

Министерство образования и науки Российской Федерации  
Ярославский государственный университет им. П. Г. Демидова

# СОВРЕМЕННЫЕ ПРОБЛЕМЫ МАТЕМАТИКИ И ИНФОРМАТИКИ

*Сборник научных трудов  
молодых ученых, аспирантов и студентов*

ВЫПУСК 19

Ярославль  
ЯрГУ  
2019

УДК 51+004(060.55)  
ББК В1я43+3973.2я43  
С 56

*Рекомендовано  
редакционно-издательским советом ЯрГУ  
в качестве научного издания. План 2019 года*

**Современные проблемы математики и информатики :**

С 56 сборник научных трудов молодых ученых, аспирантов и студентов  
/ Яросл. гос. ун-т им. П. Г. Демидова. — Ярославль : ЯрГУ, 2019.  
— Вып. 19. — 60 с.  
ISBN 978-5-8397-1186-0

В сборнике представлены работы молодых ученых, аспирантов и студентов.

В статьях рассматриваются различные проблемы теории динамических систем, информационных технологий, разработки программных средств и вычислительной математики.

Сборник подготовлен с использованием издательской системы ЛАТ<sub>E</sub>X.

ББК В1я43+3973.2я43  
УДК 51+004(060.55)

**Редакционная коллегия:**

канд. физ.-мат. наук Д. В. Глазков (отв. редактор)  
д-р физ.-мат. наук И. С. Кащенко  
канд. физ.-мат. наук П. Н. Нестеров

# Содержание

<i>Алферов Р. И.</i> Сравнение эффективности средств защиты информации от несанкционированного доступа . . . . .	4
<i>Гайдук Э. В.</i> Периодические решения одного класса функциональных уравнений . . . . .	16
<i>Гибадulin Р. А.</i> Алгоритм поиска вывода в Исчислении Высказываний и его программная реализация . . . . .	28
<i>Кудрявцев И. С.</i> Линейная интерполяция на шаре в $\mathbb{R}^n$ . . . . .	38
<i>Маслеников И. Н.</i> Локальная динамика модели оптико-электронного осциллятора . . . . .	44
<i>Федулов Д. Д., Ухалов А. Ю.</i> Оценки минимального коэффициента поглощения $n$ -мерного симплекса . . . . .	53

© Алферов Р. И., 2019

УДК 336:004

**Р. И. Алферов**

## Сравнение эффективности средств защиты информации от несанкционированного доступа

Цель работы – выработка критериев оценки для сравнения эффективности средств защиты информации от несанкционированного доступа и исследование некоторых сертифицированных средств защиты информации на предмет соответствия этим критериям.

В результате были разработаны критерии оценки эффективности и проведен сравнительный анализ эффективности некоторых средств защиты информации от несанкционированного доступа.

### Введение

В настоящее время, во время жесткой информационной войны, объекты информационной инфраструктуры Российской Федерации каждый день сталкиваются с большим количеством компьютерных атак. Для обеспечения устойчивого функционирования информационной инфраструктуры России при проведении в отношении нее компьютерных атак необходимо использовать технические, программные, программно-аппаратные средства противодействия компьютерным атакам. Одним из важных составляющих комплекса программных средств защиты информации являются средства защиты информации от несанкционированного доступа (СЗИ от НСД).

Во многих государственных и коммерческих организациях России (в том числе занимающихся хранением и обработкой конфиденциальной информации, а также сведений, составляющих государственную тайну) используются операционные системы (ОС) семейства Windows от компании Microsoft<sup>1</sup>. Поэтому сравнение эффективности работы представленных на российском рынке СЗИ от НСД для ОС семейства Windows является довольно актуальной задачей. В статье рассматриваются понятие НСД, понятия нарушителя и модели нарушителя, методы осуществления НСД, приводится сравнительный анализ эффективности работы некоторых СЗИ от НСД для ОС семейства Windows.

---

<sup>1</sup>Штаб-квартира компании Microsoft находится в городе Редмонд, штат Вашингтон, США.

## Основные понятия

Согласно ГОСТу [2] **несанкционированный доступ к информации (НСД)** — это получение защищаемой информации заинтересованными субъектами с нарушением установленных нормативными и правовыми документами (актами) или обладателями информации прав или правил управления доступом к защищаемой информации. Заинтересованными субъектами, осуществляющими НСД к защищаемой информации, могут быть:

- государство;
- юридическое лицо;
- группа физических лиц или одно физическое лицо;
- общественная организация.

Проблеме защиты информации от НСД посвящен ряд руководящих документов (РД) Гостехкомиссии России. Определение НСД в этих документах более узкое, чем в ГОСТе [2]. В соответствии с руководящим документом [3] **несанкционированный доступ к информации** — доступ к информации, нарушающий правила разграничения доступа с использованием штатных средств, предоставляемых средствами вычислительной техники (СВТ) или автоматизированными системами (АС). Под штатными средствами понимается совокупность программного и технического обеспечения средств вычислительной техники или информационных систем. Однако действие внедренной троянской программы, результатом которого стало попадание защищаемой информации к нарушителю, также можно рассматривать как факт НСД.

Далее по тексту вместо понятия «автоматизированная система» мы будем использовать более узкое понятие «информационная система» (ИС), т. к. в рамках темы работы будут рассматриваться вопросы технического характера, не связанные с воздействием на персонал автоматизированной системы в целях получения доступа к защищаемой информации.

Таким образом, НСД заключается в получении заинтересованным субъектом доступа к объекту, на который у него нет разрешения в соответствии с принятой в организации политикой безопасности, т. е. в отношении доступа к данному объекту нарушитель является нелегитимным пользователем. Основная проблема заключается в том, чтобы определить, кто и к каким наборам данных должен иметь доступ, а кто нет. Другими словами, необходимо определить объем понятия «несанкционированный».

По характеру воздействия НСД является активным воздействием, использующим любую ошибку в архитектуре и средствах информационной системы. Объектом НСД обычно выступает либо непосредственно требуемый набор данных (защищаемая информация), либо технологическая информация о санкционированном доступе, целью воздействия на которую является легализация НСД в контексте операционной среды. НСД может быть подвержен любой объект системы. НСД может быть осуществлен как стандартными, так и специально разработанными программными средствами к объектам в любом состоянии.

К основным событиям, следствием которых может быть НСД к информации, можно отнести следующие:

- проникновение в операционную среду компьютера с использованием штатного программного обеспечения (средств операционной системы или прикладных программ общего назначения);
- создание нештатных режимов работы программных (программно-аппаратных) средств за счет преднамеренных изменений служебных данных, игнорирования предусмотренных в штатных условиях ограничений на состав и характеристики обрабатываемой информации, искажения (модификации) самих данных и т. п.;
- внедрение вредоносных программ.

Методика реализации НСД в значительной мере зависит от организации обработки информации в ИС, применяемой в ИС политики безопасности, возможностей установленных средств защиты, а также добросовестности администраторов и пользователей ИС.

Для реализации НСД специалисты выделяют несколько способов:

- Наблюдение за тем, что «плохо лежит», то есть какие наборы данных, представляющие интерес для злоумышленника, открыты для доступа по недосмотру или умыслу администратора безопасности, а также в результате некорректной настройки СЗИ от НСД. Такой доступ, хотя и с некоторой натяжкой, тоже можно назвать несанкционированным, его легко осуществить (путем непосредственного обращения к объектам доступа с использованием штатных средств ОС), но от него легко (с точки зрения технических мер) и защититься.
- Обход СЗИ от НСД, то есть создание программного средства, выполняющего обращение к объектам доступа некоторым нестандартным, непредусмотренным и неотслеживаемым СЗИ от НСД способом.

- Модификация средств защиты, позволяющая осуществить НСД, то есть, например, несанкционированное изменение правил разграничения доступа, модификация (патчинг) исполняемых файлов СЗИ от НСД, удаление файлов СЗИ от НСД с целью достижения его неработоспособности.
- Внедрение программных и аппаратных закладок.

В подавляющем большинстве случаев НСД становится возможным из-за непродуманного выбора средств защиты, их некорректной установки и настройки, отсутствия контроля (мониторинга) их функционирования, наличия в них уязвимостей, а также при небрежном отношении к защите данных.

**Защита от НСД** (согласно руководящему документу [3]) – предотвращение или существенное затруднение НСД. Защита от НСД может осуществляться через применение программно-технических средств и организационных мер. В общем случае, комплекс программно-технических средств и организационных мер по защите информации от НСД реализуется в рамках системы защиты информации от НСД, условно состоящей из следующих четырех основных подсистем (из руководящего документа [6]):

- управления доступом;
- регистрации и учета;
- криптографической;
- обеспечения целостности.

Пример архитектуры СЗИ от НСД (Dallas Lock 8.0-C), включающий перечисленные выше подсистемы, представлен на рисунке 1.

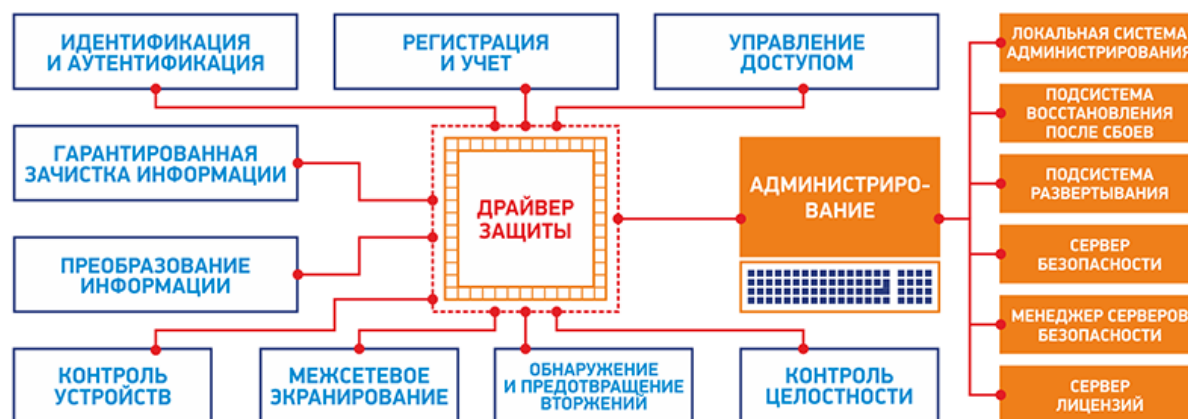


Рис. 1: Архитектура СЗИ от НСД на примере программного комплекса средств защиты информации Dallas Lock 8.0-C

В данной работе не рассматриваются организационные меры защиты, поэтому далее будем использовать для описания анализируемого программного обеспечения термин «средство защиты информации от несанкционированного доступа». В данной работе **средством защиты информации от несанкционированного доступа** мы будем называть программное средство, предназначенное для предотвращения или существенного затруднения несанкционированного доступа.

Источником НСД могут быть действия нарушителя. Рассмотрим понятия нарушителя и модели нарушителя.

## Модель нарушителя

**Нарушитель** — это лицо, предпринявшее попытку выполнения запрещенных операций (действий) по ошибке, незнанию или осознанно со злым умыслом (из корыстных интересов) или без такового (ради игры или удовольствия, с целью самоутверждения и т. п.) и использующее для этого различные возможности, методы и средства.

Следует заметить, что при отсутствии правил (запретов и ограничений) не существует и нарушителей (если нет правил, значит, нечего и нарушать). Поэтому борьба с нарушениями всегда должна начинаться с установления четких правил (ограничений, политики безопасности).

**Злоумышленник** — это нарушитель, намеренно (умышленно, со злым умыслом) идущий на нарушение. В контексте данной работы злоумышленником (нарушителем) будет считаться субъект доступа, осуществляющий НСД к информации.

Определение конкретных значений характеристик возможных нарушителей в значительной степени субъективно. Модель нарушителя, построенная с учетом особенностей конкретной предметной области и технологии обработки информации, может быть представлена перечислением нескольких вариантов нарушителей.

Модель нарушителя тесно связана с моделью угроз и, по сути, является ее частью. Модель угроз содержит максимально полное описание угроз безопасности объекта. **Модель нарушителя** содержит описание предположений о возможностях нарушителя, которые он может использовать для разработки и проведения атак, а также об ограничениях на эти возможности.

В качестве нарушителей мы будем рассматривать нарушителей первого, второго и третьего уровня возможностей из классификации нарушителей руководящего документа [5].

В качестве нарушителя первого уровня можно привести в пример пользователя, работающего в замкнутой программной среде. Он мо-



жет запускать программы из некоторого фиксированного набора (белого списка, списка разрешенных приложений).

Примером нарушителя второго уровня является пользователь, не имеющий административных привилегий и права доступа к защищаемым объектам, запускающий вредоносное программное обеспечение (ПО), а также эксплойты, которые используют уязвимости как компонентов ядра операционной системы, так и уязвимости прикладного ПО.

Примером нарушителя третьего уровня является системный администратор (ИТ-администратор), который имеет возможность влиять на функционирование ОС, возможно, может устанавливать и запускать свои драйверы, имеет соответствующие привилегии, но не может настраивать, удалять СЗИ от НСД, не имеет прав доступа к защищаемым объектам. Согласно аналитическому исследованию компании «Код безопасности» (декабрь 2016) [7] в 69% организаций за информационную безопасность отвечает профильный сотрудник, т. е. выделенный администратор безопасности (или целое подразделение, отвечающее за информационную безопасность). В таких организациях отдельно существует подразделение, занимающееся системным и сетевым администрированием (ИТ-администраторы), но не отвечающее за информационную безопасность. Процент организаций, имеющих профильных специалистов по ИБ, от общего количества организаций, опрошенных в рассматриваемой отрасли, приведен на диаграмме 2.

Профильные специалисты в различных отраслях

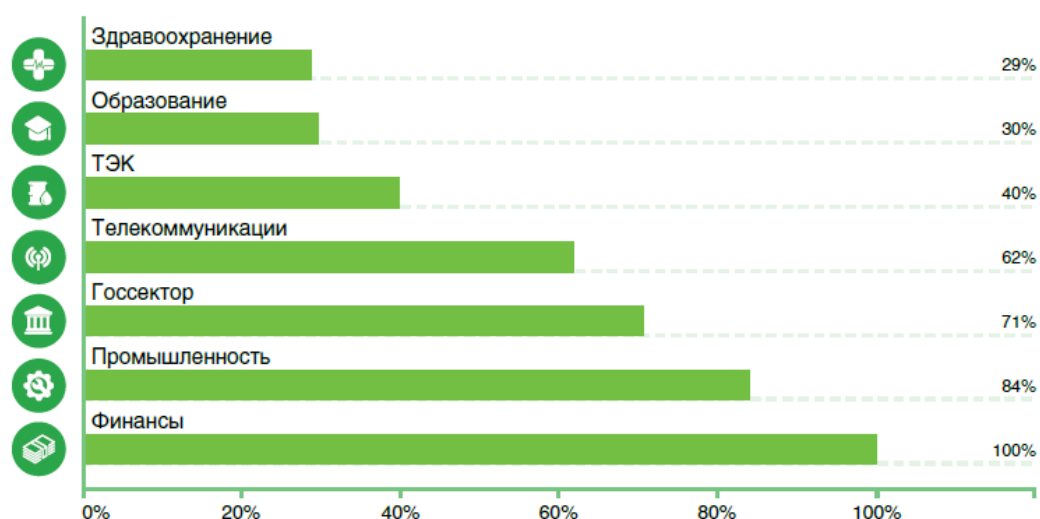


Рис. 2: Профильные специалисты в различных отраслях

Из диаграммы видно, что нарушитель третьего уровня возможностей является наиболее актуальным для таких отраслей как промышленность и финансы.

Нарушители четвертого уровня возможностей не рассматриваются, т. к. сравниваемые в данной работе СЗИ от НСД предназначены для защиты от нарушителей с более низкой мотивацией и потенциалом, чем подразумевается для четвертого уровня нарушителей, что отражено в формулярах на исследуемые СЗИ от НСД. Также действует предположение 4.3 руководящего документа [5]: «В своем уровне нарушитель является специалистом **высшей** квалификации, знает все об АС и, в частности, о системе и средствах ее защиты».

## Критерии оценки эффективности

Согласно руководящему документу [5]: «**Неотъемлемой** частью работ по защите является **оценка эффективности средств защиты**, осуществляемая по **методике**, учитывающей всю совокупность технических характеристик оцениваемого объекта, включая технические решения и практическую реализацию средств защиты.». Для формирования такой методики оценки эффективности СЗИ от НСД необходимо разработать перечень критериев для проверки соответствия требованиям руководящего документа [4], а также для выявления наличия недостатков и уязвимостей средств защиты.

Для сравнения эффективности были выбраны СЗИ от НСД, являющиеся сертифицированными по третьему классу защищенности СВТ, т. е. полностью соответствующие требованиям руководящего документа [4] для данного класса защищенности. Поэтому из требований к показателям защищенности для третьего класса СВТ вышеуказанного руководящего документа был сформирован перечень критериев оценки эффективности. Сравнение эффективности СЗИ от НСД для ОС семейства Windows проводилось с использованием следующих критериев:

- для показателей «Дискреционный принцип контроля доступа» и «Мандатный принцип контроля доступа»:
  - контроль обращений к защищаемым объектам файловой системы (ФС) с использованием штатных средств ОС;
  - устойчивость системы с установленным СЗИ от НСД к атакам типа НСД, осуществляемым из-под системных учетных записей (например, NT AUTHORITY\SYSTEM и т.д.);
  - контроль доступа к защищаемым объектам ФС, реализуемого с помощью неочевидных для большинства пользователей средств (использование NTFS-потоков из-под учетной записи с правами пользователя, прямое чтение с жесткого диска,

- доступ к объектам ФС после смены идентификатора тома, получение доступа к теневым копиям защищаемых файлов);
- наличие противодействия атакам типа НСД в режиме ядра (попытка чтения файлов с использованием стандартных функций ядра ОС);
  - контроль каналов обмена информацией (предотвращение утечки защищаемой информации через различные объекты межпроцессного и сетевого взаимодействия (pipe, mailslot, socket, отправка сообщений и т.д.), актуально только для мандатного управления доступом);
- для показателя «Изоляция модулей»:
    - противодействие НСД к памяти других процессов;
    - защита от создания полного дампа оперативной памяти;
  - для показателя «Тестирование»:
    - устойчивость системы самозащиты СЗИ от НСД (противодействие деактивации модулей СЗИ от НСД из пользовательского режима, противодействие управлению настройками СЗИ из-под системных учетных записей);
    - стойкость защиты конфигурационных файлов СЗИ от НСД (проверка наличия в СЗИ от НСД «защитных» криптографических ключей для защиты своей конфигурации и аутентификационной информации);
    - функционирование СЗИ от НСД в безопасном режиме.

