

Министерство науки и высшего образования РФ  
Ярославский государственный университет им. П. Г. Демидова  
Математический факультет



## **ПУТЬ В НАУКУ. МАТЕМАТИКА**

**Тезисы докладов конференции**

Всероссийская молодежная конференция  
**11 – 16 мая 2021 г., Ярославль**

Ярославль  
ЯрГУ  
2021

УДК 51(063)  
ББК В1я43  
П90

*Печатается в соответствии с решением оргкомитета  
Всероссийской молодежной конференции*

**Путь в науку. Математика** : тезисы докладов конференции / под. ред. А. А. Кащенко, И. С. Кащенко; Яросл. гос. ун-т им. П. Г. Демидова. – Ярославль : ЯрГУ, 2021. – 88 с. – (Всероссийская молодежная конференция, 11–16 мая 2021г., Ярославль).  
ISBN 978-5-8397-1214-0

В сборнике представлены материалы и тезисы докладов Всероссийской молодежной конференции «Путь в науку. Математика», проходившей 11–16 мая 2021г. в Ярославском государственном университете им. П. Г. Демидова.

УДК 51(063)  
ББК В1я43

**Организатор конференции:**

Ярославский государственный университет им. П. Г. Демидова

**Редакционная коллегия:**

И. С. Кащенко – д-р физ.-мат. наук, доц., ЯрГУ  
А. А. Кащенко – канд. физ.-мат. наук, ЯрГУ

ISBN 978-5-8397-1214-0

©ЯрГУ, 2021

# СОДЕРЖАНИЕ

<b>СЕКЦИЯ ЧИСТАЯ И ПРИКЛАДНАЯ АЛГЕБРА</b>	<b>6</b>
<i>Безуглова И. И.</i> Обобщения цепных дробей . . . . .	6
<i>Куликов Е. А.</i> Некоторые универсальные конструкции для векторных пространств . . . . .	9
<i>Медведев Е. А.</i> Обобщённая поточечная функция Гильберта когерентного пучка на многообразии и её первые свойства . . . . .	12
<b>СЕКЦИЯ ДИФФЕРЕНЦИАЛЬНЫЕ УРАВНЕНИЯ</b>	<b>16</b>
<i>Володина Е. И.</i> Об одной математической модели искусственной генной сети . . . . .	16
<i>Герасимов Е. Ю.</i> Об экспериментальной проверке в задаче о колебаниях мембраны . . . . .	19
<i>Головкин И. А.</i> Управление полетом квадрокоптера . .	20
<i>Канцидал Е. С.</i> К вопросу о влиянии младших резонансов в задаче о синхронизации колебаний . . . .	23
<i>Колесов В. А.</i> Преобразования Дарбу и уравнение тэт-раздров . . . . .	24
<i>Мкртчян А. А.</i> Автоколебания в одной задаче из радиофизики . . . . .	27
<i>Никитина А. А.</i> Уравнение Янга–Бакстера и приложения . . . . .	30
<i>Уланова А. Д., Уланов П. А.</i> Компьютерная визуализация фазовых портретов динамических систем на плоскости . . . . .	33
<i>Фисенко К. И.</i> Уравнения в квад-графах и преобразование Бэклунда . . . . .	35

## СЕКЦИЯ ПРОБЛЕМЫ РАЗРАБОТКИ И ЗАЩИТЫ ПРОГРАММ И ДАННЫХ 39

<i>Алексеев Д. И.</i> Проблемы разработки программ на примере создания приложения для визуализации LDAP объектов . . . . .	39
<i>Баулина А. А.</i> О важности переопределения метода GetHashCode . . . . .	42
<i>Бильчугова А. А.</i> Штрих-коды . . . . .	43
<i>Гибадулин Р. А.</i> Алгоритм Тарского. Система ввода-вывода . . . . .	45
<i>Гребенчук Д. С.</i> Определение доли плодовой ДНК . . .	46
<i>Грибкова И. С.</i> Исследование свойств решеток, используемых в качестве криптографических примитивов . . . . .	49
<i>Иванов И. О.</i> Уязвимости веб приложений. Cross-site scripting . . . . .	50
<i>Кобяков С. А.</i> Описание уязвимостей на сайте обучающей SQL системы . . . . .	52
<i>Никешина А. С.</i> Структуры данных на основе кучи . .	53
<i>Скородумов С. А.</i> Модель вируса на языке программирования Python . . . . .	55

## СЕКЦИЯ МАТЕМАТИЧЕСКИЙ АНАЛИЗ 58

<i>Петров И. А.</i> Эйлеровы числа и их приложения . . . .	58
<i>Сычев Р. С.</i> Поиск приближенного равновесия в играх с неполной информацией . . . . .	60
<i>Федулов Д. Д.</i> Обучение глубоких нейронных сетей . .	63
<i>Федотова О. Н.</i> Некоторые алгоритмы обработки изображений . . . . .	65
<i>Краснова Е. А., Чубукова К. Н.</i> Алгоритм Тренча в задачах аппроксимации . . . . .	68

## СЕКЦИЯ МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ 71

<i>Давыдов А. А.</i> О решении задач методами теории подобия и размерностей . . . . .	71
---	----

<i>Костерин Д. С.</i> Кусочно-гладкие решения одного пространственно-распределенного уравнения с кубической нелинейностью . . . . .	74
<i>Куликов В. А.</i> Особенности бифуркаций автоколебательных решений нелинейного параболического уравнения с оператором поворота пространственного аргумента и запаздыванием . . . . .	76
<i>Курбако А. В., Боровкова Е. И., Кульминский Д. Д., Храмков А. Н., Ежов Д. М.</i> Сравнение методов диагностики в реальном времени фазовой синхронизации автогенераторов по их нестационарным временным рядам . . . . .	77
<i>Маслеников И. Н.</i> Локальная динамика уравнения второго порядка с запаздыванием в производной . . .	80
<i>Погребняк М. А.</i> Моделирование движения транспортного потока . . . . .	81
<i>Татанова Е. М.</i> Нормальная форма дифференциального уравнения с запаздыванием и непостоянными коэффициентами . . . . .	84

## СЕКЦИЯ ЧИСТАЯ И ПРИКЛАДНАЯ АЛГЕБРА

### Обобщения цепных дробей

**И. И. Безуглова (ЯрГУ им. П.Г. Демидова,  
Ярославль)**

*Научный руководитель: к.ф.-м.н. Сорокина М. Е.*

Одним из важных понятий теории чисел является понятие непрерывной дроби. В ВУЗе предполагается изучение непрерывных дробей вида

$$a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \frac{1}{a_4 + \dots}}}},$$

где  $a_0 \in \mathbb{Z}$ ,  $a_i \in \mathbb{N}$  при  $i = 1, 2, \dots$ , которые будем называть *обыкновенными непрерывными* или *цепными* дробями.

Цель работы – изучить вопрос о том, какие еще виды непрерывных дробей существуют, их приложения, а также рассмотреть пример их применения.

Впервые понятие цепной дроби ввел итальянский математик Р. Бомбелли в книге “Алгебра”, вышедшей в 1572 году. Как самостоятельный раздел математики теория цепных дробей оформилась в работах Л. Эйлера. Также в данном направлении работали многие математики XVIII-XIX вв. и в настоящее время оно является актуальным. Исследования посвящены как поиску новых обобщений непрерывных дробей и их приложениям, так и модификации и обобщению существующих алгоритмов (см., например, [1]).

### Виды обобщенных непрерывных дробей

Определим непрерывную дробь при помощи определителей

матриц:

$$\omega = \frac{\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} & \cdots \\ a_{21} & a_{22} & \cdots & a_{2n} & \cdots \\ \cdot & \cdot & \cdots & \cdot & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & \cdots \\ \cdot & \cdot & \cdots & \cdot & \cdots \end{vmatrix}}{\begin{vmatrix} a_{22} & \cdots & a_{2n} & \cdots \\ \cdot & \cdots & \cdot & \cdots \\ a_{n2} & \cdots & a_{nn} & \cdots \\ \cdot & \cdots & \cdot & \cdots \end{vmatrix}},$$

где  $a_{ij}$  – независимые переменные (это могут быть действительные или комплексные числа, функции, или элементы другой природы). Подходящие дроби к ней задаются при помощи определителей следующим образом:

$$\frac{P_1}{Q_1} = \frac{a_{11}}{1}, \quad \frac{P_2}{Q_2} = \frac{\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}}{|a_{22}|}, \quad \frac{P_3}{Q_3} = \frac{\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}}{\begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix}} \dots$$

Выбирая подходящий вид определителей, мы получим частные виды непрерывных дробей:

- классические непрерывные дроби:

$$a_{11} + \frac{a_{12}}{a_{22} + \frac{a_{23}}{a_{33} + \frac{a_{34}}{a_{44} + \frac{a_{45}}{a_{55} + \dots}}}}, \quad (1)$$

где  $a_{ij} \in \mathbb{Z}$ ,  $i, j \in \mathbb{N}$ ;

- ветвящиеся дроби:

$$\omega = b_0 + \frac{a_1}{b_1 + \frac{a_3}{b_3 + \dots} + \frac{a_4}{b_4 + \dots}} + \frac{a_2}{b_2 + \frac{a_5}{b_5 + \dots} + \frac{a_6}{b_6 + \dots}};$$

- дроби Хессенберга:

$$\omega = b_0 + \frac{a_1 + \frac{a_2 + \frac{a_4 + \frac{a_6 + \dots}{b_6}}{b_4}}{b_2}}{b_1 + \frac{a_3}{b_3 + \frac{a_5}{b_5 + \frac{a_7}{b_7 + \dots}}}}.$$

Обобщения непрерывных дробей имеют множество различных применений, например, они используются для представления алгебраических иррациональностей высших степеней, для решения систем линейных алгебраических уравнений и для приближения функций многих переменных.

### $r/\varphi$ алгоритм

Непрерывная дробь *сходится* в классическом смысле, если  $\lim_{n \rightarrow \infty} \frac{P_n}{Q_n} = \alpha \in \mathbb{R}$ . Для цепных дробей доказано, что это выполняется всегда (см., например, [2]), но для непрерывных дробей (1) это не так. Поэтому в работе [3] предложен альтернативный способ определения сходимости –  $r/\varphi$  алгоритм.

Непрерывная дробь (1) сходится и имеет своим значением в общем случае комплексное число  $z = r_0 e^{i\phi_0}$ , если существуют пределы

$$\lim_{n \rightarrow \infty} \sqrt[n]{\prod_{i=1}^n \left| \frac{P_i}{Q_i} \right|} = r_0, \quad \pi \lim_{n \rightarrow \infty} \frac{k_n}{n} = |\phi_0|,$$

где  $\frac{P_i}{Q_i}$  – значения  $i$ -й подходящей дроби, а  $k_n$  – число отрицательных подходящих дробей из  $n$  подходящих дробей.

### Примеры разложений

Рассмотрим разложение  $e^x$  в цепную дробь.

Разложение степенной функции в цепную дробь приведено в работе [4] и имеет вид  $y - y_0 = \frac{x - x_0}{(dx/dy)_0 + \frac{x - x_0}{(dx/dz_2)_0 + \dots}}$ , где  $z_1 = (x - x_0)/(y - y_0)$ ,  $z_{i+1} = (x - x_0)/[z_i - (z_i)_0]$ ,  $i \in \mathbb{N}$ ,  $(z_i)_0 = z_i|_{x_0}$ ,  $(dx/dy)_0 = (dx/dy)|_{y_0}$ .

Пусть  $y = e^x$ . Для получения  $n$  коэффициентов разложения необходимо вычислить  $n$  последовательных производных функции  $y$  при  $x_0 = 0$ , затем  $n - 1$  производную функции  $z_1 = x/(e^x - 1)$ , затем  $n - 2$  производные функции  $z_2$  и так далее до первой производной  $z_{n-1}$ . Имеем разложение

$$e^x = 1 + \frac{x}{1 + \frac{x}{-2 + \frac{x}{-3 + \frac{x}{2 + \frac{x}{5 + \frac{x}{-2 + \dots}}}}}},$$

которое при  $x = 1$  позволяет получить значение  $e$  с любой наперед заданной точностью.



## Литература

1. Брюно А. Д., “Универсальное обобщение алгоритма цепной дроби” Чебышевский сб, **16**, No. 2, 35–65 (2015).
2. Бухштаб А. А., Теория чисел, Просвещение, Москва (1966).
3. Шмойлов В. И., Коваленко В. Б., “Некоторые примеры алгоритма суммирования расходящихся непрерывных дробей” Вестник южного научного центра РАН, **8**, No. 4, 3–13 (2012).
4. Ауслендер Г. “О разложении функции в ряды и непрерывные дроби” Ж. вычисл. матем. и матем. физ., **3**, No. 3, 752–757 (1963).

### Некоторые универсальные конструкции для векторных пространств

**Е. А. Куликов** (ЯрГУ им. П.Г. Демидова, Ярославль)  
*Научный руководитель: д.ф.-м.н., доцент Тимофеева Н. В.*

При рассмотрении классического прямого произведения векторных пространств выясняется, что оно не является векторным пространством.

Была поставлена следующая задача: построить такой аналог прямого произведения векторных пространств, что он будет сам являться векторным пространством.

Пусть даны два векторных пространства  $\mathbf{V}$  и  $\mathbf{U}$  (размерностей  $n$  и  $m$  соответственно) над полем  $\mathbf{F}$ .

Обозначим искомое векторное пространство как  $\mathbf{V} \otimes \mathbf{U}$  и определим некоторые его свойства:

- 1)  $\forall x, y \in \mathbf{V}, \forall z \in \mathbf{U} : (x + y) \otimes z = x \otimes z + y \otimes z,$
- 2)  $\forall x \in \mathbf{V}, \forall y, z \in \mathbf{U} : x \otimes (y + z) = x \otimes y + x \otimes z,$
- 3)  $\forall x \in \mathbf{V}, \forall y \in \mathbf{U}, \forall \alpha \in \mathbf{F} : (\alpha x) \otimes y = \alpha(x \otimes y) = x \otimes (\alpha y).$

Нетрудно заметить, что если  $\{e_1, \dots, e_n\}$  и  $\{f_1, \dots, f_m\}$  – это базисы  $\mathbf{V}$  и  $\mathbf{U}$  соответственно, то общий вид элемента в  $\mathbf{V} \otimes \mathbf{U}$  можно представить в виде:

$$\sum_{i=1}^n \sum_{j=1}^m \alpha_i \beta_j (e_i \otimes f_j).$$

Очевидно, базисом векторного пространства  $\mathbf{W} = \mathbf{V} \otimes \mathbf{U}$  будет  $\mathbf{B}_W = \{e_i \otimes f_j | i = 1, \dots, n, j = 1, \dots, m\}$ , а само пространство имеет следующий вид:

$$\mathbf{W} = \langle e_1 \otimes f_1, \dots, e_1 \otimes f_m, \dots, e_n \otimes f_1, \dots, e_n \otimes f_m \rangle.$$

**Утверждение 1.** Пусть в  $V$  выбраны базисы

$\mathbf{B}_V = \{e_1, \dots, e_n\}$  и  $\mathbf{B}'_V = \{e'_1, \dots, e'_n\}$ , связанные соотношениями

$$e'_t = \sum_{i=1}^n a_{it} e_i, \quad t = 1, \dots, n,$$

а в  $U$  – базисы  $\mathbf{B}_U = \{f_1, \dots, f_m\}$  и  $\mathbf{B}'_U = \{f'_1, \dots, f'_m\}$ , связанные соотношениями:

$$f'_k = \sum_{j=1}^m b_{jk} f_j, \quad k = 1, \dots, m.$$

Тогда если  $(a_{ij})$  – матрица перехода  $\mathbf{B}_V \rightarrow \mathbf{B}'_V$ , а  $(b_{tk})$  – матрица перехода  $\mathbf{B}_U \rightarrow \mathbf{B}'_U$ :

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}, \quad B = \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \dots & b_{mm} \end{pmatrix},$$

то матрица перехода к новому базису  $\mathbf{B}_W \rightarrow \mathbf{B}'_W$  в  $W$  будет иметь вид:

$$\Lambda = \begin{pmatrix} a_{11}b_{11} & a_{11}b_{12} & \dots & a_{11}b_{1m} & \dots & a_{1n}b_{11} & \dots & a_{1n}b_{1m} \\ a_{11}b_{21} & a_{11}b_{22} & \dots & a_{11}b_{2m} & \dots & a_{1n}b_{21} & \dots & a_{1n}b_{2m} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{11}b_{m1} & a_{11}b_{m2} & \dots & a_{11}b_{mm} & \dots & a_{1n}b_{m1} & \dots & a_{1n}b_{mm} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{n1}b_{11} & a_{n1}b_{12} & \dots & a_{n1}b_{1m} & \dots & a_{nn}b_{11} & \dots & a_{nn}b_{1m} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{n1}b_{m1} & a_{n1}b_{m2} & \dots & a_{n1}b_{mm} & \dots & a_{nn}b_{m1} & \dots & a_{nn}b_{mm} \end{pmatrix}.$$

Очевидно, что процедура перехода к новому базису в новом пространстве  $\mathbf{B}_W \rightarrow \mathbf{B}'_W$  происходит естественным образом, т.е.

