

Министерство образования и науки Российской Федерации
МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ
(государственный университет)
ФАКУЛЬТЕТ АЭРОФИЗИКИ И КОСМИЧЕСКИХ
ИССЛЕДОВАНИЙ

**Дистанционное управление стендом при
проведении лабораторной работы
по управлению ориентацией
макета микроспутника**

Выпускная магистерская квалификационная работа
студента 636 группы
Нуждина Дмитрия Олеговича

Научный руководитель
д.ф-м.н., Овчинников М.Ю.

Научный консультант
аспирант 2 года, Иванов Д.С.

г. Москва

2012

Содержание

_Тос327365012

| | |
|--|-----------|
| СПИСОК УСЛОВНЫХ СОКРАЩЕНИЙ..... | 4 |
| ВВЕДЕНИЕ..... | 5 |
| 1 МЕТОДЫ И СРЕДСТВА ПОСТРОЕНИЯ И ИССЛЕДОВАНИЯ ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМ ДИСТАНЦИОННОГО УПРАВЛЕНИЯ УЧЕБНЫМ И НАУЧНЫМ ЭКСПЕРИМЕНТОМ | 9 |
| 1.1 Современные тенденции в области информационно-телекоммуникационных технологий..... | 9 |
| 1.2 Обзор существующих систем дистанционного управления учебными и научными экспериментами | 10 |
| 1.2.1 AIM- Lab (Automated Internet Measurement Laboratory) и LAB-on-WEB | 11 |
| 1.2.2 PEARL (Practical Experimentation by Accessible Remote Learning) | 15 |
| 1.2.3 REMOT (Remote Experiment Monitoring and control)..... | 19 |
| 1.2.4 uCV-Lab | 20 |
| 1.2.5 Distant Lab 1.0..... | 22 |
| 2 ЛАБОРАТОРНЫЕ РАБОТЫ С ИСПОЛЬЗОВАНИЕМ МАКЕТА СИСТЕМЫ ОРИЕНТАЦИИ | 25 |
| 2.1 Модели движения макета под действием лопастных двигателей | 27 |
| 2.1.1 Модель на основе закона Бернулли | 28 |
| 2.1.2 Модель на основе закона сохранения импульса | 30 |
| 2.1.3 Свободное движение макета | 31 |
| 2.1.4 Определение параметров моделей | 32 |
| 2.2 Алгоритмы управления ориентацией | 33 |
| 2.2.1 Оптимальное управление | 33 |

| | | |
|----------|--|-----------|
| 2.2.2 | Пропорционально-дифференциальный регулятор | 41 |
| 2.2.3 | Характеристики алгоритмов управления | 44 |
| 3 | СИСТЕМА ДЛЯ ПРОВЕДЕНИЯ ЛАБОРАТОРНЫХ РАБОТ | 46 |
| 3.1 | Описание стенда | 46 |
| 3.2 | Программное обеспечение макета | 47 |
| 3.2.1 | Программа на макете (серверная часть) | 47 |
| 3.2.2 | Пользовательская программа (клиентская часть) | 49 |
| 3.2.3 | Интерфейс программы | 49 |
| 4 | ПРОВЕДЕНИЕ ЛАБОРАТОРНЫХ РАБОТ | 52 |
| 4.1 | Порядок выполнения лабораторных работ | 52 |
| 4.1.1 | Общие действия при проведении локальной лабораторной работы | 52 |
| 4.1.2 | Общие действия при проведении удаленной лабораторной работы | 52 |
| 4.1.3 | Определение параметров моделей лопастных двигателей ориентации | 53 |
| 4.1.4 | Исследование алгоритмов управления ориентацией | 54 |
| 4.2 | Результаты проведения лабораторной работы "Определение параметров лопастных двигателей ориентации" | 55 |
| 4.3 | Результаты проведения лабораторной работы: "Исследование алгоритмов управления ориентацией" | 58 |
| | ЗАКЛЮЧЕНИЕ | 62 |
| | СПИСОК ЛИТЕРАТУР | 63 |
| | ПРИЛОЖЕНИЕ А | 65 |

Список условных сокращений

| | |
|-------------|--|
| АИС | Автоматизированная измерительная система |
| БИУЭ | Базовый информационно-управляющий элемент |
| ВЛР | Виртуальная лабораторная работа |
| ВМ | Виртуальная модель |
| ГС | Главный сервер |
| ДЛ | Дистанционная лаборатория |
| ИЗ | Источник запросов |
| ИПВ | Источник повторных запросов |
| ИС | Измерительный сервер |
| ЛВС | Локальная вычислительная сеть |
| МАИС | Мониторинг активности измерительных серверов |
| ОУ | Объект управления |
| ПВЗ | Поток входных запросов |
| ПОЗ | Поток обслуженных запросов |
| ПО | Программное обеспечение |
| ПД | Пропорционально-дифференциальный |
| СДУ | Система дистанционного управления |
| СМО | Система массового обслуживания |
| УП | Удаленный пользователь |

Введение

Современный уровень развития телекоммуникационных технологий и средств предоставляет широкие возможности, связанные с передачей данных на большие расстояния, создание распределенных приложений, работающих через сеть, обмен аудио и видео информацией и многое другое. Подобные достижения в последнее десятилетие стали возможными благодаря интеграции компьютерных систем с сетевыми технологиями.

Настоящее время характеризуется стремительным появлением и развитием новых информационных технологий. Одной из таких новых и революционных технологий является технология виртуальных приборов, позволяющая создавать системы измерения, управления и диагностики различного назначения практически любой производительности и сложности. Суть этой технологии состоит в том, что измерительная и управляющая часть приборов и систем реализуется на аппаратной основе (устройств ввода-вывода аналоговых и цифровых сигналов), а их функциональная часть и пользовательский интерфейс - программными способами.

Преимущество и эффективность виртуальных измерительных технологий состоит в возможности программным путем, опираясь на мощь современной компьютерной техники, создавать разнообразные приборы, измерительные системы и программно-аппаратные комплексы, легко перестраивать их к изменяющимся требованиям, уменьшить материальные затраты и время на разработку. При этом создаваемая измерительная система может быть оптимальным образом адаптирована для решения поставленных задач с учетом их особенностей.

Использование виртуальных измерительных технологий в современных автоматизированных измерительных системах является устойчивой мировой тенденцией последних лет. Об этом свидетельствует огромное количество разработок, а также множество зарубежных и отечественных публикаций, посвященных решениям задач в области автоматизации измерений, контроля и управления техническими и технологическими системами.

При использовании виртуальных измерительных технологий ЭВМ становится неотъемлемым компонентом автоматизированных измерительных и управляющих систем. Это дает возможность аппаратно-программного совмещения измерительных систем с телекоммуникационными сетями и обеспечения дистанционного доступа к измерительному и управляющему оборудованию. Подобная интеграция двух современных информационных технологий, а именно, телекоммуникационных сетевых технологий и технологии виртуальных приборов, качественно и количественно расширяет функциональные возможности систем, построенных на их основе. Позволяет связывать в единую систему большое число различ-

ных измерительных и управляющих устройств, удаленных друг от друга на большие расстояния, а также строить системы дистанционного управления (СДУ) различного назначения.

Весьма важным является продвижение дистанционных технологий в лабораторные практикумы и в учебный эксперимент как с целью повышения эффективности, так и снижения материальных затрат на обучение в сфере инженерного и общепедагогического образования. При этом достигаются следующие принципиальные преимущества дистанционной учебной лаборатории:

- *круглосуточная автоматическая работа;*
- *индивидуализация и повышение качества обучения;*
- *доступность дистанционной лаборатории из любой географической точки.*

В настоящей работе реализована система ДУ на примере стенда, имеющегося на кафедре теоретической механики МФТИ (ГУ) и разработанного в ИПМ им.М.В.Келдыша РАН. Разработан и внедрен курс лабораторных работ по управлению макетом системы управления малого спутника. Курс апробирован в рамках технического курса по выбору "Управление и динамика сложных информационно-механических систем".

Целью квалификационной работы является разработка системы дистанционного управления физическим экспериментом на основе аппаратно- программно совмещения телекоммуникационных и измерительно-управляющих систем, повышающих доступность к образовательным ресурсам и расширяющих образовательное пространство ВУЗов.

Основная задача научного исследования - создание системы многопользовательского доступа и дистанционного управления физическим экспериментом через телекоммуникационные сети общего пользования (локальные сети, Интернет).

Решаемые задачи:

- разработка и обоснование структуры системы дистанционного управления, совмещающей телекоммуникационные и виртуальные измерительные технологии;
- разработка алгоритмического, программного и аппаратного обеспечения системы дистанционного управления;
- разработка обучающих лабораторных работ по определению характеристик моделей по результатам калибровочных экспериментов с помощью метода наименьших квадратов и по исследованию алгоритмов управления одноосной ориентацией: оптимального по быстродействию алгоритма управления и алгоритма на основе пропорционально-дифференциального регулятора.

Методы исследований

При решении поставленных задач использовались методы теории систем массового обслуживания, теории автоматического управления, методы компьютерного моделирования,

виртуальных измерительных технологий, а также методы экспериментальных исследований в телекоммуникационных сетях.

Научная новизна работы

- предложена и разработана структура системы дистанционного управления, на основе аппаратно-программного совмещения телекоммуникационной и измерительно-управляющей системы;

- разработана и создана типовая дистанционная автоматизированная лаборатория, а также ее аппаратно-программное обеспечение с поддержкой многопользовательского распределенного доступа через телекоммуникационные сети (Интернет, ЛВС).

Практическая ценность работы

Созданная система дистанционного управления позволяет обеспечить доступ удаленных пользователей к реальным учебным и научным лабораторным ресурсам в многопользовательском режиме с разделением времени через телекоммуникационные сети (Интернет, ЛВС).

Апробация работы

Основные положения и промежуточные результаты работы докладывались, обсуждались и представлялись на следующих конференциях:

- Вторая всероссийская научно-техническая конференция «Современные проблемы определения ориентации и навигации космических аппаратов», г. Таруса, 13-17 сентября, 2010;

- Taiwan-Russian bilateral symposium on problems in advanced mechanics, Inst. of mechanics, LMSU, Moscow, Russia, Sept. 05-11, 2010;

- Всероссийская конференция с элементами научной школы для молодежи «Специфика формирования сети научно-образовательных центров. Сетевое взаимодействие молодых исследователей в рамках информационного обмена НОЦ», Москва, 12 ноября, 2010;

- 53-я научная конференция МФТИ «Современные проблемы фундаментальных и прикладных наук», Долгопрудный, 24-29 ноября, 2010;

- X всероссийский съезд по фундаментальным проблемам теоретической и прикладной механики, Н. Новгород, 24-30 августа 2011;

- 54-я научная конференция МФТИ «Проблемы фундаментальных и прикладных естественных и технических наук в современном информационном обществе», Долгопрудный, 25-26 ноября, 2011.

Основные научные и практические результаты работы опубликованы в 13 работах, в том числе одна статья из списка ВАК, 6 - в трудах конференций, 5 - в сборниках тезисов докладов, 1 - препринт ИПМ им.М.В.Келдыша (см. Приложение А).

1 Методы и средства построения и исследования телекоммуникационных систем дистанционного управления учебным и научным экспериментом

1.1 Современные тенденции в области информационно-телекоммуникационных технологий

В настоящее время происходит достаточно стремительное появление и развитие новых технологий. Особенно ярко это проявляется в области высоких информационных технологий. Одной из таких новых технологий является технология виртуальных приборов, позволяющая создавать системы измерения, управления и диагностики различного назначения практически любой производительности, включая математическое моделирование и тестирование этих систем [1]. Суть этой технологии состоит в том, что измерительная часть приборов и систем реализуется на основе устройства ввода-вывода аналоговых и цифровых сигналов, а функциональная и пользовательский интерфейс - программно. В связи с этим появляется понятие виртуального измерительного прибора.

Преимущество виртуальных измерительных технологий состоит в возможности программным путем, опираясь на мощь современной компьютерной техники, создавать разнообразные приборы, измерительные системы и программно аппаратные комплексы, легко их адаптировать к изменяющимся требованиям, уменьшить затраты и время на разработку. При этом создаваемая измерительная система может быть достаточно точно адаптирована для решения поставленных задач с учетом всех их особенностей. Также, помимо решения задач измерений/управления, сразу же могут решаться и задачи обработки измерительной информации и представления результатов (автоматизированное формирование отчетов), что также позволяет сэкономить значительное количество времени.

Использование этой технологии при практической реализации современных автоматизированных измерительных систем (АИС) является на сегодняшний день мировой тенденцией. Об этом говорит большое количество отечественных [2][3][4][5] и зарубежных [6][7][8][9] публикаций, посвященных решениям задач в области автоматизации.

Ключевую роль в развитии виртуальных измерительных технологий играет фирма National Instruments [10], создающая программные и аппаратные средства, применяемые в сфере автоматизации технологических процессов, а также учебных и научных экспериментах.

При использовании виртуальных измерительных технологий ПЭВМ становится неотъемлемым компонентом автоматизированной измерительной системы. Это дает возможность интеграции измерительных систем с телекоммуникационными сетями и обеспечения дистанционного доступа к измерительному оборудованию, что было принципиально неосуществимо при построении измерительных систем на основе традиционных измерительных приборов. При достижении такой интеграции можно говорить о системе дистанционного управления (СДУ).

Благодаря указанным преимуществам, помимо решения промышленных задач автоматизации, виртуальные измерительные технологии находят широкое применение в сфере традиционного и дистанционного (обзору СДУ посвящен раздел 1.2) инженерного образования. В работе [3] отмечаются следующие преимущества, получаемые за счет использования в учебном процессе виртуальных лабораторий.

- За счет круглосуточной автоматизированной работы дистанционной лаборатории (без преподавателя и лаборанта, нет необходимости в посадочных местах и столах для студентов и т.п.) достигается экономия учебных площадей, оптимизируется учебное расписание, достигается экономия средств за счет сокращения часов, выделенных на проведение занятий (до 30-40% фонда заработной платы).

- Индивидуализируется и повышается качество обучения, так как студент вынужден выполнять лабораторную работу индивидуально. Появляются качественно новые возможности для самостоятельной работы студентов.

- За счет расширения пространства обучения, не локализованного пределами какого-либо здания, виртуальная лаборатория равнодоступна в любое время для всех кафедр университета, его многочисленных филиалов, а также индивидуально для студентов имеющих домашний компьютер и выход в Интернет, что позволяет исключить дублирование лабораторных работ (сейчас часто имеет место неоправданное дублирование лаб. работ на родственных кафедрах на разных факультетах, и даже в пределах одного факультета).

- Появляется возможность коммерческого дистанционного использования виртуальных лабораторий и уникальных установок другими ВУЗами.

1.2 Обзор существующих систем дистанционного управления учебными и научными экспериментами

В силу современного уровня развития измерительных, компьютерных и телекоммуникационных технологий и средств, а также их взаимной интеграции, управление реальными физическими объектами дистанционно через телекоммуникационные сети общего пользования (Интернет, ЛВС предприятий) является на сегодняшний день мировой тенденцией. Достаточно большое количество работ в этой области посвящено созданию дистанционных

практикумов по инженерным общетехническим специальностям, являющихся неотъемлемой частью дистанционного инженерного образования.

1.2.1 AIM- Lab (Automated Internet Measurement Laboratory) и LAB-on-WEB

В работе [11] представлены результаты совместных проектов, норвежского университета Науки и Технологии (NTNU) и политехнического института города Хьеллер. Первая версия их дистанционной лаборатории *AIM- Lab (Automated Internet Measurement Laboratory)* предназначена для исследования характеристик полевых транзисторов. В данной работе измерительная система построена на базе GPIB-устройств, подключаемых к серверу дистанционной измерительной системы. Используются следующие измерительные устройства:

- измерительная станция крейтового типа HP 4142B DC Source/Monitor в составе с тремя измерительными модулями (два модуля ввода/вывода HP 41421B; источник ОБ GNDU);
- матричный коммутатор HP 34970 A;
- осциллограф Tektronix TDS 3052 до 500 МГц. Структурная схема данной системы представлена на рис. 1.1.

Дистанционная лаборатория второй версии (LAB-on-WEB) ориентирована на исследование параметров и характеристик различных радиоэлектронных схем и элементов, реализованных посредством специализированной интегральной схемы Alpha-chip. Данная микросхема была разработана в рамках программы обмена между университетами Европы и Латинской Америки. Она содержит такие элементы как диоды, конденсаторы, транзисторы, инверторы и т.д.

Для обеих версий представленных систем предусмотрена возможность реализации пользовательского интерфейса посредством отдельного клиентского приложения, либо в стандартном интернет-обозревателе посредством загрузки и исполнения соответствующего Java-скрипта. В последнем случае, представление результатов измерения на стороне удаленного пользователя осуществляется посредством C01-приложения.

Реализация клиентской части в виде отдельного приложения, а также CGI- приложения на сервере, осуществлялась в среде LabVIEW [10].

К недостаткам данной работы можно отнести ограниченную гибкость измерительной системы, связанную с невозможностью задействования дополнительных лабораторий не прибегая к изменению конфигурации и доработке ПО измерительной системы. Также не приводятся никаких данных о средствах и способах реализации многопользовательского режима работы системы.

В работе [6] представлена система дистанционного управления, разработанная в рамках совместного проекта университетов г. Падовы и г. Турина (Италия). Характерным для этой системы является возможность многопользовательского режима работы, определяющего ее пригодность к полноценному использованию как в научно-технической, так и в образовательной сфере. Эта система представляет собой многоуровневую структуру, как показано на Рис. 1.1. Многопользовательский режим обеспечивается за счет организации очереди запросов и наличия механизмов ее обработки. Использование TCP/IP протокола для передачи данных между сервером системы и УП, обеспечивает возможность расположения последних в различных географических точках независимо от расстояния.



Рис. 1.1 Многоуровневая структура системы дистанционного управления, разработанная университетами г. Падовы и г. Турина (Италия)

Измерительная часть системы обеспечивает взаимодействие с измерительными устройствами через шину PC1, а также используя различные протоколы, такие как IEEE 488, RS 232 и т.д., что обеспечивает ее большую гибкость по сравнению с системой Норвежского университета, рассмотренной в работах [11].

Использование таких протоколов как IEEE 488 (RS 485) и RS 232 позволяет создать распределенную измерительную систему, однако согласно техническим характеристикам этих протоколов, площадь распределения измерительной системы будет достаточно малой по отношению к площадям университета. В силу этого становится невозможным задействование в системе всех необходимых лабораторий ВУЗа.

В работе [5] представлена система дистанционных лабораторий *Cyber Lab*, разработанная в Стендфордском университете США. Для тестирования и исследования эффективности данной системы в техническом и образовательном планах была использована «пилотная» лабораторная работа по исследованию оптических явлений. Экспериментальная часть данной лабораторной работы реализована на основе специализированной лазерной установки.

