r$, then RS_i^s and RS_j^s can intersect at U^* . If such situation as $s_r = U^*$ arises and U^* exits, when T_k varies, the relation Θ_i^s and Θ_j^s will change in the remaining area. In this case, in the neighboring boundary root nodes, the sequence of the exit angles of the edge branches will not be preserved (Fig. 2.3b).

Based on the above, it is possible to install the steps that form the algorithm itself for the search for boundary edge routing P_m .

It is assumed that the system (1.10) with affine indeterminacy has no solution for any of the facets P_m with the chosen common vertex ($U^* \notin S_r$). In this case, with known GU_q and GRS_i^q emerging from it, the subsequent boundary edge branch is determined by the modulus of the difference between its own exit angle and the analogous for GRS_i^q , namely, its smallest (largest) value. Thus, having information about measures of the angle Θ_i^q , $i = \overline{1, m}$ for any boundary root node, and then arranging them in the increasing (decreasing) order, it is possible to determine the sequence of branches RS_i^q , $q = \overline{1, 2m}$, $i = \overline{1, m}$ that bound the area of localization of the complex root.

This sequence is an ordered set of edges P_m , that is, a closed boundary edge route, consisting of $2m$ edges, connecting m pairs of vertices P_m , where the coordinates of pairs have opposite boundary values of interval parameters.

For system (1.10), a single solution is sufficient, so that it is possible to testify about the presence U^* in the area S_r . If the described scenario is implemented, it will be necessary to find a node GU_q , for which the sequence Θ_i^q and the corresponding sequence of edges P_m are defined. As a result, an edge route is obtained which must also

include all edges of the facet G_{ij} , provided that the next edge and the next one are formed by the parameters T_i and T_j whereby the considered system has a solution.

Summing up, it is possible to generate the following algorithm of system analysis with interval parameters:

1. It is necessary to transform the characteristic polynomial to the form (1.5).
2. Determine the coordinates of the vertices of the edge polyhedron P_m formed by the boundary values of the interval parameters.
3. Define the boundary node U^* from (1.10) by finding the exit angles from the projection of the vertex of the parametric polyhedron on the root plane.
4. Construct the edge route.
5. Analyze the location of the root localization area, determine indicators of robust system quality.

More detailed analysis of the construction of the edge route is presented below.

For the research, an affine characteristic polynomial with interval parameters is taken, the maximum degree of the polynomial is l .

$$P(s) = B(s) + \sum_{i=0}^m (T_i A_i^*(s)), \quad (2.3)$$

where $B(s) = \sum_{j=0}^k (b_j s^j)$, $A_i^*(s) = \sum_{j=0}^1 (a_{i,j} s^j)$, $\underline{T}_i < T_i < \bar{T}_i$.

Since the polynomial $B(s)$ can be of any degree, there is no such pair of interval parameters (2.3) so that it has solutions, therefore, the presence of U^* is impossible in the area S_r .

The zeros of the characteristic polynomial are displayed onto the real axis (Figure 2.4).

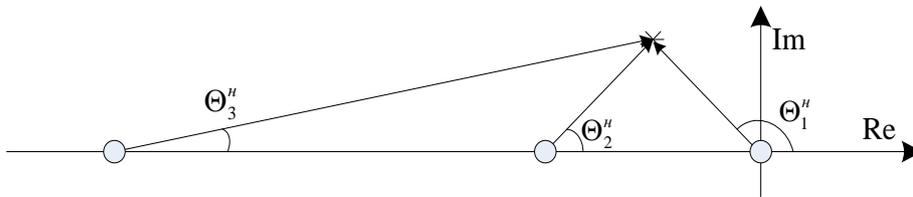


Figure 2.4 - An example of the location of the zeros of the characteristic polynomial with affine indeterminacy with the maximum value of the degree of a polynomial with interval parameters equal to 1

This leads to the fact that there is a sequence of exit angles $\Theta_1^H > \Theta_2^H > \dots > \Theta_n^H$, when the zeros of the transfer functions form the corresponding sequence $s_1 > s_2 > \dots > s_m$. The node considered in Figure 2.4 is a boundary node, since the difference of the extreme angles for the represented sequence is always less than 180° .

Earlier, the equations of the phase relations (2.1) and (2.2) were described. According to them, the difference between the extreme exit angles from the vertex formed by the boundary values of the interval coefficients (by all minimal or all maximal ones) will be less than 180° since the minimum angle from zero to the pole is 0° (in case the zero tends to minus ∞), and the maximum angle is 180° (in case the zero tends to plus ∞).

Further, it is necessary to consider the vectors of exit angles from the boundary vertex (Figure 2.5).

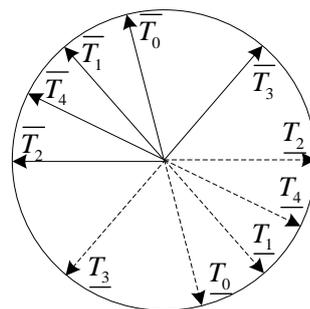


Figure 2.5 - An example of the image of the vectors of exit angles from the boundary vertex

If we sequentially change the interval parameters to the opposite ones in the order of its increasing (decreasing), it will be possible to get all the boundary vertices (Figure 2.6) from which the required minimal edge route consists.

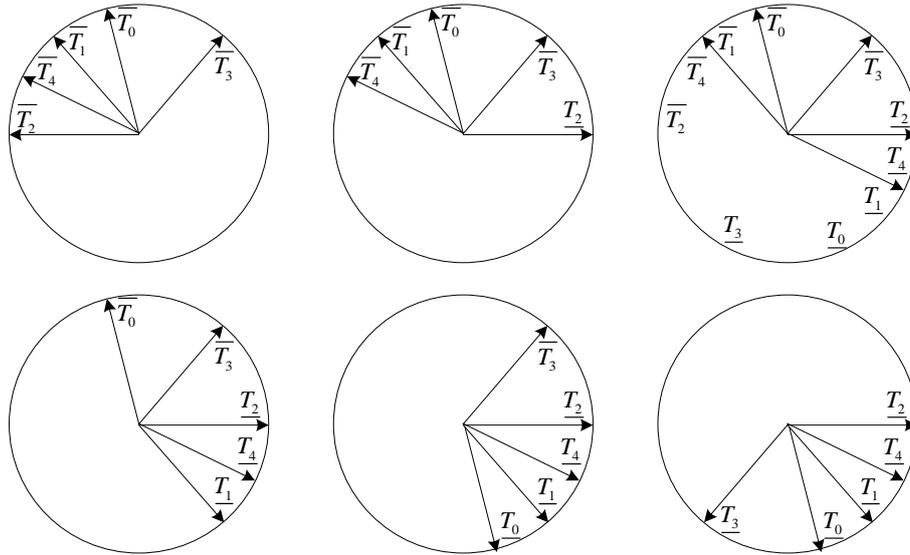


Figure 2.6 - The vectors of exit angles from the boundary vertices

The algorithm for determining the boundaries of the areas of localization of the roots of a polynomial with interval parameters involves the following stages:

1. Transforming the characteristic polynomial of the system to the form (2.3).
2. Determination of the coordinates of the zeros of the characteristic polynomial.
3. Formation of a sequence of the coordinates of the zeros of the characteristic polynomial in the order of increasing (decreasing).
4. Formation of the vertex with the maximum values of intervals of parameters written in order of increasing (decreasing) of zeros.
5. Formation the edge route by successively changing the boundary values of the parameter intervals in the order described in item 4.
6. Display the edge route on the root plane.