выражением новых базисных векторов через старые:

$$(e_t \otimes f_k)' = \sum_{i=1}^n \sum_{j=1}^m \alpha_{it} \beta_{jk} (e_i \otimes f_j), \quad t = 1, \dots, n, \quad k = 1, \dots, m.$$

Полученные векторы «укладываются» в матрицу столбцами, и таким образом мы получаем матрицу  $\Lambda$  перехода от  $\mathbf{B}_W$  к  $\mathbf{B}'_W$ .

Итак, определим отображение  $\gamma : \mathbf{V} \times \mathbf{U} \hookrightarrow \mathbf{V} \otimes \mathbf{U}$ , действующее по правилу:  $\gamma(x, y) = x \otimes y$ . Элементы вида  $x \otimes y$ , т.е. элементы образа  $\mathbf{V} \times \mathbf{U}$ , называются приводимыми тензорами.

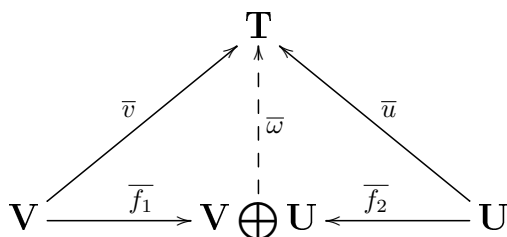
Теперь обратимся к свойству универсальности векторного пространства  $\mathbf{V} \otimes \mathbf{U}$ . Пусть задано билинейное отображение  $\beta$  прямого произведения  $\mathbf{V} \times \mathbf{U}$  в  $\mathbf{F}$ -векторное пространство  $\mathbf{T}$ . Всегда имеется отображение  $\gamma : \mathbf{V} \times \mathbf{U} \hookrightarrow \mathbf{V} \otimes \mathbf{U}$ , где  $\gamma(x, y) = x \otimes y$ . Каждое отображение  $\beta : \mathbf{V} \times \mathbf{U} \longrightarrow \mathbf{T}$  можно «пропустить» через  $\mathbf{V} \otimes \mathbf{U}$ , которое определено однозначно, а, значит, универсально для любой пары  $(\mathbf{V}, \mathbf{U})$ : существует единственное  $\mathbf{F}$ -линейное отображение, определенное на приводимых тензорах соответствием  $\omega(x \otimes y) = \beta(x, y)$ , включаемое в коммутативную диаграмму.

$$\begin{array}{ccc} \mathbf{V} \times \mathbf{U} & \xrightarrow{\beta} & \mathbf{T} \\ & \searrow \gamma & \nearrow \omega \\ & \mathbf{V} \otimes \mathbf{U} & \end{array}$$

Построенное нами произведение векторных пространств известно под названием тензорного произведения векторных пространств.

Существует свойство универсальности и для прямой суммы векторных пространств.

Для любого  $\mathbf{F}$ -векторного пространства и любых отображений  $\bar{u} : \mathbf{U} \longrightarrow \mathbf{T}$  и  $\bar{v} : \mathbf{V} \longrightarrow \mathbf{T}$  существует единственное  $\mathbf{F}$ -линейное отображение  $\bar{\omega} : \mathbf{V} \oplus \mathbf{U} \longrightarrow \mathbf{T}$ , включаемое вместе с отображениями  $\bar{f}_1 : \mathbf{V} \longrightarrow \mathbf{V} \oplus \mathbf{U} : x \longrightarrow (x, 0)$  и  $\bar{f}_2 : \mathbf{U} \longrightarrow \mathbf{V} \oplus \mathbf{U} : y \longrightarrow (0, y)$  в коммутативную диаграмму



### Литература

1. Жаринов В.В., Лекционные курсы НОЦ. Выпуск 9, Математический институт им. В.А. Стеклова, Москва (2008).

## Обобщённая поточечная функция Гильберта когерентного пучка на многообразии и её первые свойства

*Е. А. Медведев (ЯрГУ им. П.Г. Демидова, Ярославль)*

*Научный руководитель: д.ф.-м.н., доцент Тимофеева Н.В.*

Пусть  $X$  – алгебраическое многообразие или нетерова схема конечного типа над полем  $k = \bar{k}$ ,  $\mathcal{O}_X$  – его структурный пучок. Пусть задан некоторый когерентный пучок  $\mathcal{O}_X$ -модулей  $\mathcal{F}$ . Обобщенной поточечной функцией Гильберта пучка  $\mathcal{F}$  в замкнутой точке  $p \in X$  назовем следующее отношение

$$\pi_p^{\mathcal{F}}(n) = \frac{\dim(\mathcal{F} \otimes \mathcal{O}_X/\mathfrak{m}^n)_p}{\dim(\mathcal{O}_X/\mathfrak{m}^n)_p},$$

где  $\mathfrak{m}$  – пучок максимальных идеалов, соответствующий точке  $p$ . Несколько более общий вид этой функции был введен в [3] как средство диагностики свойства пучка быть плоским. Отметим, что  $(\mathcal{F} \otimes \mathcal{O}_X/\mathfrak{m}^n)_p$  и  $(\mathcal{O}_X/\mathfrak{m}^n)_p$  являются конечномерными  $k$ -векторными пространствами.

Далее в докладе под функцией Гильберта будем понимать обобщённую поточечную функцию Гильберта.

В рамках работы были поставлены следующие задачи:

- Вычислить значения обобщенной поточечной функции Гильберта в некоторых ключевых ситуациях.

- Сформулировать первые наблюдения и первые свойства функции.

Справедливо утверждение (утверждение о постоянстве функции является частным случаем результата из [3], а утверждение о её значении – доказанное нами добавление):

**Теорема 1.** *Обобщенная поточечная функция Гильберта локально свободного  $\mathcal{O}_X$  пучка  $\mathcal{F}$  является константой по  $n$ , одинаковой во всех точках многообразия  $X$ , и равна рангу пучка  $\mathcal{F}$ .*

Приведем пример не локально свободного пучка на аффинной прямой  $\mathbb{A}_k^1 = \operatorname{Spec}(k[x])$  и покажем, что функция Гильберта отлична от константы.

Рассмотрим следующий пучок (пучок-небоскреж)  $\mathcal{F}$ , заданный своими пространствами сечений следующим образом:

$$\mathcal{F}(U) = \begin{cases} k, & \text{если } 0 \in U, \\ 0, & \text{если } 0 \notin U, \end{cases}$$

где  $U$  – открытое подмножество  $U \subset \mathbb{A}_k^1$ . Пусть  $p = 0$ , тогда  $\mathfrak{m} = (x) \subset k[x]$ . Имеем  $(\mathcal{F} \otimes \mathcal{O}_X/\mathfrak{m}^n)_p \simeq (\mathcal{F}(X) \otimes_{k[x]} k[x]/(x^n)) \otimes_{k[x]} k[x]_{(x)} \simeq k$ . Поэтому  $\dim(\mathcal{F} \otimes \mathcal{O}_X/\mathfrak{m}^n)_p = 1$ .

Далее,  $(\mathcal{O}_X/\mathfrak{m}^n)_p = (k[x]/(x)^n) \otimes_{k[x]} k[x]_{(x)} \simeq k[x]/(x^n)$ . Последний изоморфизм алгебр справедлив в силу локальности алгебры  $k[x]/(x^n)$ . Так как  $k[x]/(x^n) = \langle \bar{1}, \bar{x}, \dots, \bar{x}^{n-1} \rangle_k$ , то  $\dim(\mathcal{O}_X/(x^n))_p = \dim k[x]/(x^n) = n$ . Таким образом,

$$\pi_p^{\mathcal{F}}(n) = \frac{1}{n}.$$

Функцию Гильберта можно использовать как инструмент для изучения носителя пучка.

Пусть задана аффинная плоскость  $X = \operatorname{Spec}(k[x, y])$  и прямая  $x = 0$  с пучком колец  $\mathcal{O}_l = \mathcal{O}_X/(x)$ . Вычислим функцию Гильберта для этого пучка в точке  $(0, 0)$  с максимальным идеалом  $\mathfrak{m} = (x, y)$ .

Вычислим знаменатель. Так как  $(x, y)^n = (x^n, x^{n-1}y, \dots, y^n)$ , то

$$k[x, y]/(x, y)^n \simeq \langle \bar{x}^s \bar{y}^t | 0 \leq s + t < n \rangle_k.$$

Полученная алгебра является локальной, и потому изоморфна своей локализации. Таким образом,  $\dim(\mathcal{O}_X/(x, y)^n)_p = n(n+1)/2$ . Вычислим числитель. Так как пространство глобальных сечений  $\mathcal{O}_l$  имеет вид  $k[x, y]/(x)$ , то

$$\mathcal{O}_X/(x, y)^n \otimes_{k[x, y]} k[y] \simeq \langle \bar{x}^s \bar{y}^t \otimes y^r | 0 \leq s + t < n, r \geq 0 \rangle_k.$$

В силу универсальности тензорного произведения,

$$\mathcal{O}_X/(x, y)^n \otimes_{k[x, y]} k[y] \simeq \langle \bar{1}, \bar{y}, \dots, \bar{y}^{n-1} \rangle_k \simeq k[y]/(y^n).$$

Имеем  $\dim(\mathcal{O}_X/(x, y)^n \otimes_{k[x, y]} k[y])_p = n$ . Таким образом,

$$\pi_p^{\mathcal{O}_l}(n) = \frac{n}{\frac{n(n+1)}{2}} = \frac{2}{n+1} \sim \frac{2}{n^1} \text{ при } n \rightarrow \infty.$$

Теперь проведем те же самые действия, но для другого подмногообразия  $c$ , которое задано уравнением  $xy = 0$ . Пучком колец этого подмногообразия будет  $\mathcal{O}_c = \mathcal{O}_X/(xy)$ .

Вычислим числитель функции Гильберта пучка  $\mathcal{O}_c$  в точке  $(0, 0)$ . Имеем изоморфизм  $k$ -векторных пространств:

$$k[x, y]/(xy) \simeq \langle 1, x^s, y^t | s, t > 0 \rangle_k \simeq k[x] \oplus yk[y].$$

Обозначим  $k[x, y]/(x, y)^n \otimes_{k[x, y]} k[x] =: T_n[x]$ . Заметим, что  $T_n[x] \simeq \langle \bar{1}, \bar{x}, \dots, \bar{x}^n \rangle_k$ . Умножим полученную сумму тензорно на  $k[x, y]/(x, y)^n$ .  $k[x, y]/(xy) \otimes_{k[x, y]} k[x, y]/(x, y)^n \simeq T_n[x] \oplus yT_n[y]$ . Заметим,  $T_n[x] \oplus yT_n[y] \simeq \langle 1, \bar{x}^s, \bar{y}^t | 0 < s, t < n \rangle_k$ . Полученная алгебра будет локальной, и  $\dim(T_n[x] \oplus yT_n[y]) = 2n - 1$ . Таким образом,

$$\pi_p^{\mathcal{O}_c}(n) = \frac{2n-1}{\frac{n(n+1)}{2}} = 2 \cdot \frac{2n-1}{n(n+1)} \sim \frac{4}{n^1}, \text{ при } n \rightarrow \infty.$$

Аналогичные вычисления проведены для других подмногообразий аффинных пространств  $\mathbb{A}_k^N$  и представлены в таблице 1. Можно сформулировать гипотезу:

$N$	Многообразие	Значение функции	$n \rightarrow \infty$
2	прямая $x = 0$	$\frac{2}{n+1}$	$\frac{2}{n}$
2	крест $xy = 0$	$\frac{2(2n-1)}{n(n+1)}$	$\frac{4}{n}$
3	плоскость $x = 0$	$\frac{3}{n+1}$	$\frac{3}{n}$
3	прямая $x = 0, y = 0$	$\frac{6}{n^2+3n+2}$	$\frac{6}{n^2}$

Таблица 1: Значения обобщенной поточечной функции Гильберта структурных пучков некоторых подмногообразий аффинных пространств

**Гипотеза 1.** Пусть  $M \subset \mathbb{A}_k^N$  – многообразие,  $p \in M$  – точка. Тогда  $\pi_p^{\mathcal{O}_M}(n) \sim \frac{\alpha}{n^\beta}$  при  $n \rightarrow \infty$ , где  $\beta$  – коразмерность носителя пучка  $\mathcal{O}_M$ .

Нами были вычислены значения обобщенной поточечной функции Гильберта в ряде ключевых ситуаций, на основе которых была сформулирована гипотеза 1. Если эта гипотеза будет доказана, то по значениям функции Гильберта можно получить значение коразмерности носителя когерентного пучка. Так же было сформулировано добавление к уже известному результату из [3].

Работа выполнена в рамках реализации программы развития регионального научно-образовательного математического центра (ЯрГУ им. П.Г. Демидова) при финансовой поддержке Министерства науки и высшего образования РФ (Дополнительное соглашение № 075-02-2020-1514/1 к Соглашению о предоставлении из федерального бюджета субсидии № 075-02-2020-1514).

### Литература

1. Айзенбад, Д. Коммутативная алгебра с прицелом на алгебраическую геометрию, МЦНМО, Москва (2017).
2. Атья, М. Введение в коммутативную алгебру, Мир, Москва (1972).
3. Тимофеева Н.В., “Инфинитезимальный критерий плоскости для проективного морфизма схем”, Алгебра и анализ, **26**, No. 1, 185–195 (2014).
4. Хартсхорн Р., Алгебраическая геометрия, Мир, Москва (1981).

## СЕКЦИЯ ДИФФЕРЕНЦИАЛЬНЫЕ УРАВНЕНИЯ

### Об одной математической модели искусственной генной сети

**Е. И. Володина (ЯрГУ им. П.Г. Демидова, Ярославль)**

*Научный руководитель: д.ф.-м.н., профессор Колесов А. Ю.*

Рассматривается система обыкновенных дифференциальных уравнений, представляющая собой упрощенную математическую модель некоторой генной сети [1]

$$\begin{cases} u'_1 = r[1 - u_1 - au_3]u_1 + \alpha \\ u'_2 = r[1 - u_2 - au_1]u_2 + \alpha \\ u'_3 = r[1 - u_3 - au_2]u_3 + \alpha \end{cases}, \quad a = \text{const} > 1, \quad r \gg 1, \quad \alpha > 0 \quad (1)$$

с положительным состоянием равновесия

$$u_* = \frac{r + r\sqrt{1 + 4\alpha(1 + a)/r}}{2r(1 + a)}.$$

Матрица Якоби данной системы имеет три собственных значения

$$\lambda_1 = r - 2ru_* - 2aru_*, \quad \lambda_{2,3} = r - 2ru_* - \frac{aru_*}{2} \pm iaru_* \frac{\sqrt{3}}{2}.$$

При

$$u_* = \frac{2}{4 + a}$$

$\lambda_{2,3}$  будут чисто мнимыми. С помощью найденного  $u_*$  находим  $\alpha$ :

$$\alpha_* = \frac{2r(a - 2)}{(4 + a)^2}.$$

Посмотрим, что происходит с системой при изменении  $\alpha$  на малый параметр  $\varepsilon$ . Систему (1) можно переписать в виде

$$x' = \mathcal{F}(x, \varepsilon), \quad (2)$$



где  $x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$ ,  $\alpha = \alpha_* - \varepsilon$  ( $0 < \varepsilon \ll 1$ ).