Работа остальных показателей проверялась при помощи штатных средств ОС с использованием инструкций из документации СЗИ от НСД. Далее описывается ход только тех проверок, в результате выполнения которых удалось успешно обойти систему защиты, а общие результаты всех проверок приведены в сравнительной таблице 1. Рассмотрим гипотетическую организацию, в которой к конфиденциальной информации в соответствии с действующей политикой безопасности имеет доступ только ограниченный круг пользователей, включая администратора безопасности, но не включая системных администраторов (ИТ-администраторов), чьи профессиональные задачи заключаются лишь в поддержании работоспособности вычислительной сети организации и никак не связаны с настройкой и управлением СЗИ от НСД и другими средствами защиты

информации (т. е. рассматриваем третий уровень нарушителей). В организации также есть пользователи, имеющие разные уровни конфиденциальности (формы допуска), возможно, работающие в параллельных сеансах работы (первый и второй уровни нарушителей). На всех компьютерах в гипотетической организации есть доступ в Интернет.

Для проведения испытаний использовалась ОС Windows 7 Профессиональная Service Pack 1 сборка 7601 без последних обновлений, установленная в среде виртуальной машины VMWare Workstation версии 12.5.1 build-4542065. Ход испытаний приведен в статье [1].

## Сравнительная таблица эффективности СЗИ от НСД

Сведем полученные результаты анализа эффективности СЗИ от НСД в общую таблицу. Знаком + обозначено успешное прохождение проверки, соответствующее критерию оценки эффективности. Знаком ± обозначено частичное прохождение проверки. Знаком – обозначено несоответствие критерию оценки эффективности. Символом \* обозначены критерии, которые не представляется возможным проверить в рамках выбранного подхода. Выполнение критериев, не имеющих в таблице подпунктов, проверялось (при наличии технической возможности) с использованием штатных средств ОС и официальной документации.

Таблица 1: Сравнительный анализ эффективности СЗИ от НСД

Критерий	Dallas Lock	Secret Net
<b>Дискреционный принцип контроля доступа</b>	±	–
– контроль обращений к защищаемым объектам файловой системы (ФС) с использованием штатных средств ОС	+	+
– устойчивость системы с установленным СЗИ от НСД к атакам типа НСД, осуществляемым из-под системных учетных записей	–	+
– защита от несанкционированного чтения из NTFS-поток из-под учетной записи с правами пользователя	+	+
– запрет прямого чтения с жесткого диска	±	–
– защита объектов ФС после смены идентификатора тома	–	+
– защита теневых копий защищаемых файлов от НСД	±	–

– наличие противодействия атакам типа НСД в режиме ядра	–	+
<b>Мандатный принцип контроля доступа</b>	–	–
– контроль обращений к защищаемым объектам файловой системы (ФС) с использованием штатных средств ОС	+	+
– устойчивость системы с установленным СЗИ от НСД к атакам типа НСД, осуществляемым из-под системных учетных записей	–	–
– защита от несанкционированного чтения из NTFS-потоков из-под учетной записи с правами пользователя	+	+
– запрет прямого чтения с жесткого диска	±	–
– защита объектов ФС после смены идентификатора тома	–	+
– защита теневых копий защищаемых файлов от НСД	±	+
– наличие противодействия атакам типа НСД в режиме ядра	–	–
– противодействие утечкам через различные объекты ОС	–	–
<b>Очистка памяти</b>	+	+
<b>Изоляция модулей</b>	±	–
– противодействие НСД к памяти других процессов	±	–
– защита от создания полного дампа оперативной памяти	–	–
<b>Маркировка документов</b>	+	+
<b>Защита ввода и вывода на отчуждаемый физический носитель информации</b>	+	+
<b>Сопоставление пользователя с устройством</b>	+	+
<b>Идентификация и аутентификация</b>	+	+
<b>Гарантии проектирования</b>	*	*
<b>Регистрация</b>	+	+
<b>Взаимодействие пользователя с КСЗ</b>	+	+
<b>Надежное восстановление</b>	+	+
<b>Целостность КСЗ</b>	+	+
<b>Тестирование</b>	±	–
– противодействие деактивации модулей СЗИ от НСД	–	–

– противодействие управлению настройками СЗИ из-под системных учетных записей	–	±
– стойкость криптографической защиты конфигурационных файлов	–	–
– функционирование СЗИ от НСД в безопасном режиме	+	–
<b>Руководство для пользователя</b>	+	+
<b>Руководство по КСЗ</b>	+	+
<b>Тестовая документация</b>	*	*
<b>Конструкторская (проектная) документация</b>	*	*

## Заключение

Современные универсальные ОС семейства Windows не реализуют в принципе, либо реализуют лишь частично те формальные требования по защите информации, которые устанавливаются руководящими документами в отношении информационных систем, в которых ведется обработка как конфиденциальной информации, так и сведений, составляющих государственную тайну. Это вызвано принципиальными причинами, обусловленными архитектурой подсистемы безопасности ОС семейства Windows и реализуемой в ОС семейства Windows концепцией защиты. Концепция эта основана на реализации распределенной схемы администрирования механизмов защиты, что само по себе является невыполнением формальных требований к основным механизмам защиты. В рамках этой схемы пользователь рассматривается как доверенное лицо, являющееся элементом схемы администрирования и имеющее возможность назначать/изменять правила разграничения доступа. При этом он не воспринимается как потенциальный нарушитель, который может осуществить НСД к информации, следовательно, без использования **дополнительных средств защиты информации** современные ОС семейства Windows не могут применяться даже для защиты конфиденциальной информации. Назначение механизмов дополнительной защиты ОС (при помощи СЗИ от НСД) состоит в реализации централизованной схемы администрирования механизмов защиты, в рамках которой будет осуществляться противодействие НСД пользователя к информации.

Была проанализирована эффективность СЗИ от НСД Dallas Lock 8.0-C и Secret Net. Полученные результаты сравнения эффективности данных СЗИ от НСД, по мнению автора, свидетельствуют о том, что разработчики СЗИ от НСД не учитывают в своих продуктах многие неявные механизмы работы современных ОС семейства Windows, что отрицательно сказывается как на выполнении формальных требований

руководящих документов, так и на реальной защищенности ИС, в которых используются эти средства защиты.

## Литература

1. Алферов Р. И., Горохов А. А. Сравнение эффективности средств защиты информации от несанкционированного доступа. URL: [http://yarsec.ru/bitrix/Статья сравнение СЗИ.pdf](http://yarsec.ru/bitrix/Статья%20сравнение%20СЗИ.pdf).
2. ГОСТ Р 50922-2006. Защита информации. Основные термины и определения. М.: Стандартинформ, 2008. 12 с.
3. Гостехкомиссия России. Руководящий документ. Защита от несанкционированного доступа к информации. Термины и определения. М.: Военное издательство, 1992. 7 с.
4. Гостехкомиссия России. Руководящий документ. Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации. М.: Военное издательство, 1992. 21 с.
5. Гостехкомиссия России. Руководящий документ. Концепция защиты средств вычислительной техники и автоматизированных систем от несанкционированного доступа к информации. М.: Военное издательство, 1992. 7 с.
6. Гостехкомиссия России. Руководящий документ. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации. М.: Военное издательство, 1992. 29 с.
7. Защита информации на рабочих станциях и серверах. URL: <https://www.securitycode.ru/documents/analytics/zaschita-informacii-rabochih-stanciy>.

*Ярославский государственный университет им. П. Г. Демидова*

Э. В. Гайдук

## Периодические решения одного класса функциональных уравнений

В работе исследуется функциональное уравнение, которое содержит функцию  $f$ , являющуюся  $T$ -периодическим решением некоторого нелинейного дифференциального уравнения второго порядка, и функцию  $g$ , являющуюся искомой. Функциональное уравнение сводится к дифференциальному. Установлено, что функции  $f$  и  $g$  обладают схожими динамическими свойствами.

### Введение

Изучение параболических задач вида

$$\frac{\partial u}{\partial t} = d(t) \frac{\partial^2 u}{\partial x^2}$$

с некоторыми краевыми условиями, в которых коэффициент диффузии зависит от времени, сопряжено с существенными трудностями при анализе (см., например, [3]), и один из способов облегчить их изучение — убрать зависимость от времени. Для этого можно сделать замену  $s = t + g(t)$ . Замена, конечно же, должна быть обратимой. В результате получаем систему:

$$\begin{cases} s = t + g(t), \\ t = s + f(s). \end{cases} \quad (1)$$

Заметим, что в этой системе уравнения являются симметричными, а значит все результаты, полученные в дальнейшем для функции  $g$ , справедливы и для функции  $f$ , и наоборот.

Пусть  $f(s)$  известно. Тогда, подставляя выражение для  $s$ , получаем

$$t = t + g(t) + f(t + g(t)),$$

что приводит к необходимости решать функциональное уравнение для нахождения неизвестной периодической функции  $g(t)$

$$g(t) + f(t + g(t)) = 0.$$



Основное предположение состоит в том, что  $f(t)$  является  $T$ -периодическим решением некоторого нелинейного дифференциального уравнения второго порядка вида

$$f'' = F(f, f').$$

В работе будет показано, что тогда функция  $g(t)$  является  $T$ -периодическим решением нелинейного уравнения второго порядка вида

$$g''(t) = G(F, g, g') = -F\left(-g(t), -\frac{g'(t)}{1+g'(t)}\right)[1+g'(t)]^3.$$

Полученные результаты о соответствии динамики решений данных уравнений имеют важное прикладное значение. Так, если исходное уравнение для искомой функции является достаточно сложным для изучения, то возможно с помощью относительно простой функциональной замены перейти к другому уравнению с такой же динамикой, но которое проще исследовать.

## 1. Вывод дифференциального уравнения для функции $g(t)$

Рассмотрим функциональное уравнение

$$g(t) + f(t + g(t)) = 0, \quad (2)$$

где  $f(t)$  — некоторая фиксированная функция, которая является  $T$ -периодическим решением некоторого нелинейного дифференциального уравнения второго порядка вида

$$f'' = F(f, f'). \quad (3)$$

Найдем вид дифференциального уравнения для функции  $g(t)$ . Для этого продифференцируем тождество (2):

$$g'(t) + f'(t + g(t))[1 + g'(t)] = 0,$$

откуда

$$f'(t + g(t)) = -\frac{g'(t)}{1 + g'(t)}, \quad g'(t) = -\frac{f'(t + g(t))}{1 + f'(t + g(t))}. \quad (4)$$

Продифференцируем его же еще раз:

$$g''(t) + f''(t + g(t))[1 + g'(t)] + g''(t)f'(t + g(t)) + g'(t)f''(t + g(t))[1 + g'(t)] = 0.$$

Сделав замену  $f'(t + g(t))$  согласно (4), получаем:

$$g''(t) \left[ 1 - \frac{g'(t)}{1 + g'(t)} \right] + f''(t + g(t)) [1 + g'(t)]^2 = 0.$$

Все замены и преобразования являются корректными при условии, что выполняется теорема о неявной функции, а значит, знаменатели уравнений (4) не должны обращаться в 0. Функция  $f$  — периодическая и непрерывная, следовательно производная  $f'$  конечна. По этому и с учетом (4) получаем, что  $g'(t) \neq -1$  при  $\forall t$ . Кроме того, в силу своей периодичности, все функции  $f$  можно разделить на 2 категории: 1) те, для которых  $f'(t) > -1$  при  $\forall t$ ; 2) те, для которых  $\exists t : f'(t) = -1$ . Из (4) следует, что теорема о неявной функции выполняется только для первой категории функций  $f$ .

Учитывая (3), приходим к итоговой формуле

$$g''(t) = -f''(t + g(t)) [1 + g'(t)]^3 = -F \left( -g(t), -\frac{g'(t)}{1 + g'(t)} \right) [1 + g'(t)]^3, \quad (5)$$

при условии  $f'(t + g(t)) > -1$ .

Пусть  $z$  — корень уравнения  $z = f(z)$ . Тогда начальные условия для уравнения (5) имеют вид:

$$g(0) = -z, \quad g'(0) = -\frac{f'(z)}{1 + f'(z)}.$$

## 2. Случай $f'(t) = -1$

Выше было показано, что если выполняется теорема о неявной функции, то  $g$  удовлетворяет равенству (5). Изучим, что будет происходить для остальных функций  $f$  (для которых  $\exists t : f'(t) = -1$ ).

В общем случае решить эту задачу аналитически не представляется возможным, поэтому проверка проводилась на нескольких конкретных функциях  $f$  численно. Была написана программа, работающая по ниже-описанному алгоритму. На вход программе дается дифференциальное уравнение второго порядка на функцию  $f$ . Методом Рунге-Кутты четвертого порядка (подробнее о методе [1]) оно решается, и все значения сохраняются в массив  $F$ . Вычисляется также  $T$  — период функции  $f$ . Массив  $G$  хранит значения функции  $g$ , которые вычисляются в процессе работы программы.

Для каждого фиксированного  $t$  из массива  $F$  перебором и с учетом (2) находится соответствующее ему значение  $g(t)$  следующим образом:

1) поочередно выбирается значение  $g \in [0, T]$ , начиная с 0 и далее с фиксированным шагом  $h$ ;

2) находится значение выражения  $F(t + g) + g$  и сравнивается с 0 ( $F(t+g)$  берется из сформированного ранее массива значений  $F$  функции  $f$ ):

- если  $= 0$  (с точностью до  $\varepsilon$ ), то данное  $g$  принадлежит искомой функции  $g$  и запоминается в массив  $G$ ; возврат к пункту 1;
- если  $> 0$ , то  $g$  заменяется на  $g - T$  (сдвиг влево); повтор пункта 2;
- если  $< 0$ , то  $g$  заменяется на  $g + T$  (сдвиг вправо); повтор пункта 2;
- если при сдвиге (по сравнению с предыдущим значением) произошла смена знака, то значит что данное  $g$  (при сдвиге на любое целое число периодов) не принадлежит функции  $g$ ; возврат к пункту 1.

Если  $g$  пробежало все значения из диапазона  $[0, T]$ , но ни одно (с учетом сдвигов на периоды) не удовлетворило условию  $F(t + g) + g = 0$ , значит необходимо увеличить  $\varepsilon$  или уменьшить шаг  $h$  и повторить все еще раз.

В результате выполнения алгоритма сформирован массив  $G$ , содержащий значения функции  $g$ . Затем для каждого значения вычислена (с помощью формулы центральной разностной производной) производная  $g'(t)$ .

На Рисунках 1 и 2 представлены графики функций  $f$  и  $g$ , вычисленных с помощью указанного выше алгоритма, в случае, когда  $f$  задается уравнением  $f'' = -f$ . Для сравнения, на первом рисунке  $f$  выбрана таким образом, что  $f'(t) > -1$  при  $\forall t$ , а на втором — с пересечением  $-1$ .

На Рисунке 3а представлены значения  $g, g'$ , полученные в результате работы вышеописанной программы. С другой стороны, этот же фазовый портрет можно построить, формально решив дифференциальное уравнение (5), полученное аналитически. Результат представлен на Рисунке 3б. Несмотря на то, что фазовые портреты вычислены разными способами, они совпадают (погрешностью пренебрегаем). А значит для функции  $g$  выполняется соотношение (5) и в случае, когда теорема о неявной функции не работает.



Рис. 1: Графики функций  $f$  и  $g$ , где  $f'(t) > -1$  при  $\forall t$ .

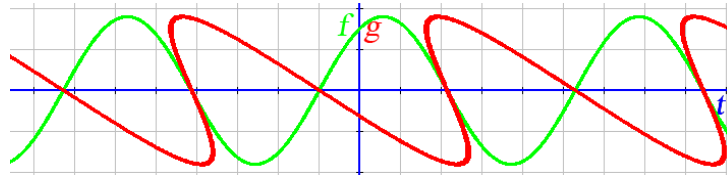
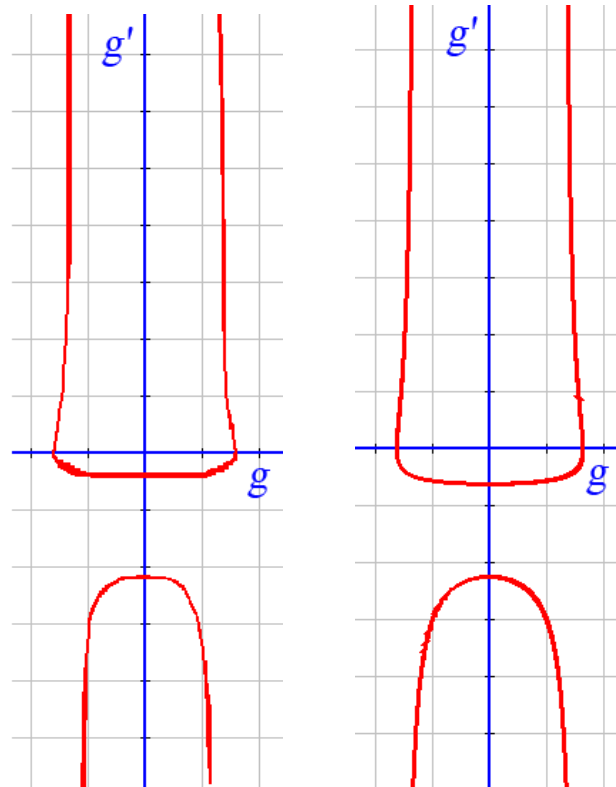


Рис. 2: Графики функций  $f$  и  $g$ , где  $\exists t : f'(t) = -1$ .