Большое внимание в работе уделяется методическому обеспечению лабораторных практикумов, контролю уровня знаний студентов перед допуском к выполнению работы, а также вопросам индивидуализации их работы. Одним из концептуальных подходов, использованным авторами при разработке и создании *Cyber Lab*, было использование возможно-

стей мультимедийных и web-технологий, что в сочетании с реальным экспериментированием могло бы в значительной степени повысить эффективность и качество обучения. На рис. 1.2 представлен пользовательский интерфейс созданной дистанционной лаборатории системы *Cyber Lab*, доступный студентам через Интернет.

В работе [12] представлена система дистанционного управления *Remote Lab*, созданная при сотрудничестве Центра инновационного дистанционного образования университета г. Атлантики и колледжа *Loyola* г. Мэрилэнда (США). Система *Remote Lab* ориентирована главным образом на использование в дистанционном инженерном образовании для прохождения лабораторных практикумов.



Рис. 1.2 Пользовательский интерфейс дистанционной лаборатории системы *Cyber Lab* с указанием составных компонентов

Особенностью системы *Remote Lab* является реализация ПО серверной и клиентской частей посредством языка программирования Java. Это обеспечивает гибкость работы клиентского ПО: для работы удаленного пользователя необходим только стандартный Интернет-обозреватель, осуществляющий загрузку (с сервера системы) и исполнение Java-апплета, реализующего пользовательский интерфейс для дистанционного управления лабораторным экспериментом. Кроме этого, использование Java обеспечивает независимость функциональности клиентского ПО от операционной системы.

Для взаимодействия «клиент-сервер» была использована технология RM1 [15], которая в отличие от, ставших традиционными, CGI-решений [46] обеспечивает более оптимальную работу сервера и позволяет снизить загрузку сети. К преимуществам системы *Remote Lab* также можно отнести возможность работы с различными типами баз данных, таких как Microsoft ODBL, Oracle, My SQL и др. База данных в системе *Remote Lab* используется для

хранения учетных записей студентов. Учетные записи необходимы для допуска студентов к выполнению лабораторных экспериментов, а также накопления статистики их работы.

Функциональная схема системы дистанционного управления *Remote Lab* представлена на Рис. 1.3.

ПО сервера данной системы включает в себя несколько блоков Remote Object, являющихся программными модулями, взаимодействующими с соответствующими физическими объектами. Блоки Stab служат для приема/передачи данных между сервером и удаленным пользователем, в то время как блоки Skeleton - между сервером и программным модулем, управляющим экспериментом (Remote Object).

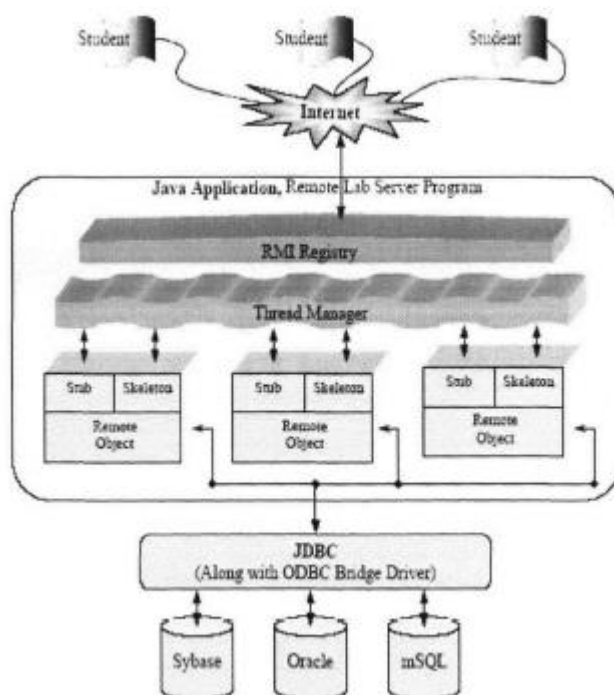


Рис. 1.3 Функциональная схема системы дистанционного управления *Remote Lab*

К недостаткам рассмотренной системы можно отнести потенциально невысокую функциональность клиентского ПО, что обусловлено спецификой Java-апплетов.

В работе [13], выполненной в Департаменте компьютерных наук университета г. Берлина, представлена система, имеющая тоже название, что и в предыдущем рассмотренном случае - *RemoteLab*, предназначенная для дистанционного управления движением робота *Khepera*. Эта система ориентирована на использование в инженерном дистанционном образовании. При ее реализации особое внимание было уделено средствам защиты от некорректных действий студента, ведущих к сбоям либо выходу экспериментального оборудования из строя.

Управление лабораторным экспериментом, а также обеспечение безопасного режима работы оборудования осуществляется посредством специализированной системы *Simplex*,

разработанной ранее. Для передачи команд управления от удаленного студента к серверу используется CORBA-архитектура, служащая для создания распределенных программных приложений.

Функциональная схема системы *RemoteLab* (г. Берлин) представлена на Рис. 1.4.

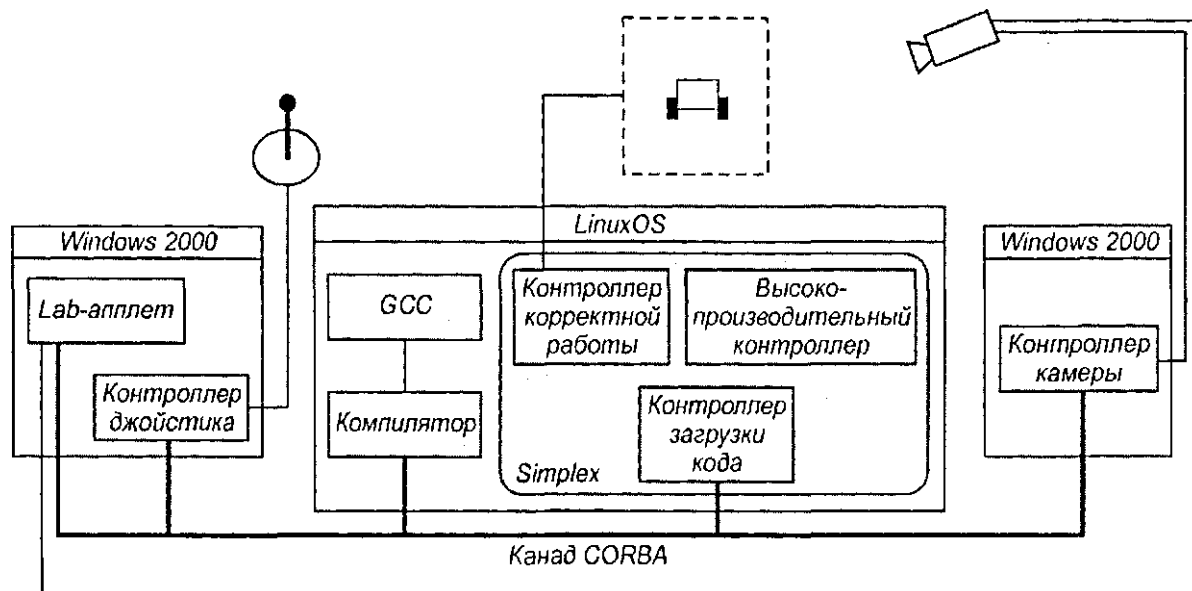


Рис. 1.4 Функциональная схема системы дистанционного управления движением робота *RemoteLab* (г. Берлин)

Перед началом работы удаленный студент осуществляет загрузку собственного программного алгоритма, используемого для управления роботом. Формируемые посредством джойстика команды управления, передаются по каналу *CORBA* в модуль *Simplex*. Данный модуль помимо компонента, управляющего движением робота (по командам удаленного пользователя) содержит компонент защиты, выполняющий слежение за процессом управления, не допуская возникновения критических режимов в работе оборудования. Результат управления фиксируется видео камерой и передается удаленному пользователю по HTTP-протоколу.

Согласно рассмотренным особенностям работы системы, выполнение дистанционных экспериментов может осуществляться только в однопользовательском режиме. Данный факт исключает возможность использования системы *RemoteLab* в лабораторных курсах, предполагающих наличие достаточно большого количества студентов.

1.2.2 PEARL (Practical Experimentation by Accessible Remote Learning)

В работе [14], выполненной в колледже Троицы (Trinity College) г. Дублин (Ирландия), представлена распределенная система дистанционного управления *PEARL* (Practical Experimentation by Accessible Remote Learning), предназначенная для выполнения реальных экспериментов в дистанционных лабораторных практикумах. В системе *PEARL*, также как и

в ряде рассмотренных ранее случаях, в качестве клиентского ПО используется стандартный Интернет-обозреватель, посредством которого загружается и исполняется соответствующий Java-апплет. Функциональная схема распределенной системы дистанционного управления *PEARL* представлена на Рис. 1.5.

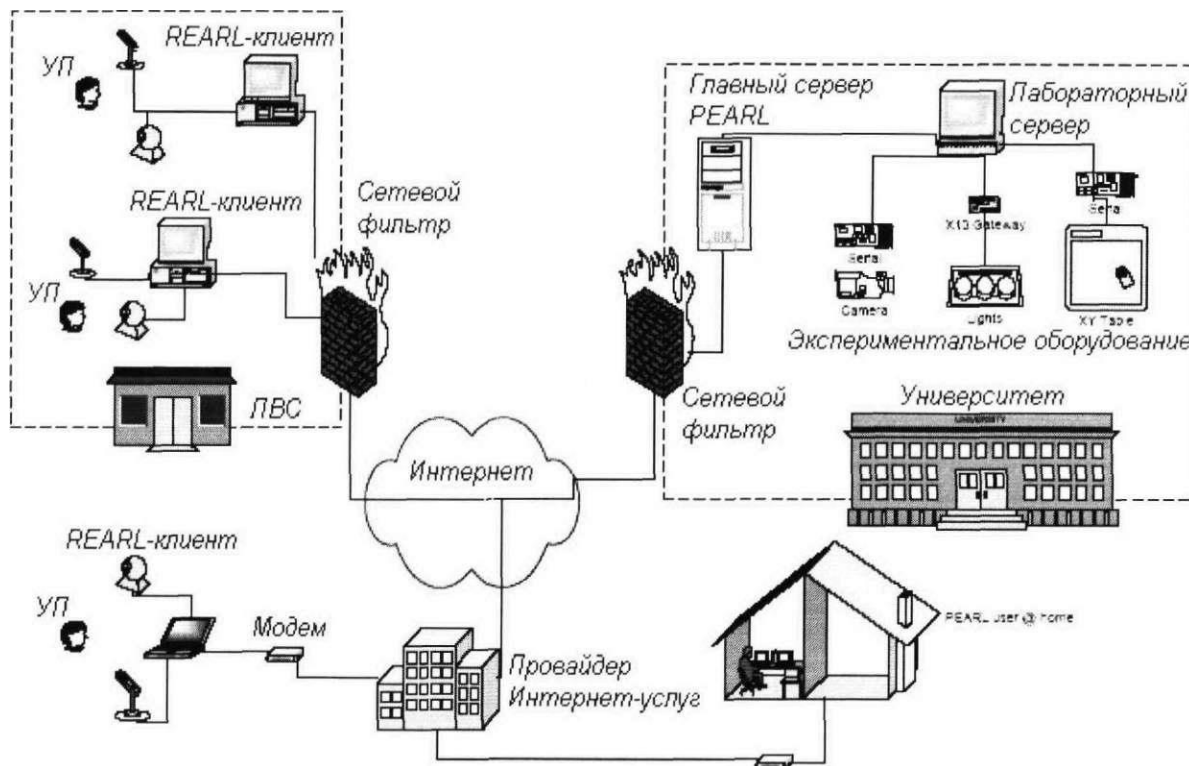


Рис. 1.5 Функциональная схема распределенной системы дистанционного управления *PEARL*

Дистанционная лаборатория (ДЛ) системы *PEARL* строится на основе лабораторного сервера, контролирующего экспериментальное оборудование. Для контроля доступа УП к ресурсам ДЛ используется отдельная серверная машина, выполняющая роль WEB-сервера. Использование в СДУ отдельной машины главного сервера обеспечивает возможность создания распределенной измерительной системы, состоящей из нескольких ДЛ. Кроме этого достигается структурирование и централизация системы, а также разделение выполняемых функций между соответствующими узлами. Функции измерения и управления выполняются лабораторными серверами, а функции контроля доступа УП, а также функции информационной и методической поддержки - главным сервером (ГС).

Организация взаимодействия ГС с ДЛ осуществляется на основании архитектуры CORBA. В отличие от системы *Remote Lab* (г. Берлин, Германия) где канал CORBA был использован для организации взаимодействия УП с непосредственно ДЛ, в системе *PEARL* CORBA используется лишь на участке ГС - ДЛ. При этом передача данных между УП и ГС осуществляется по протоколу HTTP. Такое разделение сетевой архитектуры системы на

уровни, использующие различные протоколы (протокол CORBA - участок ГС-ЛС и протокол HTTP - участок УП-ГС) позволяет решить проблему, возникающую при передаче данных в случае наличия в сети сетевых фильтров (firewall) и прокси-серверов, а также участков с малой пропускной способностью.

Первая дистанционная лабораторная работа, запущенная на основе системы *PEARL*, посвящена изучению работы системы технического зрения. Согласно Рис. 1.6, основными компонентами лабораторной установки являются: видео-камера; заданное количество независимых источников освещения; стол с XY-системой координат, служащий для размещения исследуемых объектов.



Рис. 1.6 Структурная схема лабораторной установки дистанционного лабораторного практикума, реализованного в системе PEARL

Один из экспериментов, выполняемых студентами, заключается в вычислении чувствительности каналов красного, зеленого и синего цветов камеры. Данный эксперимент состоит из следующих шагов:

- включение источников света;
- настройка фокуса и зума камеры;
- съемка изображения тестовой цветовой таблицы;
- отключение одного из источников освещения и съемка изображения;
- вычисление чувствительности каналов цветопередачи.

Также выполняются эксперименты по вычислению фокусного расстояния камеры и поиску радиокомпонентов на печатной плате по заданному шаблону.

Система дистанционного управления *RVLabX* [15] разработанная в университете LICEF-CIRTA г.Монреалья (Канада), так же как и ряд систем рассмотренных выше, ориентирована на использование в сфере дистанционного инженерного образования.

Отличительной особенностью системы дистанционного управления *RVLabX* является возможность создания удаленными пользователями (как преподавателями так и студентами)

собственных пользовательских интерфейсов, служащих для выполнения дистанционных экспериментов. Ключевыми компонентами *RVLabX*, обеспечивающими основную функциональность данной системы, являются:

- *Interface Designer* - приложение разработки пользовательских интерфейсов;
- *PolymorphiX* - серверный компонент для реализации созданного ПИ как компонента СДУ;
- средства администрирования.

Разработка пользовательского интерфейса посредством приложения *Interface Designer* осуществляется путем размещения на наборном поле соответствующих средств управления и индикации, извлекаемых из палитр. Предусмотрена возможность выбора места расположения и размеров каждого из объектов создаваемой лицевой панели, а также их свойств, связанных с форматом задаваемых/отображаемых данных.

По окончании процесса разработки формируется файл в формате XML, содержащий полное описание созданного ПИ. Данный файл передается в web- сервер системы. С помощью компонента *PolymorphiX* осуществляется представление созданного ПИ (по данным XML-файла) в виде ActiveX- объекта, встроенного в HTML-страницу. Функциональная схема системы *R VLabX* представлена на Рис. 1.7.

В системе *RVLabX*, также как и в системе PEARL рассмотренной ранее, предусмотрено наличие отдельного узла, реализующего функции лабораторного сервера, управляющего экспериментальным оборудованием. Взаимодействие ГС с лабораторными серверами осуществляется через Data Socket-сервер.

Также в системе *RVLabX* предусмотрены средства администрирования, служащие для контроля действий УП.

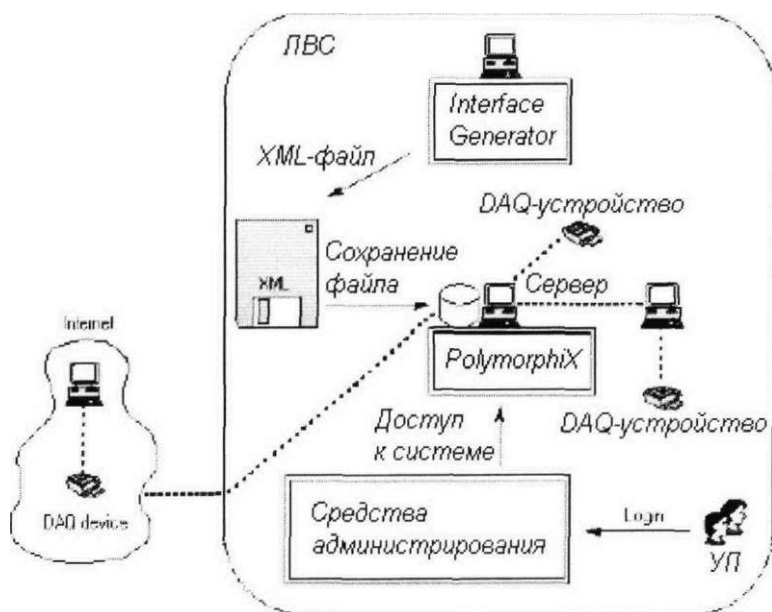


Рис. 1.7 Функциональная схема системы R VLabX

1.2.3 REMOT (Remote Experiment Monitoring and control)

В отличие от рассмотренных ранее СДУ, ориентированных главным образом на применение в сфере инженерного образования, в работе [16] представлена система *REMOT* (Remote Experiment Monitoring and control) ориентированная на обеспечение дистанционного управления научными экспериментами. Данная система создана в рамках программы *TELEMATRIX* Европейского комитета DG XIII.

Система *REMOT*, как и каждая из рассмотренных СДУ, построена по принципу «клиент-сервер». Для реализации распределенной архитектуры системы был применен стандарт *CORBA*.

С целью исследования и демонстрации возможностей, разработанная система была использована для дистанционного управления экспериментальным оборудованием двух независимых научных лабораторий: лаборатория астрономии (управление телескопом) [96]; лаборатория физики плазмы, содержащая дорогостоящее экспериментальное оборудование. Структурная схема системы *REMOT* представлена на Рис. 1.8.