Теперь необходимо перейти к нормальной форме [2]. Для этого перепишем систему (2) в виде

$$x' = (A_0 + \varepsilon A_1)x + F_2(x, x), \quad (3)$$

где  $F_2(x, x)$  — квадратичная форма, а матрицы  $A_0$  и  $A_1$  находятся как первая и вторая производные  $\mathcal{F}$  по  $\varepsilon$  в точке  $(0, 0)$ . Полученная система удовлетворяет стандартным бифуркационным ограничениям. А именно,

**во-первых**, матрица  $A_0$  имеет на мнимой оси пару простых собственных значений, а оставшаяся ее точка спектра лежит в комплексной полуплоскости  $\{\lambda : \operatorname{Re} \lambda < 0\}$

**во-вторых**, тейлоровское разложение в нуле вектор-функции  $\mathcal{F}$  имеет вид

$$\mathcal{F}(x) = (A_0 + \varepsilon A_1)x + F_2(x, x).$$

При сделанных допущениях автоколебания системы (3), бифурцирующие из ее нулевого состояния равновесия, будем искать в виде ряда по степеням  $\sqrt{\varepsilon}$ :

$$x = \sqrt{\varepsilon}x_0(t, \tau) + \varepsilon x_1(t, \tau) + \varepsilon^{\frac{3}{2}}x_2(t, \tau) + \dots, \quad \tau = \varepsilon t, \quad (4)$$

где

$$x_0 = \xi_1 v e^{i\omega t} + \overline{\xi_1} \bar{v} e^{-i\omega t}. \quad (5)$$

Здесь  $v = \begin{pmatrix} 1 \\ \frac{-1+i\sqrt{3}}{2} \\ \frac{-1-i\sqrt{3}}{2} \end{pmatrix}$  — собственный вектор матрицы  $A_0$ , отвечающий ее собственному значению  $i\omega$  и нормированный условиями  $(v, b) = 1$ ,  $(\bar{v}, b) = 0$ , где  $\xi = \xi(\tau)$  — пока произвольные комплексные амплитуды, а  $b$  находится из уравнения  $A_0^* b = -i\omega b$ .

Подставляя соотношения (4), (5) в уравнение (3), приравнявая слева и справа коэффициенты при  $\varepsilon$ , для отыскания  $x_1$  получаем линейную неоднородную систему

$$\frac{\partial x_1}{\partial t} - A_0 x_1 = g_1(t, \tau), \quad (6)$$

где  $g_1$  является суммой нулевых и вторых гармоник  $\omega t$ . Из уравнения (6) функция  $x_1$  однозначно определяется как

$$x_1 = x_{11} \xi^2 e^{2i\omega t} + x_{12} |\xi|^2 + x_{11} \bar{\xi}^2 e^{-2i\omega t}.$$

Приравниваем затем коэффициенты при степени  $\varepsilon^{\frac{3}{2}}$ . В результате для  $x_2$  приходим к аналогичному (6) уравнению:

$$\frac{\partial x_2}{\partial t} = A_0 x_2 + K_0 e^{3i\omega t} + K_1 e^{i\omega t} + \bar{K}_1 e^{-i\omega t} + \bar{K}_0 e^{-3i\omega t}.$$

Данное уравнение разрешимо при  $(g_2, b) = 0$ . Это условие приводит, в свою очередь, к системе вида

$$\frac{\partial \xi}{\partial \tau} = [(A_1 v, b) + d|\xi|^2] \xi, \quad (7)$$

для нахождения неизвестных амплитуд  $\xi$ .

Для того, чтобы сформулировать строгий результат, перейдем от (7) к вспомогательной системе для  $\eta = |\xi|^2$ :

$$\frac{\partial \eta}{\partial \tau} = [\beta + \beta_1 \eta] \eta, \quad (8)$$

где

$$\beta = 2 \left( \frac{4}{3} + \frac{8}{3a} + \frac{a}{6} \right),$$

$$\beta_1 = -\frac{(-16 + 4a + 6a^2 + a^3)r}{2a}.$$

Тогда при всех  $a \geq 2$  коэффициент  $\beta_1$  будет отрицательным. В этом случае данное уравнение имеет единственное устойчивое положительное состояние равновесия

$$\eta = -\frac{\beta}{\beta_1} = \frac{2(4+a)}{3r(-4+a(2+a))}.$$

В исходной системе уравнений этому состоянию равновесия соответствует экспоненциально орбитально устойчивый цикл.

### Литература

1. Глызин С. Д., Колесов А. Ю., Розов Н. Х. “Квазиустойчивые структуру в кольцевых генных сетях,” Журнал вычислительной математики и математической физики, **58**, No. 5, 682–704 (2018).
2. Глызин С. Д., Колесов А. Ю. Локальные методы анализа динамических систем, ООО "Ремдер Ярославль (2006).

### Об экспериментальной проверке в задаче о колебаниях мембраны

*Е. Ю. Герасимов (ЯрГУ им. П.Г. Демидова,  
Ярославль)*

*Научный руководитель: к.ф.-м.н. Очиров А. А.*

Существует большое количество технических каналов утечки информации. Злоумышленник может получить информацию с помощью устройств, перехватывающих колебания мембран и пластин, возникающие вследствие давления атмосферных акустических волн на плоские поверхности [1]. Для оценки эффективности защиты речевой информации от утечки по техническим акустическим и виброакустическим каналам важно понимать, как отвечает на речевое воздействие поверхность. Для исследования этого процесса было рассмотрено колебание окна под воздействием синусоидального акустического воздействия. Задачей эксперимента было практическое исследование теории колебания данной мембраны с закрепленными краями при внешнем воздействии на нее чистым тоном. Для этого в исследуемом окне было условно выделено 200 точек, в которых закреплялся пьезоэлектрический акселерометр, измеряющий виброускорение в данных областях. На расстоянии в 1 метр от центра была поставлена специализированная колонка с известными показателями из набора «Смарт», которая излучала чистый тон. Используя программное обеспечение «Смарт – ПО», сняли значения с интервалом 10 секунд при частоте дискретизации в 50 кГц. Полученные значения сохранили в файлах формата .txt в виде чисел с плавающей запятой. Для каждой отдельной точки был получен свой файл значений. Приступив к обработке данных файлов, выполнили преобразование Фурье. На полученных результатах обнаружили отсутствие побочных гармоник и

выделили нужную нам частоту чистого тона. Взяв среднее значение амплитуды во всех точках, получили таблицу значений для каждой области. По данным из этой таблицы был построен график зависимости амплитуды от координат (Рис. 1).

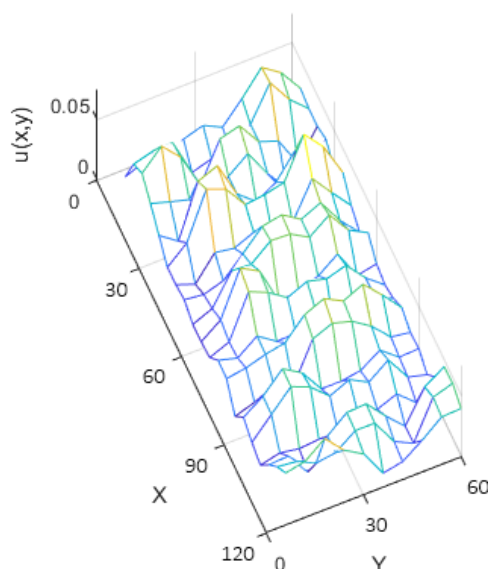


Рис. 1: Зависимость амплитуды от координат.

Таким образом экспериментальным путем были определены характерные особенности колебания рассматриваемого окна.

### Литература

1. Каторин Ю. Ф., Разумовский А. В., Спивак А. И., Защита информации техническими средствами. Учебное пособие, НИУ ИТМО, Санкт-Петербург (2012).

## Управление полетом квадрокоптера

**И. А. Головкин (ЯГТУ, Ярославль)**

*Научный руководитель: к.т.н., доцент Марьясин О. Ю.*

Рассмотрим движение квадрокоптера относительно стартовой (неподвижной) и связанной (подвижной) системы координат. С учетом симметрии аппарата и считая, что центр масс

расположен в начале координат связанной системы, уравнения динамики вращательного движения можно записать в виде [1]:

$$\begin{cases} \dot{w}_x = \frac{(I_y - I_z)}{I_x} w_y w_z + M_{R_x}/I_x, \\ \dot{w}_y = \frac{(I_z - I_x)}{I_y} w_x w_z + M_{R_y}/I_y, \\ \dot{w}_z = \frac{(I_x - I_y)}{I_z} w_x w_y + M_{R_z}/I_z, \end{cases} \quad (1)$$

где  $w_x, w_y, w_z$  — проекции вектора угловой скорости,  $I_x, I_y, I_z$  — моменты инерции,  $M_{R_x}, M_{R_y}, M_{R_z}$  — проекции результирующего момента сил.

Если пренебречь инерционностью винтов, то проекции результирующего момента можно выразить следующим образом:

$$\begin{cases} M_{R_x} = (F_3 - F_1)l + I_m w_x (-\omega_1 + \omega_2 - \omega_3 + \omega_4), \\ M_{R_y} = (F_2 - F_4)l + I_m w_y (\omega_1 - \omega_2 + \omega_3 - \omega_4), \\ M_{R_z} = -F_1 + F_2 - F_3 + F_4, \end{cases} \quad (2)$$

где  $F_i$  — сила тяги  $i$ -го винта,  $F_i = k_F \omega_i^2$ ,  $\omega_i$  — угловая скорость вращения  $i$ -го винта,  $l$  — расстояние от центра масс до оси винта,  $I_m$  — момент инерции ротора.

Изменения углов Эйлера определяются из кинематических уравнений Эйлера

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \operatorname{tg}(\theta) \sin(\phi) & \operatorname{tg}(\theta) \cos(\phi) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\psi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{bmatrix} \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix}, \quad (3)$$

где  $\phi$  — угол крена,  $\theta$  — угол тангажа,  $\psi$  — угол рыскания.

Динамика движения центра масс квадрокоптера в стартовой системе отсчета описывается уравнениями

$$\begin{cases} \ddot{x} = \frac{1}{m} (F(\sin(\phi) \sin(\psi) - \cos(\phi) \cos(\psi) \sin(\theta)) - f_x), \\ \ddot{y} = \frac{1}{m} (F(\sin(\phi) \cos(\psi) + \cos(\phi) \sin(\psi) \sin(\theta)) - f_y), \\ \ddot{z} = \frac{1}{m} (F(\cos(\phi) \cos(\theta)) - mg - f_z), \end{cases} \quad (4)$$

где  $m$  — масса квадрокоптера,  $F = \sum_{i=1}^4 F_i$ ,  $f = [-f_x, -f_y, -f_z]$  — сила сопротивления воздуха,  $g$  — ускорение свободного падения.

Математическая модель квадрокоптера (1)-(4) была реализована в системе MATLAB/Simulink. Для стабилизации режимов квадрокоптера (углов крена, тангажа, рыскания и высоты полета) использовались стандартные ПИД-регуляторы. Изменения параметров состояния квадрокоптера, полученные в результате моделирования показаны на рис. 1.

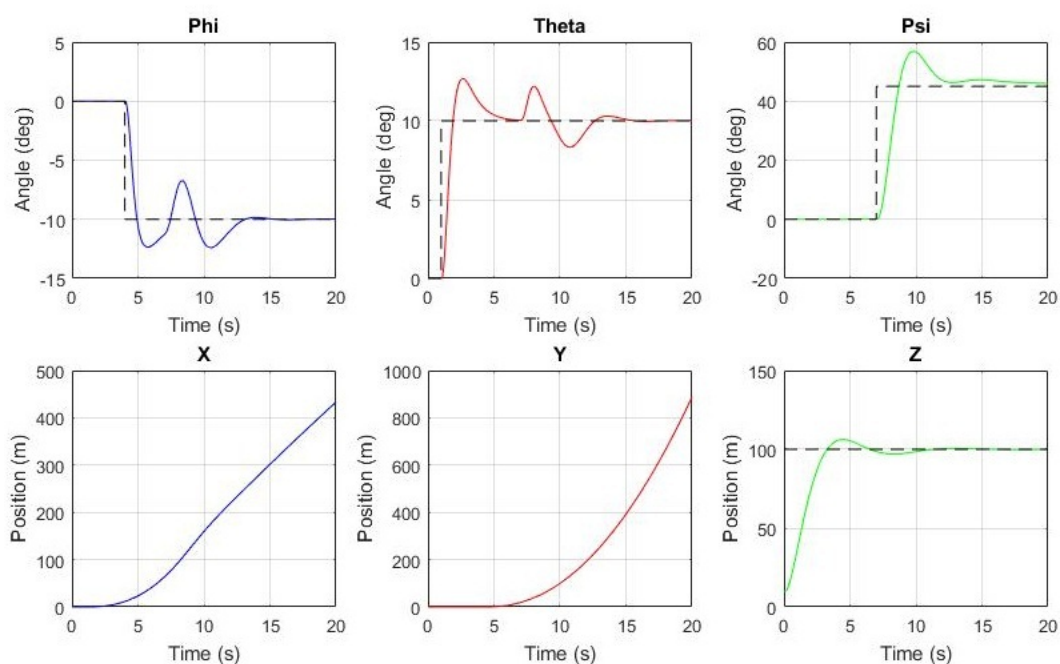


Рис. 1: Результаты моделирования

Недостатком модели квадрокоптера (1)-(4), основанной на использовании углов Эйлера, является наличие сингулярностей при достижении углов крена и тангажа значений  $\pm 180^\circ$ . Для преодоления указанного недостатка используют отключение регуляторов при приближении параметров к опасным значениям. Другим вариантом является применение кватернионной модели квадрокоптера.

## Литература

1. Огольцов И. И., Рожнин Н. Б., Шеваль В. В. “Разработка математической модели пространственного полета квадрокоптера”, Труды МАИ, **6**, No. 83, 1–26 (2015).

**К вопросу о влиянии младших резонансов  
в задаче о синхронизации колебаний**  
*Е. С. Канцидал (ЯрГУ им. П.Г. Демидова, Ярославль)*  
*Научный руководитель: к.ф.-м.н., доцент Куликов Д. А.*

Изучается вопрос о взаимодействии двух осцилляторов Ван дер Поля-Дуффинга при наличии прямой связи [1,2,3]

$$\begin{cases} \ddot{x}_1 - 2\varepsilon\dot{x}_1 + (1 + \delta_1\varepsilon)x_1 + c(1 + \delta_3\varepsilon)x_2 = -\dot{x}_1x_1^2 + ax_1^3, \\ \ddot{x}_2 - 2\varepsilon\dot{x}_2 + c(1 + \delta_4\varepsilon)x_1 + (1 + \delta_2\varepsilon)x_2 = -\dot{x}_2x_2^2 + ax_2^3. \end{cases} \quad (1)$$

Здесь  $a, \delta_1, \delta_2, \delta_3, \delta_4 \in \mathbb{R}$ ,  $\varepsilon$  — малый неотрицательный параметр,  $c \in (0, 1)$ .

Отметим, что если  $\varepsilon = 0$ , то у линеаризованной на нулевом состоянии равновесия системы (1) спектр устойчивости содержит собственные числа  $\lambda_{1,3} = \pm i\sigma_1$ ,  $\lambda_{2,4} = \pm i\sigma_2$ ,  $\sigma_1 = \sqrt{1+c}$ ,  $\sigma_2 = \sqrt{1-c}$ . При всех  $c \in (0, 1)$  система (1) имеет два орбитально асимптотически устойчивых предельных цикла, а также двумерный инвариантный неустойчивый тор.

Отметим, что при  $c = \frac{4}{5}$  в системе дифференциальных уравнений (1) реализуется резонанс собственных частот 1:3.