(а) Дифференцирование функции  $g$ , явно построенной по  $f$ .

(б) Решение  $g'' = -g(g' + 1)^3$ .

Рис. 3: Графики производной  $g'$ , полученные разными способами.

### 3. Первый интеграл

Пусть в равенстве (3) функция  $F$  зависит только от первого аргумента:  $F(f, f') \equiv F(f)$ . Таким образом, дифференциальное уравнение, определяющее  $f(t)$ , имеет вид  $f'' = F(f)$ . Тогда, как известно (см., например, [2]), для такого уравнения существует первый интеграл

$$H(f, f') = \frac{1}{2}f'^2 - \int_0^f F(s)ds. \quad (6)$$

Покажем, что для уравнения  $g'' = -F(-g)[1 + g'(t)]^3$  также существует первый интеграл. Используя равенства из (4) в уравнении (6),

построим функцию

$$H_1(g, g') = \frac{1}{2} \left( -\frac{g'}{1+g'} \right)^2 - \int_0^{-g} F(s) ds.$$

Покажем, что  $H_1$  является первым интегралом для уравнения (5). Для этого найдем его производную в силу системы:

$$\frac{dH_1}{dt} = \frac{g'g''}{(1+g')^3} + F(-g)g'.$$

Используя (5), получим

$$\frac{dH_1}{dt} = \left[ -F(-g) + F(-g) \right] g' = 0,$$

следовательно,  $H_1(g, g')$  является первым интегралом для уравнения (6). Этот факт может быть несколько обобщен. Справедлива следующая теорема.

**Теорема 1** (Необходимое и достаточное условие существования первого интеграла дифференциального уравнения (5)). *Для существования первого интеграла дифференциального уравнения (5) необходимо и достаточно, чтобы у связанного с ним дифференциального уравнения (3) существовал первый интеграл.*

Доказательство. Пусть  $H(f, f')$  — первый интеграл уравнения (3). Следовательно

$$\frac{dH}{dt} = \frac{\partial H}{\partial f} f' + \frac{\partial H}{\partial f'} f'' = \frac{dH}{df} f' + \frac{dH}{df'} F(f, f') \equiv 0. \quad (7)$$

Докажем, что для уравнения (5) первый интеграл имеет вид

$$H_1(g, g') = H\left(-g, -\frac{g'}{1+g'}\right).$$

Вычислим его полный дифференциал:

$$\begin{aligned} \frac{dH_1}{dt} &= \frac{\partial H}{\partial f} \left( -g'(t) \right) + \frac{\partial H}{\partial f'} \left( -\frac{g'(t)}{1+g'(t)} \right)' = \\ &= \frac{f(t+g'(t))}{1+f'(t+g(t))} \frac{\partial H}{\partial f} - \frac{g''(t)}{(1+g'(t))^2} \frac{\partial H}{\partial f'} = \\ &= \frac{1}{1+f'(t+g(t))} \left[ \frac{\partial H}{\partial f} f'(t+g(t)) + \frac{\partial H}{\partial f'} F(f, f') \right]. \end{aligned}$$

Учитывая (7), получаем:

$$\frac{dH_1}{dt} \equiv 0.$$

Достаточность доказана. В силу симметричности (1) необходимость доказывается аналогично (с точностью до обозначений).

## 4. Примеры

Ниже представлено несколько примеров, которые иллюстрируют зависимость между функциями  $f(t+g(t))$  и  $g(t)$ . Во всех случаях функция  $f$  задается уравнением  $f'' = F(f, f')$  и затем по формуле (5) определяется уравнение на  $g''$ . В примерах 1, 2 и 3 также указаны первые интегралы. С помощью программы строятся фазовые портреты данных уравнений.

Пример 1. Линейный математический маятник.

$$f'' = -f, \quad H(f, f') = \frac{1}{2}f'^2 + \frac{1}{2}f^2,$$

$$g'' = -g(g' + 1)^3, \quad H_1(g, g') = \frac{1}{2} \left[ -\frac{g'}{1+g'} \right]^2 + \frac{1}{2}g^2.$$

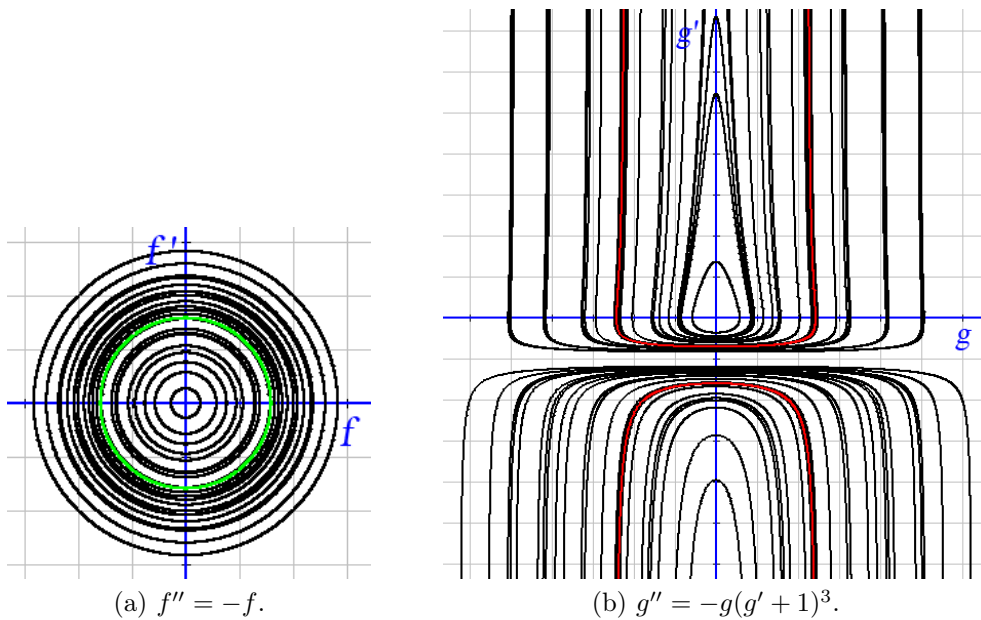


Рис. 4: Фазовые портреты для уравнений относительно  $f$  и  $g$  (линейный математический маятник).

Пример 2. Маятник без трения.

$$f'' = -\sin f, \quad H(f, f') = \frac{1}{2}f'^2 - \cos f,$$

$$g'' = -\sin g(g' + 1)^3, \quad H_1(g, g') = \frac{1}{2} \left[ -\frac{g'}{1+g'} \right]^2 - \cos g.$$

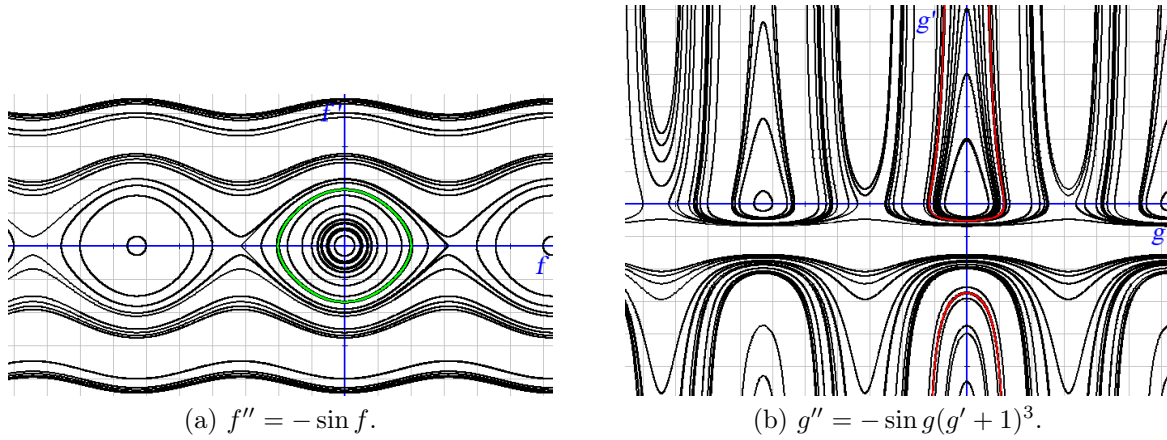


Рис. 5: Фазовые портреты для уравнений относительно  $f$  и  $g$  (маятник без трения).

Пример 3. Уравнение Дуффинга.

$$f'' = -f\omega^2 + \mu f^3, \quad \omega > 0, \quad \mu > 0, \quad H(f, f') = \frac{1}{2}f'^2 + \frac{1}{2}\omega^2 f^2 - \frac{1}{4}\mu f^4,$$

$$g'' = (-g\omega^2 + \mu g^3)(1 + g')^3, \quad H_1(g, g') = \frac{1}{2} \left[ -\frac{g'}{1+g'} \right]^2 + \frac{1}{2}\omega^2 g^2 - \frac{1}{4}\mu g^4.$$

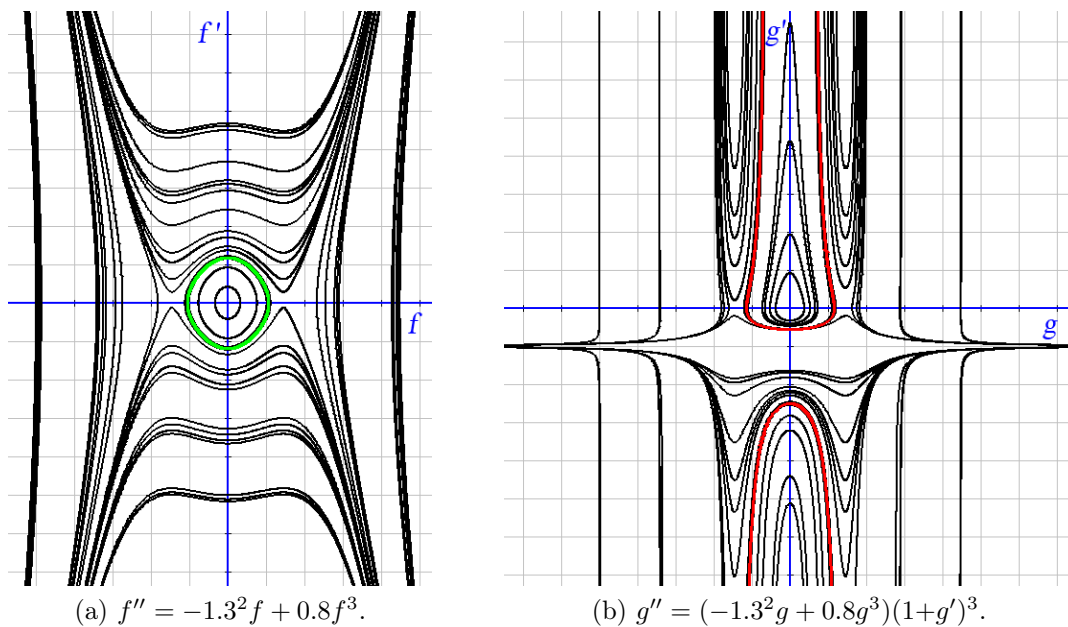


Рис. 6: Фазовые портреты для уравнения Дуффинга для  $f$  и соответствующего ему уравнения для  $g$  ( $\omega = 1.3, \mu = 0.8$ ).

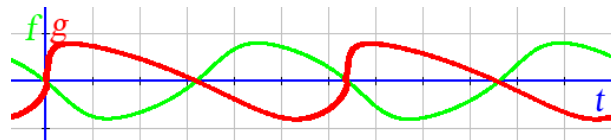
Пример 4. Уравнение Ван дер Поля

$$f'' = \mu(1 - \alpha f^2)f' - f, \quad \alpha \neq 0, \quad 0 < \mu < 1,$$

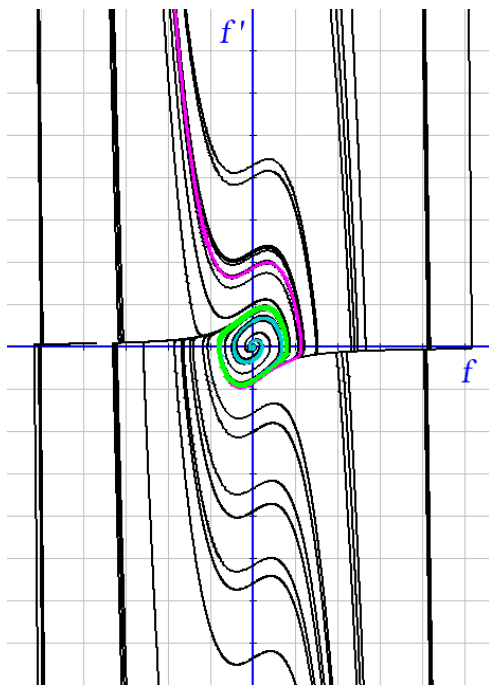
$$g'' = \mu(1 - \alpha g^2)g'(1 + g')^2 - g(1 + g')^3.$$

В этом примере правые части зависят и от производных, что уже отличает его от предыдущих примеров. Кроме того, в зависимости от параметра  $\alpha$  периодическое решение, к которому стремятся все остальные, может как пересекать  $-1$ , так и полностью находится выше ее. А значит фазовые портреты в обоих случаях будут отличаться. В виду этих соображений для уравнения Ван дер Поля был сделан подробный анализ.

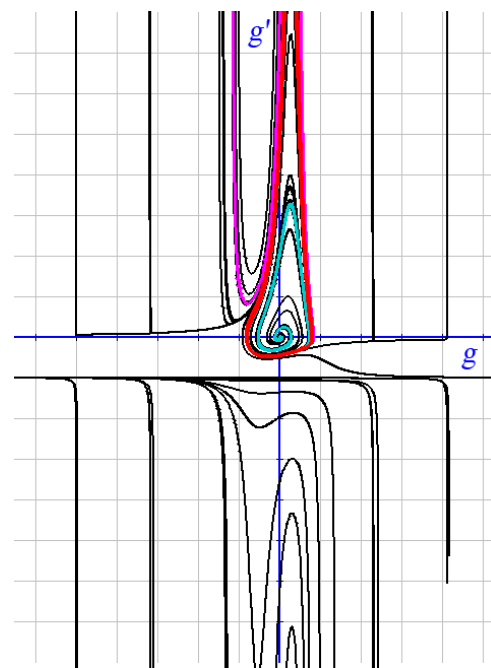
На Рисунке 7 представлен случай  $\alpha = 6.2$ ,  $\mu = 0.6$ , при котором цикл расположен полностью выше  $-1$ .



(а) Графики функций.



(b)  $f'' = 0.6(1 - 6.2f^2)f' - f$ .



(c)  $g'' = 0.6(1 - 6.2g^2)g'(1 + g')^2 - g(1 + g')^3$ .

Рис. 7: Графики функций и фазовые портреты для уравнения Ван дер Поля для  $f$  и соответствующего ему уравнения для  $g$  (цикл находится выше  $-1$ ).



Далее рассмотрен случай  $\alpha = 2.0$ ,  $\mu = 0.9$ , когда периодическое решение пересекает  $-1$ . Нахождение функций  $f$  и  $g$ , производной  $g'$  осуществляется по алгоритму, уже описанному ранее в разделе . Результат представлен на Рисунке 8. На Рисунке 9 изображены фазовые портреты для  $f$  и  $g$ .

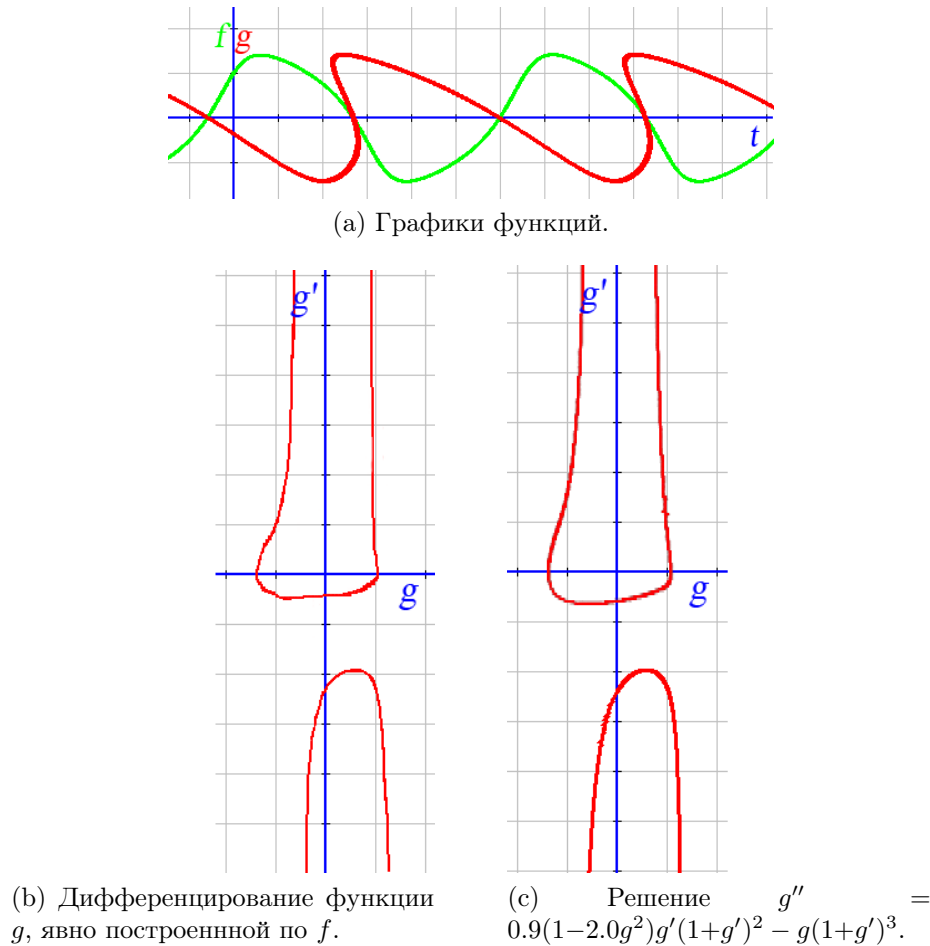


Рис. 8: Графики функций  $f$  и  $g$  и производной  $g'$  для уравнения Ван дер Поля для  $f$  и соответствующего ему уравнения для  $g$  (цикл пересекает  $-1$ ).