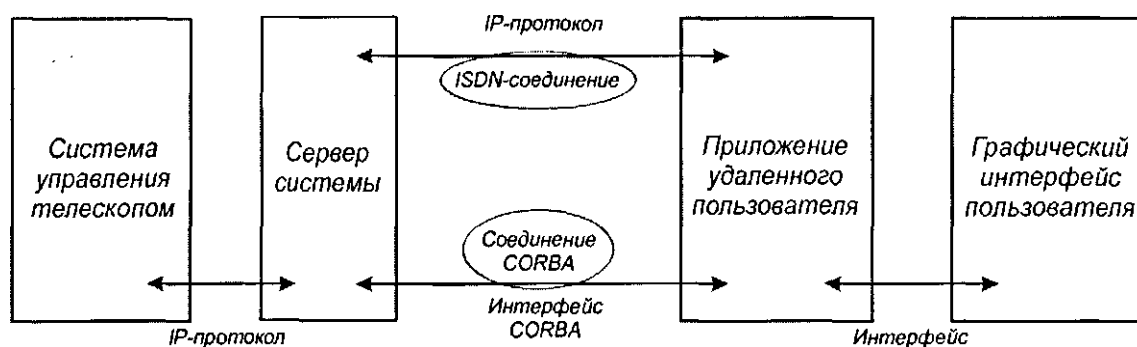


Рис. 1.8 Структурная схема СДУ научными экспериментами *REMOT*

В связи с ориентированностью данной системы на использование в научной сфере, обеспечиваются лишь средства дистанционного доступа и управления экспериментом, а также представление результатов экспериментов. В работе показаны основные принципы построения системы, однако не приводятся результаты реализации данного проекта, а также описание автоматизированных измерительных систем, применяемых на локальном уровне в лабораториях.

1.2.4 uCV-Lab

В работе [2] представлена дистанционная автоматизированная web- лаборатория *uCV-Lab* для изучения и исследования микроконтроллеров и сигнальных процессоров. Базовая часть автоматизированной измерительной системы данной web-лаборатории строится по, ставшей на сегодняшний день классической, схеме ПК + устройство ввода/вывода. В сочетании со специализированным оценочным модулем обеспечивается возможность проектирования программного обеспечения, а также контроля состояния и исследования рабочих процессов микроконтроллерных устройств и построенных на их основе программно-аппаратных средств.

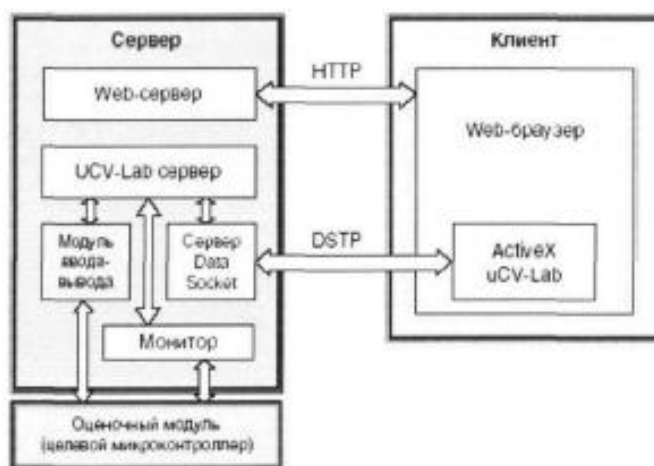


Рис. 1.8. Структурная схема web-лаборатории на основе сервера *uCV-Lab*

Клиентское ПО реализовано как ActiveX-объект, встроенный в *html*- страницу. Загрузка этого ПО осуществляется через стандартный Интернет- обозреватель по протоколу HTTP. Обмен данными между серверным и

клиентским ПО осуществляется по протоколу DSTP. Компонент «монитор», входящий в состав серверного ПО, выполняет функции управления оценочным модулем.

На основании представленной структуры web-лаборатории *uCV-Lab*, создан центр коллективного пользования, представляющий собой сеть дистанционных лабораторий.

Следует отметить достаточно тщательную проработку вопросов, касающихся методического обеспечения дистанционных работ. Так, например, согласно [2], при подключении к главному серверу центра (диспетчеру) студент выбирает нужную дисциплину, после чего загружается страница с изображением лаборатории со стеллажами, где размещены все доступные для работы измерительные приборы, а также объекты исследований, как показано на рис. 1.12. После выбора требуемой дисциплины, посредством указанной выше страницы, выполняется компоновка рабочего места, путем перемещения на лабораторный стол необходимого оборудования.

К недостаткам системы можно отнести ограничения многопользовательского режима работы, связанные с достаточно небольшим количеством (учитывая востребованность представленных лабораторных дисциплин) удаленных пользователей (не более восьми), способных одновременно получать доступ к ресурсам лабораторий.

1.2.5 Distant Lab 1.0

Одним из наиболее интересных решений представляется система, созданная в Казанском ГУ [1], рис. 1.9. Центральным компонентом системы является главный сервер, обеспечивающий доступ УП к ресурсам ДЛ, а также управляющий работой распределенной измерительной системы, построенной на основе дистанционных лабораторий. Взаимодействие главного сервера с измерительными средствами ДЛ осуществляется по ЛВС. Передача данных от установок к серверу и далее клиенту использует в работе Data-Socket сервер. Передатчик запросов ГС и передатчик результатов ИС осуществляют запись сформированных для передачи данных в соответствующие разделы Data Socket - сервера. Приемники результатов и запросов соответственно ГС и ИС необходимы для чтения данных, публикуемых в соответствующих разделах Data Socket – сервера, и передачу их клиентской стороне.

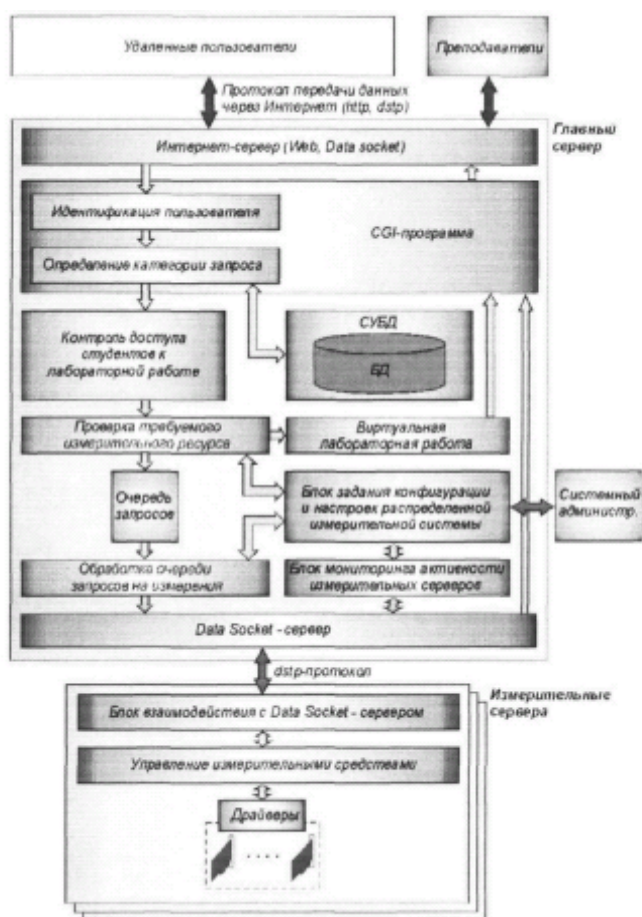


Рис. 1.9 Схема СДУ

Для выполнения работ дистанционных лабораторных практикумов используется клиентское ПО Distant Lab 1.0. Особенностью среды Distant Lab 1.0 является то, что количество доступных лабораторных работ не является фиксированным – непосредственно «тело» лабораторной работы (ее функциональность, математические модели исследуемых схем и устройств, их графические изображения и пр.) не является внедренным в программный код

среды Distant Lab 1.0, а реализуется посредством отдельной Jlb-библиотеки, которая инициализируется в Distant Lab 1.0 перед началом работы. Данная особенность позволяет значительно более легко увеличивать количество доступных дистанционных работ (каждой лабораторной работе соответствует отдельная Jlb-библиотека), а также, при необходимости, модернизировать ранее созданные, исключая трудоемкие модификации исходной среды Distant Lab 1.0 и, как следствие, обновление ее новой версией. Другой важной особенностью Distant Lab 1.0 является гибкость – расположение на лицевой панели виртуальных электронно-измерительных приборов, таких как многоканальный осциллограф, функциональный генератор и др., вызываемых из специальной библиотеки, их подключение к исследуемой схеме не являются фиксированными и производятся студентом в соответствии с методикой лабораторной работы. Стоит заметить что данная система совмещает в себе средства моделирования с лабораторными установками, которые задействуются по необходимости.

В таблице 1.1 приведены основные данные по рассмотренным системам дистанционного управления учебными и научными экспериментами.

Таблица 1.1 Системы СДУ

| | Название СДУ | Расположение ДЛ | Используемые телекоммуникац. технологии | Ориентированность | Особенности |
|---|---------------------|-----------------------------|---|---|--|
| 1 | <i>AIM-Lab</i> | г. Хьеллер, Норвегия | Средства Java, Lab VIEW, CGI, HTML. | Изучение свойств схем на полевом транзисторе | Возможность создания только одной дистанционной лаборатории вследствие неразделенности серверного и измерительного уровней системы |
| 2 | <i>LAB-on-WEB</i> | г. Хьеллер, Норвегия | Технологии ActiveX/DCOM. | Изучение характеристик и параметров различных радиоэлектронных схем | |
| 3 | <i>без названия</i> | г. Турин, г. Падова, Италия | Протокол TCP/IP. | Представлена общая концепция создания распределенных измерительных систем | |
| 4 | <i>Cyber Lab</i> | г. Стенфорд, США | Описание технологий не приведено, ориентировочно ActiveX, HTTP. | Изучение оптических явлений на основе лазерной установки | Уделяется большое внимание вопросам методической и информационной поддержки, а также контролю знаний удаленных пользователей. |

| | | | | | |
|----|-------------------|---------------------------------|--|---|---|
| 5 | <i>Remote Lab</i> | г. Балтимор, США | RMI, Java, HTTP. | Дистанционная работа с реальными объектами (для дистанционного инженерного образования) | Делается акцент на возможностях использования технологии ИМ1 при построении системы дистанционного управления |
| 6 | <i>RemoteLab</i> | г. Берлин, Германия | HTTP, CORBA. | Управление движением робота | Возможность загрузки на сервер системы собственных алгоритмов управления роботом. Обеспечивается режим защиты оборудования от некорректных действий пользователей |
| 7 | <i>PEARL</i> | г. Дублин, Ирландия | HTTP, Java, CORBA. | Дистанционная работа с реальными объектами (для дистанционного инженерного образования) | Возможность создания набора дистанционных лабораторий ВУЗа. Решение проблем присутствия ргоху- серверов и сетевых фильтров. |
| 8 | <i>RVLabX</i> | г. Монреаль, Канада | HTTP, ActiveX, XML. | Дистанционная работа с реальными объектами (для дистанционного инженерного образования) | Возможность быстрого создания собственного пользовательского интерфейса для дистанционных лаб. практикумов |
| 9 | <i>REMOT</i> | При сотрудничестве ВУЗов Европы | HTTP, CORBA. | Доступ и дистанционное управление к оборудованию обсерватории и лаборатории физики плазмы | Ориентированность только на научные исследования |
| 10 | <i>uCV-Lab</i> | г. Новосибирск, Россия | HTTP, DSTP, ActiveX, средства LabVIEW. | Исследование микропроцессорных систем | Использование измерительной системы, позволяющей работать со сложными устройствами. Возможность увеличения числа дистанционных лабораторий |

| | | | | | |
|----|------------------------|-------------------|----------------------------|--|--|
| 11 | <i>Distant Lab 1.0</i> | Г. Казань, Россия | LabVIEW, HTTP, Data Socket | Дистанционные лаборатории по электронике | Простота увеличения количества работ и числа дистанционных лабораторий |
|----|------------------------|-------------------|----------------------------|--|--|

2 Лабораторные работы с использованием макета системы ориентации

Для управления движением относительно центра масс космического аппарата могут использоваться реактивные двигатели малой тяги. Они устанавливаются на теле аппарата таким образом, чтобы, не влияя на движение центра масс, создавать моменты относительно главных осей тела (осей, в которых тензор инерции имеет диагональный вид). Поэтому, как правило, такие двигатели устанавливаются парами, например, так, как изображено на Рис. 2.1. Для создания положительного момента, например, относительно оси Z одновременно включаются двигатели №2 и 8. Для произвольного поворота аппарата в пространстве производится серия одноосных поворотов. [EQUATION SECTION 2](#)

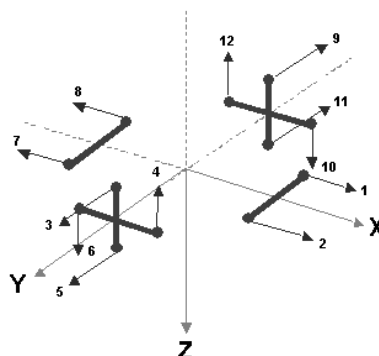


Рис. 2.1. Расположение и направления двигателей на спутнике-инспекторе Mini AERCAM

Система ориентации на двигателях малой тяги имеет ряд преимуществ и недостатков. Основным преимуществом такой системы является то, что величина прикладываемого момента не зависит от ориентации (в отличие от магнитной системы ориентации, например) и может иметь достаточно большую величину. Однако, как правило, модуль вектора тяги остается постоянным во время подачи импульса, то есть возможно только импульсное управление, что накладывает определенные ограничения на максимальную достижимую точность поддержания ориентации, которая зависит от минимальной величины управляющего момента и минимального времени между импульсами, а также от продолжительности импульса последствия. Ещё одним недостатком такой системы ориентации является то, что

для её работы требуется топливо, запас которого ограничивает время активного существования аппарата.

Двигатели малой тяги могут также входить в систему ориентации совместно с другими исполнительными элементами и использоваться в некоторых режимах работы, таких как, например, демпфирование начальной угловой закрутки, закручивание аппарата вокруг некоторой оси или разгрузка кинетического момента маховиков.

В лабораторных условиях трудно создать условия для работы реактивных двигателей, для которых требуется либо безвоздушное пространство (например, для работы плазменных двигателей), либо сжатый газ. Однако в качестве имитации реактивных двигателей могут выступать лопастные двигатели, которые работают в воздушной среде, но выполняют ту же функцию - создают управляющий момент.

Для расчета управляющего момента необходимо иметь правдоподобную и, желательно, простую модель работы актюатора. Чем проще модель - тем проще её реализовать на бортовом компьютере, который имеет ограничения по производительности и объему оперативной памяти. У любой модели есть некоторые параметры, которые определяются с помощью калибровочных экспериментов следующим образом. Воздействие актюатора обуславливает движение управляемого объекта, которое определяется с помощью установленных на нем датчиков. Далее сравнивается траектория движения в фазовом пространстве, полученная экспериментально, с траекторией, полученной интегрированием уравнений движения, в которые включена модель актюатора. С помощью метода наименьших квадратов определяются такие параметры модели, при которых сумма квадратов разницы двух кривых траекторий на сетке точек будет минимальной.

Основными задачами управления ориентацией являются поворот на заданный угол и поддержание заданной ориентации (в общем случае - отслеживание программной траектории). Для решения этих задач могут использоваться различные алгоритмы управления двигателями. Если при маневре поворота дополнительно требуется минимизировать некоторый функционал (например, сэкономить топливо или повернуть за минимальное время), то используется оптимальное управление. Если же на тело воздействуют значительные по величине возмущения, а требуется отслеживать некоторую траекторию, то часто используются алгоритмы, основанные на пропорционально-дифференциальном регуляторе.

Вследствие неидеальности исполнительных органов, воздействия неучтенных в алгоритме возмущений, задержек в исполнительном контуре, неточности измерений датчиков и т.д. система управления в целом неспособна идеально точно выполнять управляющие команды. Поэтому для оценки качества работы системы управления ориентацией можно ис-

пользовать такую величину как точность выполнения маневра ориентации и точность поддержания заданного положения.

В настоящей лабораторной работе используется макет системы ориентации подвижного объекта, подвешенного на струне. На макете установлены лопастные двигатели, с помощью которых производится управление его ориентацией относительно вертикальной оси. При выполнении лабораторных работ преследуются следующие цели:

- на примере определения параметров модели лопастных двигателей приобрести навыки работы с методом наименьших квадратов;
- оценить точность определения полученных параметров;
- провести сравнительный анализ двух моделей лопастных двигателей;
- познакомиться с оптимальным по быстродействию алгоритмом управления и с пропорционально-дифференциальным регулятором;
- исследовать алгоритмы с точки зрения точностных характеристик и сравнить их.
- получить навыки работы в среде MatLab.

2.1 Модели движения макета под действием лопастных двигателей

Рассматривается управляемое вращательное движение твердого тела вокруг вертикали. Тело подвешено на струне, верхний конец которой неподвижно закреплен. Управление его движением относительно центра масс осуществляется с помощью двух лопастных двигателей, реализующих разнонаправленные управляющие моменты вокруг вертикальной оси. Информация о текущей ориентации и угловой скорости вычисляются с помощью измерений оптоволоконного датчика угловой скорости. Тело подвержено линейному по углу поворота моменту от упругости струны и линейному по угловой скорости моменту вязкого трения от сопротивления атмосферы. Предполагается, что лопастной двигатель мгновенно набирает номинальные обороты. Одновременно может работать только один двигатель. Влияние работающего двигателя на движении центра масс тела не учитывается и не рассматривается. Атмосфера считается однородной и неподвижной. Рассмотрим сначала различные модели движения макета под действием лопастных двигателей, а затем модель свободного движения тела на струне.

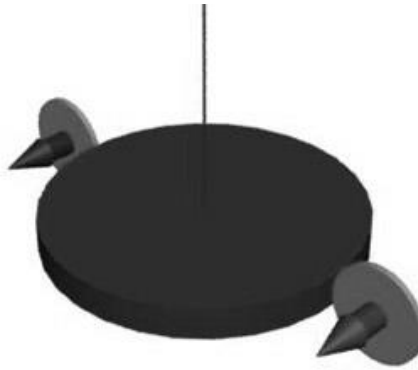


Рис. 2.2 Макет аппарата

2.1.1 Модель на основе закона Бернулли

Получим выражение для управляющего момента, который создается двигателями [17]. Сила, действующая на аппарат со стороны одного двигателя, $F = S\Delta p$, где S - площадь диска вентилятора, Δp - разность давлений воздуха до и после прохождения вентилятора. Используя уравнение Бернулли для обоих состояний

$$p_1 = p_0 + \frac{1}{2}\rho V_0^2, \quad p_2 = p_0 + \frac{1}{2}\rho V_e^2,$$

можно найти эту разность давлений. Здесь p_1 , p_2 - соответственно давления воздуха до и после прохождения вентилятора, p_0 - статическое давление, V_0 - скорость воздуха до входа в вентилятор, V_e - скорость воздуха после выхода из вентилятора, ρ - плотность воздуха. Тогда управляющий момент M определяется выражением

$$M = \frac{1}{2}RS\rho(V_e^2 - V_0^2),$$

где R - расстояние от оси вращения тела вокруг вертикали до вентилятора. Считаем, что размер вентилятора значительно меньше R . Уравнения, описывающие осевое вращение тела, имеют вид

$$J\dot{\omega} + \delta\omega + \sigma\alpha = \pm \frac{1}{2}RS\rho(V_e^2 - V_0^2), \quad \dot{\alpha} = \omega. \quad (2.1)$$

Здесь J - момент инерции тела относительно оси вращения; δ - коэффициент вязкого трения тела о воздух; σ - коэффициент упругости струны; точкой обозначена операция дифференцирования по времени t . Знак “+” или “-” выбирается в зависимости от того, положительный или отрицательный момент создается двигателем.