Особенностью данной задачи является то обстоятельство, что наличие резонанса не влияет на динамику решений, принадлежащих окрестности нулевого состояния равновесия данной системы. Аналогичный вывод справедлив при реализации резонанса 1:2, который реализуется, если  $c = \frac{3}{5}$ .

Анализ задачи проводился с помощью метода интегральных многообразий и теории нормальных форм Пуанкаре.

### Литература

1. Блэкьер О., Анализ нелинейных систем, Мир, Москва (1969).
2. Пиковский А., Розенблюм М., Куртс Ю., Синхронизация: Фундаментальные нелинейные явления, Техносфера, Германия (2003).
3. Куликов Д.А., “Автомодельные циклы и их локальные бифуркации в задаче о двух слабосвязанных осцилляторах,” Прикладная математика и механика, **74**, No. 4, 543-559 (2010).

## Преобразования Дарбу и уравнение тетраэдров

**В. А. Колесов (ЯрГУ им. П.Г. Демидова, Ярославль)**

*Научный руководитель: PhD Константин-Ризос С.*

Рассмотрим отображение  $T : X^3 \mapsto X^3$

$$T : (x, y, z) \mapsto (u(x, y, z), v(x, y, z), w(x, y, z)).$$

Такое отображение называется отображением тетраэдров, если оно удовлетворяет уравнению тетраэдров

$$T^{123} \circ T^{145} \circ T^{246} \circ T^{356} = T^{356} \circ T^{246} \circ T^{145} \circ T^{123}, \quad (1)$$

где  $T^{ijk}(x_1, x_2, x_3, x_4, x_5, x_6)$  действует на  $x_i, x_j, x_k$  как  $T(x_i, x_j, x_k)$ , а на остальные аргументы действует тривиально. Например,

$$T^{145}(x, y, z, r, s, t) = (u(x, r, s), y, z, v(x, r, s), w(x, r, s), t).$$

Отображение тетраэдров связано с задачей матричной трифакторизации [1] следующим образом: пусть  $L(x, a; \lambda)$  — матрица, зависящая от переменной  $x \in X$ , параметра  $a \in \mathbb{C}$ , и спектрального параметра  $\lambda \in \mathbb{C}$ , вида

$$L(x, a; \lambda) = \begin{pmatrix} A(x, a; \lambda) & B(x, a; \lambda) \\ C(x, a; \lambda) & D(x, a; \lambda) \end{pmatrix}.$$



Определим следующие матрицы

$$\begin{aligned} L_{12} &= \begin{pmatrix} A(x, a; \lambda) & B(x, a; \lambda) & 0 \\ C(x, a; \lambda) & D(x, a; \lambda) & 0 \\ 0 & 0 & 1 \end{pmatrix}, \\ L_{13} &= \begin{pmatrix} A(x, a; \lambda) & 0 & B(x, a; \lambda) \\ 0 & 1 & 0 \\ C(x, a; \lambda) & 0 & D(x, a; \lambda) \end{pmatrix}, \\ L_{23} &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & A(x, a; \lambda) & B(x, a; \lambda) \\ 0 & C(x, a; \lambda) & D(x, a; \lambda) \end{pmatrix}. \end{aligned}$$

Следующее уравнение определяет задачу матричной трифакторизации

$$\begin{aligned} L_{12}(u, a; \lambda) L_{13}(v, b; \lambda) L_{23}(w, c; \lambda) = \\ = L_{23}(z, c; \lambda) L_{13}(y, b; \lambda) L_{12}(x, a; \lambda). \end{aligned} \quad (2)$$

Если уравнение (2) имеет единственное решение для  $u$ ,  $v$  и  $w$ , т.е. определяет единственно отображение  $T : (x, y, z) \mapsto (u(x, y, z), v(x, y, z), w(x, y, z))$ , то  $T$  — это отображение тетраэдров.

Дальше мы объясним, как можно строить уравнения тетраэдров, используя преобразования Дарбу. Для этого мы даем определения интегрируемого уравнения в частных производных и преобразования Дарбу [2].

Уравнение эволюционного типа  $u_t = f(u, u_x, u_{xx}, \dots)$  называют интегрируемым, если для него существует пара Лакса. Пусть  $L, A$  — пара линейных операторов следующего вида:

$$\begin{aligned} L &= D_x + U(u, u_x, u_{xx}, \dots), \\ A &= D_t + V(u, u_x, u_{xx}, \dots), \end{aligned}$$

$\psi$  — вспомогательная функция. Запишем систему линейных уравнений для  $\psi$

$$\begin{cases} L\psi = 0, \\ A\psi = 0, \end{cases} \Leftrightarrow \begin{cases} \psi_x = -U\psi, \\ \psi_t = -V\psi, \end{cases} \quad \text{— переопределенная система.}$$

Чтобы система могла иметь решение, нужно чтобы выполнялось условие совместности системы:  $\psi_{xt} = \psi_{tx}$ .

$$\begin{cases} \psi_{xt} = -U_t\psi - U\psi_t, \\ \psi_{tx} = -V_x\psi - V\psi_x, \end{cases} \Leftrightarrow \begin{cases} \psi_{xt} = -U_t\psi + UV\psi, \\ \psi_{tx} = -V_x\psi + VU\psi, \end{cases}$$

$$-U_t\psi + UV\psi = -V_x\psi + VU\psi \Leftrightarrow U_t - V_x - UV + VU = 0.$$

Если полученное условие совместности системы эквивалентно уравнению  $u_t = f(u, u_x, u_{xx}, \dots)$ , то операторы  $L$  и  $A$  называются парой Лакса для данного уравнения. Например, для нелинейного уравнения Шрёдингера  $p_t = p_{xx} + 4p^2q$ ,  $q_t = -q_{xx} - 4pq^2$ , существует следующая пара Лакса:

$$L = D_x + \lambda \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} + \begin{pmatrix} 0 & 2p \\ 2q & 0 \end{pmatrix},$$

$$A = D_t + \lambda^2 \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} + \lambda \begin{pmatrix} 0 & 2p \\ 2q & 0 \end{pmatrix} + \begin{pmatrix} -2pq & p_x \\ -q_x & 2pq \end{pmatrix}.$$

Для оператора Лакса  $L = D_x + U(p, q; \lambda)$  преобразование Дарбу определяется невырожденной матрицей  $M$  (матрица Дарбу), такой что

$$L \rightarrow MLM^{-1} = D_x + U(\tilde{p}, \tilde{q}; \lambda) = \tilde{L},$$

где  $\tilde{p}, \tilde{q}$  — новые потенциальные функции. Матрицу Дарбу можно получить из определения  $MLM^{-1} = \tilde{L} \Rightarrow ML = \tilde{L}M$ .

Для оператора  $L$  пары Лакса для нелинейного уравнения Шрёдингера матрица Дарбу имеет следующий вид:

$$M = \begin{pmatrix} f + \lambda & p \\ \tilde{q} & 1 \end{pmatrix}, \quad \text{где} \quad \begin{cases} f_x = 2(pq - \tilde{p}\tilde{q}) \\ p_x = 2(pf - \tilde{p}) \\ \tilde{q}_x = 2(q - f\tilde{q}) \end{cases}.$$

Используя матрицы Дарбу, можно строить новые решения для уравнения тетраэдров. Для этого можно подставить матрицу Дарбу в задачу матричной трифакторизации. Если выразить переменные  $u, v, w$  через  $x, y, z$  (возможно, наложив некоторые ограничения), в результате получим отображение  $(x, y, z) \mapsto (u, v, w)$ , которое удовлетворяет уравнению тетраэдров.

- Преобразование Дарбу тесно связано с преобразованием Беклунда, которое позволяет строить решения нелинейных систем, имея даже самые тривиальные.
- Для отображения тетраэдров можно определить дифференциал, который также является отображением тетраэдров, что позволяет строить целые последовательности отображений, начиная с одного [3].
- Как отображения Янга-Бакстера связаны с уравнениями на Квад-графах, так отображения тетраэдров связаны с разностными уравнениями на 3d-решетке.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта No. 20-71-10110.

### Литература

1. S. Konstantinou-Rizos, “Nonlinear Schrödinger type tetrahedron maps”, Nucl. Phys. B **960** 115207 (2020).
2. S. Konstantinou-Rizos and A.V. Mikhailov. “Darboux transformations, finite reduction groups and related Yang–Baxter maps” // Journal of Physics A. 2013. V. 46, №425201.
3. S. Igonin, V. Kolesov, S. Konstantinou-Rizos, M.M. Preobrazhenskaia. “Tetrahedron maps, Yang–Baxter maps, and partial linearisations”, 2021, arXiv:2106.09130

### Автоколебания в одной задаче из радиофизики

**А. А. Мкртчян (ЯрГУ им. П.Г. Демидова, Ярославль)**

*Научный руководитель: к.ф.-м.н., доцент Куликов Д. А.*

В работе рассматривается система дифференциальных уравнений

$$\begin{cases} \ddot{x}_1 - 2\varepsilon\gamma\dot{x}_1 + x_1 = -dx_1^2\dot{x}_1 + 2\varepsilon\alpha\dot{x}_2 + cx_1^3, \\ \ddot{x}_2 - 2\varepsilon\gamma\dot{x}_2 + x_2 = -dx_2^2\dot{x}_2 + 2\varepsilon\alpha\dot{x}_1 + cx_2^3, \end{cases} \quad (1)$$

где  $\gamma > 0$ ,  $\alpha \neq 0$ ,  $\alpha \in R$ ,  $x_1 = x_1(t)$ ,  $x_2 = x_2(t)$ ,  $d$  — любое.

Считаем, что  $\varepsilon$  — малый неотрицательный (положительный) параметр. Отметим, что при  $\alpha = 0$  получим два отдельных автономных уравнения системы (1). При этом каждое из уравнений — осциллятор Ван дер Поля, и в каждом таком уравнении можно найти устойчивые периодические решения, если, конечно,  $\gamma > 0$ ,  $\varepsilon > 0$ ,  $d$  — любое.

В работе будет рассмотрен вопрос о существовании и устойчивости периодических решений у системы дифференциальных уравнений (1), если  $\varepsilon_0$  достаточно мало.

Вопрос о существовании периодических решений был сведен к анализу вспомогательной системы обыкновенных дифференциальных уравнений — нормальной формы [1,2].

$$\begin{cases} z_1' = \gamma z_1 + \alpha z_2 - \left(\frac{d}{2} - \frac{3c}{2}i\right) z_1^2 \overline{z_1}, \\ z_2' = \gamma z_2 + \alpha z_1 - \left(\frac{d}{2} - \frac{3c}{2}i\right) z_2^2 \overline{z_2}, \end{cases} \quad (2)$$

где  $z_1 = z_1(s)$ ,  $z_2 = z_2(s)$ ,  $s = \varepsilon t$  — «медленное» время.

Для анализа нормальной формы (2) выполним замены:

$$z_1 = \rho_1 e^{i\varphi_1}, z_2 = \rho_2 e^{i\varphi_2}, \psi = \varphi_2 - \varphi_1.$$

В результате вопрос о периодических решениях нормальной формы (2) удалось свести к анализу состояний равновесия системы дифференциальных уравнений

$$\begin{cases} \dot{\rho}_1 = \gamma \rho_1 + \alpha \rho_2 \cos \psi - \frac{d}{2} \rho_1^3, \\ \dot{\rho}_2 = \gamma \rho_2 + \alpha \rho_1 \cos \psi - \frac{d}{2} \rho_2^3, \\ \dot{\psi} = -\alpha \left[ \frac{\rho_2^2 + \rho_1^2}{\rho_1 \rho_2} \right] \sin \psi + \frac{3c}{2} [\rho_2^2 - \rho_1^2]. \end{cases} \quad (3)$$

Система дифференциальных уравнений (3) имеет следующие состояния равновесия:

$$A : \rho_1 = \rho_2 = \sqrt{\frac{2(\gamma + \alpha)}{d}}, \psi = 0;$$

$$B : \rho_1 = \rho_2 = \sqrt{\frac{2(\gamma - \alpha)}{d}}, \psi = \pi.$$

При этом состояние равновесия  $A$  существует при  $\frac{\gamma + \alpha}{d} > 0$  и устойчиво при

$$\alpha + \gamma > 0, \alpha > 0.$$

Состояние равновесия  $B$  существует при  $\frac{\gamma - \alpha}{d} > 0$  и устойчиво при

$$-\alpha + \gamma > 0, \alpha < 0.$$

Также найдены иные состояния равновесия:

$$C(\sqrt{1,5}; \sqrt{1,5}; 1, 4),$$

которое неустойчиво всегда,

$$D(\sqrt{0,75}; \sqrt{0,75}; 1, 4),$$

которое неустойчиво всегда,

$$E(\sqrt{0,375}; \sqrt{0,375}; -1, 4),$$

которое также всегда неустойчиво,

$$F(\sqrt{0,75}; \sqrt{0,75}; -1, 4),$$

которое устойчиво при  $\gamma = 1, \alpha = \frac{1}{4}, d = 2, c = -2$ .

Состояниям равновесия системы (3) соответствуют периодические решения системы (1) с наследованием устойчивости.

### Литература

1. Куликов Д.А., “Автомодельные циклы и их локальные бифуркации в задаче о двух слабосвязанных осцилляторах,” Прикладная математика и механика, **74**, No. 4, 543-559 (2010).
2. Куликов Д.А., “Автомодельные периодические решения и бифуркации от них в задаче о взаимодействии двух слабосвязанных осцилляторов,” Известия высших учебных заведений. Прикладная нелинейная динамика, **14**, No. 5, 120-132 (2006).

## Уравнение Янга–Бакстера и приложения

**А. А. Никитина (ЯрГУ им. П.Г. Демидова,  
Ярославль)**

*Научный руководитель: PhD Константину-Ризос С.*

Уравнение Янга–Бакстера является одним из фундаментальных уравнений в математической физике. В данной работе покажем, как можно построить решение этого уравнения, используя преобразование Дарбу.

Для начала определим отображение Янга–Бакстера  $Y$  на некотором множестве  $A$ :

$$Y : A \times A \rightarrow A \times A, \quad (x, y) \rightarrow (u(x, y), v(x, y)) \quad (1)$$

Уравнение Янга–Бакстера определяется следующим образом:

$$Y^{12} \circ Y^{13} \circ Y^{23} = Y^{23} \circ Y^{13} \circ Y^{12}, \quad (2)$$

где  $Y^{ij} : A \times A \times A \rightarrow A \times A \times A, i \neq j, i < j$ . Элемент  $Y^{ij}$  действует на  $ij$  переменные набора  $(x, y, z)$ , как отображение (1), а на третью переменную тождественно. В частности,

$$\begin{aligned} Y^{12}(x, y, z) &= (u(x, y), v(x, y), z) \\ Y^{13}(x, y, z) &= (u(x, z), y, v(x, z)) \\ Y^{23}(x, y, z) &= (x, u(y, z), v(y, z)) \end{aligned}$$

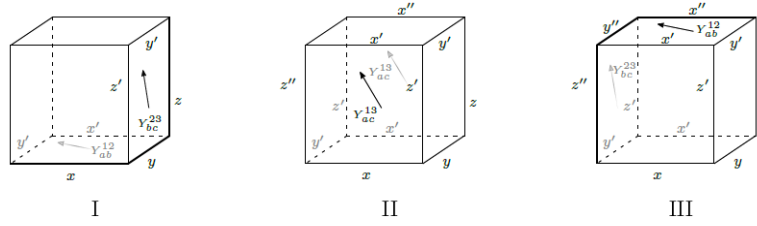
Если присвоить переменным  $x, y$  и  $z$  параметры  $a, b$  и  $c \in \mathbb{C}$ , соответственно, тогда можно рассматривать параметрическое отображение Янга–Бакстера следующего вида:

$$Y_{ab} : (A \times \mathbb{C}) \times (A \times \mathbb{C}) \rightarrow (A \times \mathbb{C}) \times (A \times \mathbb{C}), \quad (3)$$

$$\begin{aligned} ((x, a), (y, b)) &\rightarrow ((u((x, a), (y, b)); a), (v((x, a), (y, b)); b)) = \\ &= ((u, a), (v, b)). \end{aligned}$$

Параметрическое уравнение Янга–Бакстера:

$$Y_{ab}^{12} \circ Y_{ac}^{13} \circ Y_{bc}^{23} = Y_{bc}^{23} \circ Y_{ac}^{13} \circ Y_{ab}^{12}, \quad (4)$$



первому аргументу соответствует параметр  $a$ , второму  $b$ , третьему  $c$ .