Проанализировав представленные примеры, можно сделать вывод, что динамика решений уравнений относительно  $f(t + g(t))$  и  $g(t)$  схожа. Циклы  $f$  переходят в циклы  $g$ , колебательные движения  $f$  остаются такими же и в  $g$ . Отдельно хочется отметить, что в случае, когда теорема о неявной функции не работает, производная  $g'(t) = \pm\infty$ , но, несмотря на это, решение все равно остается периодическим, просто делает «прыжок» из  $+\infty$  в  $-\infty$  или наоборот. Для примеров 1, 2 и 3 это подтверждается еще и тем фактом, что указанные в них уравнения на  $f$  имеют первый интеграл, а значит по теореме 1 существует первый интеграл и для соответствующих уравнений на  $g$ .

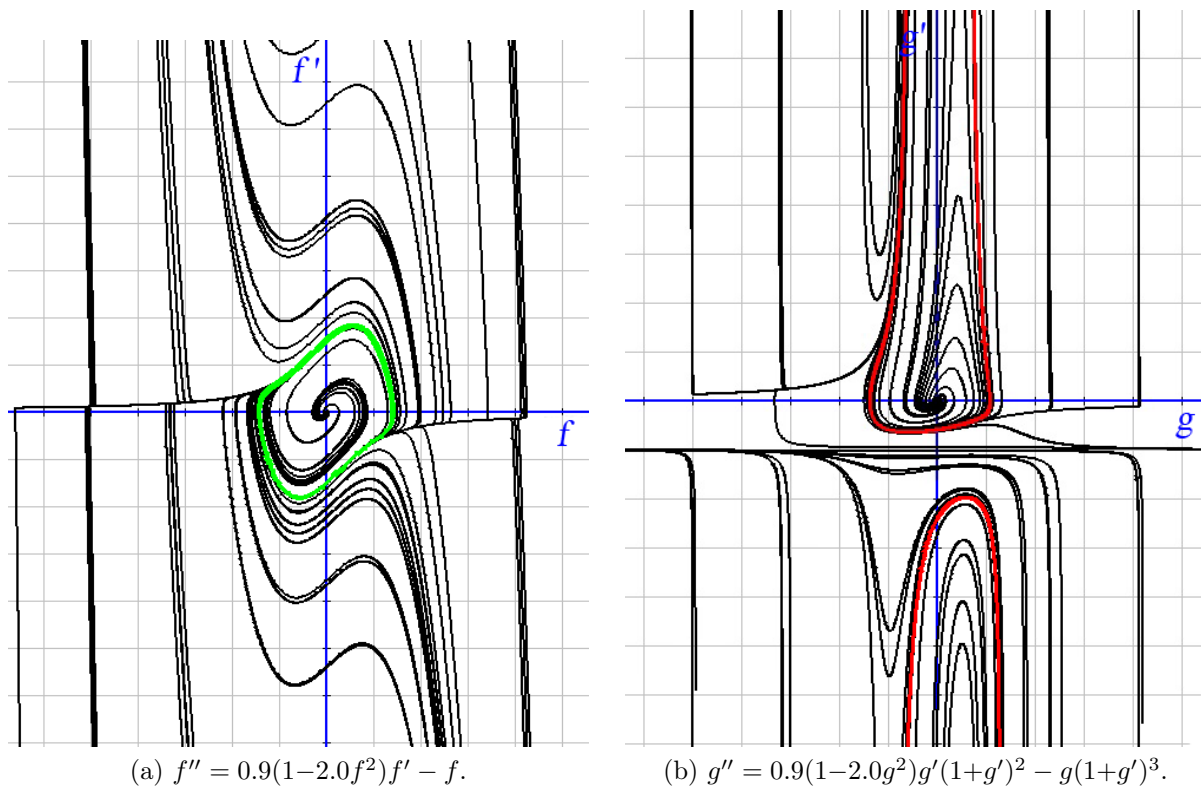


Рис. 9: Фазовые портреты для уравнения Ван дер Поля для  $f$  и соответствующего ему уравнения для  $g$  (цикл пересекает  $-1$ ).

## Заключение

В работе проведено сравнение поведения функций  $f$  и  $g$ , связанных соотношением

$$g(t) + f(t + g(t)) = 0,$$

где функция  $f$  являлась решением дифференциального уравнения второго порядка

$$f'' = F(f, f').$$

Установлено, что в таком случае дифференциальное уравнение второго порядка для  $g$  имеет вид

$$g''(t) = -F\left(-g(t), -\frac{g'(t)}{1 + g'(t)}\right)[1 + g'(t)]^3$$

(причем даже в случае, когда теорема о неявной функции не выполняется). В ходе исследования данных уравнений частично теоретически, частично численно показано, что динамика их решений совпадает. Разобран ряд примеров пар функций  $f$  и  $g$ . Написана программа, которая строит фазовые портреты соответствующих систем дифференциальных уравнений.

## Литература

1. *Бахвалов Н.С.* Численные методы / М.: Наука, 1975.
2. *Моисеев Н.Н.* Асимптотические методы нелинейной механики / М.: Наука, Физматлит, 1969.
3. *Тихонов А.Н., Самарский А.А.* Уравнения математической физики / М.: Изд-во МГУ, 1999.

*Ярославский государственный университет им. П. Г. Демидова*

**Р. А. Гибадулин**

## Алгоритм поиска вывода в Исчислении Высказываний и его программная реализация

Цель работы – создание компьютерной программы, которая по выводимой формуле Исчисления Высказываний находит её вывод.

В результате написана компьютерная программа, соответствующая заявленной цели. Получены выводы для некоторых формул, среди которых есть выводы, не встреченные автором в изученной им литературе.

### Исчисление Высказываний

Для того, чтобы задать язык  $L$ , необходимо определить алфавит  $\Sigma$ , а также определить множество формул  $F$  – подмножество множества всех выражений языка  $L$ .

**Определение 1.** Алфавит  $\Sigma_{\text{ИВ}}$  языка  $L_{\text{ИВ}}$  Исчисления Высказываний является объединением трех множеств символов:  $\Sigma_1, \Sigma_2, \Sigma_3$ , то есть  $\Sigma_{\text{ИВ}} = \Sigma_1 \cup \Sigma_2 \cup \Sigma_3$ . Множество  $\Sigma_1 = \{A_1, A_2, \dots\}$  – счетное множество **пропозициональных переменных**. Множество  $\Sigma_2 = \{\neg, \vee, \&, \rightarrow\}$  – конечное множество **пропозициональных связок**. А множество **технических символов**  $\Sigma_3$  состоит из двух символов: ( – левая скобка, и ) – правая скобка.

**Определение 2.** Формула языка  $L_{\text{ИВ}}$  – выражение, которое удовлетворяет следующим условиям.

1. Каждая пропозициональная переменная  $A_i$  – формула языка  $L_{\text{ИВ}}$ . Такие формулы называются **элементарными формулами**.
2. Если  $\mathcal{A}$  и  $\mathcal{B}$  – формулы языка  $L_{\text{ИВ}}$ , то и выражения:

$$(\neg \mathcal{A}), (\mathcal{A} \vee \mathcal{B}), (\mathcal{A} \& \mathcal{B}), (\mathcal{A} \rightarrow \mathcal{B})$$

являются формулами языка  $L_{\text{ИВ}}$ .

Число символов в формуле (или длину формулы) будем обозначать  $|\mathcal{A}|$ .

**Определение 3.** Формулы  $\mathcal{A}$  и  $\mathcal{B}$  языка  $L_{\text{ИВ}}$  называются **равными**, если они равны как слова языка  $L_{\text{ИВ}}$ .

**Определение 4.** Заменой переменных  $A_1, A_2, \dots, A_n$  на формулы  $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_n$  в формуле  $\mathcal{A}$  называется одновременная замена всех вхождений  $A_1, A_2, \dots, A_n$  в слово  $\mathcal{A}$  на слова  $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_n$  соответственно. В результате замены переменных получается формула  $\mathcal{A}' = \mathcal{A}_{A_1, A_2, \dots, A_n} [\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_n]$ .

**Определение 5.** Множество  $Sub(\mathcal{A})$  подформул формулы  $\mathcal{A}$  языка  $L_{\text{ИВ}}$  определяется тремя пунктами.

1.  $\mathcal{A} \in Sub(\mathcal{A})$ .
2. Если формула  $\mathcal{A}$  имеет вид  $(\neg \mathcal{B})$ , то  $Sub(\mathcal{B}) \subset Sub(\mathcal{A})$ .
3. Если формула  $\mathcal{A}$  имеет вид  $(\mathcal{B} \vee \mathcal{C})$ , или  $(\mathcal{B} \& \mathcal{C})$ , или  $(\mathcal{B} \rightarrow \mathcal{C})$ , то  $Sub(\mathcal{B}) \subset Sub(\mathcal{A})$  и  $Sub(\mathcal{C}) \subset Sub(\mathcal{A})$ .

**Теорема 1.** Число различных подформул формулы Исчисления Высказываний не превосходит числа символов в этой формуле.

*Доказательство.* Рассмотрим формулу  $\mathcal{A}$ .

Если формула  $\mathcal{A}$  – есть элементарная формула, то число символов в формуле равно единице, и при этом, по определению, имеется всего одна подформула, которая совпадает с самой формулой. Таким образом, для формул длины равной единице теорема доказана.

Пусть теорема доказана для всех формул, длина которых меньше  $n$ . Пусть  $\mathcal{A}$  содержит  $n$  символов и  $n \neq 1$ . Значит, формула  $\mathcal{A}$  либо имеет вид  $(\neg \mathcal{B})$ , тогда имеем

$$\left. \begin{array}{l} \overline{Sub(\mathcal{A})} = 1 + \overline{Sub(\mathcal{B})} \\ |\mathcal{A}| = 3 + |\mathcal{B}| \\ |\mathcal{B}| < n \Rightarrow \overline{Sub(\mathcal{B})} \leq |\mathcal{B}| \end{array} \right\} \Rightarrow \begin{array}{l} \overline{Sub(\mathcal{A})} \leq 1 + |\mathcal{B}| = |\mathcal{A}| - 2 \leq |\mathcal{A}| \Rightarrow \\ \overline{Sub(\mathcal{A})} \leq |\mathcal{A}|, \end{array}$$

либо имеет один из видов  $(\mathcal{B} \vee \mathcal{C})$ ,  $(\mathcal{B} \& \mathcal{C})$ ,  $(\mathcal{B} \rightarrow \mathcal{C})$ , тогда

$$\left. \begin{array}{l} \overline{Sub(\mathcal{A})} \leq 1 + \overline{Sub(\mathcal{B})} + \overline{Sub(\mathcal{C})} \\ |\mathcal{A}| = 3 + |\mathcal{B}| + |\mathcal{C}| \\ |\mathcal{B}| < n \Rightarrow \overline{Sub(\mathcal{B})} \leq |\mathcal{B}| \\ |\mathcal{C}| < n \Rightarrow \overline{Sub(\mathcal{C})} \leq |\mathcal{C}| \end{array} \right\} \Rightarrow \begin{array}{l} \overline{Sub(\mathcal{A})} \leq 1 + |\mathcal{B}| + |\mathcal{C}| = \\ = |\mathcal{A}| - 2 \leq |\mathcal{A}| \Rightarrow \\ \overline{Sub(\mathcal{A})} \leq |\mathcal{A}|. \end{array}$$

Таким образом, для любой формулы  $\mathcal{A}$  Исчисления Высказываний верно, что

$$\overline{Sub(\mathcal{A})} \leq |\mathcal{A}|.$$



Прежде чем перейти к понятию вывода в Исчислении Высказываний, необходимо дать определение аксиомы Исчисления Высказываний:

**Определение 6. Аксиома Исчисления Высказываний** – это любая формула любого из приведённых далее вида, где  $\mathcal{A}$ ,  $\mathcal{B}$ ,  $\mathcal{C}$  – произвольные формулы Исчисления Высказываний:

1.  $(\mathcal{A} \rightarrow (\mathcal{B} \rightarrow \mathcal{A}))$ ,
2.  $((\mathcal{A} \rightarrow (\mathcal{B} \rightarrow \mathcal{C})) \rightarrow ((\mathcal{A} \rightarrow \mathcal{B}) \rightarrow (\mathcal{A} \rightarrow \mathcal{C})))$ ,
3.  $((\mathcal{A} \& \mathcal{B}) \rightarrow \mathcal{A})$ ,
4.  $((\mathcal{A} \& \mathcal{B}) \rightarrow \mathcal{B})$ ,
5.  $((\mathcal{C} \rightarrow \mathcal{A}) \rightarrow ((\mathcal{C} \rightarrow \mathcal{B}) \rightarrow (\mathcal{C} \rightarrow (\mathcal{A} \& \mathcal{B}))))$ ,
6.  $(\mathcal{A} \rightarrow (\mathcal{A} \vee \mathcal{B}))$ ,
7.  $(\mathcal{B} \rightarrow (\mathcal{A} \vee \mathcal{B}))$ ,
8.  $((\mathcal{A} \rightarrow \mathcal{C}) \rightarrow ((\mathcal{B} \rightarrow \mathcal{C}) \rightarrow ((\mathcal{A} \vee \mathcal{B}) \rightarrow \mathcal{C})))$ ,
9.  $((\mathcal{A} \rightarrow (\neg \mathcal{B})) \rightarrow (\mathcal{B} \rightarrow (\neg \mathcal{A})))$ ,
10.  $((\neg(\neg \mathcal{A})) \rightarrow \mathcal{A})$ .

В Исчислении Высказываний используется всего одно правило вывода – **Правило Отделения** (*Modus Ponens*). Обозначать его будем **MP**, а запись:

$$\frac{\mathcal{A}, (\mathcal{A} \rightarrow \mathcal{B})}{\mathcal{B}}$$

будет означать, что по правилу **MP** из формул  $\mathcal{A}$  и  $(\mathcal{A} \rightarrow \mathcal{B})$  получается формула  $\mathcal{B}$ .

**Определение 7. Выводом в Исчислении Высказываний** называется любая конечная последовательность формул Исчисления Высказываний  $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_n$ , где для любого  $i$ , ( $i = 1, \dots, n$ )

- либо формула  $\mathcal{B}_i$  является аксиомой Исчисления Высказываний,
- либо найдутся натуральные числа  $j$  и  $k$  меньшие, чем  $i$ , такие, что

$$\frac{\mathcal{B}_j, \mathcal{B}_k}{\mathcal{B}_i}.$$

**Определение 8.** Формула  $\mathcal{B}$  называется **выводимой в Исчислении Высказываний**, если существует вывод в Исчислении Высказываний  $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_n$ , оканчивающийся формулой  $\mathcal{B}$ .

## Алгоритм поиска вывода в Исчислении Высказываний

Задача: найти вывод формулы  $\mathcal{A}$  Исчисления Высказываний.

**Предложение 1.** Множество переменных можно ограничить набором переменных, которые входят в формулу  $\mathcal{A}$ .

**Предложение 2.** Любой вывод содержит аксиомы. Поэтому имеет смысл генерировать большое число аксиом, чтобы как можно скорее получить те аксиомы, которые войдут в вывод формулы  $\mathcal{A}$ .

Таким образом, предлагается следующий алгоритм:

---

### Алгоритм 1 поиска вывода формулы Исчисления Высказываний

---

```

1: for all  $\mathcal{B}$  – подформула формулы  $\mathcal{A}$  do
2:    $F \leftarrow F \cup \{\mathcal{B}\}$ 
3: while вывод не найден do
4:   for all  $\mathcal{B} \in F$  do
5:     if формулу  $\mathcal{B}$  можно добавить в конец вывода  $\mathcal{D}$  then
6:       добавить  $\mathcal{B}$  в конец вывода  $\mathcal{D}$ 
7:       if  $\mathcal{B}$  и  $\mathcal{A}$  равны then
8:         завершить выполнение алгоритма
9:    $F' \leftarrow \emptyset$ 
10:  for all  $(\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3) \in F^3$  do
11:    for all  $\mathcal{B}$  – формула из списка аксиом do
12:      for all  $\mathcal{C}$  – подформула формулы  $\mathcal{B}_{\mathcal{A}, \mathcal{B}, \mathcal{C}}[\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3]$  do
13:         $F' \leftarrow F' \cup \{\mathcal{C}\}$ 
14:   $F \leftarrow F \cup F'$ 

```

---

Символ  $\leftarrow$  обозначает присвоение значения переменной. После завершения работы алгоритма  $\mathcal{D}$  – искомый вывод формулы  $\mathcal{A}$ .

Важно подчеркнуть, что алгоритм ищет вывод только для **выводимых** формул. Нетрудно заметить, что если на вход этому алгоритму подать формулу, которая не является выводимой, то алгоритм никогда не завершит свою работу.

Аналогичным образом можно искать вывод из множества гипотез: во множество  $F$  изначально добавить подформулы всех гипотез.

Пусть  $f_n$  – число элементов в множестве  $F$  во время  $n$ -й итерации цикла **while** алгоритма 1. До цикла в множестве  $F$  столько формул, сколько подформул в формуле  $\mathcal{A}$ , а их по теореме 1 не более чем  $|\mathcal{A}|$ . Во

время  $n$ -й итерации цикла **while** в множество  $F$  добавляется не более  $10 * f_{n-1}^3$  аксиом. Тогда, в первом приближении, будем считать, что

$$f_n \approx 10 * f_{n-1}^3, \quad \text{где} \quad 1 \leq f_0 \leq |\mathcal{A}|.$$

Данная оценка является **грубой**: никак не учитывается, что в множество  $F$  добавляются также все подформулы сгенерированных аксиом. Однако эта оценка позволяет предположить, что попытка реализовать этот алгоритм «в лоб» не завершится успехом, поскольку множество растёт слишком быстро.