В силу предположения о неподвижности атмосферы $V_0 = |\omega R|$, поэтому первое уравнение системы (2.1) можно записать так:

$$\dot{\omega} + \frac{\delta}{J} \omega \pm \frac{1}{2} \frac{\rho S R^3}{J} \omega^2 + \frac{\sigma}{J} \alpha = \pm \frac{1}{2} \frac{\rho S R V_e^2}{J}. \quad (2.2)$$

Перейдем к безразмерным угловой скорости Ω и времени τ по формулам $\omega = V_e/R \Omega$, $t = R/V_e \tau$ и введем безразмерные параметры

$$k = \pm \frac{1}{2} \frac{\rho S R^3}{J}, \quad \varepsilon_1 = \frac{\delta R}{J V_e}, \quad \varepsilon_2 = \frac{\sigma R^2}{J V_e^2}. \quad (2.3)$$

Тогда уравнение (1.2) принимает вид

$$\Omega' + \varepsilon_1 \Omega + k \Omega^2 + \varepsilon_2 \alpha = k. \quad (2.4)$$

Перейдем к решению уравнения (2.4) в частных случаях. Будем считать, что

$$|k| \gg \varepsilon_1, \quad |k| \gg \varepsilon_2, \quad (2.5)$$

т. е. в присутствии управления пренебрегаем моментом вязкого трения тела о воздух и моментом от скручивания струны. В этом случае уравнение (2.4) принимает вид

$$\Omega' = k - k \Omega^2. \quad (2.6)$$

Его решение записывается следующим образом:

$$\Omega = \frac{\tilde{C} \exp(2k\tau) - 1}{\tilde{C} \exp(2k\tau) + 1}.$$

Переходя обратно к размерным величинам и переменным, имеем

$$\omega = \frac{V_e}{R} \frac{C \exp(2k \frac{V_e}{R} t) - 1}{C \exp(2k \frac{V_e}{R} t) + 1}. \quad (2.7)$$

Здесь

$$C = \frac{V_e + \omega_0 R}{V_e - \omega_0 R}.$$

Заметим, что если плотность атмосферы зависит от времени, а это может иметь место при исследовании динамики тела, закрепленного на меняющую высоту полета воздушном шаре [5] (тогда плотность зависит от высоты и, в конечном итоге, от времени), т. е.

$k(t) = \pm \frac{S R^3 \rho(t)}{2J}$, тогда решение уравнения (2.6) можно записать так:

$$\omega = \frac{V_e}{R} \frac{C \exp \left\{ \pm \frac{S R^2 V_e}{J} \int_0^t \rho(t') dt' \right\} - 1}{C \exp \left\{ \pm \frac{S R^2 V_e}{J} \int_0^t \rho(t') dt' \right\} + 1}.$$

Если ввести обозначения

$$A = \frac{V_e}{R}, B = k \frac{V_e}{R}, \quad (2.8)$$

то выражение (2.7) имеет вид

$$\omega = A \frac{C \exp(2Bt) - 1}{C \exp(2Bt) + 1}. \quad (2.9)$$

2.1.2 Модель на основе закона сохранения импульса

Модель действующей со стороны вентиляторов на макет силы, полученная из закона Бернулли, справедлива для установившегося, стационарного потока воздуха, проходящего через вентилятор. Однако лопасти вентилятора создают вихревые закрученные потоки на выходе из двигателя, поэтому рассмотренная модель дает лишь некоторое приближение реальной действующей силы со стороны вентиляторов.

Рассмотрим другую модель действующей силы со стороны вентиляторов. Если принять, что в состоянии "включен" через вентилятор проходит направленный перпендикулярно плоскости диска вентилятора поток воздуха с фиксированной по величине скоростью V_e относительно вентилятора, действующая на вентилятор сила может быть найдена из закона сохранения импульса

$$F = -\frac{dp}{dt} = -(V_0 + V_e) \frac{dm}{dt},$$

откуда

$$F = -(V_0 + V_e) \rho S V_e,$$

где dp , dm - импульс и масса воздуха, прошедшего через вентилятор за время dt , V_0 - скорость вентилятора относительно неподвижной атмосферы, ρ - плотность воздуха, S - площадь диска вентилятора.

Уравнения, описывающие осевое вращение тела, имеют вид, аналогичный (2.1),

$$J \dot{\omega} + \delta \omega + \sigma \alpha = \pm (V_0 + V_e) \rho S R V_e, \quad \dot{\alpha} = \omega. \quad (2.10)$$

Обозначения совпадают с обозначениями выражения (2.1).

В силу предположения о неподвижности атмосферы $V_0 = |\omega R|$, поэтому первое уравнение системы (2.10) можно записать так:

$$\dot{\omega} + \frac{\delta}{J} \omega \pm \frac{\rho S R^2 V_e^2}{J} \omega + \frac{\sigma}{J} \alpha = \pm \frac{\rho S R V_e^2}{J}. \quad (2.11)$$

Снова перейдем к безразмерным угловой скорости Ω и времени τ по формулам

$\omega = \frac{V_e}{R} \Omega$, $t = \frac{R}{V_e} \tau$ и введем безразмерные параметры. Тогда уравнение (1.2) принимает

вид

$$\Omega' + \varepsilon_1 \Omega + k \Omega + \varepsilon_2 \alpha = k. \quad (2.12)$$

Будем считать, что

$$|k| \gg \varepsilon_1, |k| \gg \varepsilon_2, \quad (2.13)$$

т. е. в присутствии управления пренебрегаем моментом вязкого трения тела о воздух и моментом от скручивания струны. В этом случае уравнение (2.12) принимает вид

$$\Omega' = k - k \Omega. \quad (2.14)$$

Его решение записывается следующим образом:

$$\Omega = \tilde{C} \exp(-k\tau) + 1.$$

Возвращаясь обратно к размерным переменным, получаем

$$\omega = \frac{V_e}{R} C \exp(-k \frac{V_e}{R} t) + \frac{V_e}{R}, \quad (2.15)$$

где

$$C = \frac{R}{V_e} \omega_0 - 1.$$

Если ввести обозначения

$$\tilde{A} = \frac{V_e}{R}, \tilde{B} = k \frac{V_e}{R}, \quad (2.16)$$

то выражение (2.15) можно записать в виде

$$\omega = \omega_0 \exp(-\tilde{B}t) + \tilde{A}(1 - \exp(-\tilde{B}t)) \quad (2.17)$$

2.1.3 Свободное движение макета

Рассмотрим свободное движение тела, подвешенного на струне. Это предположение отвечает условию $k=0$ в уравнении (2.4). Тогда его решение в виде выражений для угловой скорости и угла поворота тела при свободных затухающих колебаниях имеет следующий вид:

$$\alpha = \alpha_0 \exp(-\frac{\varepsilon_1}{2} \tau) \cos(\sqrt{\varepsilon_2 - \frac{\varepsilon_1^2}{4}} \tau) + \frac{\Omega_0 + \frac{\varepsilon_1}{2} \alpha_0}{\sqrt{\varepsilon_2 - \frac{\varepsilon_1^2}{4}}} \exp(-\frac{\varepsilon_1}{2} \tau) \sin(\sqrt{\varepsilon_2 - \frac{\varepsilon_1^2}{4}} \tau),$$

$$\Omega = -\frac{\alpha_0 \varepsilon_2 + \frac{\varepsilon_1}{2} \Omega_0}{\sqrt{\varepsilon_2 - \frac{\varepsilon_1^2}{4}}} \exp(-\frac{\varepsilon_1}{2} \tau) \sin(\sqrt{\varepsilon_2 - \frac{\varepsilon_1^2}{4}} \tau) + \Omega_0 \exp(-\frac{\varepsilon_1}{2} \tau) \sin(\sqrt{\varepsilon_2 - \frac{\varepsilon_1^2}{4}} \tau).$$

При переходе к размерной скорости получим

$$\omega = -\frac{\alpha_0 \tilde{\varepsilon}_2 + \frac{\tilde{\varepsilon}_1}{2} \omega_0}{\sqrt{\tilde{\varepsilon}_2 - \frac{\tilde{\varepsilon}_1^2}{4}}} \exp\left(-\frac{\tilde{\varepsilon}_1}{2} t\right) \sin\left(\sqrt{\tilde{\varepsilon}_2 - \frac{\tilde{\varepsilon}_1^2}{4}} t\right) + \omega_0 \exp\left(-\frac{\tilde{\varepsilon}_1}{2} t\right) \sin\left(\sqrt{\tilde{\varepsilon}_2 - \frac{\tilde{\varepsilon}_1^2}{4}} t\right), \quad (2.18)$$

где

$$\tilde{\varepsilon}_1 = \varepsilon_1 \frac{V_e}{R}, \quad \tilde{\varepsilon}_2 = \varepsilon_2 \frac{V_e^2}{R^2}. \quad (2.19)$$

2.1.4 Определение параметров моделей

Чтобы воспользоваться моделями движения под действием вентиляторов для управления ориентацией, необходимо определить их параметры (2.8) и (2.16), проверить правдоподобность предположений (2.5) и с помощью экспериментов верифицировать модель, то есть определить насколько близка используемая модель к реальности.

Для определения характеристик моделей необходимо провести калибровочные эксперименты. Допустим, движение твердого тела можно определить с помощью измерений датчиков, установленных на этом теле. Введем следующие обозначения: \mathbf{x} - вектор фазового состояния, \mathbf{z} - вектор измерений. Тогда уравнения движения записываются следующим образом

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{p}, t), \quad (2.20)$$

где $\mathbf{f}(\mathbf{p})$ - некоторая в общем случае нелинейная вектор-функция от вектора параметров модели \mathbf{p} и времени t . Если проинтегрировать уравнение (2.20), то зависимость вектора состояния от времени можно записать так:

$$\mathbf{x} = \boldsymbol{\varphi}(\mathbf{p}, \mathbf{x}(t_0), t).$$

Модель измерений в общем случае можно записать в виде

$$\mathbf{z} = \mathbf{h}(\mathbf{x}, t),$$

где $\mathbf{h}(\mathbf{x}, t)$ - некоторая нелинейная функция от вектора состояния и времени t .

При проведении эксперимента в некоторые дискретные моменты времени t_i имеем вектор измерений \mathbf{z}_i . Для определения вектора параметров \mathbf{p} запишем следующий функционал:

$$\Phi(\mathbf{p}) = \sum_{i=1}^N (\mathbf{z}_i - \mathbf{h}(\mathbf{x}(\mathbf{p}, t_i), t_i))^T (\mathbf{z}_i - \mathbf{h}(\mathbf{x}(\mathbf{p}, t_i), t_i)). \quad (2.21)$$

Здесь N - это количество векторов измерений, в скобках стоит разница измерений \mathbf{z}_i от прогнозируемого значения измерений $\mathbf{h}(\mathbf{x}(\mathbf{p}, t_i), t_i)$ согласно моделям. Найдем такой вектор

параметров, при котором функционал будет минимален. Так как функционал Φ является квадратичной формой, то минимум будет наблюдаться в точке, в которой производные функционала будут равны нулю:

$$\frac{\partial \Phi(\mathbf{p})}{\partial \mathbf{p}} = 0. \quad (2.22)$$

В общем случае система уравнений (2.22) нелинейная и не всегда удается её решить аналитически [18]. Численно система уравнений (2.22) поддается решению, например, с помощью метода Ньютона, который можно записать следующим образом:

$$\mathbf{p}_j = \mathbf{p}_{j-1} - \left. \frac{\partial \mathbf{J}}{\partial \mathbf{p}} \right|_{\mathbf{J}=\mathbf{J}(\mathbf{p}_{j-1})} \cdot \mathbf{J}(\mathbf{p}_{j-1}), \quad (2.23)$$

где $\mathbf{J}(\mathbf{p}) = \frac{\partial \Phi(\mathbf{p})}{\partial \mathbf{p}}$, $\left. \frac{\partial \mathbf{J}}{\partial \mathbf{p}} \right|_{\mathbf{J}=\mathbf{J}(\mathbf{p}_{j-1})}$ - якобиан вектор функции \mathbf{J} в точке \mathbf{p}_{j-1} , \mathbf{p}_j - значение вектора параметров на j -й итерации. Итерационная процедура (2.23) прекращается, когда $|\mathbf{p}_j - \mathbf{p}_{j-1}| < \varepsilon$, где ε - некоторый заданный вектор ошибок. Для работы метода Ньютона необходимо задать начальное приближение вектора \mathbf{p} , которое выбирается либо нулевым, либо из каких-либо разумных предположений.

Полученное значение вектора параметров модели $\tilde{\mathbf{p}}$ будет доставлять минимум функционалу (2.21), а само значение функционала $\Phi(\tilde{\mathbf{p}})$ является некоторым критерием правдоподобности модели: чем меньше его значение, тем лучше измерений ложится на модельную кривую $\mathbf{h}(\mathbf{x}(\mathbf{p}, t), t)$ и тем ближе модель к реальности.

2.2 Алгоритмы управления ориентацией

2.2.1 Оптимальное управление

Классификация оптимального управления

Каждому заданному закону управления соответствует закон изменения координат вектора состояния, то есть траектория “движения” управляемого объекта в фазовом пространстве [19]. Зачастую процесс управления осуществляется с “ограниченными ресурсами”, то есть закон управления не может быть произвольным, а должен выбираться из некоторого множества Ω . Математически задача оптимального управления может быть сформулирована так. Дан управляемый динамический объект, вектор состояния которого подчиняется системе уравнений:

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, \mathbf{u}), \quad \mathbf{u} \in \Omega. \quad (2.24)$$

Информацию о текущем векторе состояния $x(t)$ можно получить из наблюдений вида

$$z = h(x).$$

Закон управления $u(t)$ определяется с помощью процедуры минимизации функционала вида

$$Q(x, z, u, t) = \min.$$

Этот функционал и определяет цель управления. Рассмотрим классификацию задач управления.

Пусть дано начальное значение вектора состояния $x(t_0)$. Если задача управления формулируется так, что окрестность должна $x(t_1)$ достигаться за некоторый фиксированный отрезок времени $T = t_1 - t_0$. Тогда задачу относят к типу задач с *фиксированным временем и свободным концом траектории*.

Другая постановка задачи. Конец траектории строго фиксирован, то есть $x(t_1)$ задано. Требуется найти такое управление u , которое сообщает динамическому объекту траекторию, минимизирующую функционал Q . Время перехода от начального состояния к конечному не фиксировано. Тогда это задача с *закрепленными концами фазовой траектории и свободным временем*.

В частном случае, взяв в качестве функционала время T , получим задачу на *максимальное быстрое действие*. Например, задача управления состоит в том, чтобы повернуть аппарат из одного положения в некоторое другое, но время маневра при этом не фиксировано. Имеем задачу со свободным временем и закрепленными концами.

Другой пример – набор второй космической скорости для межпланетного перелета. Аппарат находится на круговой орбите, требуется за фиксированное время набрать параболическую скорость и потратить минимальное количество топлива. Здесь концы траектории не закреплены, а ведется поиск закона управления с минимальной энергией, решающего задачу достижения результата за ограниченное допустимое время.

Минимизация функционала – классическая задача вариационного исчисления. В разных прикладных задачах те или иные преимущества имеют

- а) метод динамического программирования Р. Беллмана,
- б) принцип максимума Л. Понтрягина.

Последний метод более распространен в небесной механике и астродинамике.

Принцип максимума Понтрягина

Запишем систему дифференциальных уравнений – математическую модель управляемого динамического объекта (ограничимся автономными системами):

$$\frac{dx_j}{dt} = f_j(x_1, \dots, x_n; u_1, \dots, u_l), \quad u_r \in \Omega, \quad j = 1, n, \quad r = 1, l. \quad (2.25)$$

Компоненты вектора управления u_r нужно выбрать так, чтобы минимизировать функционал Q , который запишем в виде

$$Q = \int_{t_0}^{t_1} f_0(x(\xi), u(\xi)) d\xi. \quad (2.26)$$

Введем $n+1$ -ю компоненту вектора состояния x следующим образом

$$x_0(t) = \int_{t_0}^t f_0(x(\xi), u(\xi)) d\xi, \quad (2.27)$$

так что

$$\frac{dx_0(t)}{dt} = f_0(x_1, \dots, x_n; u_1, \dots, u_l).$$

Очевидно, что начальные условия для компоненты x_0 имеют вид $x_0(t_0) = 0$, а цель управления $x_0(t_1) = Q$. Теперь расширенная система уравнений имеет вид

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, \mathbf{u}),$$

где

$$\mathbf{x} = (x_0, x_1, \dots, x_n)^T,$$

$$\mathbf{u} = (u_1, u_2, \dots, u_l)^T.$$

Пусть u^0 и x^0 – оптимальное управление и оптимальная траектория. Тогда для любого, мало отличающегося от оптимального, процесса можно написать уравнение в вариациях

$$\delta \dot{\mathbf{x}} = \frac{\partial \mathbf{f}(\mathbf{x}^0, \mathbf{u}^0)}{\partial \mathbf{x}} \delta \mathbf{x}$$

и сопряженную систему

$$\dot{\boldsymbol{\psi}} = - \left(\frac{\partial \mathbf{f}(\mathbf{x}^0, \mathbf{u}^0)}{\partial \mathbf{x}} \right)^T \boldsymbol{\psi}, \quad \boldsymbol{\psi} = (\psi_0, \psi_1, \dots, \psi_n)^T.$$

Заметим, что $\frac{\partial \mathbf{f}}{\partial \mathbf{x}}$ – квадратная матрица.