Уравнение (4) можно представить схематически:

Дальше, мы рассматриваем соотношение уравнения Янга–Бакстера с задачами матричной рефакторизации.

**Определение 1.** Пусть задана квадратная матрица  $L = L(x; a, \lambda) = L_a(x)$ , где  $x$  — переменная,  $a$  — параметр,  $\lambda$  — спектральный параметр. Матрица  $L(x; a, \lambda)$  называется матрицей Лакса [1] для отображения  $(x, y) \xrightarrow{Y_{a,b}} (u(x, y; a, b), v(x, y; a, b))$ , если удовлетворяет задаче “матричной рефакторизации”:

$$L(u; a, \lambda)L(v; b, \lambda) = L(y; b, \lambda)L(x; a, \lambda). \quad (5)$$

Например, для отображения Адлера [2]:

$$Y_{ab}^{12} : (x, y) \rightarrow \left( y - \frac{a-b}{x+y}, x + \frac{a-b}{x+y} \right), \quad (6)$$

матрица Лакса, выглядит так:  $L(x; a, \lambda) = \begin{pmatrix} x & 1 \\ x^2 + a - \lambda & x \end{pmatrix}$ .

Если эту матрицу подставить в задачу (5) и решить её относительно  $u$  и  $v$ , то получим отображение (6).

Далее, определим пару Лакса и преобразование Дарбу.

Пусть операторы  $L(u, \lambda) = D_x + U(u; \lambda)$  и  $A(u, \lambda) = D_t + V(u; \lambda)$ . Мы рассматриваем линейную систему 
$$\begin{cases} \Psi_x = L\Psi, \\ \Psi_t = A\Psi, \end{cases}$$

где  $\Psi$  — вспомогательная матрица. Это переопределенная система, и условие совместимости этой системы:

$$L_t - A_x + (LA - AL) = 0. \quad (7)$$

**Определение 2.** Если можно найти операторы  $L$  и  $A$ , для которых нелинейное уравнение  $u_t = F(u, u_x, u_{xx}, \dots; x, t)$  эквивалентно уравнению (7), то это уравнение называется интегрируемым. Пара операторов  $(L, A)$  называется парой Лакса нелинейного уравнения, а условие (7) — это его представление Лакса.

**Определение 3.** Пусть  $L = D_x + U(u; \lambda)$ . Преобразование Дарбу — это преобразование вида  $MLM^{-1}$ , которое оставляет оператор  $L$  ковариантным. Т.е.  $MLM^{-1} = \tilde{L}$ , где  $\tilde{L} = D_x + U(\tilde{u}; \lambda)$ .

Из  $MLM^{-1} = \tilde{L}$  следует  $ML = \tilde{L}M$ , которое является задачей матричной рефакторизации, как (5). Это наводит на мысль рассматривать задачу матричной рефакторизации для конкретных матриц Дарбу  $M$ , чтобы построить отображения Янга–Бакстера.

Для построения отображения Янга–Бакстера возьмём оператор NLS, связанный с нелинейным уравнением Шрёдингера. Этот оператор имеет следующий вид:

$$L(p, q; \lambda) = D_x + U(u, \lambda) = D_x + \begin{pmatrix} \lambda & 2p \\ 2q & -\lambda \end{pmatrix}, \quad (8)$$

Одна матрица Дарбу для оператора NLS:

$$M(x_1, x_2; a) = \lambda \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} a + x_1 x_2 & x_1 \\ x_2 & 1 \end{pmatrix} \quad (9)$$

Подставим матрицу (9) в задачу матричной рефакторизации (5):

$$M(\vec{u}; a) M(\vec{v}; b) = M(\vec{y}; b) M(\vec{x}; a), \quad (10)$$

где  $\vec{u} = (u_1, u_2)$ ,  $\vec{v} = (v_1, v_2)$ ,  $\vec{x} = (x_1, x_2)$ ,  $\vec{y} = (y_1, y_2)$ , получится отображение Янга–Бакстера [3], [4]  $\vec{x} \rightarrow \vec{u}(\vec{x}, \vec{y})$ ,  $\vec{y} \rightarrow \vec{v}(\vec{x}, \vec{y})$ :

$$(\vec{x}, \vec{y}) \xrightarrow{Y_{a,b}} \left( y_1 - \frac{a-b}{x_1 y_2 + 1} x_1, y_2; x_1, x_2 + \frac{a-b}{x_1 y_2 + 1} y_2 \right) \quad (11)$$

Легко проверить, что найденное отображение (11) удовлетворяет уравнению (4).



Таким образом, в данной работе мы рассмотрели отображение и уравнение Янга–Бакстера и их параметрические версии, а также показали способ нахождения отображения Янга–Бакстера с помощью преобразования Дарбу на основе оператора NLS.

### Литература

1. Suris Y. B. and Veselov A. P., “Lax matrices for Yang–Baxter maps”, *Journal of Nonlinear Mathematical Physics*, **10**, 223–230 (2003).
2. Adler V. E., “Recuttings of polygons”, *Functional Analysis and Its Applications*, **27**, No. 2, 141–143 (1993).
3. Adler V. E. and Yamilov R. I., “Explicit auto-transformations of integrable chains”, *Journal of Physics A: Mathematical and General*, **27**, No. 2, 477 (1994).
4. Konstantinou-Rizos S. and Mikhailov A. V., “Darboux transformations, finite reduction groups and related Yang–Baxter maps”, *Journal of Physics A*, **46**, No. 425201 (2013).

## Компьютерная визуализация фазовых портретов динамических систем на плоскости

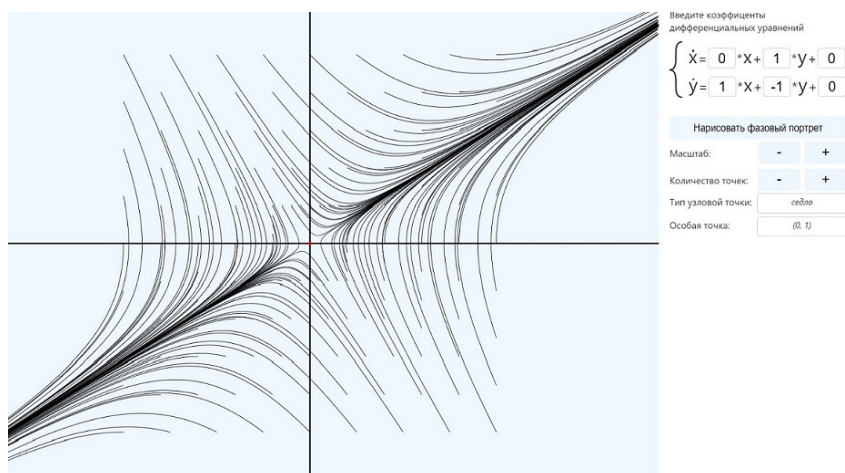
*А. Д. Уланова, П. А. Уланов (ЯрГУ им. П.Г. Демидова, Ярославль)*

*Научный руководитель: д.ф.-м.н., доцент Куликов А. Н.*

Целью данной работы было создание программы, которая позволяла бы строить фазовые портреты для любой линейной системы дифференциальных уравнений второго порядка без предварительного ее исследования. Наша программа основана на фреймворке ElectronJS. ElectronJS — это фреймворк для разработки настольных приложений с использованием HTML, CSS и JavaScript [1]. Такие приложения могут работать на различных платформах. Среди них — Windows, Mac и Linux.

В основе ElectronJS лежат проекты Chromium и Node.js, объединенные в единую среду, обеспечивающую работу приложений. Это даёт возможность применять веб-технологии при разработке настольных программ. Первым делом был создан макет программы на Axure 8. Пользовательский интерфейс программы представляет собой окно, в левой части которого располагается холст для отрисовки фазового портрета, а в правой

части располагаются поля ввода и кнопки управления. После того как пользователь ввел коэффициенты уравнений и нажал кнопку отрисовки, на экране отображается фазовый портрет введенной системы. Он строится при помощи метода Рунге-Кутты [2]. Сначала находится особая точка системы, затем в качестве начальных точек метода Рунге-Кутты берутся точки удаленные от особой на некоторое число порядка  $\varepsilon = 0,001$ . Затем производится сдвиг координатной оси относительно особой точки, чтобы она всегда была в центре холста отрисовки. Это делается для большей наглядности и для того, чтобы в некоторых случаях точка, которая лежит за пределами координат холста, все равно корректно отображалась на экране. Также пользователю доступны кнопки изменения масштаба и количества начальных точек. При помощи первой пары кнопок можно увеличивать либо уменьшать размеры области в которой строится фазовый портрет. Делается это путем увеличения расстояния между точками, полученными в результате работы метода Рунге-Кутты. При помощи второй пары кнопок можно увеличивать или уменьшать количество начальных точек. При этом фазовый портрет сопровождается следующей информацией: указываются координаты особой точки, а также ее тип. Данный разрабатываемый комплекс может быть использован в процессе обучения студентов физико-математических специальностей. Внешний вид и пример работы программы представлен на рисунке.



## Литература

1. Документация Electron.js [Электронный ресурс]. <https://www.electronjs.org/docs>
2. Численное решение системы дифференциальных уравнений [Электронный ресурс]. <https://toehelp.ru/theory/informat/lecture14.html>

## Уравнения в квад-графах и преобразование Бэклунда

**К. И. Фисенко (ЯрГУ им. П.Г. Демидова, Ярославль)**

*Научный руководитель: PhD Константину-Ризос С.*

Мы рассматриваем особый класс разностных уравнений, так называемые «уравнения в квад-графах». Для данного класса уравнений можно построить решения, используя простую линейную алгебру.

Актуальность темы заключается в том, что с помощью этих уравнений можно изучать решения нелинейных уравнений в частных производных для которых, как известно, нет общих методов построения решения. В частности, условие совместимости преобразований Дарбу для интегрируемого уравнения в частных производных вокруг квадрата, может быть рассмотрено как уравнение в квад-графах.

Уравнение в квад-графах имеет вид:

$$Q(u, u_{10}, u_{01}, u_{11}; a, b) = 0, \quad (1)$$

где  $Q$  — линейная по каждой переменной  $u_{ij}$ .

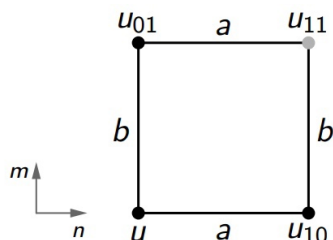


Рис. 1: Представление уравнений в квад-графах

В качестве иллюстративного примера мы рассматриваем дискретное потенциальное уравнение Кортевега–де Фриза – одно

из самых известных примеров данного класса:

$$(u - u_{11})(u_{10} - u_{01}) = a - b. \quad (2)$$

Такие уравнения можно понять как разностные уравнения следующим способом: пусть  $u = u(n, m)$ , через  $u_{ij}$  обозначаем  $u_{ij} = u(n + i, m + j)$ ,  $i, j \in \mathbb{Z}$ . т.е.,  $u_{00} \equiv u$ ,  $u_{10} = u(n + 1, m)$ ,  $u_{01} = u(n, m + 1)$  и  $u_{11} = u(n + 1, m + 1)$  и т.д.

Нет общего определения понятия интегрируемости дискретных систем, но есть «рабочие» определения. Пусть квад матрицы  $L = L(u, u_{10}; a, \lambda)$  и  $M = M(u, u_{01}; b, \lambda)$ .  $\Psi = \Psi(n, m)$  – вспомогательная матрица. Мы рассматриваем линейную систему:

$$\begin{cases} \Psi_{10} = L\Psi, \\ \Psi_{01} = M\Psi, \end{cases} \Rightarrow \begin{cases} \Psi_{11} = L_{01}\Psi_{01} = L_{01}M\Psi, \\ \Psi_{11} = M_{10}\Psi_{10} = M_{10}L\Psi, \end{cases} \quad (3)$$

**Определение 1.** Мы называем уравнение (1) интегрируемым, если существует пара матриц  $L(u, u_{10}; a, \lambda)$  и  $M(u, u_{01}; b, \lambda)$  с помощью которых уравнение (1) может быть переписано в эквивалентном виде:

$$L(u_{01}, u_{11}; a, \lambda)M(u, u_{01}; b, \lambda) = M(u_{10}, u_{11}; b, \lambda)L(u, u_{10}; a, \lambda). \quad (4)$$

Это значит, что нелинейное уравнение (1) является условием совместимости линейной системы (4).

**Определение 2.** Уравнение (1) является интегрируемым, если оно имеет свойство трехмерной совместимости, т.е. если мы можем его переписать совместимо на гранях куба.

Определение 2 сильнее 1 поскольку, во-первых, из определения 2 следует определение 1, во-вторых, если уравнение 1 имеет свойство трехмерной совместимости, то для него можно строить преобразование Бэклунда. С помощью этого преобразования мы можем строить интересные решения для нелинейной системы уравнений в квад-графах, используя найденное тривиальное решение. Мы это продемонстрируем на примере уравнения (2).

Перепишем уравнение (2) в нижней, передней и левой грани куба, и решаем для  $u_{110}$ ,  $u_{101}$  и  $u_{011}$ , соответственно:

$$u_{110} = u - \frac{a - b}{u_{010} - u_{100}}, \quad (5a)$$

$$u_{101} = u - \frac{a - c}{u_{001} - u_{100}}, \quad (5b)$$

$$u_{011} = u - \frac{c - b}{u_{010} - u_{001}}. \quad (5c)$$

Чтобы получить  $u_{111}$ , либо сдвинем (5a) по третьему направлению, либо (5b) по среднему направлению, либо (5c) по первому направлению. Можно проверить, что мы получаем одинаковые значения для  $u_{111}$ , следовательно уравнение (2) обладает свойством трехмерной совместимости.

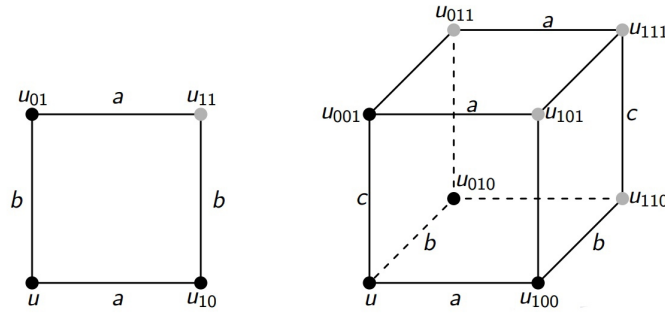


Рис. 2: Свойство трехмерной совместимости

Чтобы получить преобразование Бэклунда ставим  $v = u_{001}$ . Отсюда,  $v_{10} = u_{101}$ ,  $v_{01} = u_{011}$ . Нетрудно доказать, что если  $u$  является решением уравнения, то и  $v$  так же решение. После такой замены можно переписать уравнения (5b) и (5c):

$$\begin{aligned} (u - v_{10})(u_{10} - v) &= a - c, \\ (u - v_{01})(v - u_{01}) &= c - b. \end{aligned} \quad (6)$$

Система (6) — это наше отображение Бэклунда, с параметром  $c$ .