## Программная реализация. Программа «Latium»

Прежде чем перейти непосредственно к реализации алгоритма 1, необходимо решить несколько вспомогательных задач.

Задача распознавания формулы: *является ли данная строка формулой Исчисления Высказываний?*

Для решения этой задачи применяется тот же подход, что используется для распознавания арифметических выражений:

1. Разбиваем данную строку на токены, то есть отделяем пропозициональные переменные, пропозициональные связки и технические символы друг от друга.
2. Проверяем расположение пропозициональных связок в строке, например, отрицание должно стоять только перед пропозициональной переменной или левой скобкой.
3. И наконец, по токенам строится обратная польская запись формулы, в ходе построения которой проверяется баланс скобок в строке.

Если все перечисленные шаги выполняются успешно, без ошибок, то строка является формулой Исчисления Высказываний, иначе – не является.

Пусть каждая формула – это

- либо массив из одного элемента – пропозициональной переменной, если формула является элементарной формулой,
- либо массив из двух элементов – формула  $\mathcal{B}$  и символ отрицания, если формула имеет вид  $(\neg \mathcal{B})$ ,
- либо массив из трёх элементов – формула  $\mathcal{B}$ , формула  $\mathcal{C}$  и пропозициональная связка, если формула имеет вид  $(\mathcal{B} \vee \mathcal{C})$ , или  $(\mathcal{B} \& \mathcal{C})$ , или  $(\mathcal{B} \rightarrow \mathcal{C})$ .



Такая структура, во-первых, позволяет быстро определять какой же вид имеет формула, что полезно, например, при сравнении формул. Во-вторых, позволяет быстро получить подформулы, что необходимо при применении правила **МР**. В-третьих, позволяет сократить расходы памяти, например, при выполнении замены переменных в формуле необходимо лишь заменить ссылки на переменные ссылками на формулы, на которые происходит замена.

В результате работы алгоритма 1 в вывод добавляется большое число аксиом, однако не все из них необходимы для вывода итоговой формулы, то есть можно удалить некоторые формулы из полученной последовательности формул, при этом она не перестанет быть выводом. Так и возникла задача упрощения (сокращения) вывода, которую решает алгоритм 2. После завершения работы алгоритма  $\mathcal{D}'$  – сокращенный вывод.

---

### Алгоритм 2 сокращения вывода формулы Исчисления Высказываний

---

- 1:  $\mathcal{D}$  – исходный вывод,  $\mathcal{D}'$  – новый пустой вывод
  - 2:  $\mathcal{A}$  – последняя формула вывода  $\mathcal{D}$
  - 3: ADDFORMULA( $\mathcal{A}$ )
  - 4: **procedure** ADDFORMULA(формула  $\mathcal{A}$ )
  - 5:     **if**  $\mathcal{A}$  – не аксиома **then**
  - 6:          $\mathcal{A}$  получена из формул  $\mathcal{B}$  и  $\mathcal{C}$  по правилу **МР**,  
            которые располагаются левее формулы  $\mathcal{A}$  в выводе  $\mathcal{D}$
  - 7:         ADDFORMULA( $\mathcal{B}$ )
  - 8:         ADDFORMULA( $\mathcal{C}$ )
  - 9:     добавить  $\mathcal{A}$  в конец вывода  $\mathcal{D}'$
- 

Решение отмеченных выше вспомогательных задач и программная реализация алгоритмов 1 и 2 образуют приложение «Latium», которое написано и отлажено автором работы. Если ввести в Яндекс.Переводчик «логически рассуждающий», то на латыни получится «*Latium*», отсюда и произошло название. Программа написана на языке C# версии 7.0 в среде разработки Visual Studio 2017.

Ранее было выдвинуто предположение, что реализация алгоритма 1 «в лоб» не самая лучшая идея. Так и получилось: программа зависала на самых простых формулах. Для решения этой проблемы необходимо «ускорить» алгоритм.

Заметим, что может возникнуть ситуация, когда необходимые для вывода формулы уже сгенерированы, однако сам процесс генерации ещё не завершён. Поэтому предлагается выполнять в разных потоках процессы генерации формул и их добавления в конец вывода, т. е. воспользоваться методами многопоточного программирования языка C# [2]. В аксиомах

замену переменных также можно производить в разных потоках. Таким образом, необязательно дожидаться окончания очередной итерации цикла генерации аксиом, а, как только вывод будет построен, завершить работу программы.

Для начала найдём вывод для формулы  $(A_1 \rightarrow (B_1 \rightarrow A_1))$ , которая, очевидно, является аксиомой. Программа определила, что эта формула оказывается аксиомой и выдала состоящий из одной формулы вывод (см. рисунок 1).

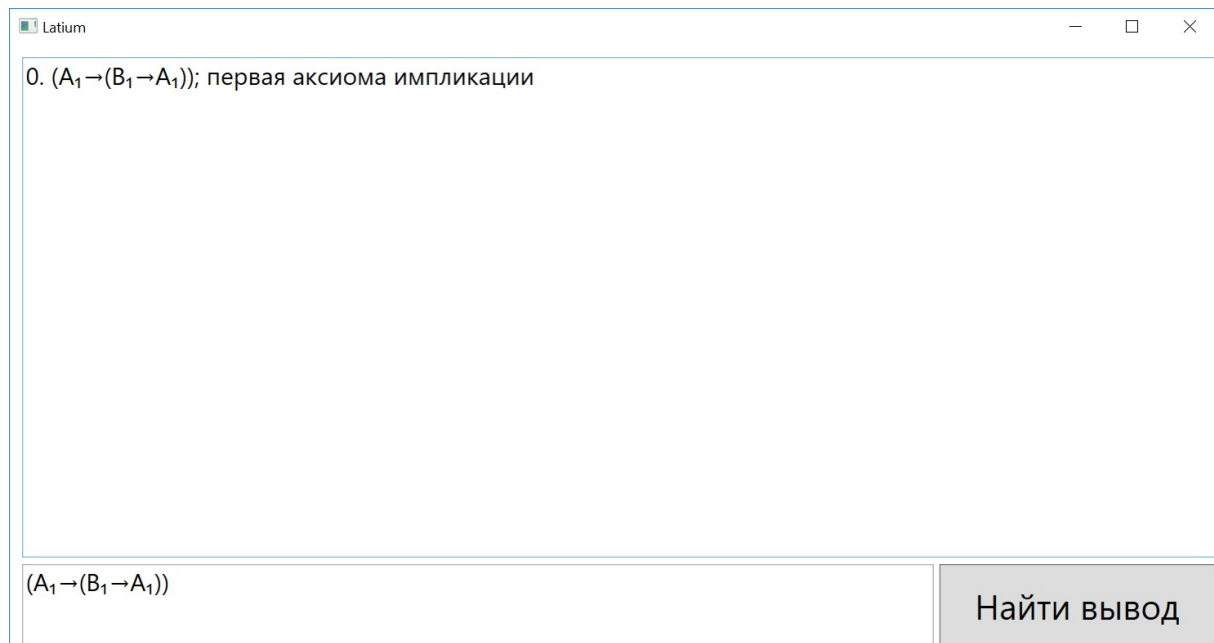
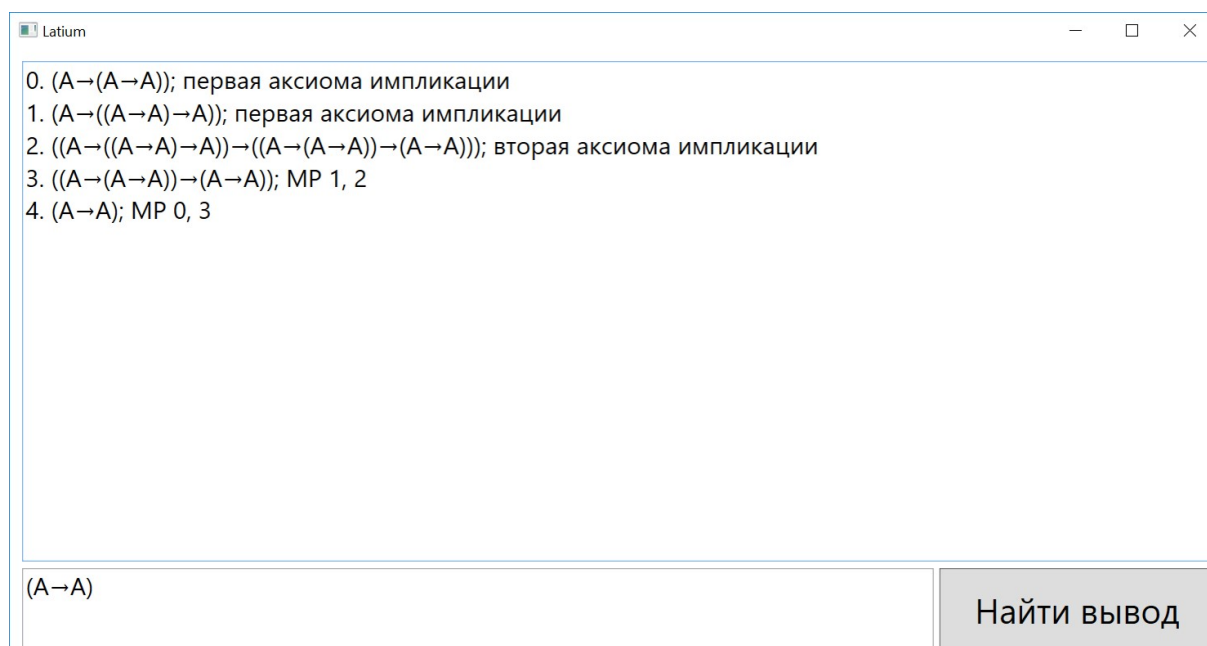
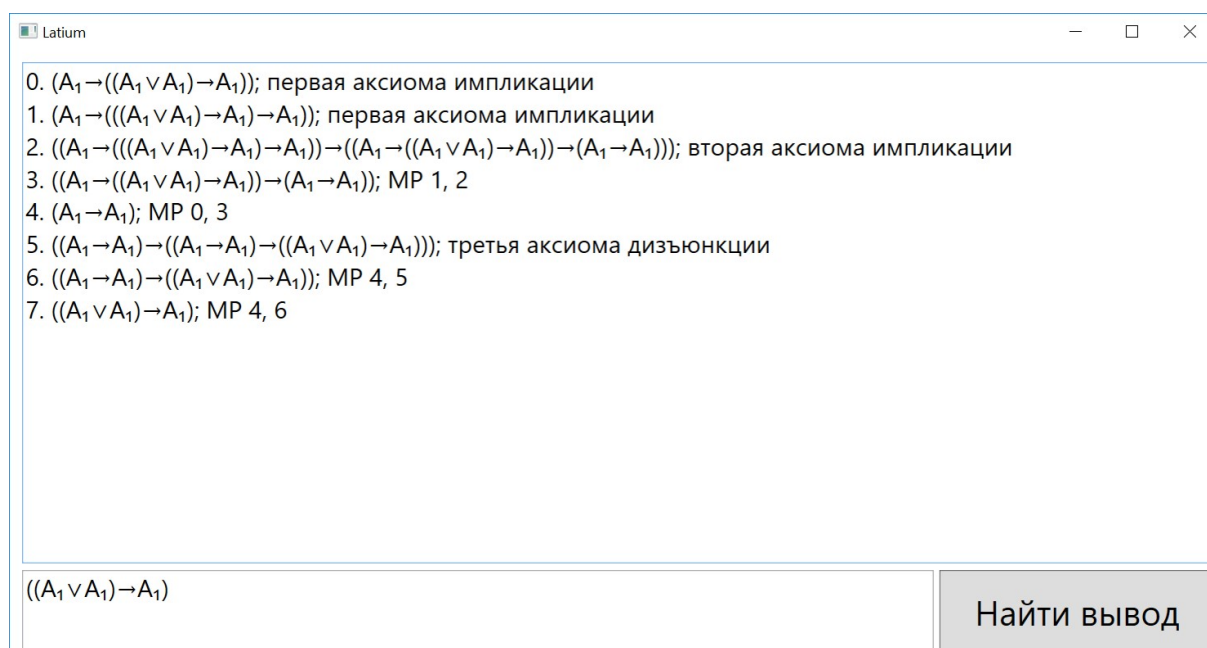


Рис. 1: Вывод формулы  $(A_1 \rightarrow (B_1 \rightarrow A_1))$

Теперь введём формулу  $(A \rightarrow A)$ . Программа нашла вывод, идентичный выводу этой формулы в книге [1] (см. рисунок 2).

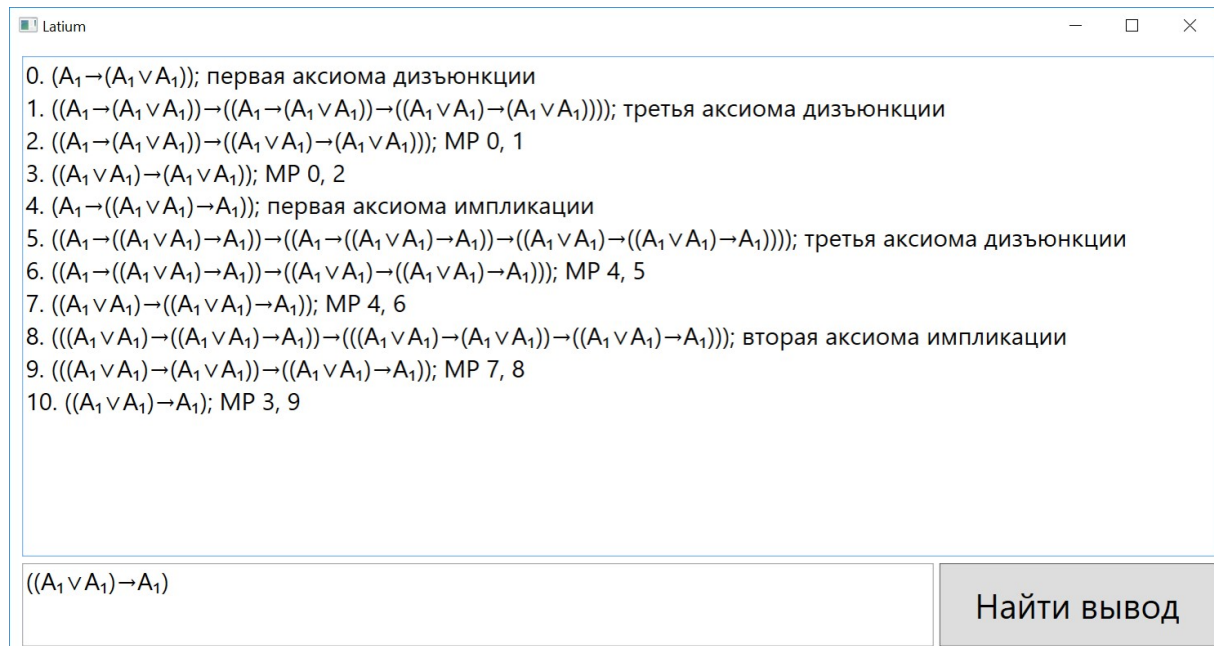
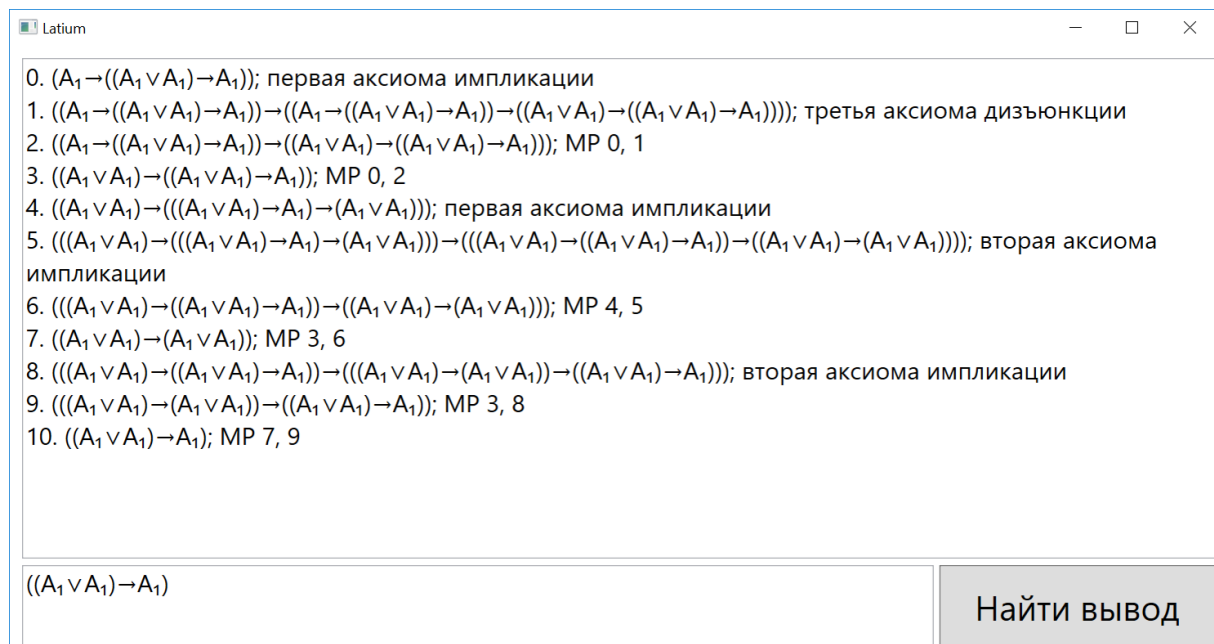
Рассмотрим формулу  $((A_1 \vee A_1) \rightarrow A_1)$ . Оказалось, что программа «Latium» от запуска к запуску может выдавать разные выводы, что и произошло в случае рассматриваемой формулы (см. рисунки 3, 4 и 5). Это объясняется тем, что операционная система может предоставлять потокам различные интервалы времени, от чего формулы могут генерироваться, а, значит, и добавляться в вывод в разном порядке от запуска к запуску.