Определим функцию Понтрягина (расширенную функцию Гамильтона)

$$\tilde{H}(\boldsymbol{\psi}, \mathbf{x}, \mathbf{u}) = \boldsymbol{\psi}^T \mathbf{f}(\mathbf{x}, \mathbf{u}). \quad (2.28)$$

Теперь систему (2.25) можно записать в канонической форме

$$\begin{aligned}\frac{dx_j}{dt} &= \frac{\partial \tilde{H}}{\partial \psi_j}, \quad j = 0, 1, \dots, n; \\ \frac{d\psi_j}{dt} &= -\frac{\partial \tilde{H}}{\partial x_j},\end{aligned}\tag{2.29}$$

так как

$$\begin{aligned}\frac{\partial \tilde{H}}{\partial \psi_j} &= \frac{\partial}{\partial \psi_j} \left(\sum_{k=0}^n \psi_k f_k \right) = f_j(x, u), \\ \frac{\partial \tilde{H}}{\partial x_j} &= \sum_{k=0}^n \psi_k \frac{\partial f_k}{\partial x_j} = -\frac{\partial f_j}{\partial t}.\end{aligned}$$

Л.С. Понтрягиным доказано [20], что оптимальное управление $u = u^0$ достигается при максимуме функции $\tilde{H}(\psi, x, u)$:

$$\tilde{M}(\psi, x^0) = \sup_{u \in \Omega} \tilde{H}(\psi, x, u).\tag{2.30}$$

Сформулируем теперь принцип максимума. Пусть $u(t)$ ($t_0 \leq t \leq t_1$) – такое допустимое управление, что соответствующая ему траектория $x(t)$, исходящая в момент t_0 из точки $x(t_0)$, проходит в момент t_1 через точку $x(t_1)$. Для оптимальности управления $u(t)$ и траектории $x(t)$ необходимо существование такой ненулевой непрерывной вектор-функции $\psi(t)$, что

1⁰ – если канонические уравнения (2.29) для любого момента $t \in [t_0, t_1]$ удовлетворены, функция $\tilde{H}(\psi, x, u)$ переменной $u \in \Omega$ достигает в точке $u = u^0$ максимума;

2⁰ – в конечный момент t_1 выполнены соотношения

$$\tilde{M}(\psi(t_1), x(t_1)) = 0, \quad \psi_0(t_1) \leq 0.$$

Оптимальное быстроедействие

Решим задачу наибыстрейшего перевода фазовой точки из одного состояния в другое. Сформулируем эту задачу следующим образом. Пусть точка движется вдоль оси Ox по закону $\ddot{x}(t) = u(t), (t \geq t_0)$. Требуется найти кусочно-непрерывное управление $u(t)$, $|u(t)| \leq u^{\max}, (t_0 \leq t \leq T)$ такое, чтобы точка, выйдя из начального положения $x(t_0) = x_0, \dot{x}(t_0) = \dot{x}_0$, пришла за минимальное время в точку $x(T) = x_T, \dot{x}(T) = \dot{x}_T$ [8].

Положим, что $x_1 = x, x_2 = \dot{x}$ – фазовые координаты точки. Тогда задачу можно переформулировать так: быстрейшим образом перевести фазовую точку (x_1, x_2) из состояния (x_0, \dot{x}_0) в состояние (x_T, \dot{x}_T) , считая, что движение подчиняется уравнениям

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), \\ x_2(t) &= u(t). \end{aligned} \quad (2.31)$$

Так как мы минимизируем время, то функционал Q запишется в следующем виде:

$$Q = -T = \int_0^T (-1) dt.$$

Согласно формуле (2.26) в нашем случае $f_0 = -1$, а согласно формуле (2.27)

$$x_0(t) = \int_{t_0}^t f_0(x(\xi), u(\xi)) d\xi = \int_{t_0}^t (-1) d\xi. \text{ Запишем функцию Понтрягина (2.28):}$$

$$\tilde{H}(\psi, x, u) = \psi^T f(x, u) = -\psi_0 + \psi_1 x_2 + \psi_2 u. \quad (2.32)$$

Найдём сопряжённую систему из уравнений (2.29)

$$\dot{\psi}_0 = -\frac{\partial H}{\partial x_0} = 0,$$

$$\dot{\psi}_1 = -\frac{\partial H}{\partial x_1} = 0,$$

$$\dot{\psi}_2 = -\frac{\partial H}{\partial x_2} = -\psi_1.$$

Проинтегрировав, получаем

$$\psi_0 = C_1,$$

$$\psi_1 = C_2,$$

$$\psi_2 = -C_2 t + C_3.$$

Итак, согласно принципу максимума Понтрягина (2.30) оптимальным управлением, максимизирующем Гамильтониан (2.32), является управление

$$u_{opt}(t) = u^{\max} \text{sign}(\psi_2) = u^{\max} \text{sign}(-C_2 t + C_3). \quad (2.33)$$

Таким образом, оптимальное управление является кусочно-постоянной функцией, принимающей значения $+u^{\max}$, $-u^{\max}$ и имеющей не более одной точки переключения t_n , при переходе через которую $u_{opt}(t)$ меняет знак.

Для отрезка времени, на котором $u = u^{\max}$ в силу системы (2.31) получаем

$$x_2 = u^{\max} t + s_2,$$

$$x_1 = \frac{t^2}{2} + s_2 t + s_1 = \frac{1}{2} (t + s_2)^2 + \left(s_1 - \frac{s_2^2}{2} \right),$$

где s_1, s_2 - некоторые константы, отсюда получаем

$$x_1 = \frac{1}{2} x_2^2 + s, \quad (2.34)$$

где s - константа. Таким образом, кусок фазовой траектории, для которого $u \equiv u^{\max}$, представляет собой дугу параболы (Рис. 2.3).

Аналогично, для отрезка времени, на котором $u \equiv -u^{\max}$ получаем

$$x_1 = -\frac{1}{2}x_2^2 + r, \quad (2.35)$$

где r - константа.

По параболам (2.34) фазовые точки движутся снизу вверх, а по параболам (2.35) сверху вниз. Как было указано выше, оптимальное управление является кусочно-постоянной функцией t , принимающей значения $\pm u^{\max}$ и имеющей не более двух интервалов постоянства. Если управление

$u_{opt}(t)$ сначала, в течение некоторого времени, равно u^{\max} , а затем равно $-u^{\max}$, то фазовая траектория состоит из двух кусков парабол, примыкающих друг к другу, причем второй из этих кусков лежит на той из парабол, которая проходит через начало координат (ибо искомая траектория должна вести в начала координат). Если же наоборот, то фазовая траектория заменяется центрально-симметричной (см. Рис. 2.3).

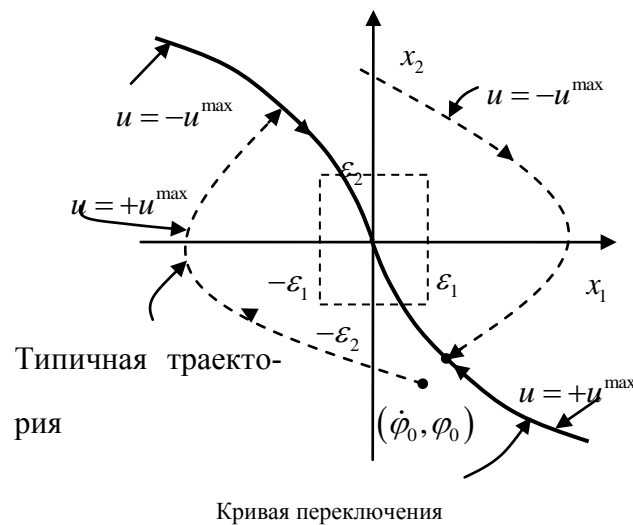


Рис. 2.3 Фазовые траектории и кривые переключения

Управление ориентацией

Рассмотрим задачу оптимального быстродействия при управлении ориентацией макета. В этом случае

$$x_1 = \varphi,$$

$$x_2 = \omega.$$

Пренебрегая сопротивлением со стороны воздуха и модулем кручения нити, уравнение движения одноосного вращения примет вид

$$\ddot{\varphi} = u(t),$$

где φ – угол поворота связанной с аппаратом системы относительно некоторой инерциальной, ω – угловая скорость вращения.

Задача управления состоит в наибоыстрейшем переводе точки фазового состояния из положения $(\varphi_0, \dot{\varphi}_0)$ в состояние $(\varphi_T, \dot{\varphi}_T)$ с тем условием, что точка $(\varphi_T, \dot{\varphi}_T)$ – лежит на некоторой заранее определённой кривой, которую будем называть *опорная* траектория. В случае, когда $(\varphi_T, \dot{\varphi}_T) = (0, 0)$, решением такой задачи будет следующее управление, которое следует из принципа максимума Понтрягина

$$u = -u^{\max} \operatorname{sign} \left[u^{\max} \varphi + \frac{1}{2} |\dot{\varphi}| \dot{\varphi} \right] = \begin{cases} +u^{\max}, & \text{если } \varphi < \frac{|\dot{\varphi}| \dot{\varphi}}{2u^{\max}} \\ -u^{\max}, & \text{если } \varphi > \frac{|\dot{\varphi}| \dot{\varphi}}{2u^{\max}} \end{cases} \quad (2.36)$$

Уравнение (2.36) обуславливает собой кривую переключения между управлениями, а проинтегрировав уравнение движения (2.9), можно получить типичную фазовую траекторию движения тела при включенном одном или другом импульсном двигателе (см. Рис. 2.3).

Далее рассмотрим оптимальное по быстродействию управление, основываясь на модели лопастных двигателей из раздела 1.1. Интегрируем (2.9), считая

$$\omega = \dot{\alpha},$$

и находим

$$\alpha = -At + \frac{A}{B} \ln \frac{1 + C \exp(2Bt)}{1 + C} + \alpha_0. \quad (2.37)$$

Выразим из (2.9) время t и подставим его в (2.37). Получим

$$\alpha = \frac{A}{2B} \ln \frac{A^2 - \omega_0^2}{A^2 - \omega^2} + \alpha_0. \quad (2.38)$$

Выражение (2.38) дает два семейства кривых на фазовой плоскости (α, ω) : для положительного вращения ($k > 0$, а следовательно, и $B > 0$) и для отрицательного ($k < 0$, $B < 0$).

Задача поворота на заданный угол и/или торможения – это переход из одной точки фазовой плоскости в другую. Пример задачи, в рамках которой осуществляется переход из точки 1 в точку 2, приведен на рис.1, где пунктирная кривая соответствует положительному значению параметра B , а сплошная – отрицательному.

На рис.1 видно, что сначала происходит движение по пунктирной кривой вверх до точки пересечения со сплошной линией, а затем по сплошной линии вниз до точки 2. Это значит, что сначала работает один двигатель, а когда достигается определенное начальными условиями и точкой на фазовой плоскости значение угловой скорости, происходит переключе-

чение, выключается первый двигатель и включается второй. Второй двигатель отключается при достижении нулевой скорости.

Рассмотрим теперь ситуацию, когда двигатели не идентичны и величины параметров A и B различны для каждого из двигателей. Введём обозначения для двигателя, создающего положительный момент, $A_1 > 0$, $B_1 > 0$, а для двигателя, создающего отрицательный момент, $A_2 < 0$, $B_2 < 0$. Выражение для кривой переключения будет иметь вид

$$f(\alpha, \omega) = \begin{cases} \alpha - \frac{A_2}{2B_2} \ln \frac{A_2^2 - \omega^2}{A_2^2}, & \alpha \leq 0, \\ \alpha - \frac{A_1}{2B_1} \ln \frac{A_1^2 - \omega^2}{A_1^2}, & \alpha > 0. \end{cases} \quad (2.39)$$

Обе ветви получаются подстановкой в (2.38) $\omega = 0$ и $\alpha = 0$ при разных значениях параметров A и B (график этой функции изображен на рис. 2, пунктирная линия соответствует параметрам A_1 и B_1 , сплошная - A_2 и B_2).

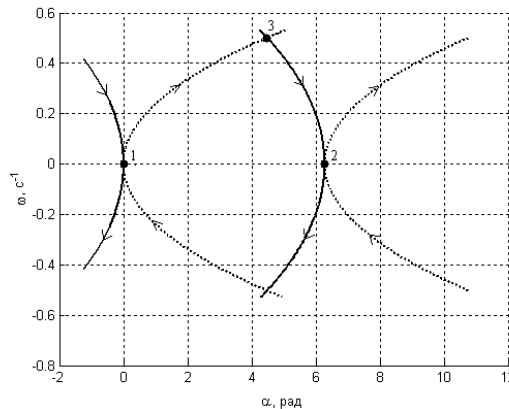


Рис. 2.4 Траектории движения на фазовой плоскости

Алгоритм переключения работает так.

1. Определяется начальное фазовое состояние (α_0, ω_0) , при этом выделяется ветвь кривой переключения

$$\alpha_0 \leq 0 \Rightarrow f = \alpha - \frac{A_2}{2B_2} \ln \frac{A_2^2 - \omega^2}{A_2^2},$$

$$\alpha_0 > 0 \Rightarrow f = \alpha - \frac{A_1}{2B_1} \ln \frac{A_1^2 - \omega^2}{A_1^2}.$$

Если же задана величина поворота $\Delta\alpha$, то

$$\Delta\alpha \geq 0 \Rightarrow f = \alpha - \frac{A_2}{2B_2} \ln \frac{A_2^2 - \omega^2}{A_2^2},$$

$$\Delta\alpha < 0 \Rightarrow f = \alpha - \frac{A_1}{2B_1} \ln \frac{A_1^2 - \omega^2}{A_1^2}.$$

2. Вычисляется $f(\alpha_0, \omega_0)$. Если $f(\alpha_0, \omega_0) > 0$, то работает двигатель с A_2, B_2 , если $f(\alpha_0, \omega_0) < 0$, то работает двигатель с A_1, B_1 .

3. Измеряются (α, ω) и определяется знак выражения $\Delta f = |f(\alpha, \omega)| - f_0$; здесь f_0 - допустимое значение отключения от расчетной кривой. При изменении знака Δf производится переключение двигателей.

4. Осуществляется движение вдоль кривой переключения. При этом проводится проверка $|\omega| < \varepsilon_2$ (ε_2 - точность определения угловой скорости). Если достигается малое значение угловой скорости, то проводится проверка $|f(\alpha, \omega)| < \varepsilon_1$ (ε_1 - точность по углу). Если и это условие выполняется, то это значит, что система пришла в нужное положение и цикл управления заканчивается, если нет, то повторяются п. 1 - 4.

2.2.2 Пропорционально-дифференциальный регулятор

Рассмотрим алгоритм управления ориентацией, основанный на пропорционально-дифференциальном регуляторе (ПД-регуляторе). ПД-регулятор формирует управляющий сигнал, являющийся суммой двух слагаемых, первое из которых пропорционально входному сигналу, второе — производная входного сигнала. В случае управления ориентацией тела, подвешенного на струне, входным сигналом является рассогласование по углу $\Delta\alpha$, а производная — это вектор угловой скорости аппарата ω . Рассмотрим задачу поворота на заданный угол и стабилизации в этом состоянии.

Если задать ориентацию аппарата углом поворота от связанной с телом системы координат относительно некоторой инерциальной, то формирование управляющего сигнала $u(t)$ происходит по следующей формуле:

$$u(t) = k_\alpha \Delta\alpha + k_\omega \omega, \quad (2.40)$$

где k_α, k_ω — коэффициенты усиления управления по рассогласованию угла и угловой скорости соответственно [21].

Модуляция управляющего сигнала

Из формулы (2.40) следует, что вектор управления $u(t)$ является непрерывной функцией от времени, а имеется импульсное управление, которое является дискретным и может принимать значения $\pm U$ и 0. Эту проблему можно решить модуляцией сигнала $u(t)$ импульсами $\pm U$, которая заключается в том, что интеграл по времени $\int_0^T u(t) dt$ приблизительно

равен сумме импульсов $\sum_{i=1}^N U_i \Delta t_i$, где N – количество импульсов за время T , $U_i = \pm U$, Δt_i – длительность i -го импульса, то есть

$$\int_0^T u(t) dt \approx \sum_{i=1}^N U_i \Delta t_i.$$

Такая модуляция может быть реализована несколькими способами. Например, рассчитывается интеграл $\int_{T_{i-1}}^{T_i} u(t) dt$, и в тот момент, когда его величина будет равна $U \Delta t_{\min}$, где Δt_{\min} –

минимальное время подачи импульса, производится импульс, и после этого снова считается интеграл $\int_{T_{i-1}}^{T_i} u(t) dt$. Или можно подавать импульсы с некоторой заданной периодичностью.

При этом длительность каждого импульса $\Delta t_{\text{имп}}$ будет считаться, исходя из равенства

$$\Delta t_{\text{имп}} = \frac{\int_{T_{i-1}}^{T_i} u(t) dt}{U}, \quad (2.41)$$

где $T_i - T_{i-1} = \Delta t$ – периодичность подачи импульсов. Пример такой модуляции представлен на Рис. 2.5.

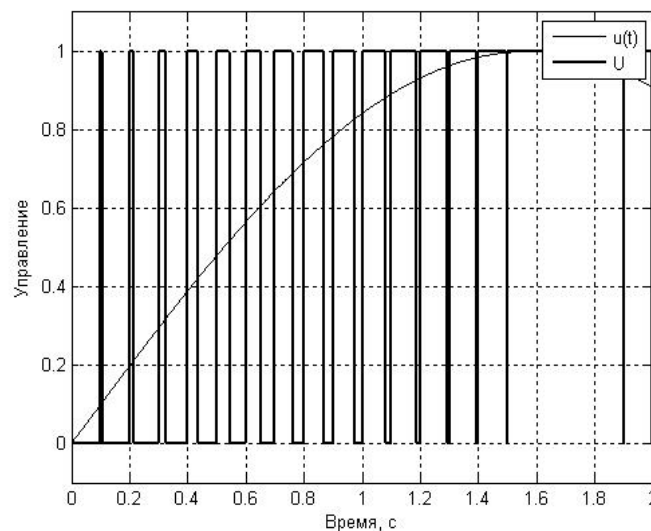


Рис. 2.5 Модуляция сигнала $u(t)$ с помощью импульсов U

Обеспечение асимптотической устойчивости

Теперь найдем такие коэффициенты k_α , k_ω , чтобы система управления осуществляла поворот в заданное положение и обеспечивала его асимптотическую устойчивость. Будем строить управление с помощью функции Ляпунова, выбранной в виде

$$V = \frac{1}{2}\omega^2 + \frac{1}{2}k_\alpha\Delta\alpha^2,$$

где $\Delta\alpha, \omega$ – рассогласование по углу и угловая скорость. Уравнение движения (2.1) можно переписать так:

$$J\dot{\omega} + \delta\omega + \sigma\Delta\alpha = Jk(A^2 - \omega^2). \quad (2.42)$$

Здесь будем считать, что можно управлять скоростью вращения вентиляторов, т.е. изменять V_e , а следовательно, параметр A . Кроме того, также можно переключать двигатели, т.е. осуществлять выбор между $k > 0$ и $k < 0$. Тогда с учетом уравнений движения (2.42) \dot{V} будет иметь вид

$$\dot{V} = -\delta\omega^2 + \omega(Jk(A^2 - \omega^2) - \delta\omega - \sigma\Delta\alpha + k_\alpha\Delta\alpha).$$

Для асимптотической устойчивости положения $\omega = 0, \Delta\alpha = 0$ потребуем (теорема Барбашина-Красовского [22]) выполнения условия

$$k(A^2 - \omega^2) - \delta\omega + k_\alpha\Delta\alpha = -k_\omega\omega,$$

где $k_\omega > 0$. Введем обозначение

$$Q = \sigma\Delta\alpha - k_\alpha\Delta\alpha - k_\omega\omega$$

и выразим из условия $k(A^2 - \omega^2) = Q$

$$A = \sqrt{\frac{Q}{Jk} + \omega^2}, \text{ где } \begin{cases} k > 0, \text{ если } \frac{Q}{J|k|} + \omega^2 \geq 0, \\ k < 0, \text{ если } \frac{Q}{J|k|} + \omega^2 < 0. \end{cases} \quad (2.43)$$

Это и есть условие переключения, так как A должно быть вещественным. Подставив (2.43) в (2.42), получим

$$J\dot{\omega} + (\delta + k_\omega)\omega + k_\alpha\Delta\alpha = 0. \quad (2.44)$$

На практике, как правило, $A \leq A_{\max}$, поэтому возникает ограничение на возможные траектории. Рассмотрим ситуацию, когда задача выхода на положение уже решена, т.е. $\omega = 0$ и $\Delta\alpha = 0$. В этом случае получаем ограничение на возможные траектории

$$\omega^2 \leq A_{\max}^2. \quad (2.45)$$

Параметры k_ω и k_α выберем таким образом, чтобы максимизировать степень устойчивости [22] корней характеристического полинома уравнения (2.44)

$$J\lambda^2 + (\delta + k_\omega)\lambda + k_\alpha\lambda = 0. \quad (2.46)$$

Корни этого уравнения имеют вид

$$\lambda_{1,2} = \frac{-(\delta + k_\omega) \pm \sqrt{(\delta + k_\omega)^2 - 4Jk_\alpha}}{2J}. \quad (2.47)$$

Максимальная степень устойчивости будет достигаться в случае, когда подкоренное выражение будет равно нулю

$$(\delta + k_\omega)^2 - 4Jk_\alpha = 0,$$

т.е.

$$k_\omega = \sqrt{4Jk_\alpha} - \delta. \quad (2.48)$$

В противном случае один из корней будет ближе к мнимой оси, и уже он будет определять степень устойчивости. Таким образом, остается один независимый параметр управления k_α . Его нужно выбирать исходя из ограничений, наложенных на управляющий момент (например, в виде неравенства (2.45)).