Уравнение Кортевега-де Фриза

$$(u - u_{11})(u_{10} - u_{01}) = a^2 - b^2$$

имеет одно простое решение

$$u(n, m) = an + bm + \text{const.}$$

Используя преобразование Бэклунда, мы сможем найти второе интересное решение вида:

$$v(n, m) = na + mb + c \frac{1 + \rho_{n,m}}{1 - \rho_{n,m}}, \quad \rho_{n,m} = \gamma \left( \frac{a+c}{a-c} \right)^n \left( \frac{b+c}{b-c} \right)^m.$$

## СЕКЦИЯ ПРОБЛЕМЫ РАЗРАБОТКИ И ЗАЩИТЫ ПРОГРАММ И ДАННЫХ

### Проблемы разработки программ на примере создания приложения для визуализации LDAP объектов

*Д. И. Алексеев (ЯрГУ им. П.Г. Демидова, Ярославль)*

*Научный руководитель: Савинов Д. А.*

Сейчас повсеместно используются службы каталогов — системы программного обеспечения, работающие с данными в каталогах операционных систем. Благодаря каталогам администраторы имеют возможность получать, редактировать, добавлять и удалять информацию о пользователях, а также и о других ресурсах компании.

Однако, существует ограниченное количество ПО, реализующих понятный и удобный интерфейс для выполнения описанных выше функций. Более того, основной проблемой существующих приложений является закрытость их исходного кода, что не только не позволяет обеспечить инфраструктуру систем безопасностью, но и приводит к их неоптимальному использованию. Заметим, что эти приложения не реализованы с учетом используемых в РФ систем, а следовательно не могут применяться отечественными производителями систем и ПО.

Целью исследования является разработка приложения для работы с сервером Active Directory, систематизация и решение проблем, возникших в процессе разработки, а также определение оптимального метода расширения функций приложения для дальнейшей разработки.

Задачами исследования являются конфигурация рабочих станций Alt Linux Client и Server, разработка прототипов клиентской и серверной частей приложения, систематизация полученных проблем разработки и оценка полученных результатов.

В начале у нас имеется 2 станции: Alt Linux Client, на которой будет запущено наше приложение и Alt Linux Server, где будет работать LDAP сервер.

Приложение состоит из двух частей. Первая — клиентская, вторая — серверная. Взаимодействие между клиентом и сервером происходит по HTTPS.

Разберем, из чего состоят части приложения. Клиентская часть приложения разработана на NodeJS. Для отрисовки интерфейса используется библиотека React, для управления состоянием приложения используется Redux. Жизненный цикл этой части приложения выглядит так: получив от пользователя запрос на выполнение какой-либо операции, соответствующий компонент обращается к объекту методов общения с серверной частью приложения — API. Подготовив данные, он посылает HTTPS REQUEST серверной части приложения. После этого клиентской части посылается HTTPS RESPONSE с требуемыми данными, изменяется соответствующая часть состояния приложения, а вместе с ней и зависимая часть интерфейса.

Разберем теперь как работает серверная часть приложения. Она также разработана на NodeJS, используя технологию NestJS. Эта часть состоит из следующих компонентов: контроллер, сервисы, а также объект Active Directory. Опишем жизненный цикл серверной части приложения. Получив HTTPS запрос от клиента, данные запроса отправляются в соответствующий сервис, где подготавливаются к отправке на LDAP сервер, затем передаются объекту Active Directory и отправляются серверу как LDAPS запрос. Получив ответ, сервер снова отправляет данные через сервис на контроллер, откуда они отправляются клиенту в качестве HTTPS RESPONSE.

Опишем проблемы разработки, с которыми мы столкнулись во время создания нашего приложения.

Первой проблемой является то, что фирмы не предоставляют описания уже существующих приложений. К каким последствиям приводит эта проблема было описано выше. Наше приложение решает эту проблему тем, что оно является открытым. Весь исходный код находится в открытом доступе, что дает возможность не только полного понимания работы приложения, но и позволяет разработчикам указывать на места потенциальных проблем. Все библиотеки, используемые в нашем приложении, также являются открытыми.

В качестве второй проблемы стоит привести сопротивление



распространителей отечественного ПО. Данная проблема была выявлена на этапе вступления в отечественный репозиторий разработки пакетов, или сборок приложений, для Alt Linux. Руководители проекта высказали опасение, связанное с использованием активно разрабатываемых библиотек, что может привести к ошибкам, связанным с зависимостью версий [1], а также возможной подменой сервера — репозитория библиотек [2]. Эти проблемы можно решить с помощью утилиты управления пакетами *rpm*. Этот подход используется в репозитории Alt Linux и позволяет добавить все необходимые зависимости с конкретными версиями приложения в один пакет. Альтернативным подходом является создание своего собственного установочного скрипта с определением необходимого набора зависимостей и их версий, устанавливающего данные библиотеки с доверенного сервера. Последний подход был нами реализован.

Третьей проблемой является защищенность данных, которые передаются от различных частей приложения. Проблема закрытости приложений других фирм не позволяет полностью определить, какие технологии используются для общения частей приложения, без применения анализаторов. В нашем же приложении четко понимается, что для общения используются HTTPS и LDAPS, что позволяет с уверенностью говорить о защищенности передаваемых данных.

Таким образом, в результате данной исследовательской работы был обнаружен ряд проблем, связанных с разработкой данного типа приложений, а также предложены и реализованы меры по их разрешению.

В качестве продолжения данной работы нам бы хотелось добиться вступления в репозиторий Alt Linux, настройки *rpm* пакета для установки приложения, добавления новых функций не только для работы с пользователями, но и с другими ресурсами системы. Также, необходимо испытать данное приложение в реальной системе.

### Литература

1. Проблема мелких зависимостей: <http://www.opennet.ru/opennews/art.shtml?num=54402>

2. Атака на зависимости библиотек приложений:

<http://www.opennet.ru/opennews/art.shtml?num=54566>

## О важности переопределения метода GetHashCode

**А. А. Баулина (ЯрГУ им. П.Г. Демидова, Ярославль)**

*Научный руководитель: к.ф.-м.н., доцент Якимова О. П.*

Хеш-код — это число, которое используется для вставки или идентификации элемента в таких коллекциях, как `HashSet<T>` и `Dictionary<TKey,TValue>` платформ `.Net` и `.Net Core`.

Метод `Object.GetHashCode()` возвращает хеш-код, нужный алгоритмам для проверки равенства элементов. Для оптимизации времени работы MSDN рекомендует переопределять `GetHashCode()` для значимых типов [1].

Цель исследования: изучить, в каких случаях нужно переопределять метод `GetHashCode()` и каким способом это сделать для достижения лучшей производительности.

По итогам изучения монографий и статей был выбран способ вычисления хеш-кода впервые предложенный Джошуа Блохом [2] (алгоритм 1). Кроме этого предлагается использовать модификацию этого алгоритма, использующую в качестве константы большое простое число 486187739 (алгоритм 2) и способ на основе криптографического генератора случайных чисел, реализуемого классом `RNGCryptoServiceProvider` (алгоритм 4). Сравнение производится с стандартной реализацией метода, предоставляемой платформой `.Net`.

Алгоритм	n=1000, ms	n=10000, ms	n=100000, ms	n=1000000, ms
1	0	2	27	579
2	0	2	27	449
3	1065	87057	8164491	>36000000
4	0	1	24	526

Рис. 1: Результаты тестирования наборов структур, где первые два поля — строки

Тестирование проводилось на наборах структур размером 1000, 10000, 100000 и один миллион элементов, причем значения

полей генерировались случайным образом. Часть результатов тестирования приведена на рисунках 1 и 2.

Алгоритм	n=1000, ms	n=10000, ms	n=100000, ms	n=1000000, ms
1	0	3	38	770
2	0	3	38	560
3	943	63397	6206299	>36000000
4	0	2	29	622

Рис. 2: Результаты тестирования наборов структур, где первое поле – строка, второе – массив строк

В результате проведенных экспериментов были получены следующие выводы: при использовании структур, первое поле которых строковое, а второе – строка или массив строк, необходимо переопределять метод GetHashCode, иначе возникают серьезные проблемы с временем работы коллекций, основанных на хеш-таблицах. Из вышеперечисленных способов переопределения наилучшую производительность показывают модификация способа Д. Блоха и на основе криптографического генератора случайных чисел.

### Литература

1. Object.GetHashCode Method:  
[https://docs.microsoft.com/en-us/dotnet/api/system.object.gethashcode?view=net-5.0#System\\_Object\\_GetHashCode](https://docs.microsoft.com/en-us/dotnet/api/system.object.gethashcode?view=net-5.0#System_Object_GetHashCode) (дата обращения: 08.05.2021).
2. Блох Д., Java: Эффективное программирование, ООО “Диалектика”, Санкт-Петербург (2019).

### Штрих-коды

**А. А. Бильчугова (ЯрГУ им. П.Г. Демидова,  
Ярославль)**

*Научный руководитель: Власова О. В.*

Цель работы: Исследование штрих-кодов и изучение алгоритмов их генерации.

Информационные и коммуникационные технологии активно развиваются во многих сферах нашей жизни. В экономике так

же постоянно идет развитие и разработка всё новых и новых способов контроля за производством и реализацией, качеством и продажами товаров народного потребления. В целях упрощения торговых процессов для быстрого считывания информации о товарах используют штрих-код.

Штриховые коды разделяют на три категории – одномерные (линейные) (UPC/EAN-128, EAN-13, UPC-E, Interleaved 2 of 5, EAN-8 и др.), двухмерные (Aztec Code, Data Matrix, MaxiCode, PDF417, QR код и др.) и радиометки [1]. Они отличаются графическим изображением, емкостью, т.е. «вместимостью» записываемых символов, способом шифрования информации и чтения. Также выбор типа штрих-кода зависит от вида данных, которые необходимо записать : цифры, буквы (заглавные и строчные, кириллица и латиница). Исходя из этих критериев одномерные и двумерные штрих-коды используют в разных сферах, а для их считывания может понадобиться разное оборудование.

Актуальность штрих-кодов в настоящее время состоит в том, что они широко используются не только при производстве и в торговле товарами, но и во многих отраслях промышленного производства для идентификации объектов, обозначения мест хранения, в туристических отраслях, в почтовых ведомствах, транспорте, сетевых технологиях.

В работе были реализованы следующие алгоритмы построения линейных штрих-кодов: EAN-13, Code 128, Interleaved 2 of 5 [2]. При обработке последовательностей выполнялось вычисление контрольных сумм в соответствии с правилами выбранного алгоритма. Прорисовка штрих-кодов выполнялась с помощью специальных шрифтов, которые содержат необходимые кодировочные таблицы (ean13.ttf, code128.ttf, code25I.ttf). Реализация выполнена на языке C#. Правильность генерации изображения штрих-кода была проверена на <https://www.imgonline.com.ua/> путем загрузки изображения. Все сгенерированные штрих-коды были распознаны.

### Литература

1. Афонин А.М., Царегородцев Ю.Н., Петрова А.М., Ефремова Ю.Е., Теоретические основы разработки и моделирования систем автоматизации, Форум, Москва, (2011).

2. Зульфугарзаде Т., Тушиев М., “Основные правила штрихового кодирования товаров на территории Российской Федерации. Аналитическое исследование,” *Право: теория и практика*, **14**, 10-14, (2003).

**Алгоритм Тарского. Система ввода-вывода**  
***Р. А. Гибадулин (ЯрГУ им. П.Г. Демидова,***  
***Ярославль)***

*Научный руководитель: д.ф.-м.н., профессор Дурнев В. Г.*

В предыдущей работе [1] рассматривался алгоритм Тарского. Для того, чтобы иметь возможность использовать этот алгоритм, необходимо реализовать систему ввода-вывода логических формул. Поэтому **цель работы** — реализовать систему ввода-вывода формул языка элементарной алгебры. Выбор языка обусловлен тем, что алгоритм Тарского принимает на вход формулы этого языка.

Для достижения цели необходимо было предложить и реализовать абстракции и интерфейсы их взаимодействия, решить задачу лексического анализа, решить задачу синтаксического анализа и реализовать обратное преобразование формулы в строку символов. Задачи были решены программно, на языке программирования C# версии 9.0.

Были выделены и реализованы в виде классов и интерфейсов следующие объекты: символ, оператор, слово, терм и формула. Каждый символ реализован как класс, который наследуется от абстрактного класса *Symbol*. Операторы реализованы как интерфейс *IOperator*, который определяет арность оператора и его тип: префиксный, инфиксный, постфиксный или функциональный. Слова и термы реализованы как интерфейсы *IWord* и *ITerm* соответственно. Следуя определению терма, интерфейс *ITerm* реализуют классы *FunctionTerm*, *ObjectVariable* и *IndividualConstant*. Следуя определению, абстрактный класс *Formula* реализуют три класса.

Задачу лексического анализа традиционно решают с помощью автоматов Мили [2]. Для построения автомата, распознающего символы языка элементарной алгебры, используется алгоритм построения префиксного дерева.

Синтаксический анализ условно можно разделить на четыре этапа. Сначала необходимо подготовить последовательность к парсингу, после чего проверить её на соответствие грамматике входного языка, затем привести её к обратной польской записи алгоритмом Дейкстры, и наконец непосредственно вычислить формулу.

Для того, чтобы преобразовать формулу в строку, были реализованы два класса. Первый по формуле на входе выдает последовательность символов на выходе согласно правилам грамматики выходного языка. Вторым по символу — строку, которой в системе  $\text{\LaTeX}$  обозначается данный символ.

В результате проделанной работы была реализована система ввода-вывода формул языка элементарной алгебры, которая в дальнейшем будет интегрирована с реализацией алгоритма Тарского.

### Литература

1. Гибадулин Р. А., Алгоритм Тарского, Путь в науку. Математика, тезисы докладов конференции, 39—40, 2020.
2. Соколов В. А., Введение в теорию формальных языков, Ярослав. гос. ун-т им. П. Г. Демидова., Ярославль, 2014.

### Определение доли плодовой ДНК Д. С. Гребенчук (ЯрГУ им. П.Г. Демидова, Ярославль)

*Научный руководитель: к.ф.-м.н., доцент Якимова О. П.*

Плодовая ДНК — это ДНК ребёнка в крови матери. Начиная с 10 недели беременности уже можно из крови матери получить 4% плодной внеклеточной ДНК, что позволяет оценить, с определенной долей вероятности, риск развития хромосомных аномалий у плода. Самым точным методом определения этой доли является определение по Y-хромосоме [1]. Однако, этот метод работает только в тех случаях, когда ребенок мужского пола, поэтому необходимо было протестировать новый метод определения доли плодовой ДНК, под названием *hetff*, который должен работать вне зависимости от пола ребёнка.

Для работы `hetff` необходимо было получить список позиций часто встречаемых вариантов генома, то есть отличий генома конкретного человека от референсного генома человека. Для этого были скачаны базы данных вариантов с известными частотами встречаемости с базы GnomAd, откуда были извлечены позиции с частотой более 1%. Было получено около 8,5 миллионов позиций.

Далее использовался ВАМ-файл с ДНК пациента, то есть файл с короткими последовательностями ДНК, выровненными на референсный геном. С помощью программы `Rysam` каждая позиция из ВАМ-файла сравнивалась с референсной. Если количество прочтений позиции от 2 до 4, то если эта позиция находится в частых вариантах, записываем ее сразу в выходной промежуточный файл, если нет, добавляем в общую сумму референсных и альтернативных аллелей. В конце прохода по ВАМ-файлу подсчитывается ошибка секвенирования: количество всех альтернативных аллелей делится на сумму количества всех альтернативных и всех референсных аллелей.

В выходном промежуточном файле находятся частые варианты, найденные в конкретном ВАМ-файле, с указанием количества референсных и альтернативных аллелей в каждом из вариантов.

Введем следующие обозначения [2]:

$$G_i = (c_i^A, c_i^B)$$

— количество референсных и альтернативных аллелей в  $i$ -м варианте,  $x$  — частота аллеля  $A$ ;

$$B(x, G_i) = \frac{(c_i^A + c_i^B)!}{c_i^A! c_i^B!} x^{c_i^A} (1 - x)^{c_i^B}$$

— биномиальная вероятность для  $G_i$ ;

$$M_i(h, F_1, F_2, e) = \sum_{j \in MMFF} P(j) B(f_A(j), G_i)$$

— предельная вероятность — биномиальная вероятность для совместных генотипов матери и ребенка, взвешенная по веро-

ятности совместного генотипа;

$$L(h, F_1, F_2, e) = \prod_{i \in S} M_i(h, F_1, F_2, e)$$

составная итоговая вероятность — произведение всех предельных вероятностей.