Первый вывод формулы  $((A_1 \vee A_1) \rightarrow A_1)$  (рисунок 3) можно найти в книгах, а вот два оставшихся вывода (рисунки 4 и 5) не найдены автором в доступной ему литературе.

Рис. 2: Вывод формулы  $(A \rightarrow A)$ Рис. 3: Первый вывод формулы  $((A_1 \vee A_1) \rightarrow A_1)$ 

## Заключение

Таким образом, заявленная цель достигнута: написана и отлажена компьютерная программа «Latium», которая ищет вывод введенной с клавиатуры формулы Исчисления Высказываний. Программа умеет распознавать формулы ИВ, а для представления формулы в компьютере

Рис. 4: Второй вывод формулы  $((A_1 \vee A_1) \rightarrow A_1)$ Рис. 5: Третий вывод формулы  $((A_1 \vee A_1) \rightarrow A_1)$ 

реализована нестандартная структура данных, основанная на идее обратной польской записи. Также разработаны и реализованы алгоритм 1, который строит вывод формулы, и алгоритм 2, который упрощает найденный вывод. Наконец, удалось получить выводы для трёх формул, при этом для формулы  $((A_1 \vee A_1) \rightarrow A_1)$  найдены три различных вывода.

## Литература

1. Дурнев В. Г. Элементы теории множеств и математической логики: учеб. пособие / Ярославль : ЯрГУ, 2009.
2. Якимова О. П., Якимов И. М., Дольников В. Л. Языки программирования. Ч. 2 : лабораторный практикум / Ярославль : ЯрГУ, 2012.

*Ярославский государственный университет им. П. Г. Демидова*

**И. С. Кудрявцев**

## Линейная интерполяция на шаре в $\mathbb{R}^n$

В работе рассматривается задача линейной интерполяции непрерывной функции, заданной на шаре в  $\mathbb{R}^n$  с помощью линейных функций. Целью работы является построение интерполяционных проекторов, имеющих минимальную норму (минимальных проекторов). С помощью численных методов приближенно найдены минимальные проекторы для  $2 \leq n \leq 7$ .

### Задача линейной интерполяции для $n$ -мерного шара

В дальнейшем будем считать, что  $n \in \mathbb{N}$ , элемент  $x \in \mathbb{R}^n$  будем записывать в виде  $x = (x_1, \dots, x_n)$ . Введем обозначение

$$B_n := \{x : x \in \mathbb{R}^n, x_1^2 + \dots + x_n^2 \leq 1\}.$$

$\Pi_1(R^n)$  обозначает пространство многочленов  $n$  переменных степени не выше 1.

Пусть  $f(x)$  — непрерывная функция  $f : B_n \rightarrow \mathbb{R}$ . Набор точек

$$x^j = (x_1^j, \dots, x_n^j) \in B_n, \quad j = 1, \dots, n+1$$

будем называть допустимым набором узлов для интерполяции с помощью  $\Pi_1(\mathbb{R}^n)$ , если  $x^j$  являются вершинами невырожденного симплекса  $S$ . Задача линейной интерполяции состоит в построении многочлена  $p(x) \in \Pi_1(\mathbb{R}^n)$ , удовлетворяющего условиям

$$p(x^j) = f(x^j).$$

### Интерполяционный проектор и его норма

Приведем используемые в работе теоретические сведения. Более подробное изложение этих вопросов, в частности — доказательство формулы для вычисления нормы проектора (2), можно найти в монографии М. В. Невского [1].

Пусть  $S \in \mathbb{R}^n$  — невырожденный симплекс с вершинами  $x^j = (x_1^j, \dots, x_n^j)$ ,  $j = 1, \dots, n+1$ . Рассмотрим матрицу

$$A := \begin{pmatrix} x_1^{(1)} & \dots & x_n^{(1)} & 1 \\ x_1^{(2)} & \dots & x_n^{(2)} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{(n+1)} & \dots & x_n^{(n+1)} & 1 \end{pmatrix}. \quad (1)$$

Пусть  $A^{-1} = (l_{ij})$ . Базисными многочленами Лагранжа называются функции

$$\lambda_j = l_{1j}x_1 + \dots + l_{nj}x_n + l_{n+1j}.$$

Коэффициентами  $\lambda_j$  являются элементы  $j$ -го столбца матрицы, обратной матрице  $A$ :

$$A^{-1} = (l_{ij}) = \begin{pmatrix} \dots & l_{1,j} & \dots \\ \vdots & \vdots & \vdots \\ \dots & l_{n,j} & \dots \\ \dots & l_{n+1,j} & \dots \end{pmatrix}.$$

Для любого многочлена  $p \in \Pi_1(\mathbb{R}^n)$  выполняется равенство

$$p(x) = \sum_{j=1}^{n+1} p(x^{(j)})\lambda_j(x).$$

Таким образом, интерполяционный многочлен может быть однозначно представлен в виде линейной комбинации многочленов  $\lambda_j$ . По этой причине они и называются базисными многочленами Лагранжа симплекса  $S$ .

Пусть дан невырожденный симплекс  $S$  с вершинами  $x^{(j)} \in B_n$ . Интерполяционный проектор  $P : C(B_n) \rightarrow \Pi_1(R_n)$  по системе узлов  $x^{(j)}$  определяется равенством

$$Pf(x^{(j)}) = f(x^{(j)}), \quad j = 1, \dots, n+1.$$

Результат действия проектора  $P$  на функцию  $f$  можно записать через аналог интерполяционной формулы Лагранжа

$$Pf(x) = \sum_{j=1}^{n+1} f(x^{(j)})\lambda_j(x).$$

Доказано (см. [1]), что норма интерполяционного проектора  $P$  как оператора из  $C(B_n)$  в  $C(B_n)$  может быть найдена по формуле

$$\|P\| = \max_{x \in B_n} \sum_{j=1}^{n+1} |\lambda_j(x)|. \quad (2)$$

В теории приближений важную роль играет так называемый минимальный проектор, то есть такой проектор, норма которого минимальна. Обозначим через  $\theta_n(B_n)$  норму минимального интерполяционного проектора по узлам из  $B_n$ . Таким образом  $\theta_n(B_n)$  определяется формулой

$$\theta_n(B_n) = \min_{x^{(j)} \in B_n} \|P\|. \quad (3)$$

Целью настоящей работы является построение оценок величин  $\theta_n(B_n)$  и отыскание узлов интерполяции соответствующего интерполяционного проектора. Из общей теории (см. [1]) следует, что  $\theta_1(B_1) = 1$ .

Задача о минимальном проекторе для функций, определенных на кубе  $Q_n = [0, 1]^n$  рассматривалась в работах [1]–[3]. Отметим, что для случая куба была известна формула, позволяющая вычислять норму проектора по данному набору узлов. Задача для шара является более сложной, т. к. в этом случае точная формула для вычисления нормы в литературе на данный момент не описана. Для вычисления нормы в настоящей работе используется итерационный процесс, позволяющий приближенно вычислять норму проектора с использованием формулы (2). Итерационный процесс приходится применять также и для поиска проектора, имеющего минимальную норму. Таким образом, полученные в работе значения являются только оценками нормы минимального проектора. Отметим, что соответствующие вычислительные процедуры являются весьма трудоемкими. Трудоемкость существенно возрастает с ростом размерности. В силу этого, оценки для норм удалось получить только для  $n = 2, \dots, 7$ .

## Задача оценивания $\theta_n(B_n)$

В настоящее время неизвестен аналитический подход для вычисления  $\theta_n(B_n)$ . Автором была написана компьютерная программа на языке *Node.JS* для поиска симплекса, которому соответствует проектор с наименьшей нормой. Для вычисления нормы проектора в программе реализован метод градиентного подъема для вычисления максимума правой части (2). Для оценивания  $\theta_n(B_n)$  применяется метод градиентного спуска для вычисления минимума в (3).

С помощью этой программы были приближенно вычислены такие симплексы для  $n = 2, 3$ . В ходе анализа полученных результатов была замечена следующая закономерность. Длины ребер полученных симплексов либо все равны между собой, либо очень мало отличаются друг от друга. Это позволяет высказать следующую гипотезу: для данного  $n$  узлы минимального проектора располагаются в вершинах правильного симплекса, вписанного в граничную сферу шара  $B_n$ .



Результаты для  $n = 4, \dots, 7$  получены на основании данного предположения. Была разработана компьютерная программа, которая для данного  $n$  строит правильный симплекс, вписанный в граничную сферу  $B_n$  и приближенно вычисляет норму проектора по узлам в вершинах этого симплекса. Полученное значение можно рассматривать как приближенное значение для верхней границы  $\theta_n(B_n)$ .

## Результаты

Результаты работы программы приведены в Таблице 1 и Таблице 2. Для  $n = 2, \dots, 7$  приведены верхние границы  $\theta_n(B_n)$  и соответствующие им матрицы узлов (1).

## Литература

1. *Невский М. В.* Геометрические оценки в полиномиальной интерполяции / Ярославль : Яросл. гос. ун-т им. П. Г. Демидова, 2012.
2. *Кудрявцев И. С., Озерова Е. А., Ухалов А. Ю.* Новые оценки для норм минимальных проекторов // Современные проблемы математики и информатики. Вып. 17 / Ярославль : ЯрГУ, 2017. С. 74–81.
3. *Невский М. В., Ухалов А. Ю.* Об оптимальной интерполяции линейными функциями на  $n$ -мерном кубе // Моделирование и анализ информационных систем. 2018. Том 25, № 3. С. 291–311.

*Ярославский государственный университет им. П. Г. Демидова*

Таблица 1: Оценки  $\theta_n(B_n)$ ,  $n = 2, \dots, 5$

$n$	$\theta_n(B_n) \leq$	$A$
2	1.668	$\begin{pmatrix} 0.966 & -0.258 & 1 \\ -0.258 & 0.966 & 1 \\ -0.707 & -0.707 & 1 \end{pmatrix}$
3	2	$\begin{pmatrix} 0.962 & -0.193 & -0.193 & 1 \\ -0.193 & 0.962 & -0.193 & 1 \\ -0.193 & -0.193 & 0.962 & 1 \\ -0.577 & -0.577 & -0.577 & 1 \end{pmatrix}$
4	2.202	$\begin{pmatrix} 0.963 & -0.154 & -0.154 & -0.154 & 1 \\ -0.154 & 0.963 & -0.154 & -0.154 & 1 \\ -0.154 & -0.154 & 0.963 & -0.154 & 1 \\ -0.154 & -0.154 & -0.154 & 0.963 & 1 \\ -0.5 & -0.5 & -0.5 & -0.5 & 1 \end{pmatrix}$
5	2.444	$\begin{pmatrix} 0.966 & -0.129 & -0.129 & -0.129 & -0.129 & 1 \\ -0.129 & 0.966 & -0.129 & -0.129 & -0.129 & 1 \\ -0.129 & -0.129 & 0.966 & -0.129 & -0.129 & 1 \\ -0.129 & -0.129 & -0.129 & 0.966 & -0.129 & 1 \\ -0.129 & -0.129 & -0.129 & -0.129 & 0.966 & 1 \\ -0.447 & -0.447 & -0.447 & -0.447 & -0.447 & 1 \end{pmatrix}$

Таблица 2: Оценки  $\theta_n(B_n)$ ,  $n = 5, 6$

$n$	$\theta_n(B_n) \leq$	$A$
6	2.638	$\begin{pmatrix} 0.968 & -0.112 & -0.112 & -0.112 & -0.112 & -0.112 & 1 \\ -0.112 & 0.968 & -0.112 & -0.112 & -0.112 & -0.112 & 1 \\ -0.112 & -0.112 & 0.968 & -0.112 & -0.112 & -0.112 & 1 \\ -0.112 & -0.112 & -0.112 & 0.968 & -0.112 & -0.112 & 1 \\ -0.112 & -0.112 & -0.112 & -0.112 & 0.968 & -0.112 & 1 \\ -0.112 & -0.112 & -0.112 & -0.112 & -0.112 & 0.968 & 1 \\ -0.408 & -0.408 & -0.408 & -0.408 & -0.408 & -0.408 & 1 \end{pmatrix}$
7	2.805	$\begin{pmatrix} 0.97 & -0.098 & -0.098 & -0.098 & -0.098 & -0.098 & -0.098 & 1 \\ -0.098 & 0.97 & -0.098 & -0.098 & -0.098 & -0.098 & -0.098 & 1 \\ -0.098 & -0.098 & 0.97 & -0.098 & -0.098 & -0.098 & -0.098 & 1 \\ -0.098 & -0.098 & -0.098 & 0.97 & -0.098 & -0.098 & -0.098 & 1 \\ -0.098 & -0.098 & -0.098 & -0.098 & 0.97 & -0.098 & -0.098 & 1 \\ -0.098 & -0.098 & -0.098 & -0.098 & -0.098 & 0.97 & -0.098 & 1 \\ -0.098 & -0.098 & -0.098 & -0.098 & -0.098 & -0.098 & 0.97 & 1 \\ -0.378 & -0.378 & -0.378 & -0.378 & -0.378 & -0.378 & -0.378 & 1 \end{pmatrix}$

**И. Н. Маслеников**

## Локальная динамика модели оптико-электронного осциллятора

Для модели оптико-электронного осциллятора, описываемого дифференциально-интегральным уравнением, изучена устойчивость состояния равновесия. Для этого построено характеристическое уравнение и определено положение его корней. В зависимости от значений параметров определена устойчивость состояния равновесия. Выделены критические значения параметров, при которых состояние равновесия меняет свою устойчивость. В критических случаях построены аналоги нормальных форм.

### 1. Постановка задачи

Рассмотрим дифференциально-интегральное уравнение с запаздыванием [1], которое представляет собой реализацию модифицированного уравнения Икеды с задержкой по времени:

$$\varepsilon \frac{dx}{dt} + x + \delta \int_{t_0}^t x(s) ds = \beta_1 F(x(t - \tau)). \quad (1)$$

Здесь  $\beta_1$  — параметр,  $\tau$  — параметр запаздывания, вещественный и положительный. Функция  $F$  достаточно гладкая, не ограничивая общности можно считать, что  $F(0) = 0$ . Таким образом, уравнение (1) имеет нулевое состояние равновесия, если это не так, то можно сделать соответствующую замену.

В работе [1] уравнение исследовалось численными методами, подбирались параметры, начальные условия, строились сложные квазипериодические режимы, так называемые химеры, при этом рассматривался случай, когда параметры  $\varepsilon = 0.005$  и  $\delta = 0.016$ . В силу этого будем предполагать, что параметры  $\varepsilon$  и  $\delta$  малы ( $0 < \varepsilon \ll 1$ ,  $0 < \delta \ll 1$ ) и пропорциональны:  $\varepsilon = k\delta$ ,  $k > 0$ .

Поставим задачу исследовать локальную динамику в окрестности состояния равновесия. В §2 опишем устойчивость состояния равновесия, а в §3 исследуем бифуркации, которые возникают при потере устойчивости нулевого решения.

Отметим, что в статье [2] рассмотрено похожее уравнение оптико-электронного осциллятора, в котором параметр  $\delta$  не является малым.

## 2. Исследование устойчивости состояния равновесия

В работе [5] проведено исследование расположения корней характеристического квазиполинома линеаризованного уравнения (1).

$$\varepsilon\lambda^2 + \lambda + k\varepsilon = \lambda\beta e^{-\lambda}. \quad (2)$$

Для уравнения (2) справедливы теоремы, доказанные в [5].

**Теорема 1.** При  $|\beta| < 1$  и достаточно малом  $\varepsilon$ , все корни (2) имеют отрицательные вещественные части.

**Теорема 2.** При  $\beta = 1 + \varepsilon^2\beta_1$  корни характеристического уравнения допускают асимптотическое представление вида:

$$\lambda_n = \lambda_{n0} + \varepsilon\lambda_{n1} + \varepsilon^2\lambda_{n2} + o(\varepsilon^2) \quad \text{при } n \neq 0, \quad \text{где}$$

$$\lambda_{n0} = 2\pi ni, \quad \lambda_{n1} = -\frac{\lambda_{n0}^2 + k}{\lambda_{n0}},$$

$$\lambda_{n2} = -\frac{2\lambda_{n0}\lambda_{n1} + \lambda_{n1}^2}{\lambda_{n0}} + \frac{\lambda_{n1}^2}{2} + \beta_1,$$

$$\text{и } \lambda = \sqrt{\varepsilon}\lambda_1 + \varepsilon\lambda_2 + \varepsilon^{\frac{3}{2}}\lambda_3 + o(\varepsilon^{\frac{3}{2}}) \quad \text{при } \lambda_0 = 0, \quad \text{где}$$

$$\lambda_1 = \pm\sqrt{k}i, \quad \lambda_2 = \frac{-k}{4}.$$

**Теорема 3.** При  $\beta = -(1 + \varepsilon^2\beta_1)$  корни характеристического уравнения допускают асимптотическое представление вида:

$$\lambda_n = \lambda_{n0} + \varepsilon\lambda_{n1} + \varepsilon^2\lambda_{n2} + o(\varepsilon^2), \quad \text{где}$$

$$\lambda_{n0} = \pi(2n + 1)i, \quad \lambda_{n1} = -\frac{\lambda_{n0}^2 + k}{\lambda_{n0}},$$

$$\lambda_{n2} = -\frac{2\lambda_{n0}\lambda_{n1} + \lambda_{n1}^2}{\lambda_{n0}} + \frac{\lambda_{n1}^2}{2} + \beta_1,$$

$$\text{и } \lambda = \frac{-\varepsilon k}{2} + o(\varepsilon) \quad \text{при } \lambda_0 = 0.$$

### 3. Построение нормальной формы уравнения

Уравнение (1) допускает запись в виде

$$\varepsilon \ddot{y} + \dot{y} + k\varepsilon y = \beta F(\dot{y}(t-1)). \quad (3)$$

Разложим в уравнении (3) функцию  $F$  в ряд Тейлора:

$$\varepsilon \ddot{y} + \dot{y} + k\varepsilon y = \beta_1 \dot{y}(t-1) + \beta_2 (\dot{y}(t-1))^2 + \beta_3 (\dot{y}(t-1))^3 + \dots, \quad (4)$$

где  $\beta_2, \beta_3$  некоторые постоянные.