Алгоритм управления ориентацией тела с помощью ПД-регулятора можно описать следующим образом. По измерениям датчиков алгоритм определения вектора состояния вычисляет текущий угол поворота связанной с телом системы координат относительно инерциальной и вектор угловой скорости. Далее производится проверка, находится ли вектор состояния аппарата в области допустимых значений (допустимых отклонений текущего фазового положения от желаемого). Если аппарат выходит за эти рамки, то принимается решение об управлении, рассчитывается непрерывное управление $u(t)$ (2.40), которое модулируется по одному из вышеизложенных алгоритмов, например, по (2.41).

2.2.3 Характеристики алгоритмов управления

Система управления не способна идеально точно выполнять задачи управления вследствие ряда факторов, а именно:

- ошибки исполнения актюаторов;
- ограничения на величину управляющего воздействия;
- неточность используемых моделей актюаторов и неточность знания параметров моделей;
- неточность знания параметров модели управляемого объекта;
- влияние на управляемый объект неучтенных в модели движения возмущений;

- запаздывание в управляющем контуре как в программной части, так и в механической части, вызванное инерционностью актюаторов;
- ошибки определения состояния объекта, вызванные, в том числе, и ошибками измерений;
- особенности используемых алгоритмов управления.

Устранить влияние всех этих факторов невозможно, но можно понизить уровень их влияния путем уточнения используемых моделей, улучшения программно-алгоритмической и аппаратной части системы управления. Однако для любой системы управления в целом можно ввести такие характеристики качества управления, как точность выполнения задачи управления и характерное время, которое система управления на это тратит.

Основными задачами системы ориентации являются выполнения маневра поворота в заданное положение и поддержание этого положения. При выполнении маневра поворота основной характеристикой системы управления является точность конечного положения после поворота и время, за которое аппарат совершит поворот на максимальный угол (в случае одноосного управления - на 180^0).

При поддержании ориентации представляет интерес максимальная точность, с которой система управления способна удержать аппарат в заданном положении. Дело в том, что вследствие совокупности всех вышеперечисленных факторов система управления будет иметь некоторую предельную точность поддержания. Если не задать в алгоритме управления некоторую окрестность на фазовой плоскости вблизи желаемого положения, внутри которой алгоритм управления алгоритм не будет работать (область допустимых ошибок) или задать эту окрестность достаточно малой, то фазовая траектория управляемого тела выйдет на некоторый предельный цикл - периодическую замкнутую траекторию, которая и будет обуславливать предельную точность системы управления. Пример такой траектории при работе оптимального алгоритма приведен на Рис. 2.6.

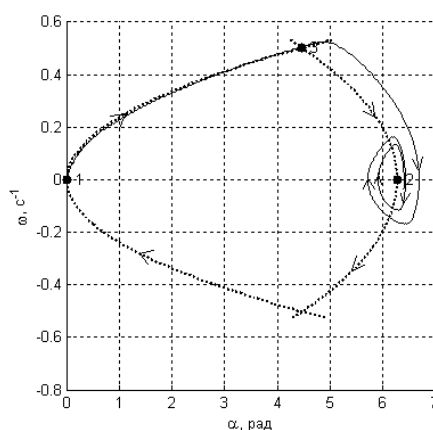


Рис. 2.6 Фазовая траектория и предельный цикл

3 Система для проведения лабораторных работ

3.1 Описание стенда

Рассмотрим схему макета системы ориентации на лопастных двигателях, созданного на кафедре теоретической механики в МФТИ. Макет представляет собой автономный прибор, включающий в себя систему электропитания, систему определения ориентации, систему управления ориентацией и систему связи. Схема макета изображена на Рис. 3.1, внешний вид макета – на Рис. 3.2.

Система электропитания включает в себя два аккумулятора и преобразователь напряжения. Преобразователь из 14 вольт от аккумуляторов формирует 5 вольт, необходимых для питания бортового компьютера и 12 вольт для питания управляющей платы.

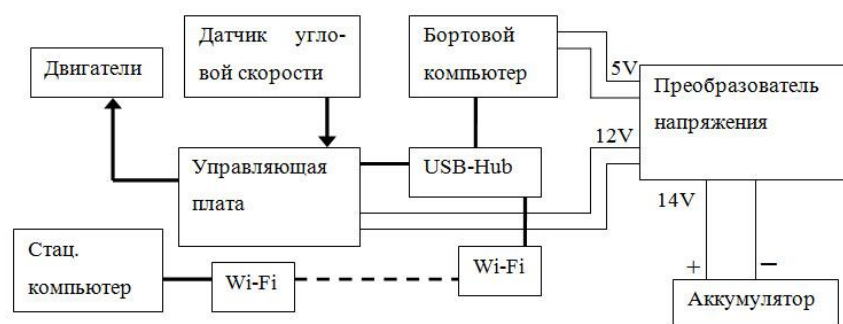


Рис. 3.1 Схема макета

Система определения ориентации состоит из бортового компьютера JRex CE (далее БК) и оптоволоконного датчика угловой скорости VG-910D. БК производит считывание информации об угловой скорости с датчика угловой скорости, обрабатывает её с помощью программного обеспечения (далее ПО) в режиме реального времени и передает данные. Передача информации производится через USB порты БК, используется разветвитель USB-Hub.

Система управления ориентацией состоит из двух лопастных двигателей, с помощью которых к макету можно приложить управляющий момент. БК выдает на управляющую плату команду на включение одного из двигателей, управляющая плата преобразует команду в напряжение, которое подается на лопастные двигатели.

Система связи состоит из устройства Wi-Fi и БК. Устройство Wi-Fi позволяет организовать доступ к БК через удалённый рабочий стол и подключиться клиентской программе. Управление ПО происходит со стационарного компьютера через удалённый рабочий стол БК или через клиентскую программу.

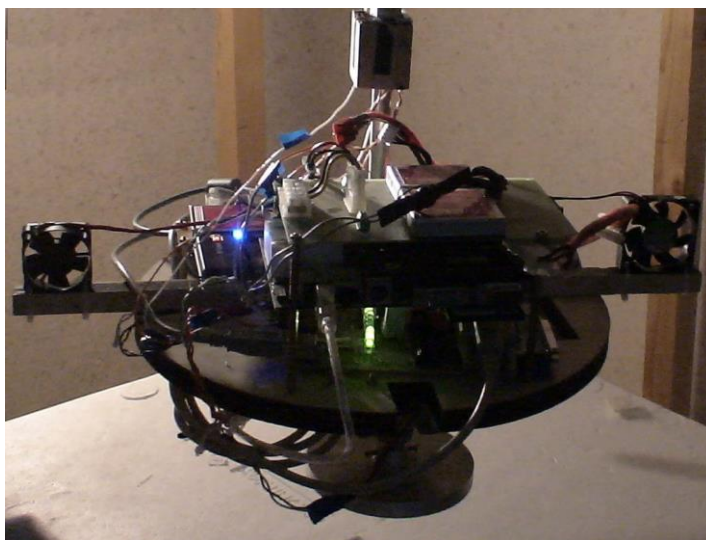


Рис. 3.2 Внешний вид макета

3.2 Программное обеспечение макета

Программное обеспечение (ПО) – это комплекс из двух программ. Первая из них, установленная на макете (серверная часть), осуществляет считывание информации с датчиков макета, её обработку и передачу результатов второй программе. Вторая программа (клиентская часть) установлена на пользовательском компьютере, осуществляет прием данных, их визуализацию и сохранение. Программа написана в среде C++ Builder 6.

3.2.1 Программа на макете (серверная часть)

Работа программы

С датчика поступает цифровой сигнал, который умножением на некоторый коэффициент и смещением (калибровка) преобразуется в угловую скорость. Значения угловой скорости за несколько измерений суммируются, после чего эта сумма делится на количество измерений, таким образом, на выходе получаем усреднённое значение угловой скорости. Усреднение производится для того, чтобы уменьшить шум измерений и уменьшить число вычислений программы в единицу времени. У программы есть верхний предел по частоте выходных данных ≈ 100 Гц, выше этого значения программа не успевает обрабатывать поступающие с датчика данные, обработка в режиме реального времени становится невозможной. И поэтому при установке параметров работы программы нужно отдельно обратить

внимание на усреднение: зная частоту подачи данных устанавливать такое усреднение (то есть выбрать количество измерений для усреднения), чтобы частота выходных данных программы не превышала предела.

Усреднённая угловая скорость поступает на вход сразу трём программным модулям: интегрирование кинематического уравнения, Фильтр Калмана (ФК) модели перманентного вращения, ФК модели крутильных колебаний. На выходе соответственно имеем: результат интегрирования, апостериори оценки угла отклонения макета от положения равновесия и угловой скорости двух ФК. Все эти значения плюс сама усреднённая угловая скорость и угол, передаются клиентской программе.

Интерфейс программы

Главное окно программы содержит три области (Рис.3.3):

- *Port settings* – установка рабочего датчика, выбор порта, через который будет поступать информация, установка порта управления и частоты отправки данных на клиенты. Задав эти параметры программы можно нажать кнопку “Open port”, программа подключится к датчикам и автоматически начнется процесс вычисления.

- *Control Settings* – управление алгоритмом действия макета (эти кнопки дублируют аналогичные на клиенте).

- *Some information* – содержит информацию о выполняемом алгоритме, число принятых байт в одной посылке с датчика, текущем времени эксперимента, текущей угловой скорости, количестве присоединенных пользователей и сведений об администраторе.

- Главное окно содержит меню со следующими разделами:

- *File* – выход из программы.

- *Sensor calibration* – установка калибровочных коэффициентов для датчиков VG-910D и ADIS16100.

- *Initial conditions* – задание начального вектора состояния макета.

- *Model of measuring* – задание параметров модели измерений (квадрата ошибки измерений σ^2 и значений элементов матрицы ошибки для ФК) и размер выборки N для алгоритма “Уточнение ковариации”. Если N=0, то “Уточнение ковариации” отключено.

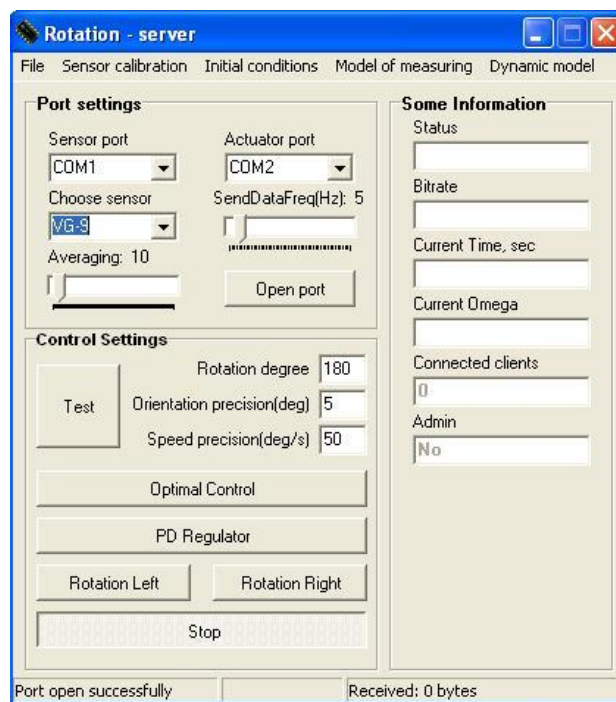


Рис.3.3. Интерфейс серверной части

- *Dynamic model* – задание периода крутильных колебаний.

Управление ПО

Управление ПО осуществляется при помощи интерфейса программы (Рис.9). Перед запуском программы необходимо произвести следующие действия:

- В области *Port settings* в выпадающем окошке *Choose sensor* выбрать датчик, с которого будет производиться чтение (VG-9, волоконно-оптический датчик или Mems, микро-механический датчик)
- В окошке *Sensor port* задать COM-порт, откуда будет считываться информация.
- В окошке *Actuator port* задать COM-порт, через который будет осуществляться управление.
- Установить значение усреднения *Averaging*.
- Установить частоту отправки данных *SendDataFreq*.
- Запустить программу, нажав на кнопку *Open port*. ПО начнет производить вычисления и отправлять данные подключенным клиентам.

3.2.2 Пользовательская программа (клиентская часть)

Работа программы

Все входные значения клиентской программы записываются в два вида файлов в папке *Results*, которая создается в директории клиентской программы. Первый вид файла с названием “Дата-Время_AllSessionReceiveddata.txt”, где «Дата» - это текущая дата, а «Время» - время запуска клиентской программы, содержит все измерения, полученные за то время, пока программа была запущена. Второй вид файла с названием “Дата-Время_ТипПрограммыReceiveddata.txt”, где дата и время аналогичны первому файлу, а «ТипПрограммы» - это тип выполняемого алгоритма управления, создается при изменении типа работы, то есть фактически при выполнении каждого отдельного эксперимента, это позволяет получить сразу результаты каждого эксперимента отдельно. Каждый блок значений записывается в отдельной строке.

Также значения обрабатываются и выводятся в виде информации о выполняемом эксперименте, в виде графиков результатов в реальном времени и значений в таблицах. Также программа позволяет удаленно наблюдать за макетом по веб камере, либо наблюдать за движением трехмерной модели макета, отражающей получаемые данные угла и угловой скорости.

3.2.3 Интерфейс программы

Главное окно программы содержит четыре области (Рис.3.4):

- *Connect properties* – настройка соединения с макетом (серверной частью программы). Здесь настраивается адрес сервера и возможность подключения в режиме «администратора» для управления, либо в режиме клиента для просмотра поступающих данных.
- *Status information* – содержит информацию о выполняемом алгоритме, текущем времени эксперимента, текущем угле поворота макета и текущей угловой скорости.
- *Lab1, Lab2, Lab3* – вкладки управления режимами макета, с помощью которых можно задавать режимы работы макета.
- *VideoCam, 3DModel* – вкладки наблюдения за макетом в режиме реального времени через Веб камеру, либо в режиме 3D модели.

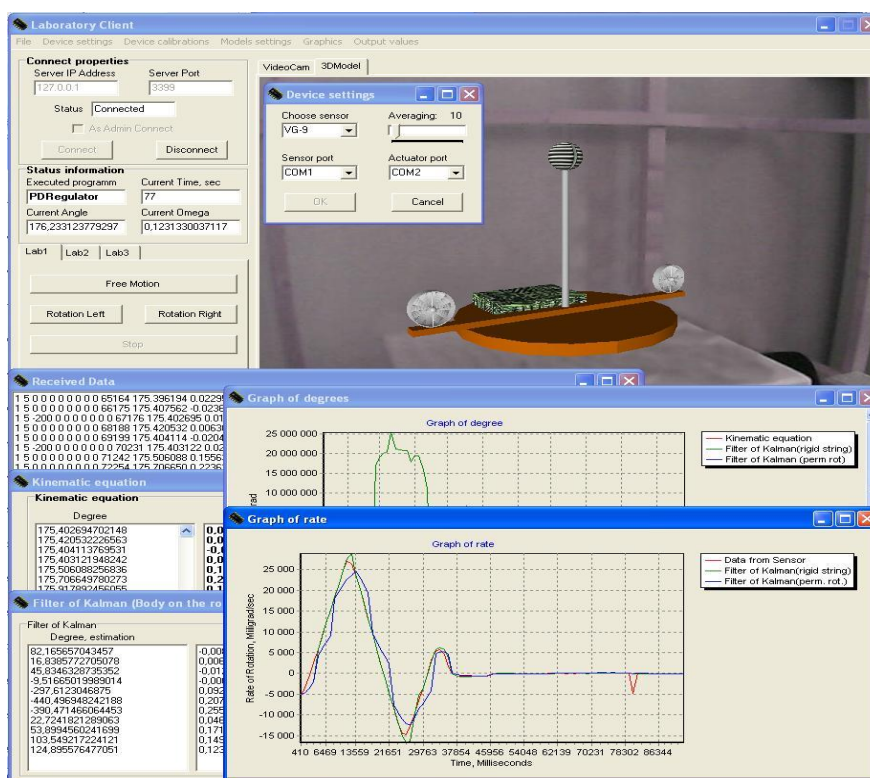


Рис.3.4. Интерфейс клиентского ПО

Главное окно содержит меню со следующими разделами:

- *File* – выход из программы.
- *Device Settings* – установка рабочего датчика, выбор порта, через который будет поступать информация и выбор порта управления.
- *Device calibration* – установка калибровочных коэффициентов для датчиков угловой скорости (ДУС) и вентиляторов.
- *Initial conditions* – задание начального вектора состояния макета.
- *Dynamic model* – задание периода крутильных колебаний.

- *Models settings* – задание начального вектора состояния макета, периода крутильных колебаний и параметров модели измерений (квадрата ошибки измерений σ^2 и значений элементов матрицы ошибки для ФК).

- *Graphics* – вывод в режиме реального времени графиков включения вентиляторов, угловой скорости и угла отклонения от положения равновесия. Графики отображают все выходные данные программы.

- *Output values* – вывод на экран принимаемых данных, выходных значений 2-х фильтров и результата интегрирования.