Таким образом, максимизируя составную итоговую вероятность, мы получаем оценку для перебираемых вариантов доли плодовой ДНК. И доля с максимальной итоговой вероятностью является ответом данного метода.

Метод был опробован на 67 мужских тестовых образцах. Для тех же образцов были известны результаты, определенные по Y-хромосоме, в качестве достоверных результатов, с которыми происходило сравнение. В итоге метод показал не очень хорошую корреляцию, всего 0,65. Поэтому было решено использовать его совместно с другим методом [3], совместно с которым корреляция значительно улучшилась и достигла значения 0,86. Особенно важно, чтобы точность определения плодовой доли ДНК была как можно выше при маленьких значениях этой доли. Совместное использование hetff и другого метода обеспечивают эту точность, теряя её при больших значениях доли плодовой ДНК, что не является критичным.

### Литература

1. Peng X.L., Jiang P., “Bioinformatics Approaches for Fetal DNA Fraction Estimation in Noninvasive Prenatal Testing”, *Int. J. Mol. Sci.*, **18**, No. 2, 453 (2017).
2. Dang M., Xu H., Zhang J., et al, “Inferring fetal fractions from read heterozygosity empowers the noninvasive prenatal screening”, *Genet Med*, **22**, 301-308 (2020).
3. Kim SK, et al, “Determination of Fetal DNA Fraction from the Plasma of Pregnant Women using Sequence Read Counts”, *Prenat Diagn*, **35**, No. 8, 810-815 (2015).



**Исследование свойств решеток,  
использующихся в качестве  
криптографических примитивов**  
*И. С. Грибкова (ЯрГУ им. П.Г. Демидова, Ярославль)*  
*Научный руководитель: к.ф.-м.н. Мурин Д. М*

Цели и задачи курсовой работы:

- Изучить свойства, которыми обладают криптографические решетки;
- Разработать программный комплекс для наглядного изучения работы алгоритмов;
- Реализовать LLL-алгоритм и алгоритм Бабаи;
- Обеспечить их наглядную реализацию.

Решёткой называется конечно порождённая подгруппа группы  $R^n$ . Если ранг группы равен  $n$ , то решётка называется полной, в противном случае — неполной. Базис группы называется в этом случае базисом решетки. Приведём примеры алгоритмов, в которых используются криптографические решётки:

1. Алгоритм Ленстры-Ленстры-Ловаса — алгоритм редукции базиса решётки;
2. Алгоритм Бабаи — алгоритм поиска ближайшей к  $x$  точки  $\lambda_x$  решетки  $L$ ;
3. NTRUEncrypt — криптографическая система с открытым ключом.

Впервые LLL-приведенные базисы были введены в рассмотрение в работе А. Ленстра, Х. Ленстра и Л. Ловаса и использовались ими для построения полиномиального алгоритма факторизации полиномов над кольцом целых чисел. С этого момента теория LLL-приведенных базисов нашла множество различных приложений, в том числе и в криптологии, где ее используют для построения «эффективных» атак на системы

защиты информации. LLL-алгоритм обладает другими многочисленными применениями. Так его можно применять при решении известной задачи об укладке рюкзака, имеющей приложения в криптографии. Именно поэтому было принято решение визуализировать работу алгоритма, приводящего ортогонализированные базисы в LLL-приведенные базисы.

При разработке визуализации вышеописанных алгоритмов были реализованы классы  $n$ -мерных векторов,  $n$ -мерных точек. Для визуализации векторов и их передвижения по координатной плоскости была выбрана графическая подсистема Windows Presentation Foundation. За создание и передвижение базиса векторов отвечают отдельные классы приложения.

Подводя итоги, можно отметить, что по результатам проделанной работы были изучены свойства криптографических решёток, реализованы алгоритмы LLL и Бабаи, визуализирована их работа в двумерном пространстве в приложении «Lattice app».

## **Уязвимости веб приложений. Cross-site scripting**

***И. О. Иванов (ЯрГУ им. П.Г. Демидова, Ярославль)***

*Научный руководитель: к.ф.-м.н., доцент Якимова О. П.*

С ростом спроса на разработку веб-приложений и ускорением времени их разработки снижается качество продуктов в сфере информационной безопасности. На сегодняшний день, огромное количество web-приложений содержат разные виды уязвимостей, которые могут привести к ущербам различного вида. Так, по информации журнала Positive Technologies, в 2019 году в 9 из 10 web-приложений преступники могут проводить атаки на пользователей, в 68% приложений существуют угрозы утечки важных данных. В 53% случаев, в таких приложениях существуют XSS уязвимости.

XSS (англ. Cross-Site Scripting — «межсайтовый скриптинг») — тип атаки на веб-системы, заключающийся во внедрении в выдаваемую веб-системой страницу вредоносного кода (который будет выполнен на компьютере пользователя при открытии им этой страницы) и взаимодействии этого кода с веб-сервером

злоумышленника [1]. Является разновидностью атаки «Внедрение кода».

Использование средств автоматического тестирования значительно упрощает процесс контроля качества продукта на разных этапах разработки. Подобные сканеры позволяют добиться приемлемого уровня безопасности путем обнаружения уязвимостей и создания специальных отчетов.

Целью данной работы является создание собственного инструмента поиска уязвимостей, являющегося веб приложением, построенного на паттерне MVC. Он позволяет быстро провести тестирование и контроль интернет-ресурса на наличие веб уязвимостей.

Приложение разработано на языке Java 8 [2]. Основным механизмом является фреймворк для автотестирования Selenium [3]. С помощью него можно имитировать действия пользователя, исполнять JS код. Само веб-приложение построено с помощью фреймворка Spring MVC, позволяющего создавать классические MVC приложения на базе Java [4].

Основной алгоритм выглядит следующим образом:

1. Построение внутренней карты сайта.
2. Поиск на каждой странице форм, с помощью которых пользователь может вводить свои данные.
3. Проверка каждой на потенциальную уязвимость. Если введенное тестовое значение осталось в URL при отправке формы — потенциальный reflect, осталось на странице — потенциальный stored.
4. Раскручивание потенциальных уязвимостей путем заполнения реальными уязвимостями, js код которых взят с сайта owasp.org [5].

### Литература

1. Межсайтовый скриптинг [Электронный ресурс]. URL: [https://www.ru.wikipedia.org/wiki/Межсайтовый\\_скриптинг](https://www.ru.wikipedia.org/wiki/Межсайтовый_скриптинг)
2. Java Platform Standard Edition 8 Documentation [Электронный ресурс]. URL: <https://docs.oracle.com/javase/8/docs/>
3. The Selenium Browser Automation Project [Электронный ресурс]. URL: <https://www.selenium.dev/documentation/en/>
4. Spring Framework [Электронный ресурс]. URL: <https://spring.io/projects/spring-framework>

5. XSS Filter Evasion Cheat Sheet [Электронный ресурс]. URL: <https://owasp.org/www-community/xss-filter-evasion-cheatsheet>

## **Описание уязвимостей на сайте обучающей SQL системы**

**С. А. Кобяков (ЯрГУ им. П.Г. Демидова, Ярославль)**  
*Научный руководитель: к.ф.-м.н. Мурин Д. М.*

### **Постановка задачи**

В данной работе я исследовал сайт обучающей SQL системы <http://sqlcontest.uniylar.ac.ru>. Данная обучающая система предназначена для обучения студентов написанию запросов на языке SQL, путём выполнения различных заданий. В ходе данной работы я исследовал этот сайт на наличие на нём каких-либо уязвимостей, которые могут привести к получению злоумышленником правильных ответов на задания, критической информации или полного контроля над сервером. Также я сформулировал рекомендации по исправлению найденных уязвимостей. Данная тема исследования является актуальной, потому что каждый год на этом сайте обучается множество студентов.

### **Результаты исследования**

В ходе данного исследования мне удалось найти уязвимости в обучающей системе.

#### *Доступ к реестру Windows*

На сайте есть возможность читать записи в реестре Windows.  
*SQL Инъекции*

На сайте присутствует возможность получать данные из базы данных обучающей системы через SQL Инъекцию, что позволяет злоумышленнику получать правильные ответы и аутентификационные данные.

#### *Reflected XSS Инъекции*

В обучающей системе есть возможность внедрить JavaScript код, который будет исполнен в браузере пользователей сайта. Это может привести к получению злоумышленником аутентификационных данных пользователей.

#### *Доступ к резервным копиям БД*

На сайте обучающей системы есть доступ к резервным копиям базы данных, что позволяет злоумышленнику получать данные из базы данных обучающей системы.

*Возможность загружать на сервер любые файлы*

На сайте присутствует возможность загружать любые файлы на веб-сервер, что может привести к получению злоумышленником полного контроля над сервером.

**Рекомендации по исправлению**

Для исправления этих уязвимостей на сайте обучающей системы требуется:

- 1) запретить пользователям SQL процедуры позволяющие читать реестр Windows;
- 2) фильтровать пользовательские параметры, которые отправляются к SQL серверу;
- 3) фильтровать данные, которые отправляются пользователю, на предмет XSS инъекций;
- 4) закрыть доступ к резервным копиям базы данных и возможности загружать любые файлы на сервер.

Особая благодарность Власовой Ольге Владимировне за возможность провести данное исследование на сайте обучающей SQL системы.

**Структуры данных на основе кучи**

***А. С. Никешина (ЯрГУ им. П.Г. Демидова,  
Ярославль)***

*Научный руководитель: к.ф.-м.н., доцент Якимова О. П.*

Куча представляет собой полное бинарное дерево, для которого выполняется основное свойство: величина родительского узла больше (или меньше) величины потомков [1]. Рассмотрим эту структуру данных на примере двоичной кучи.

В зависимости от того, максимальный или минимальный элемент расположен в корне, кучи подразделяются на *max* — *heap* и *min* — *heap*. Куча представима в виде массива: в корне кучи находится элемент *array[0]*, если узлу соответствует индекс *i*, то индекс левого потомка  $2 * i + 1$ , правого —  $2 * i + 2$ .

Рассмотрим сортировки, основанные на применении данной структуры.

Пирамидальная сортировка осуществляется в два этапа: сначала строится пирамида из элементов массива, а потом происходит упорядочивание элементов [2]. Главной составляющей этого процесса является просеивание — операция восстановления

основного свойства пирамиды, когда элемент с большей (или меньшей) величиной поднимается вверх по ветви.

Восходящая сортировка очень похожа на пирамидальную, основное отличие заключается в реализации просеивания. А именно, здесь сравниваются только потомки между собой, чтобы спуститься вниз по максимальным.

В турнирной сортировке куча используется как вспомогательная структура. С ее помощью минимальный узел, как победитель в турнирной сетке, путем сравнений поднимается вверх. Также используются два массива: в «победителей» записываем элементы из вершины кучи, а в «проигравших» тех, кто не попал в кучу. Затем объединяем эти массивы в один и снова повторяем процедуру турнира, пока массив проигравших не будет пуст.

В слабой куче у корня нет левого потомка, а каждый родитель не меньше любого потомка только из правого поддерева, в левом поддереве находятся потомки с любым приоритетом. При формировании первичной слабой кучи для каждого элемента ищем прародителя, для которого этот элемент будет правым потомком, сравниваем их. Для восстановления свойств кучи необходимо спуститься по левым потомкам на нижний уровень, поднимаясь, найти узел, который больше исходного и обменять их.

Декартово дерево является кучей, если смотреть на значения, и деревом, относительно индексов элементов в массиве. Основа данной сортировки – операция объединения, когда необходимо соединить два дерева в одно, не нарушив свойств данной структуры. Сначала строится общая куча, затем, пока в ней есть элементы, извлекаем вершину и объединяем образовавшиеся поддерева, на которые распалось начальное дерево.

Было проведено исследование зависимости временной сложности сортировки в зависимости от типа кучи, размера и характера входных данных (упорядочен входной массив или нет). Тестирование проводилось на компьютере с частотой процессора 1.60 ГГц, модель Intel Core i5-8250U CPU, объем оперативной памяти: 8 ГБ. Часть результатов, измеренных в тиках, представлена на рисунке 1.

Несмотря на то, что трудоемкость всех алгоритмов состав-

	Случайный массив			Упорядоченный массив		
Размер массива:	100	1000	10 000	100	1000	10 000
Пирамидальная	1	1	2	2	1	1
Восходящая	1	1	1	1	1	1
Слабой кучей	1	1	1	1	1	1
Турнирная	1	4	26	1	2	19
Декартовым деревом	370	6223	58226	29	296	5573

Рис. 1: Результаты тестирования

ляет  $O(n * \log n)$ , они обладают разной производительностью. По результатам данного тестирования отметим, что наиболее эффективными по времени являются восходящая сортировка, пирамидальная сортировка и сортировка слабой кучей, а самая затратная — сортировка декартовым деревом.

### Литература

1. Структуры данных: двоичная куча (binary heap) [Электронный ресурс]: <https://habr.com/ru/post/112222/>
2. Williams J. W. J., “Algorithm 232 – Heapsort”, Communications of the ACM, **6**, No. 7, 347–348 (1964).

## Модель вируса на языке программирования Python

*С. А. Скородумов (ЯрГУ им. П.Г. Демидова,  
Ярославль)*

*Научный руководитель: Власова О. В.*

Задача проекта — создать модель HLLP-вируса (High Level Language Parasitic), который не только распространяет свой код, но также архивирует недавно открытые пользователем файлы и отправляет архив на почту своего создателя.

HLLP являются самыми продвинутыми из ЯВУ-вирусов. Они внедряются непосредственно в файл-жертву, сохраняя при этом

работоспособность оригинального кода [1]. Данная модель записывает своё тело в начало файла-жертвы.

Несмотря на простоту и универсальность, Python не лучший язык для написания вирусов. Это связано с тем, что Python — интерпретируемый язык, и его сценарии — это текстовые файлы. Если записать текстовый файл в начало двоичного exe-файла, то заражения, конечно, не произойдёт.

Работу над моделью можно разделить на несколько этапов:

1. Создание основной части вируса
2. Создание архива недавно открытых файлов
3. Создание сообщения с прикреплённым архивом
4. Отправка сообщения

Основная часть вируса инфицирует файлы, находящиеся в одной директории с файлом, в который внедрён вирус. Алгоритм её работы можно описать следующим образом:

1. Получить путь к файлу, в котором находится код вируса
2. Получить список файлов, лежащих в той же директории
3. Прочитать данные вируса
4. Проверить, есть ли в файле данные оригинальной программы (их нет при первом запуске вируса)
5. Если они есть, то:
  - (a) Прочитать данные оригинальной программы
  - (b) Создать exe-файл с кодом оригинальной программы и запустить его в отдельном потоке
6. Для каждого неинфицированного exe-файла, лежащего в одной директории с файлом вируса:
  - (a) Переименовать файл
  - (b) Прочитать данные файла



- (с) Создать новый файл с именем оригинала
- (d) Записать в него сначала данные вируса, а затем данные оригинала
- (е) Удалить оригинальный файл

Второй поток используется для того, чтобы оригинальное приложение запустилось как можно быстрее, и у пользователя было меньше шансов узнать о работе модели вируса.

Для создания архива используется системная папка «Недавние документы», в которой хранятся данные о последних файлах, которые открывал данный пользователь. Из этой папки можно получить пути к искомым файлам и добавить их в архив.

Создание сообщения и его отправка — тривиальные задачи, которые решаются с помощью специальных модулей стандартной библиотеки Python. Для конвертации сценария Python в двоичный файл используется модуль `cx_Freeze`.

Исходный код доступен на github [2].

Вывод: модель вируса на языке программирования Python можно создать описанным выше способом.

### Литература

1. Цифровой журнал "Хакер": <https://xakep.ru/2011/03/13/55980/#toc02>.
2. GitHub: <https://github.com/sSkorodumov/HLLP>

## СЕКЦИЯ МАТЕМАТИЧЕСКИЙ АНАЛИЗ

### Эйлеровы числа и их приложения

**И. А. Петров (ЯрГУ им. П.Г.Демидова, Ярославль)**

*Научный руководитель: д.ф.-м.н., доцент Невский М. В.*

В работе приводятся новые оценки для некоторых величин, определяемых с помощью эйлеровых чисел.