Рассмотрим задачу построения нормальной формы для уравнения (4), при параметре  $\beta_1 = -(1 + \varepsilon^2 \beta)$ ,  $\beta > 0$ . При таком  $\beta_1$  характеристическое уравнение имеет бесконечное количество корней, стремящихся к мнимой оси, таким образом, реализуется критический случай бесконечной размерности. Для исследования поведения решений воспользуемся методом квазинормальных форм [3]. Представим функцию  $y$  в виде ряда

$$y = \varepsilon V(t) + \varepsilon W(\tau) + \varepsilon^2 U_1(t) + \varepsilon^3 U_2(t) + \dots, \quad (5)$$

где  $U_1, U_2$  периодические с периодом 1:  $U_1(t) \equiv U_1(t+1)$ ,  $U_2(t) \equiv U_2(t+1)$ . Функция  $W$  такая, что среднее значение  $\beta_2 \dot{V}^2 - kW(\tau)$  равняется 0:

$$W(\tau) = \frac{\beta_2}{k} \int_0^1 \dot{V}^2(t, \tau) dt.$$

Функция  $V(t)$  представляется в виде ряда Фурье:

$$V(t) = \sum_{-\infty}^{\infty} e^{i \operatorname{Im}(\lambda_{n0} + \varepsilon \lambda_{n1} + \varepsilon^2 \lambda_{n2} + \dots)t} V_n(\tau), \quad \text{где } \tau = \varepsilon^2 t.$$

Значения  $\lambda_{n0}, \lambda_{n1}, \lambda_{n2}$  определяются из асимптотического приближения корней уравнения (2) (см. Теорему 3).

Обозначим  $\xi_n(\varepsilon)$ :

$$\xi_n(\varepsilon) = i(\lambda_{n0} + \varepsilon \lambda_{n1} + \varepsilon^2 \lambda_{n2} + \dots).$$

Для удобства выпишем первую и вторую производную функции  $V$ .

$$\dot{V} = \sum_{-\infty}^{\infty} (\xi_n V_n(\tau) + \varepsilon^2 V_n'(\tau)) e^{\xi_n t}, \quad (6)$$

$$\ddot{V} = \sum_{-\infty}^{\infty} (\xi_n^2 V_n(\tau) + 2\xi_n \varepsilon^2 V_n'(\tau) + \varepsilon^4 V_n''(\tau)) e^{\xi_n t}. \quad (7)$$

Рассмотрим подробнее  $\dot{y}(t-1)$ :

$$\begin{aligned} \dot{y}(t-1) = \varepsilon \sum_{-\infty}^{\infty} (\xi_n V_n(\tau - \varepsilon^2) + \varepsilon^2 V_n'(\tau - \varepsilon^2)) e^{\xi_n t} e^{-\xi_n} + \\ + \varepsilon^3 \dot{W}(\tau - \varepsilon^2) + \varepsilon^2 \dot{U}_1(t-1) + \varepsilon^3 \dot{U}_2(t-1) + \dots \end{aligned} \quad (8)$$

Раскладывая в ряды Тейлора функции  $e^{-\xi_n}$ ,  $V_n(\tau - \varepsilon^2)$  и  $W(\tau - \varepsilon^2)$ , получим:

$$\begin{aligned} e^{-\xi_n} = e^{-\pi(2n+1)i - \varepsilon \lambda_{n1} - \varepsilon^2 \lambda_{n2} - \dots} = -1 + \varepsilon \lambda_{n1} + \varepsilon^2 \lambda_{n2} - \frac{1}{2} \varepsilon^2 \lambda_{n1}^2 - \dots, \\ V_n(\tau - \varepsilon^2) = V_n(\tau) - \varepsilon^2 V_n'(\tau) + \dots, \\ W(\tau - \varepsilon^2) = W(\tau) - \varepsilon^2 W'(\tau) + \dots \end{aligned} \quad (9)$$

Подставим (5), (6), (7), (8), (9) в уравнение (4), будем приравнивать коэффициенты при одинаковых степенях малого параметра  $\varepsilon$  и соответствующих  $e^{\xi_n t}$ , выпишем результат в явном виде:

$$\begin{aligned} \varepsilon \left( \varepsilon \sum_{-\infty}^{\infty} (\xi_n^2 V_n(\tau) + 2\xi_n \varepsilon^2 V_n'(\tau) + \varepsilon^4 V_n''(\tau)) e^{\xi_n t} + \varepsilon^5 W''(\tau) + \right. \\ \left. + \varepsilon^2 \ddot{U}_1 + \varepsilon^3 \ddot{U}_2 \right) + \varepsilon \sum_{-\infty}^{\infty} (\xi_n V_n(\tau) + \varepsilon^2 V_n'(\tau)) e^{\xi_n t} + \varepsilon^3 W'(\tau) + \\ + \varepsilon^2 \dot{U}_1 + \varepsilon^3 \dot{U}_2 + k\varepsilon \left( \varepsilon \sum_{-\infty}^{\infty} e^{\xi_n t} V_n(\tau) + \varepsilon W(\tau) + \varepsilon^2 U_1 + \varepsilon^3 U_2 \right) + o(\varepsilon^4) = \\ = -(1 + \varepsilon^2 \beta) \left( \varepsilon \sum_{-\infty}^{\infty} (\xi_n (V_n(\tau) - \varepsilon^2 V_n'(\tau)) + \varepsilon^2 (V_n'(\tau) - \varepsilon^2 V_n''(\tau))) e^{\xi_n t} \right. \\ \cdot (-1) (1 - \varepsilon \lambda_{n1} - \varepsilon^2 \lambda_{n2} + \frac{1}{2} \varepsilon^2 \lambda_{n1}^2) + \varepsilon^3 (W'(\tau) - \varepsilon^2 W''(\tau)) + \\ + \varepsilon^2 \dot{U}_1(t-1) + \varepsilon^3 \dot{U}_2(t-1)) + \beta_2 \left( \varepsilon \sum_{-\infty}^{\infty} (\xi_n (V_n(\tau) - \varepsilon^2 V_n'(\tau)) + \right. \\ \left. + \varepsilon^2 (V_n'(\tau) - \varepsilon^2 V_n''(\tau))) e^{\xi_n t} (-1) (1 - \varepsilon \lambda_{n1} - \varepsilon^2 \lambda_{n2} + \frac{1}{2} \varepsilon^2 \lambda_{n1}^2) + \right. \\ \left. + \varepsilon^3 (W'(\tau) - \varepsilon^2 W''(\tau)) + \varepsilon^2 \dot{U}_1(t-1) + \varepsilon^3 \dot{U}_2(t-1) \right)^2 + \\ \left. + \beta_3 \left( \varepsilon \sum_{-\infty}^{\infty} (\xi_n (V_n(\tau) - \varepsilon^2 V_n'(\tau)) + \varepsilon^2 (V_n'(\tau) - \varepsilon^2 V_n''(\tau))) e^{\xi_n t} \right. \right. \\ \left. \cdot (-1) (1 - \varepsilon \lambda_{n1} - \varepsilon^2 \lambda_{n2} + \frac{1}{2} \varepsilon^2 \lambda_{n1}^2) + \right. \\ \left. + \varepsilon^3 (W'(\tau) - \varepsilon^2 W''(\tau)) + \varepsilon^2 \dot{U}_1(t-1) + \varepsilon^3 \dot{U}_2(t-1) \right)^3. \end{aligned} \quad (10)$$

Раскроем скобки, приравняем коэффициенты при одинаковых степенях малого параметра  $\varepsilon$  и соответствующих  $e^{\xi_n t}$ .

При  $\varepsilon$  получаем верное тождество:

$$\lambda_{n0} V_n(\tau) e^{\xi_n t} = \lambda_{n0} V_n(\tau) e^{\xi_n t}.$$

При  $\varepsilon^2$  получаем:

$$\begin{aligned} \sum_{-\infty}^{\infty} \lambda_{n0}^2 V_n(\tau) e^{\xi_n t} + \sum_{-\infty}^{\infty} \lambda_{n1} V_n(\tau) e^{\xi_n t} + \dot{U}_1 + \sum_{-\infty}^{\infty} k V_n(\tau) e^{\xi_n t} + \\ + kW(\tau) = \sum_{-\infty}^{\infty} (-1)(-\lambda_{n1} V_n(\tau) + \lambda_{n1} \lambda_{n0} V_n(\tau)) e^{\xi_n t} - \\ - \dot{U}_1(t-1) + \beta_2 \sum_{-\infty}^{\infty} g_{2n}(V) e^{(\pi 2ni + o(\varepsilon))t}, \end{aligned} \quad (11)$$

где  $g_{2n}(V)$  – это коэффициенты ряда Фурье для функции

$$\left( \sum_{-\infty}^{\infty} \pi(2n+1) i V_n(\tau) e^{\xi_n t} \right)^2.$$

После сокращений из уравнения (11) определяется  $\dot{U}_1$

$$\dot{U}_1 = \frac{1}{2} (\beta_2 \sum_{-\infty}^{\infty} g_{2n}(V(t, \tau)) e^{(\pi 2ni + o(\varepsilon))t} - kW(\tau)),$$

$$\dot{U}_1 = \frac{1}{2} (\beta_2 \dot{V}^2 - kW(\tau)).$$

$U_1$  находится в виде:

$$U_1 = \frac{1}{2} \int_0^r \beta_2 \dot{V}^2(s, \tau) - W(\tau) ds.$$

Обратим внимание, что  $U_1$  периодическая, т.к. среднее значение подынтегральной функции равно нулю.



При  $\varepsilon^3$  получаем:

$$\begin{aligned} & \sum_{-\infty}^{\infty} e^{\xi_n t} 2\lambda_{n0}\lambda_{n1}V_n(\tau) + \ddot{U}_1 + \sum_{-\infty}^{\infty} (\lambda_{n2}V_n(\tau) + V'_n(\tau))e^{\xi_n t} + \dot{U}_2 + \\ & + kU_1 + W'(\tau) = \sum_{-\infty}^{\infty} (-1)e^{\xi_n t} ((-1)(\lambda_{n2}V_n(\tau) - \lambda_{n0}V'_n(\tau) + V'_n(\tau)) + \\ & + \lambda_{n1}^2 V_n(\tau) + (-\lambda_{n2} + \frac{1}{2}\lambda_{n1}^2)\lambda_{n0}V_n(\tau)) - (\dot{U}_2(t-1) + W'(\tau)) + \quad (12) \\ & + \sum_{-\infty}^{\infty} \lambda_{n0}\beta V_n(\tau)e^{\xi_n t} + \dot{U}_1(t-1) + \beta_2 \sum_{-\infty}^{\infty} \varphi_{2n}(V)e^{(\pi(2n+1)i+o(\varepsilon))t} + \\ & + \beta_2 \sum_{-\infty}^{\infty} f_{2n}(V)e^{(\pi 2ni+o(\varepsilon))t} + \beta_3 \sum_{-\infty}^{\infty} \varphi_{3n}(V)e^{(\pi(2n+1)i+o(\varepsilon))t}. \end{aligned}$$

Здесь  $\varphi_{2n}(V)$  – коэффициенты ряда Фурье для функции

$$(-2)\dot{U}_1(t-1) \sum_{-\infty}^{\infty} \pi(2n+1)iV_n(\tau)e^{\xi_n t};$$

$\varphi_{3n}(V)$  – коэффициенты ряда Фурье для функции

$$(-1) \left( \sum_{-\infty}^{\infty} \pi(2n+1)iV_n(\tau) \right)^3;$$

$f_{2n}(V)$  – коэффициенты ряда Фурье для функции

$$(-2) \sum_{-\infty}^{\infty} \lambda_{n0}V_n(\tau)e^{\xi_n t} \cdot \sum_{-\infty}^{\infty} \lambda_{m0}\lambda_{m1}V_m(\tau)e^{\xi_m t}.$$

Уравнение (12) упрощается до вида:

$$\begin{aligned} & \ddot{U}_1 + 2\dot{U}_2 + kU_1 + 2W'(\tau) - \beta_2 \sum_{-\infty}^{\infty} f_{2n}(V)e^{(\pi 2ni+o(\varepsilon))t} = \\ & = \sum_{-\infty}^{\infty} (-\lambda_{n0}V'_n(\tau) + \frac{1}{2}\lambda_{n1}^2\lambda_{n0}V_n(\tau) + \lambda_{n0}\beta V_n(\tau) + \\ & + \beta_2\varphi_{2n}(V) + \beta_3\varphi_{3n}(V))e^{(\pi(2n+1)i+o(\varepsilon))t}. \quad (13) \end{aligned}$$

В уравнении (13) левая часть периодическая, а правая антипериодическая, равенство возможно тогда и только тогда, когда левая и правая часть равна 0.

$\dot{U}_2$  определяется в виде:

$$\dot{U}_2 = \frac{1}{2}(\beta_2 \sum_{-\infty}^{\infty} f_{2n}(V) e^{(\pi 2ni + o(\varepsilon))t} - \ddot{U}_1 - kU_1 - 2W'(\tau)).$$

Рассмотрим антипериодическую часть уравнения (13) и разложим её на соответствующие степени  $e^{\xi_n t}$ :

$$\lambda_{n0} V'_n(\tau) = \frac{1}{2} \lambda_{n1}^2 \lambda_{n0} V_n(\tau) + \lambda_{n0} \beta V_n(\tau) + \beta_2 \varphi_{2n}(V) + \beta_3 \varphi_{3n}(V). \quad (14)$$

Подставим значения  $\lambda_{n1}$  и  $\lambda_{n0}$  в уравнение (14) :

$$\begin{aligned} V'_n(\tau) = & -\frac{1}{2} \left( \pi^2 (2n+1)^2 - 2k + \frac{k^2}{\pi^2 (2n+1)^2} \right) V_n(\tau) + \\ & + \beta V_n(\tau) + \frac{\beta_2}{\pi (2n+1)i} \varphi_{2n}(V) + \frac{\beta_3}{\pi (2n+1)i} \varphi_{3n}(V). \end{aligned} \quad (15)$$

Рассмотрим дифференциальное уравнение вида

$$\begin{aligned} \frac{\partial V}{\partial \tau} = & \frac{1}{2} \frac{\partial^2 V}{\partial r^2} + kV - \frac{k^2}{2} J^2(V) + \\ & + \beta V + \beta_2 J \left( U_1 \frac{\partial V}{\partial r} \right) + \beta_3 J \left( \left( \frac{\partial V}{\partial r} \right)^3 \right), \end{aligned} \quad (16)$$

с дополнительными условиями

$$\int_0^1 V(\tau, r) dr = 0, \quad V(\tau, r) \equiv -V(\tau, r+1). \quad (17)$$

Через  $J(V)$ , как и ранее, обозначена первообразная функции  $V$  с нулевым средним:

$$J^2(V) = J(J(V)), \quad (J(V))'_r \equiv V.$$

Представим функцию  $V$  из уравнения (16) в виде:

$$V = \sum_{-\infty}^{\infty} e^{\pi(2n+1)ir} V_n(\tau). \quad (18)$$

Подставив формулу (18) в уравнение (16) и приравняв коэффициенты получившихся рядов Фурье, мы получим построчное равенство, что для каждого  $n$  справедливо равенство (14).

**Теорема 4.** Пусть  $V_* = \sum_{-\infty}^{\infty} e^{\pi(2n+1)ir} V_n(\tau)$  – решение задачи (16), (17), тогда

$$y(t, \varepsilon) = \varepsilon \sum_{-\infty}^{\infty} e^{\xi_n t} V_n(\varepsilon^2 t) + \varepsilon W(\varepsilon^2 t)$$

является асимптотическим по невязке равномерно по  $t \geq 0$  решением (4).

Доказательство теоремы следует из построений, сделанных ранее.

Аналогичные действия и результаты получаются при  $\beta_1 = 1 + \varepsilon^2 \beta$ .

## 4. Заключение

Рассмотрена модель оптико-электронного осциллятора, найдено его состояние равновесия. Для изучения локальной динамики в окрестности нулевого решения уравнения построен характеристический квазиполином. Выделены критические значения параметра  $\beta$ , при которых состояние равновесия меняет свою устойчивость. Найдено асимптотическое представление корней характеристического квазиполинома при критическом значении параметра  $\beta$ . Интересным оказывается тот факт, что кроме основной «цепочки» стремящихся к мнимой оси корней существует еще один близкий к нулю корень характеристического уравнения. Построено асимптотическое представление корней при малом изменении параметра  $\beta_1$ . Главным результатом работы является построение нормальной формы дифференциального уравнения (4) при критических значениях параметра  $\beta_1$ .

## Литература

1. *Larger L., Maistrenko Y., Penkovskiy B.* Virtual Chimera States for Delayed-Feedback Systems // Physical Review Letters, 2013. Vol. 111. 054103.
2. *Григорьева Е. В., Кащенко С. А., Глазков Д. В.* Особенности локальной динамики модели оптико-электронного осциллятора с запаздыванием // Моделирование и анализ информационных систем. Т. 25, № 1. 2018. С. 71-82.
3. *Кащенко И. С.* Асимптотическое разложение решений уравнений: методические указания. Ярославль: ЯрГУ, 2011.
4. *Кащенко И. С.* Метод квазинормальных форм в уравнениях с запаздыванием: методические указания. Ярославль: ЯрГУ, 2011.

5. *Маслеников И. Н.* Исследование локальной динамики дифференциально-интегрального уравнения с запаздыванием // Современные проблемы математики и информатики. Вып. 18 / Ярославль : ЯрГУ, 2018. С. 39-45.
6. *Хейл Дж.* Теория функционально-дифференциальных уравнений. М. : Мир, 1984.