Управление ПО

Управление клиентского ПО осуществляется при помощи интерфейса программы (Рис.10). Перед запуском лабораторной работы:

1. В области *Connect Properties* ввести адрес макета и порт 3399. Если вы будете управлять проведением лабораторной работы, нужно подключиться в режиме «Администратор», для этого отметить метку «As Admin Connect», если вы будете просто наблюдать, метку нужно оставить неотмеченной (К серверу может подключиться только один управляющий, остальные подключаются наблюдателями). Далее нужно нажать кнопку *Connect* и дождаться состояния «Connected» в поле *Status*.

2. При подключении в режиме «Администратор», во вкладке главного меню *Device Settings* можно изменить окошко *Sensor port* - COM-порт, откуда будет считываться информация, в окошке *Actuator port* задать COM-порт, через который будет осуществляться управление. Также в окошке *Choose sensor* выбирается тип подключенного ДУС (VG-9 или Mems), а ползунок *Averaging* устанавливает усреднение для принимаемых данных на серверной части. Эти настройки дублируют и меняют настройки области *Port Settings* на серверной части программы.

3. При подключении в режиме «Администратор», во вкладке главного меню *Device Calibrations*, проверить настройки ДУСов и вентиляторов.

4. При подключении в режиме «Администратор», во вкладке главного меню *Models Settings*, в разделе меню *Initial conditions* проверить начальный угол отклонения от положения равновесия в градусах (если макет закручен по часовой стрелке, то добавляется знак минус). В разделе *Dynamic model* проверить период крутильных колебаний. В разделе *Model of measuring* проверить значение квадрата ошибки σ^2 для заданного усреднения датчика и размер выборки N для алгоритма “Уточнение ковариации”. Если N=0, то “Уточнение ковариации” отключено.

5. Выбрать режим наблюдения через веб камеру или 3D-модель и выбрать соответствующую вкладку *VideoCam* или *3DModel*. При выборе режима веб камеры, нужно подключить ее, настроив необходимые параметры и нажав клавишу *Play/Stop*, находящиеся в области *Camera Settings* под изображением. Если выбран режим 3D-модели, можно включить или отключить сглаживание движений кнопкой *Smoothing*.

6. Для начала эксперимента необходимо выбрать соответствующую вкладку *Lab1/Lab2/Lab3* и нажать на кнопку необходимого эксперимента. ПО начнет производить вычисления и прием-передачу данных.

Полученные данные а также результаты вычислений отображаются на графиках раздела меню *Graphics* и в значениях, в разделе *Output values*.

4 Проведение лабораторных работ

4.1 Порядок выполнения лабораторных работ

4.1.1 Общие действия при проведении локальной лабораторной работы

1. Проверить, что аккумуляторы надежно соединены проводами с платой питания. Проверить правильность подключения аккумуляторов: черный провод всегда соответствует минусу, поэтому он должен соединяться с черным проводом аккумулятора и входить в разъем платы с подписью “-”; провода с любым другим цветом (кроме белого, который тоже иногда соответствует “-”) обозначают полярность “+” и должны подключаться соответственно.

2. Включить БК. Проверить, что на макете зажглась зеленая лампочка – индикатор включения питания.

3. Включить стационарный компьютер. Подключиться к БК макета и запустить серверную часть программы.

4. Отключиться от БК макета и запустить клиентскую часть программы.

5. Настроить параметры соединения в области *Connect Properties*, выбрать свою роль (администратора – управляет проведением эксперимента или наблюдающего – просто получает данные, соответственно установкой или не установкой метки «As Admin Connect»). Далее нужно нажать кнопку «Connect» и дождаться состояния «Connected» в поле «Status».

4.1.2 Общие действия при проведении удаленной лабораторной работы

1. Запустить клиентскую часть программы.

2. Настроить параметры соединения в области *Connect Properties*, выбрать свою роль (администратора – управляет проведением эксперимента или наблюдающего – просто полу-

чает данные, соответственно установкой или не установкой метки «As Admin Connect»). Далее нужно нажать кнопку «Connect» и дождаться состояния «Connected» в поле «Status».

3. Для наблюдения за экспериментом по веб камере, выбрать вкладку *VideoCam* в правой части программы, настроить параметры камеры в области *Camera Settings* и нажать клавишу *Play/Stop*.

4. Для наблюдения за экспериментом по 3D-модели, выбрать вкладку *3DModel* в правой части программы. При наблюдении можно включить или отключить сглаживание движений кнопкой *Smoothing*.

4.1.3 Определение параметров моделей лопастных двигателей ориентации

Дальнейшие действия, связанные с запуском экспериментов или управлением макетом, доступны только при подключении в роли «Администратор». В роли наблюдателя возможно только получение данных и наблюдение за проведением эксперимента.

1. Выбрать вкладку *Lab1*.

2. Если макет неподвижен, кратковременно(<5 сек) включить один из вентиляторов, нажатием клавиши *Rotation Left* или *Rotation Right*. Для остановки нужно нажать клавишу *Stop*.

3. Запустить запись измерения датчика угловой скорости для одного полного колебания клавишей *Free Motion*. В области *Status Information* в окошке *Executed Program* должна появиться надпись «FreeMotion». А в других окнах должна отображаться информация о текущем времени эксперимента(*Current Time*), текущем угле поворота(*Current Angle*) и текущей угловой скорости макета(*Current Omega*). Зайти в меню *Graphics* и выбрать *Graph of Degree*. График будет отображать текущее значение угла поворота. Зайти в меню *Graphics* и выбрать *Graph of Rate of Rotation*. График будет отображать текущее значение выдаваемой датчиком угловой скорости и оценки угловой скорости фильтров Калмана. После получения данных одного полного колебания, для остановки нужно нажать клавишу *Stop*.

4. Включить на 2 минуты один из вентиляторов, например, для поворота против часовой стрелки, нажатием клавиши *Rotation Left*. В области *Status Information* в окошке *Executed Programm* должна появиться надпись «RotationLeft». Наблюдать за выполнением эксперимента можно по информации аналогичной пункту 3. После получения данных, для остановки нужно нажать клавишу *Stop*.

5. Повторить измерения пункта 4 для другого вентилятора, например, для поворота по часовой стрелке, нажатием клавиши *Rotation Right*. Время измерения нужно увеличить примерно в 2 раза, для того чтобы макет остановился и закрутился в другую сторону по отношению к пункту 4.

6. Найти в папке Results, согласно пункту 4.2.1 настоящего руководства, результаты измерений и скопировать себе.

7. По измерениям датчика угловой скорости определить параметры свободного движения макета $\tilde{\varepsilon}_1, \tilde{\varepsilon}_2$ из (2.18) с помощью методики, описанной в разделе 2.

8. Аналогично, по измерениям датчика угловой скорости определить параметры (2.8) и (2.16) обоих двигателей для двух моделей с помощью метода наименьших квадратов.

9. Проверить правдоподобность предположений (2.5) и (2.13).

10. По величине функционала (2.21) сравнить две модели лопастных двигателей.

11. Представить график с измерениями датчика угловой скорости и угловой скорости, которая следует из моделей движения, для обоих двигателей и для свободного движения.

4.1.4 Исследование алгоритмов управления ориентацией

Дальнейшие действия, связанные с запуском экспериментов или управлением макетом, доступны только при подключении в роли «Администратор». В роли наблюдателя возможно только получение данных и наблюдение за проведением эксперимента.

1. Выбрать вкладку Lab2.

2. Задать параметры лопастных двигателей в программе. Для этого нужно зайти в меню *Device Calibrations* и выбрать *Coolers(OptControl)*. Затем нужно задать значения параметров двигателей, соответствующих модели из раздела 1.1, и нажать кнопку *OK*.

3. Задать разворот на 180 градусов в окошке *Rotation Degree*.

4. Задать точность поддержания угла для оптимального алгоритма в окошке *Orientation Precision(deg)* в градусах.

5. Задать точность поддержания скорости для оптимального алгоритма в окошке *Speed Precision(deg/s)* в градус/секунда.

6. Запустить программу управлением нажатием на кнопку *Optimal Control*. В области *Status Information* в окошке *Executed Programm* должна появиться надпись «OptimalControl». А в других окнах должна отображаться информация о текущем времени эксперимента(*Current Time*), текущем угле поворота(*Current Angle*) и текущей угловой скорости макета(*Current Omega*). Зайти в меню *Graphics* и выбрать *Graph of Coolers*. График будет отображать текущее состояние работы вентиляторов. Зайти в меню *Graphics* и выбрать *Graph of Degree*. График будет отображать текущее значение угла поворота. Зайти в меню *Graphics* и выбрать *Graph of Rate of Rotation*. График будет отображать текущее значение выдаваемой датчиком угловой скорости и оценки угловой скорости фильтров Калмана. Пре-

кратить программу, когда визуально аппарат выйдет на предельный цикл, нажатием клавиши *Stop*.

7. Задать коэффициенты k_ω и k_α ПД-регулятора, соответствующие максимальной степени устойчивости, которые оценить по формулам (2.45), (2.48). Для этого нужно зайти в меню *Device Calibrations* и выбрать *Coolers(PDReg)*. Затем нужно задать значения и нажать кнопку *OK*.

8. Задать разворот на 180 градусов в окошке *Rotation Degree*.

9. Запустить программу управлением нажатием на кнопку *PD Regulator*. В области *Status Information* в окошке *Executed Program* должна появиться надпись «PDRegulator». А в других окнах должна отображаться информация о текущем времени эксперимента(*Current Time*), текущем угле поворота(*Current Angle*) и текущей угловой скорости макета(*Current Omega*). Зайти в меню *Graphics* и выбрать *Graph of Coolers*. График будет отображать текущее состояние работы вентиляторов. Зайти в меню *Graphics* и выбрать *Graph of Degree*. График будет отображать текущее значение угла поворота. Зайти в меню *Graphics* и выбрать *Graph of Rate of Rotation*. График будет отображать текущее значение выдаваемой датчиком угловой скорости и оценки угловой скорости фильтров Калмана. Прекратить программу, когда визуально аппарат выйдет на предельный цикл, нажатием клавиши *Stop*.

10. Повторить пункты 7-9 с коэффициентами k_ω и k_α ПД-регулятора, при которых корни характеристического уравнения (2.47) будут мнимыми.

11. Построить в Матлабе графики траектории движения на фазовой плоскости и определить точность единичного маневра, время маневра и точность поддержания положения (определить по размерам предельного цикла) для двух алгоритмов. Для ПД-регулятора точность единичного маневра считать в моменты, когда траектория впервые достигнет точек в которых угловая скорость и рассогласование по углу равны 0 (значения угла и угловой скорости в соответственных точках будут значениями точностей маневра).

4.2 Результаты проведения лабораторной работы "Определение параметров лопастных двигателей ориентации"

При проведении лабораторной работы "Определение параметров лопастных двигателей ориентации" преследуется цель получить навык работы с лабораторным оборудованием; получить оценку параметров моделей лопастных двигателей, сравнить две модели. Лабораторная работа выполнялась в соответствии с порядком, описанным в разделе 4.1.3.

После получения данных эксперимента, с помощью метода наименьших квадратов, описанного в разделе 2.1.4, получены параметры моделей движения, сведенные в табл. 4.1.

Табл. 4.1 Параметры моделей лопастных двигателей

| Параметр | 1-й двигатель | 2-й двигатель |
|------------------------------|-------------------------|-------------------------|
| A | 3.75 | -3.67 |
| B | 0.016 | 0.015 |
| $\Phi(A, B)$ | $0.50 (\text{рад/с})^2$ | $0.71 (\text{рад/с})^2$ |
| \tilde{A} | 3.98 | -3.91 |
| \tilde{B} | 0.020 | 0.019 |
| $\Phi(\tilde{A}, \tilde{B})$ | $1.97 (\text{рад/с})^2$ | $4.54 (\text{рад/с})^2$ |

Как видно из табл. 4.1 значения параметров A для модели на основе закона Бернулли несколько отличаются по модулю, что означает, что величина максимальной угловой скорости, до которой лопастной двигатель может разогнать макет разные. Это может быть обусловлено различием в трении между лопастями и осью двигателя, что влияет на скорость вращения самих лопастей. Параметры B этой модели отличаются незначительно.

Параметры модели на основе закона сохранения импульса \tilde{A} , \tilde{B} достаточно значительно отличаются от параметров модели A , B , хотя \tilde{A} также несет смысл максимальной угловой скорости макета при работающем двигателе. Но если сравнить величины функционалов (сумма квадратов разностей), то видно, что модель на основе закона сохранения импульса хуже "ложится" на результаты экспериментов. Это означает, что модель на основе закона Бернулли более правдоподобная.

На Рис. 4.1 и Рис. 4.2 представлены графики измерений угловой скорости и угловой скорости, полученной по моделям движения с параметрами из табл. 4.1, для 1-го и 2-го вентиляторов, соответственно. Визуально видно, что угловая скорость по модели на основе закона сохранения импульса заметно отличается от измеренной в ходе экспериментов.

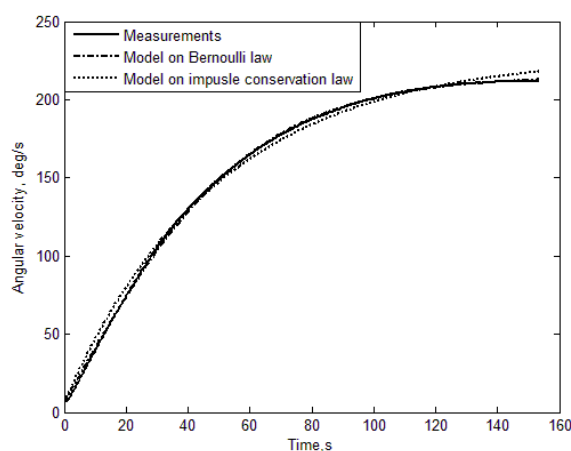


Рис. 4.1 Графики измерений угловой скорости и угловой скорости, полученной по моделям движения для 1-го вентилятора

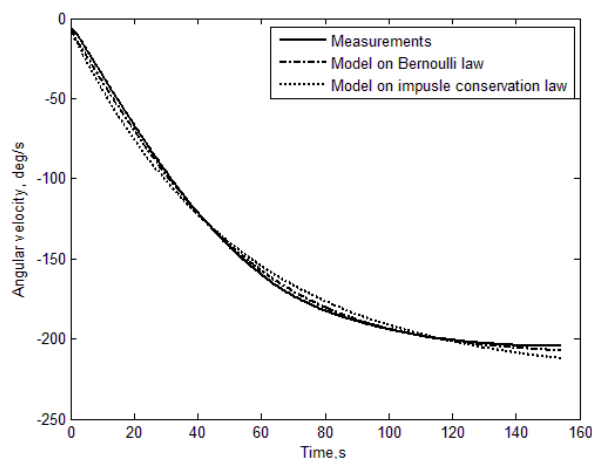


Рис. 4.2 Графики измерений угловой скорости и угловой скорости, полученной по моделям движения для 2-го вентилятора

По полученным измерениям угловой скорости с помощью метода наименьших квадратов были получены параметры свободного движения (2.19)

$$\tilde{\varepsilon}_1 = 0.0022, \quad \tilde{\varepsilon}_2 = 0.0002.$$

Далее, используя соотношения (2.16) и величины параметров из табл. 4.1 можно проверить правильность предположений (2.5):

$$|k| \gg \varepsilon_1, \quad |k| \gg \varepsilon_2.$$

Таким образом, например, для модели на основе закона Бернулли для 1-го вентилятора:

$$k = B / A = 4,2 \cdot 10^{-3}, \quad \varepsilon_1 = \tilde{\varepsilon}_1 / A = 5,8 \cdot 10^{-4}, \quad \varepsilon_2 = \tilde{\varepsilon}_2 / A = 5,3 \cdot 10^{-5},$$

то есть предположения (2.5) выполнены, отличие модуля k на порядок от ε_1 и на два порядка от ε_2 .

На Рис. 4.3 представлен график измерений угловой скорости во время свободного движения и угловая скорость, полученная по модели свободного движения. Модельные значения несколько отличаются от измеренных, так как на тело, подвешенное на струне действуют некоторые возмущения, неучтенные в модели движения. Но, в целом, характер кривой обуславливает затухающие колебания, что позволяет с помощью метода наименьших квадратов оценить параметры свободного движения.

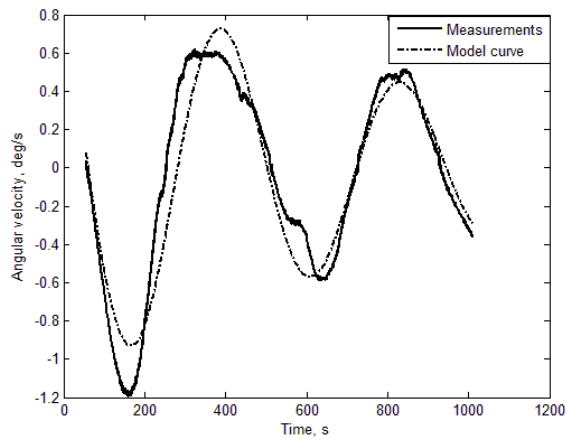


Рис. 4.3 График измерений угловой скорости и угловая скорость, полученная по модели свободного движения

4.3 Результаты проведения лабораторной работы: "Исследование алгоритмов управления ориентацией"

При проведении лабораторной работы "Определение параметров лопастных двигателей ориентации" преследуется цель получить оценку точности работы алгоритмов управления, сравнить качество работы алгоритма на основе ПД-регулятора и оптимального по быстродействию алгоритма. Лабораторная работа выполнялась в соответствии с порядком, описанным в разделе 4.1.4.

На Рис. 4.4 представлен график траектории на фазовой плоскости при работе оптимального по быстродействию управления, на Рис. 4.5 представлен график управления в зависимости от времени. Из-за того, что в алгоритме управления неучтен импульс последствия траектория выходит на предельный цикл. По размерам предельного цикла можно оценить точность поддержания заданной ориентации, которая составляет около 15 градусов и по скорости около 7 град /сек.