*Ярославский государственный университет им. П. Г. Демидова*

Д. Д. Федулов, А. Ю. Ухалов

## Оценки минимального коэффициента поглощения $n$ -мерного симплекса

В работе рассматривается задача о минимальном коэффициенте поглощения  $n$ -мерного куба симплексом при гомотетии с центром в центре масс симплекса. Обсуждаются различные численные алгоритмы оценивания этой величины. Получены новые оценки для случаев  $n = 8$  и  $n = 10$ . Приведены наиболее точные известные на данный момент границы коэффициента поглощения при  $1 \leq n \leq 10$ .

### Основные определения и постановка задачи

Пусть  $n \in \mathbb{N}$ ,  $Q_n = [0, 1]^n$ ,  $S$  — невырожденный  $n$ -мерный симплекс. Коэффициентом поглощения куба  $Q_n$  симплексом  $S$  называется величина

$$\xi(S) := \min\{\sigma \geq 1 : Q_n \subset \sigma S\}.$$

Здесь  $\sigma S$  обозначает результат гомотетии симплекса  $S$  с центром в центре масс симплекса с коэффициентом  $\sigma$ . Введем в рассмотрение величину минимального коэффициента поглощения куба  $Q_n$  симплексом

$$\xi_n = \min\{\xi(S) : S \subset Q_n\}.$$

Числа  $\xi_n$  были введены в работах М. В. Невского в связи с задачей о построении минимальных интерполяционных проекторов при линейной интерполяции непрерывных функций, определенных на кубе в  $\mathbb{R}^n$  (см., например, [1]). Задача о минимальном коэффициенте поглощения куба симплексом представляет также и самостоятельный интерес.

Приведем некоторые известные на данный момент свойства констант  $\xi_n$ . М. В. Невским (см. [1]) были установлены следующие факты. Для любого  $n \in \mathbb{N}$  справедливо неравенство  $n \leq \xi_n < n + 1$ . Если  $n > 2$ , то

$$\xi_n \leq \frac{n^2 - 3}{n - 1}. \quad (1)$$

Если  $n + 1$  есть число Адамара, т. е. существует матрица Адамара порядка  $n + 1$ , то  $\xi_n = n$ . В частности,  $\xi_1 = 1$ ,  $\xi_3 = 3$ ,  $\xi_7 = 7$ ,  $\xi_{11} = 11$ ,  $\xi_{15} = 15$  и т. д. Последнее свойство позволяет найти точное значение  $\xi_n$

для бесконечного числа нечетных  $n$ . Доказано также, что  $\xi_2 = \frac{3\sqrt{5}}{5} + 1$ . Отметим, что на данный момент  $n = 2$  является единственным четным  $n$ , для которого  $\xi_n$  известно точно. Для  $\xi_n$ , точные значения которых неизвестны, оценки сверху могут быть получены из общей оценки (1). Эти оценки, однако, не очень точны, так как с ростом  $n$  верхние границы в (1) оказываются близкими к  $n + 1$ .

Позднее в работах М. В. Невского и А. Ю. Ухалова (см. [2]–[4]) были найдены точные значения для некоторых нечетных  $n$  и уточнены оценки для некоторых четных  $n$ . В частности, сейчас известно, что

$$\xi_4 \leq \frac{19 + 5\sqrt{13}}{9} = 4.1141\dots, \quad \xi_5 = 5, \quad \xi_6 < 6.0166, \quad \xi_9 = 9. \quad (2)$$

В работе [1] высказывается предположение, что  $\xi_4 = \frac{19+5\sqrt{13}}{9}$ . На данный момент эту гипотезу не удалось ни доказать, ни опровергнуть. Пока неизвестно, возможно ли выполнение равенства  $\xi_n = n$  при четных  $n$ . Самое малое  $n$ , для которого  $\xi_n = n$ , но  $n + 1$  не является числом Адамара, есть  $n = 5$ .

Для получения оценок (2) использовалась комбинация символьных и численных методов системы Wolfram Mathematica (см., например, [5]). Опыт авторов показывает, что при больших  $n$  (в частности, для четных  $n > 6$  и для нечетных  $n > 9$ ) такой подход оказывается малоэффективным. Возникает вопрос о поиске других методов оценивания  $\xi_n$ .

В настоящей работе обсуждаются вопросы, связанные с использованием различных библиотек численных методов для поиска минимума функции  $\xi(S)$ . Приводятся новые оценки для  $\xi_8$  и  $\xi_{10}$ , полученные в ходе численных экспериментов.

## Вычисление $\xi(S)$

Пусть  $S$  — невырожденный симплекс в  $\mathbb{R}^n$  с вершинами  $x^{(1)}, \dots, x^{(n+1)}$ . Построим матрицу

$$A = \begin{pmatrix} x_1^{(1)} & \dots & x_n^{(1)} & 1 \\ x_1^{(2)} & \dots & x_n^{(2)} & 1 \\ \vdots & \dots & \dots & \vdots \\ x_1^{(n+1)} & \dots & x_n^{(n+1)} & 1 \end{pmatrix}.$$

Матрица  $A$  называется матрицей вершин симплекса  $S$ . Пусть  $A^{-1} = (l_{ij})$ . Многочлены

$$\lambda_j(x) = l_{1,j}x_1 + \dots + l_{n,j}x_n + l_{n+1,j}, \quad j = 1, \dots, n+1$$

называются базисными многочленами Лагранжа симплекса  $S$ . Справедлива формула

$$\xi(S) = (n + 1) \max_{1 \leq k \leq n+1} \max_{x \in \text{ver}(Q_n)} (-\lambda_k(x)) + 1. \quad (3)$$

Здесь  $\text{ver}(Q_n)$  – множество вершин куба  $Q_n$ . Доказательство формулы (3) и описание других свойств базисных многочленов Лагранжа можно найти в [1].

Задача вычисления  $\xi_n$  сводится к отысканию минимума функции  $\xi(S)$  при  $S \subset Q_n$ . Явное выражение правой части (3) через координаты вершин симплекса  $S$  оказывается очень громоздким и сложным для анализа. Численные эксперименты, проведенные авторами, показывают, что эта функция имеет большое количество локальных минимумов. Это обстоятельство существенно затрудняет поиск глобального минимума. В настоящей работе для уточнения оценок используются численные методы.

## Опыт применения различных алгоритмов

В таблице 1 представлены результаты, полученные при использовании различных алгоритмов минимизации: метод последовательного квадратичного программирования (SLSQP, см. [7]–[9]), усеченный алгоритм Ньютона (в частности, метод Ньютона-Ланцоша, TNC, см. [7], [10]–[12]), метод Нелдера-Мида (NM, см. [13]–[15]), метод байесовской оптимизации с использованием гауссовых процессов (SCOPT, см. [16]). В строке таблицы «Теоретическая оценка» приведены значения, получающиеся в правой части (1) при данном  $n$ .

С помощью алгоритма SLSQP при  $n = 4$  было получено  $\xi(S) = 4.1143\dots$  Это число близко к значению  $\frac{19+5\sqrt{13}}{9}$  из (2). В точности такая величина получится при замене десятичных дробей, полученных численным методом, близкими рациональными и иррациональными числами. Такой уточненный симплекс имеет вершины  $(1, \frac{5-\sqrt{13}}{6}, 1, 1)$ ,  $(\frac{1}{2}, 0, 0, \frac{1}{2})$ ,  $(0, 1, \frac{5-\sqrt{13}}{6}, 1)$ ,  $(0, \frac{5-\sqrt{13}}{5}, 1, 0)$ ,  $(1, 1, \frac{5-\sqrt{13}}{6}, 0)$ . Как оказалось, этот симплекс совпадает с точностью до ортогонального преобразования с симплексом  $T$ , полученным в работе [2]. Этот факт служит подкреплением гипотезы о том, что  $\xi_4 = \frac{19+5\sqrt{13}}{9}$ . Отметим также, что все найденные до сих пор симплексы, доставляющие это значение, оказываются эквивалентными одному из двух симплексов, найденных в [2] –  $T$  или  $R$ . Можно высказать гипотезу о том, что при  $n = 4$  имеется всего два (с точностью до ортогонального преобразования) экстремальных симплекса.

Таблица 1: Сравнение различных алгоритмов минимизации  $\xi(S)$ 

Алгоритм	$n = 4$	$n = 6$	$n = 8$
SLSQP	$4.1143... \approx \frac{19+5\sqrt{13}}{9}$	6.0303...	8.3255...
TNC	4.3744...	7.1482...	9.1732...
NM	4.2545...	6.3183...	—
SCOPT	5.5474...	—	—
Теоретическая оценка	$\frac{13}{3} = 4.3333...$	$\frac{33}{5} = 6.6$	$\frac{61}{7} = 8.7142...$

Для  $n = 6$  минимальное достигнутое с помощью алгоритма SLSQP значение 6.0303... не улучшает известную оценку  $\xi_6 < 6.0166$ , полученную в [3].

Приведем некоторые замечания, касающиеся особенностей работы различных алгоритмов.

Среди представленных алгоритмов наиболее быстрыми с точки зрения работы являются алгоритмы последовательного квадратичного программирования (SLSQP) и усеченного алгоритма Ньютона (TNC). Поскольку с увеличением размерности пространства  $n$  увеличивается размерность задачи оптимизации (задача оптимизации рассматривается в пространстве  $\mathbb{R}^{n(n+1)}$ ), только с помощью реализаций этих алгоритмов удалось провести эксперименты для  $n = 8$ . Однако, для  $n = 6$  и  $n = 8$  реализация усеченного алгоритма Ньютона (TNC) показала наихудшие результаты из представленных.

В рамках вычислительного эксперимента реализация алгоритма Нелдера-Мида (NM) показала хорошие результаты для  $n = 4$  и  $n = 6$  и смогла улучшить теоретическую оценку. Однако, для  $n = 8$  проведение вычислительного эксперимента было затруднено, так как увеличилось (по сравнению с алгоритмами SLSQP и TNC) время выполнения одной итерации. Отметим, что в 40-50 % итераций вычислительного алгоритма выполнение оптимизации прерывалось из-за попадания в немодифицируемый симплекс.

Реализация алгоритма байесовской оптимизации с использованием гауссовых процессов (SKGP) была использована только для  $n = 4$ , поскольку выполнение одной итерации алгоритма занимает в несколько раз больше времени по сравнению с реализацией алгоритма Нелдера-Мида (NM). Как видно из таблицы, полученные результаты не позволяют уточнить даже теоретическую оценку.



Таблица 2: Оценки  $\xi_n$ 

$n$	$\xi_n$	$n$	$\xi_n$
1	1	6	$6 \leq \xi_6 < 6.0166$
2	$\frac{3\sqrt{5}}{5} + 1 = 2.3416\dots$	7	7
3	3	8	$8 \leq \xi_8 < 8.1355$
4	$4 \leq \xi_4 \leq \frac{19+5\sqrt{13}}{9} = 4.1141\dots$	9	9
5	5	10	$10 \leq \xi_{10} < 10.2342$

## Новые оценки $\xi_n$

Для получения новых оценок  $\xi_8$  и  $\xi_{10}$  использовались функции численной минимизации библиотеки Dlib C++ Library (см. [6]).

Новая оценка для  $\xi_8$  получается из рассмотрения симплекса  $S_1$  с вершинами

(0., 0.487324, 0.999584, 0.000015, 0., 0., 1., 0.000054),  
 (0., 0.021247, 0., 0.994519, 0.997612, 0.007806, 0.000049, 0.000363),  
 (0.994191, 0., 1., 1., 0.093607, 1., 0., 0.404681),  
 (0.999981, 0., 0., 0.320835, 0., 0., 0.99927, 1.),  
 (0.998303, 1., 0.989783, 1., 1., 0.269099, 1., 0.),  
 (0.995842, 1., 0., 0., 0.414823, 0.999313, 0.018129, 0.),  
 (0.671663, 0.99946, 1., 0.031503, 0.99879, 0.000322, 0., 1.),  
 (0.00547, 0.021783, 0.452784, 0.077069, 1., 0.999996, 1., 0.997195),  
 (0., 0.99992, 0.123118, 1., 0.000852, 0.999914, 0.483036, 0.936099).

Вычисления по формуле (3) дают  $\xi(S_1) = 8.13545\dots$  Следовательно, можно утверждать, что  $\xi_8 < 8.1355$ .

Для уточнения оценки  $\xi_{10}$  рассмотрим симплекс  $S_2$  с вершинами  
 (0.041047, 0.037879, 1., 0.330575, 1., 0.0, 1., 0.048055, 1., 0.291904),  
 (0.960780, 0., 1., 0.250305, 0.999706, 0.916678, 0.014938, 1., 0.000089, 1.),  
 (1., 0.734044, 0., 0., 1., 0.999928, 0.204362, 0., 0.488617, 0.),  
 (1., 0.046996, 0.920973, 0.950917, 0., 1., 1., 0.999620, 1., 0.053571),  
 (0., 1., 0.004464, 1., 1., 0.374532, 1., 1., 0., 0.006593),  
 (0., 0., 0.392400, 1., 0., 0.752041, 0., 0., 0., 0.002029),  
 (1., 0.817225, 0., 1., 0.801914, 0., 0.019363, 0.221740, 0.782488, 0.991481),

(0., 0.999979, 0.017534, 0.532962, 0.581550, 1., 0.222236, 0.231594, 1., 1.),  
 (0.992805, 1., 1., 0.366910, 0., 0.644450, 1., 0.040009, 0.000828, 1.),  
 (0.305125, 0., 0.004921, 0.003458, 0.000000, 0.041917, 1., 1., 0.174159, 1.),  
 (0.051508, 0.993242, 1., 0., 0.026295, 0., 0.000001, 1., 0.928228, 0.000312).  
 Для симплекса  $S_2$  имеем  $\xi(S_2) = 10.23415\dots$ . Отсюда  $\xi_{10} < 10.2342$ .

В таблице 2 приводятся оценки первых десяти чисел  $\xi_n$ . Таблица содержит результаты работ [1]–[4] и настоящей статьи. Для величин, точные значения которых неизвестны, приводятся наиболее точные из известных неравенств.

## Благодарность

Численные эксперименты проводились на вычислительном кластере лаборатории «Дискретная и вычислительная геометрия» им. Б. Н. Делоне Ярославского государственного университета им. П. Г. Демидова. Авторы выражают благодарность ведущему программисту Университетского центра Интернет ЯрГУ Дмитрию Сергеевичу Глызину за помощь, оказанную при использовании вычислительного кластера.

## Литература

1. *Невский М. В.* Геометрические оценки в полиномиальной интерполяции / Ярославль: Яросл. гос. ун-т им. П. Г. Демидова, 2012.
2. *Невский М. В., Ухалов А. Ю.* О числовых характеристиках симплекса и их оценках // Моделирование и анализ информационных систем. 2016. Том 23, № 5. С. 603–619.
3. *Невский М. В., Ухалов А. Ю.* Новые оценки числовых величин, связанных с симплексом // Моделирование и анализ информационных систем. 2017. Том 24, № 1. С. 94–110.
4. *Nevskii M., and Ukhalov A.* Perfect simplices in  $\mathbb{R}^5$  // Beiträge zur Algebra und Geometrie / Contributions to Algebra and Geometry. 2018. Vol 59, Issue 3, pp. 501–521.
5. *Wolfram S.* An elementary introduction to the Wolfram Language. — New York: Wolfram Media, 2015. 342 p.
6. *King, Davis E.* Dlib-ml: A Machine Learning Toolkit // Journal of Machine Learning Research. 2009. Vol. 10, pp. 1755–1758.
7. *Nocedal J., Wright S. J.* Numerical Optimization. New York: Springer. 2006. ISBN: 9780387303031.

8. *Boggs P., Tolle J. W.* Sequential Quadratic Programming // *Acta Numerica*. 1995. Vol. 4. 1-51. 10.1017/S0962492900002518.
9. *Bonnans J. F., Frédéric G. J., Lemaréchal C., Sagastizábal C. A.* Numerical optimization: Theoretical and practical aspects. Berlin: Springer-Verlag. 2006. ISBN: 9783540354451.
10. *Nash S. G.* A survey of truncated-Newton methods // *Journal of Computational and Applied Mathematics*. 2000. Vol. 124. pp. 45-49.
11. *Martens J.* Deep learning via Hessian-free optimization // *Proceedings of the 27 th International Conference on Machine Learning, Haifa, Israel, 2010*.
12. *Nash S. G.* Newton-Type Minimization via the Lanczos Method // *Siam Journal on Numerical Analysis - SIAM J NUMER ANAL*. 1984. Vol. 21. 770-788. 10.1137/0721052.
13. *Nelder J. A., Mead R.* A simplex method for function minimization. // *Computer Journal*. 1965. Vol. 7, pp. 308–313.
14. *Wright M. H.* Direct Search Methods: Once Scorned, Now Respectable // *Numerical Analysis 1995, Proceedings of the 1995 Dundee Biennial Conference in Numerical Analysis*, D.F. Griffiths and G.A. Watson (Eds.), Addison Wesley Longman, Harlow, UK. 1996. pp. 191–208.
15. *Singer S., Nelder J. A.* Nelder-Mead algorithm [Electronic resource]. – URL: [http://www.scholarpedia.org/article/Nelder-Mead\\_algorithm](http://www.scholarpedia.org/article/Nelder-Mead_algorithm) Просмотрено: 31.01.2019.
16. *The scikit-learn python library contributors* Documentation for Scikit-Optimize module [Electronic resource]. – URL: [scikit-optimize.github.io](http://scikit-optimize.github.io) Просмотрено: 31.01.2019.

*Ярославский государственный университет им. П. Г. Демидова*

Научное издание

СОВРЕМЕННЫЕ ПРОБЛЕМЫ  
МАТЕМАТИКИ И ИНФОРМАТИКИ

Сборник научных трудов  
молодых ученых, аспирантов и студентов  
Выпуск 19

Компьютерный набор, верстка Д. В. Глазков

Подписано в печать 29.11.19. Формат 60×84/16.  
Усл. печ. л. 3,5. Уч.-изд. л. 3,3.  
Тираж 28 экз. Заказ

Оригинал-макет подготовлен  
в редакционно-издательском отделе ЯрГУ.

Ярославский государственный университет им. П. Г. Демидова.  
150003, Ярославль, ул. Советская, 14.