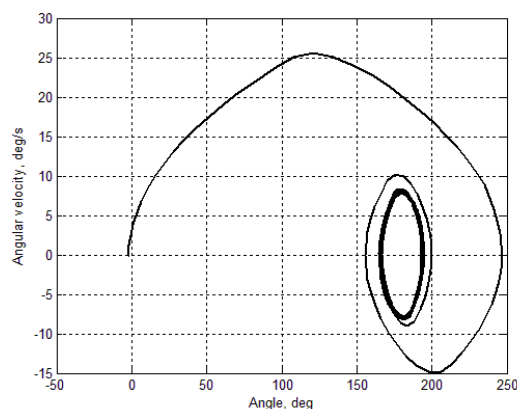


Рис. 4.4 Траектория на фазовой плоскости при работе алгоритма оптимального управления

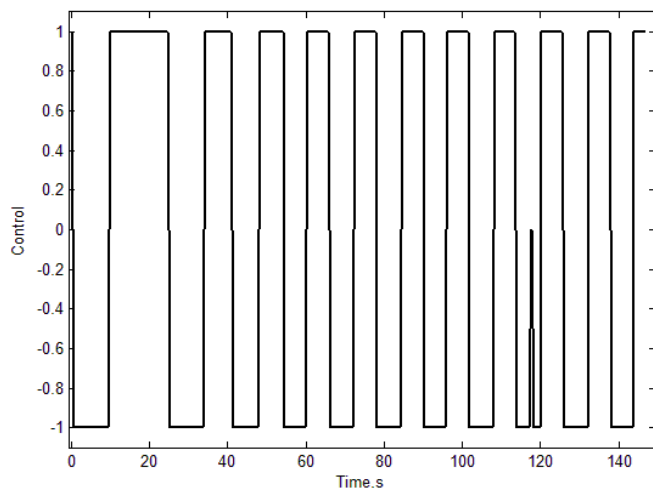


Рис. 4.5 Управление в зависимости от времени при работе алгоритма оптимального управления

На Рис. 4.6 представлен график работы ПД-регулятора с коэффициентами, обуславливающими затухающие колебания (здесь взяты $k_\alpha = 0.04, k_\omega = 0.2$), на Рис. 4.7 представлен график зависимости управления от времени. Вследствие невысокой степени устойчивости (есть мнимые корни у характеристического многочлена (2.46)), траектория макета достаточно продолжительно приходит в окрестность заданного положения -180 град. Но после прихода в нужную точку на фазовой плоскости точность поддержания составляет порядка 3^0 по углу и $1^0/\text{с}$ по угловой скорости.

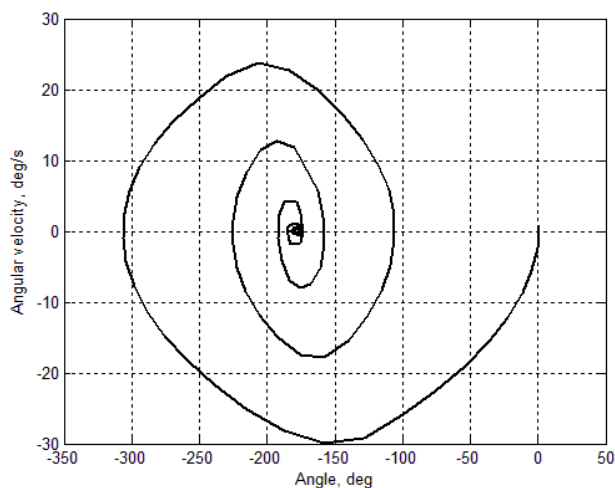


Рис. 4.6 Траектория на фазовой плоскости при работе ПД-регулятора с коэффициентами, обеспечивающими затухающие колебания

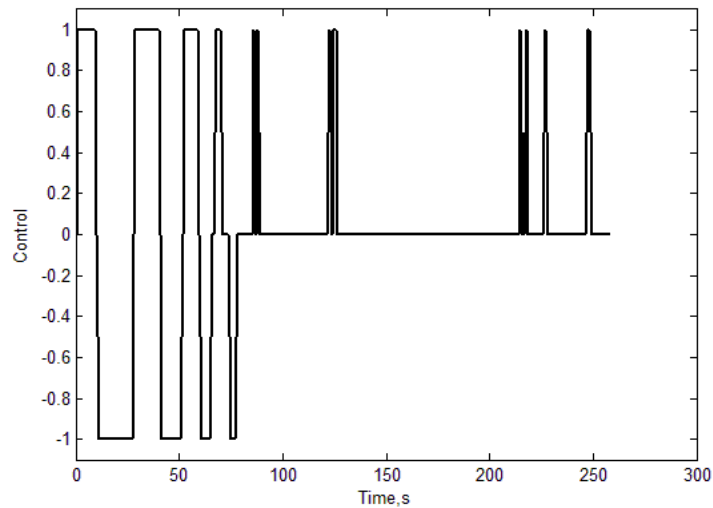


Рис. 4.7 Управление в зависимости от времени при ПД-регулятора

При задании коэффициентов ПД-регулятора, обеспечивающих максимальную степень устойчивости $k_\alpha = 0.01, k_\omega = 0.2$ (мнимая часть корней характеристического члена равна нулю (2.48), корни имеют только отрицательную действительную часть), траектория на фазовой плоскости по траектории, близкой к экспоненциальной, приходит в заданную точку на фазовой плоскости (см. Рис. 4.8) за время приблизительно 50 с. График управления представлен на Рис. 4.9. При поддержании положения достигается точность 1.0^0 по положению и $0.5^0/\text{с}$ по угловой скорости.

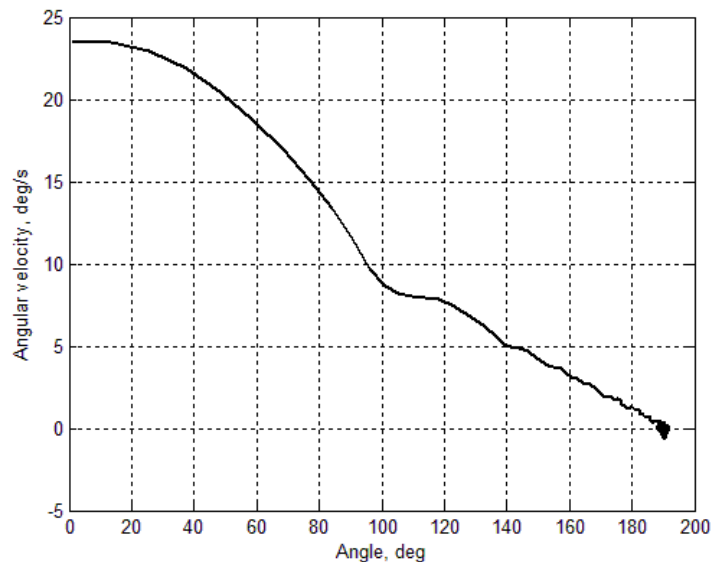


Рис. 4.8 Траектория на фазовой плоскости при работе ПД-регулятора с коэффициентами, обеспечивающими максимальную степень устойчивости

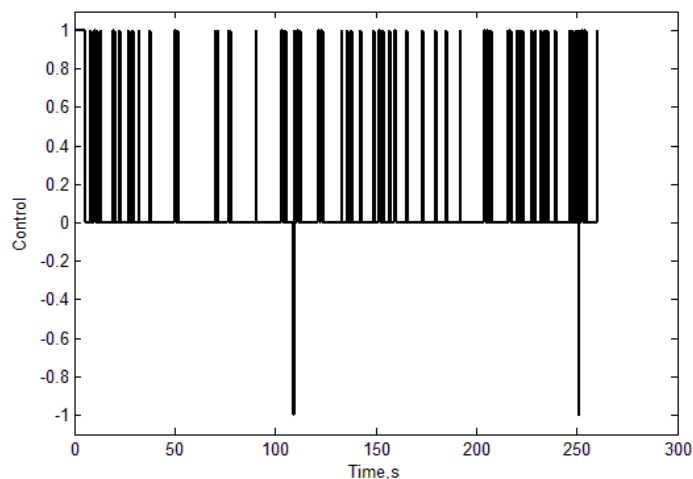


Рис. 4.9 Управление в зависимости от времени при работе ПД-регулятора

Все характеристики алгоритмов управления сведены в табл. 4.2

Табл. 4.2 Характеристики работы алгоритмов управления ориентацией

| Параметр | Оптимальный алгоритм | ПД-регулятор 1 | ПД-регулятор 2 |
|--|----------------------|-----------------|------------------|
| Точность единичного маневра поворота на 180^0 , градусы и угловая скорость | $25^0, 15^0/с$ | $100^0, 20^0/с$ | $5^0, 0.7^0/с$ |
| Время маневра, с | 25 | 25 | 50 |
| Точность поддержания положения, градусы и угловая скорость | $15^0, 7^0/с$ | $3^0, 1^0/с$ | $1.0^0, 0.5^0/с$ |

Таким образом, оптимальный алгоритм имеет самую низкую точность вследствие того, что в модели вентиляторов неучтен импульс последствия, но у него самое маленькое время единичного маневра. При работе ПД-регулятора с коэффициентами, обеспечивающими затухающие колебания, точность единичного маневра намного хуже оптимального управления при том же времени, но точность поддержания значительно лучше. Если же выбрать коэффициенты, обеспечивающие максимальную степень устойчивости, то ПД-регулятор достаточно долго приводит систему в нужное положение, но точность поддержания самая высокая.

Заключение

В настоящей работе разработана система дистанционного управления макетом системы ориентации малых аппаратов, который имеется на кафедре теоретической механики МФТИ. Разработаны и внедрены в технический курс по выбору "Управление и динамика сложных информационно-механических систем" лабораторные работы по определению параметров моделей лопастных двигателей и исследованию качества работы алгоритмов управления ориентацией. Приведена теоретическая база к лабораторным работам, создана программа, позволяющая проводить лабораторные работы удаленно и получать результаты измерений.

В результате проведения лабораторных работ было показано, что модель лопастного двигателя на основе закона Бернулли более правдоподобна, чем модель на основе закона сохранения импульса, были вычислены параметры свободного движения макета и показано, что сопротивлением воздуха и кручением нити можно пренебречь при управляемом движении. При проведении второй лабораторной работы показано, что оптимальный алгоритм выводит траекторию движения макета на предельный цикл вследствие влияния импульса последствия двигателей и имеет худшую точность, чем ПД-регулятор.

Список литературы

1. Евдокимов Ю.К., Маливанов Н.Н., Кирсанов А.. Разработка и создание региональной дистанционной автоматизированной учебной лаборатории коллективного пользования по общетехническим дисциплинам для ВУЗов и средних учебных заведений республики Татарстан на базе технологии National Instruments // Сб. тр. Междунар. науч. практич. конф. «Образовательные, научные и инженерные приложения в среде LabVIEW и технологии National Instruments». Москва. 17-18 ноября, 2006. 2006. С. 5-10.
2. Баран Е.Д., Захаров П.М., Любенко А.Ю. Web-лаборатория «Микроконтроллеры и сигнальные процессоры» // Современные технологии автоматизации. 2005. Т. 1. С. 128-135.
3. Дунаев С.П. Технологии Интернет-программирования. СПб.: БХВ- Петербург, 2001.
4. Кирсанов А.Ю. Методика экспериментального исследования динамики работы системы дистанционного управления экспериментом через информационные сети общего пользования // Электронное приборостроение. Научно-практический сборник. 2005. № 3(44). С. 42-49.
5. Назаров А.А. Исследование компьютерных сетей связи с протоколами случайного множественного доступа // Вестник Томского государственного университета. 2000. № 271. С. 72-73.
6. Balestra A. Remote Control for Galeleo Telescope and the EU // «The Three Galileos Conference". Padova, Italy, 7-10 Jan. 1997.
7. Buckman B. VI-Based Introductory Electrical Engineering Laboratory Course // International Journal of Engineering Education. 2000. Т. 16. № 3. С. 25-30.
8. Liu J. Distributed Distance Measurement for Large-Scale Networks // Computer Networks: The International Journal of Computer and Telecommunications. 2003. Т. 41. № 2. С. 177-192.
9. Ponomarev D.U. Tensor Analysis for Investigation Next Generation Networks // Proceedings of IEEE International Siberian Conference on Control and Communications, Tomsk, 20 november. 2005. С. 53-57.
10. Образовательные, научные и инженерные приложения в среде LabVIEW и технологии National Instruments. Сб. тр. Междунар. науч.-практич. конф. «Образовательные, научные и инженерные приложения в среде LabVIEW и технологии National Instruments». М.: , 2006.

11. Ferreira J.M. The PEARL Digital Electronics Lab: Full Access to the Workbench via the Web // Proceedings of the 13th Annual Conference in Education - European Association for Education in Electrical and Information Engineering. , 2002.
12. Fjeldly T.A. Lab-on-Web - A Comprehensive Electronic Device Laboratory on a Chip Assembled via Internet // International Conference on Engineering Education. August 18-19. Manchester, U.K.: , 2002. C. 1-5.
13. Schwars J. Remote Lab: A Reliable Tele- Laboratory Environment // International Conference on Internet Computing, January. , 2000.
14. Schafe T. PERL: A Generic Architecture for Live Experiments in a Remote Laboratory // SCS International Conference on Simulation and Multimedia in Engineering Education, ICSEE'03, January 2003.
15. Saliyah H.H. Resource Management Strategies for Remote Virtual Laboratory Experimentation // 2000 Frontier in Education Conference: Building a Century of Progress in Engineering Education", October 18-21. Kansas City, USA: , 2000.
16. Rugej J. The Application Of Information Services In Computer Networks To Education // IEEE Instrumentation and Measurement Technology Conference. May 18-21. St. Paul, Minnesota: , 1998.
17. Иванов Д.С., Овчинников М.Ю., Ткачев С.С. Управление ориентацией твердого тела, подвешенного на струне, с использованием вентиляторных двигателей // Известия РАН. Теория и системы управления. 2011. Т. 50. № 1. С. 107-119.
18. Иванов Д.С. и др. Калибровка датчиков для определения ориентации малого космического аппарата // Препринт ИПМ им. М.В. Келдыша РАН. 2010. № 28. С. 32.
19. Пантелеев В.Л. Наблюдение и управление динамическими системами. М.: Изд-во Московского университета, 2000.
20. Понтрягин Л.С. Избранные труды. М.: Наука, 1989.
21. Иванов Д.С. и др. Испытания алгоритмов управления ориентацией микроспутника "Чибис-М" на лабораторном стенде // Известия РАН. Теория и системы управления. 2012. № 1. С. 118-137.
22. Барбашин Е.А. Введение в теорию устойчивости. М.: Наука, 1967.

Приложение А

Список публикаций Нуждина Дмитрия Олеговича, студента 6 курса МФТИ (ГУ)

1. D. Ivanov, M. Ovchinnikov, D. Nuzhdin, S. Tkachev. Balloon's payload attitude control system with propeller thruster use. // Proceedings of Taiwan-Russian bilateral symposium on problems in advanced mechanics. М.: Изд-во МГУ, 2010. С. 85-92.

2. Д.С.Иванов, М.Ю.Овчинников, С.П.Трофимов, Д.О.Нуждин. Лабораторный стенд для отработки алгоритмов определения ориентации и навигации, основанных на обработке видеоизображений объекта и звездного неба. // Сборник тезисов второй всероссийской научно-технической конференции «Современные проблемы определения ориентации и навигации космических аппаратов», г. Таруса, 13-17 сентября, 2010, с 36.

3. Д.С.Иванов, М.Ю.Овчинников, С.П.Трофимов, Д.О.Нуждин. Лабораторный стенд для отработки алгоритмов определения ориентации и навигации, основанных на обработке видеоизображений объекта и звездного неба. // Сборник трудов второй всероссийской научно-технической конференции «Современные проблемы определения ориентации и навигации космических аппаратов» под ред. Г. А. Аванесова, г. Таруса, 13-17 сентября, 2010. М:ИКИ РАН, 2011, с 215-225.

4. Иванов Д.С., Нуждин Д.О., Егоров К.В. Лабораторное моделирование алгоритмов определения и управления ориентацией микроспутников// Сборник тезисов второй всероссийской научно-технической конференции «Современные проблемы определения ориентации и навигации космических аппаратов», г. Таруса, 13-17 сентября, 2010, с 34-35.

5. Иванов Д.С., Нуждин Д.О., Егоров К.В. Лабораторное моделирование алгоритмов определения и управления ориентацией микроспутников// Сборник трудов второй всероссийской научно-технической конференции «Современные проблемы определения ориентации и навигации космических аппаратов», г. Таруса, 13-17 сентября 2010. М:ИКИ РАН, 2011, с 239-247.

6. Иванов Д.С., Ткачев С.С., Ролдугин Д.С., Трофимов С.П., Нуждин Д.О. Лабораторные испытания алгоритмов управления спутников в задаче обучения молодых специалистов// Сборник тезисов всероссийской конференции с элементами научной школы для молодежи «Специфика формирования сети научно-образовательных центров. Сетевое взаимодействие молодых исследователей в рамках информационного обмена НОЦ», 12 ноября, Москва, 2010, с. 56.

7. Д.О.Нуждин, С.С.Ткачев, Д.С.Иванов, Дистанционное управление лабораторным стендом в целях обучения// Труды 53-й научной конференции МФТИ “Современные проблемы фундаментальных и прикладных наук”. Часть III. Аэрофизика и космические исследования. Том 1. — М.: МФТИ, 2010. — с.108-109.
8. Д.С. Иванов, С.П. Трофимов, Д.О. Нуждин, Система определения положения и ориентации макета подвижного объекта на основе блока инерциальных датчиков и звездной камеры// Труды 53-й научной конференции МФТИ “Современные проблемы фундаментальных и прикладных наук”. Часть VII. Управление и прикладная математика. Том 3. — М.: МФТИ, 2010. — с.57-58.
9. Д.С. Иванов, С.С. Ткачев, Д.С. Ролдугин, С.П. Трофимов, Д.О. Нуждин С.О.Карпенко. Аналитическое, численное и полунатурное исследование алгоритмов управление ориентацией микроспутников. Сборник тезисов докладов Второй всероссийской школы молодых ученых-механиков "Актуальные проблемы механики" в рамках X всероссийского съезда по фундаментальным проблемам теоретической и прикладной механики, Н. Новгород, 24-30 августа 2011 г., с. 45-46.
10. Д.О. Нуждин, С.С.Ткачев, Д.С. Иванов. Реализация алгоритма определения направления на Солнце с помощью экспериментального солнечного датчика на ПЗС-линейках. // Труды 54-й научной конференции МФТИ “Проблемы фундаментальных и прикладных естественных и технических наук в современном информационном обществе”. Часть III. Аэрофизика и космические исследования. Том 1. — М.: МФТИ, 2011. — с.171-173.
11. Д.О. Нуждин, С.С.Ткачев, Д.С. Иванов. Дистанционная лаборатория. Труды 54-й научной конференции МФТИ "Современные проблемы фундаментальных и прикладных наук", 25-26 ноября 2011г. Долгопрудный. Часть VII. Управление и прикладная математика. - М.: МФТИ, 2011, с.42-43.
12. Д.С. Иванов, С.С. Ткачев, Д.С. Ролдугин, С.П. Трофимов, Д.О. Нуждин С.О.Карпенко. Аналитическое, численное и полунатурное исследование алгоритмов управление ориентацией микроспутников// Вестник Нижегородского университета им. Н.И. Лобачевского, 2011, № 4 (2), с. 152–154
13. Д.Биндель, Д.С.Иванов, Д.О.Нуждин, М.Ю.Овчинников, С.П.Трофимов. Система определения положения и ориентации макета спутника на основе блока инерциальных датчиков и звездного датчика/ Препринт ИПМ им. М.В. Келдыша РАН. – М., 2011. – № 24. – 30 с.