Пусть  $k$  и  $n$  натуральные,  $k \leq n$ . Эйлеровым числом  $A_{n,k}$  называется количество перестановок порядка  $n$ , каждая из которых имеет ровно  $k - 1$  снижений, то есть инверсий своих соседних компонент. Для вычисления эйлеровых чисел можно использовать явную формулу

$$A_{n,k} = \sum_{j=0}^{k-1} (-1)^j \binom{n+1}{j} (k-j)^n,$$

или рекуррентное соотношение

$$A_{n,k} = (n - k + 1)A_{n-1,k-1} + kA_{n-1,k}.$$

Геометрическая интерпретация эйлеровых чисел связана с объемами слоев  $n$ -мерного единичного куба  $Q_n := [0, 1]^n$ . По поводу этих и других свойств см., например, [1, глава 4].

Определим числа  $Y_n$  следующим образом. Для четного  $n = 2m$  положим

$$Y_n := \sum_{j=1}^m j A_{n,m+j}.$$

Если же  $n$  — нечетное,  $n = 2m + 1$ , то по определению считаем

$$Y_n := \frac{1}{4} A_{n,m+1} + \sum_{j=2}^{m+1} \left( j - \frac{1}{2} \right) A_{n,m+j}.$$

Последовательность  $Y_n$  была введена в [1]. Там доказано, что существует константа  $\gamma \geq 2$ , такая что для всех  $n \in \mathbb{N}$  справедлива оценка

$$\frac{Y_n}{n!} > \frac{\sqrt{n}}{12\gamma}. \quad (1)$$

Запишем неравенство (1) в эквивалентном виде:

$$\gamma > \frac{n!\sqrt{n}}{12Y_n}. \quad (2)$$

Обозначим правую часть (2) через  $M_n$ . С применением компьютера нами получены следующие результаты.

$n$	$Y_n$	$M_n$
1	0,25	0,33(3)
2	1	0,2357
3	2.5	0,3464
4	13	0,3077
5	58	0,3856
...		
51	$1,652 \cdot 10^{66}$	0,5587
...		
170	$1,277 \cdot 10^{307}$	0,6215

Нас интересует наименьшая положительная константа, с которой выполняется (1). Пусть  $\gamma^*$  — точная нижняя грань тех  $\gamma > 0$ , для которых неравенство (1) выполняется при всех  $n$ . Очевидно,

$$\gamma^* \geq \sup_n M_n \geq \max_{1 \leq n \leq 170} M_n = 0,6215.$$

Итак,  $\gamma^* \geq 0,6215$ . Иными словами, неравенство (1) не может выполняться для всех  $n$  ни с какой константой, меньшей 0.6215.

Приведем пример применения наших результатов. Пусть  $H$  — ортогональный проектор, действующий из  $C(Q_n)$  на пространство многочленов от  $n$  переменных степени  $\leq 1$ , соответствующий скалярному произведению  $(f, g) = \int_{Q_n} f(x)g(x)dx$ . Как доказано в [1], норма  $H$  как оператора из  $C(Q_n)$  в  $C(Q_n)$  удовлетворяет неравенству

$$\|H\| > c\sqrt{n} - 7, \quad c = \frac{1}{\gamma}.$$

Из предыдущего следует, что по крайней мере при  $n \leq 170$  эта оценка является верной, если взять  $c = (0,6215)^{-1} = 1,6090$ .

### Литература

1. Невский М.В. Геометрические оценки в полиномиальной интерполяции. Ярославль: ЯрГУ, 2012. 218 с.

## Поиск приближенного равновесия в играх с неполной информацией

*Р. С. Сычев (ЯрГУ им. П.Г. Демидова, Ярославль)*

*Научный руководитель: к.ф.-м.н. Глазков Д. В.*

В настоящее время теория игр с неполной информацией находит все большее применение в таких отраслях как теория операций, экономика, кибербезопасность и физическая безопасность. Не в последнюю очередь это происходит благодаря постоянному совершенствованию алгоритмов и увеличению производительности вычислительной техники. Частным случаем таких игр являются игры в развернутой форме. При такой постановке задачи можно близким к естественному способом отразить в игровой форме структуру последовательного принятия решений набором участников в конфликтной ситуации.

Существенным шагом в развитии данного направления является использование алгоритма подсчета сожалений [1]. Алгоритм представляет из себя относительно простую процедуру приближения корелирующего равновесия. Нас же интересует модификация данного алгоритма для игр с неполной информацией CFR (Counterfactual Regret Minimization) [2]. В данной реализации кумулятивные сожаления представлены следующим образом

$$R_i^T(I, a) = \frac{1}{T} \sum_{t=1}^T \pi_{-i}^{\sigma^t}(I) (u_i(\sigma^t|_{I \rightarrow a}, I) - u_i(\sigma^t, I)). \quad (1)$$

В формуле (1)  $I$  представляет информационное состояние, а  $a$  обозначает доступное в информационном состоянии  $I$  действие игрока. Величина  $\pi_{-i}^{\sigma^t}(I)$  возвращает вероятность достижения информационного состояния  $I$  при стратегии  $\sigma^t$  без учета действий игрока  $i$ . Величина  $u_i$  представляет ожидаемую выплату игроку  $i$  при заданном стратегическом профиле начиная с  $I$ .

Ключевым отличием подобных сожалений является использование крайне простой модификации стратегии  $\sigma^t|_{I \rightarrow a}$ , в которой для оценки полезности действия стратегия игрока меняется для постоянного выбора  $a$  в  $I$ . Примечательно, что сумма максимумов таких сожалений по всем информационным состояниям не может быть меньше среднего общего сожаления, и для нахождения решения достаточно минимизировать каждое из таких сожалений [2].

Для этого, как правило, используют процедуру, называемую алгоритмом Блэквела [1]. Стратегия при  $t = 1$  выбирается случайным образом. Расчет последующих стратегий происходит по формуле

$$\sigma_i^{T+1}(I)(a) = \begin{cases} \frac{R_i^{T,+}(I, a)}{\sum_{a \in A(I)} R_i^{T,+}(I, a)} & \text{если } \sum_{a \in A(I)} R_i^{T,+}(I, a) > 0, \\ \frac{1}{|A(I)|} & \text{иначе.} \end{cases} \quad (2)$$

В формуле (2)  $A(I)$  представляет множество допустимых действий в  $I$ . На практике это означает, что на каждой итерации происходит одно обновление стратегии для каждого информационного состояния каждого игрока. Итоговая стратегия считается как усредненная по всем итерациям

$$\bar{\sigma}_i^T(I) = \frac{\sum_{t=1}^T \pi_i^{\sigma^t}(I) \sigma^t(I)(a)}{\sum_{t=1}^T \pi_i^{\sigma^t}(I)}. \quad (3)$$

В случае игры двух игроков с нулевой суммой, данный алгоритм сходится к равновесию Нэша [1]. Случай большего числа игроков не имеет теоретически обоснованной сходимости к равновесию Нэша, но некоторые теоретические обоснования его применения есть и в этом случае, и на практике успешно рассматриваются игры большего числа игроков [3].

Целью данной работы является отработка приведенного выше алгоритма на практике. В качестве объекта исследования была выбрана упрощенная версия классического домино. В игре участвуют 2 игрока. Набор костяшек включает 10 единиц и содержит все сочетания пар чисел от 0 до 3 без повторений и 4 дубля. В начале каждого раунда игроки получают на руки по 3

костяшки, остальные 4 остаются в резерве и добираются игроками после хода. Право первого хода разыгрывается случайным образом. Выкладка костяшек происходит по обычным правилам домино, и игра заканчивается при невозможности произвести ход.

Но даже такая сильно упрощенная версия остается довольно сложной в комбинаторном плане. В связи с этим при построении модели был использован ряд оптимизаций:

- абстракция истории игры. Это расширение информационных состояний игроков за счет частичного забывания публичной истории. На практике наборы из одних и тех же костяшек с одинаковыми числами на концах считались эквивалентными;
- метод Монте-Карло для расчета сожалений [4]. Для сокращения объема памяти и производимых вычислений используется предпросчет исходов случайных событий перед каждой итерацией алгоритма. Этот прием позволяет на порядок сократить объем занимаемой памяти.

В ходе исследования была сформирована модель, содержащая  $10^5$  информационных состояний и проведено  $10^8$  итераций алгоритма CFR. В итоге был получен профиль стратегий с расчетной эксплуатируемостью не более чем в 0.02 очков, по сравнению с 0.93 при использовании случайной стратегии.

### Литература

1. Hart S., Mas-Colell A. "A simple adaptive procedure leading to correlated equilibrium", *Econometrica*, 68(5):1127–1150, (2000).
2. Zinkevich M., Johanson M., Bowling M., Piccione C. *Regret minimization in games with incomplete information*, MIT Press, Cambridge (2008).
3. Superhuman AI for multiplayer poker: <https://science.sciencemag.org/content/365/6456/885>
4. Lanctot M., Waugh K., Zinkevich M., Bowling M. "Monte carlo sampling for regret minimization in extensive games", *Advances in Neural Information Processing Systems* 22, pages 1078–1086 (2009).

## Обучение глубоких нейронных сетей

*Д. Д. Федулов (ЯрГУ им. П.Г. Демидова, Ярославль)*

*Научный руководитель: к.ф.-м.н. Ухалов А. Ю.*

Рассмотрим следующую задачу компьютерного зрения. Необходимо разработать модель машинного обучения, которая получает на вход изображения из видеопотока и определяет, какой жест из 33 возможных показывает рука на изображении. Для обучения модели используется тренировочный датасет, содержащий изображения рук, показывающих жест из 33 возможных как на чёрном контрастном фоне, так и на произвольном неконтрастном фоне. Для проверки качества модели используется валидационный датасет, состоящий из изображений жестов руки из 33 возможных, снятых на произвольном неконтрастном фоне (см. рис. 1).

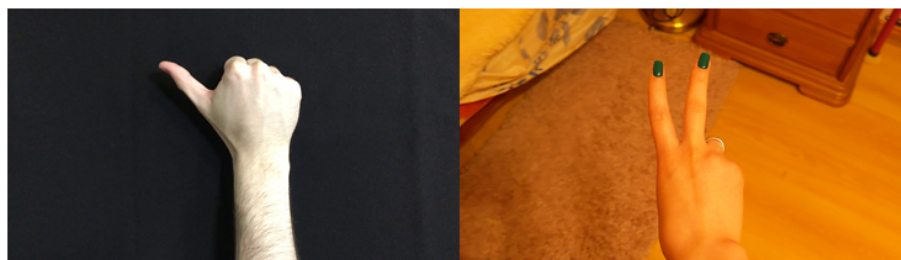


Рис. 1: Примеры изображений: из тренировочного набора данных (слева), из валидационного набора данных (справа)

В рамках первой серии экспериментов, были обучены модели глубокого обучения для классификации 33 жестов следующих архитектур нейронных сетей: VGG-16, InceptionResNetV2, MobileNetV2. Обучение производилось с помощью технологии обучения с переносом опыта на основе моделей, обученных для решения задачи классификации на 1000 классов из набора данных ImageNet, с заменой полносвязного слоя для получения вероятности принадлежности одному из 33 классов [1]. Обучение каждой нейронной сети производилось в несколько итераций — эпох. В конце каждой эпохи производился контроль на валидационной выборке. В качестве метрики оценки качества предсказания используется максимальная доля правильных ответов (Ассигасу). Для эксперимента была написана программа

Таблица 1: Значения доли правильных ответов, эксперимент 1

Архитектура сети	Доля правильных ответов
MobileNetV2	0.13853
InceptionResNetV2	0.1897
VGG-16	0.1762

на языке Python с использованием библиотек tensorflow и keras. Результаты эксперимента содержатся в таблице 1.

Мы видим довольно низкие значения метрики качества модели, которые не позволяют применить нашу модель на практике. Для дальнейших экспериментов, был предложен трёхступенчатый процесс машинного обучения:

1. Определение региона, в котором находится рука;
2. Определение ключевых точек руки, находящихся в определённом на этапе 1 регионе;
3. Классификация жеста на основе ключевых точек руки.

Последующие серии экспериментов основывались на использовании готовой модели Google Mediapipe [2] для 1 и 2 этапа и обработке ключевых точек и последующем обучении моделей — классификаторов для предсказаний классов.

Во второй серии экспериментов, координаты ключевых точек руки использовались для обучения классификаторов. В третьей серии экспериментов для обучения классификаторов, помимо ключевых точек руки, использовались расстояния между кончиками пальцев. В четвёртой серии экспериментов для обучения классификаторов, помимо ключевых точек руки, использовались расстояния между некоторыми ключевыми точками, а также некоторые углы между ними. В качестве классификаторы использовались  $K$  ближайших соседей, дерево решений, случайный лес, машина опорных векторов, логистическая регрессия, гребневая регрессия, стохастический градиентный спуск. Обучение классификаторов осуществлялось на тренировочном наборе данных, контроль качества — на валидационном. Для проверки качества классификатора использовалась доля правильных ответов и  $F_1$  мера. Для экспериментов были написа-



Таблица 2: Значения доли правильных ответов, эксперименты 2-4

Классификатор	Эксперимент 2	Эксперимент 3	Эксперимент 4
KNN	0,5907136	0,6489119	0,4940853
SVM	0,8053834	<b>0,8692434</b>	0,8095585
Random Forest	0,6465397	0,7496837	0,8347988
Decision Tree	0,5028467	0,5564588	0,6820913
LogReg	0,7649291	<b>0,8372659</b>	0,8252151
LogReg + $L_2$	<b>0,8104757</b>	<b>0,8537133</b>	0,8221154
SGD	<b>0,8166751</b>	0,8111083	<b>0,8773406</b>
SGD + $L_1$	<b>0,8153150</b>	0,8166435	<b>0,8768345</b>
SGD + $L_2$	0,7866587	0,8021255	<b>0,8713942</b>
Ridge	0,7952935	0,7899481	0,8259426

ны программы на языке Python с использованием библиотек pandas и scikit-learn. Результаты экспериментов представлены в таблице 2.

### Литература

1. A. Karpathy, Fei-Fei Li, J. Johnson. CS231n Convolutional Neural Networks for Visual Recognition Lecture Notes: <http://cs231n.github.io>
2. Zhang F., Bazarevsky V., Vakunov A., Tkachenka A., Sung G., Chang C.-L., Grundmann M. “MediaPipe Hands: On-device Real-time Hand Tracking”, arxiv:2006.10214 [cs.CV]

### Некоторые алгоритмы обработки изображений

*О. Н. Федотова (ЯрГУ им. П.Г. Демидова,  
Ярославль)*

*Научный руководитель: к.ф.-м.н. Глазков Д. В.*

Рассмотрим применение свертки в задачах обработки изображений. Под сверткой понимается операция, описывающая взаимодействие сигналов между собой. Применим к тестовым изображениям фильтры размытия, четкости и выделения границ. Реализуем это при помощи матрицы свертки. Под ней будем понимать матрицу коэффициентов, которая при «умножении»

на значение пикселей изображения дает необходимый результат.

В качестве тестовых взяты черно-белые и цветные изображения, содержащие различное количество мелких деталей.

Ниже приведены матрицы, соответствующие каждому из фильтров [1].

Матрица свертки для размытия изображения:

$$\begin{bmatrix} 0.000789 & 0.006581 & 0.013347 & 0.006581 & 0.000789 \\ 0.006581 & 0.054901 & 0.111345 & 0.054901 & 0.006581 \\ 0.013347 & 0.111345 & 0.225821 & 0.111345 & 0.013347 \\ 0.006581 & 0.054901 & 0.111345 & 0.054901 & 0.006581 \\ 0.000789 & 0.006581 & 0.013347 & 0.006581 & 0.000789 \end{bmatrix}.$$

Матрица, которую используют для улучшения четкости:

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}.$$

Для выделения границ применяются следующие две матрицы, называемые оператором Превитта:

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix},$$

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}.$$

Рассмотрим результаты применения фильтров. Вне зависимости от цвета и сложности изображения применение фильтра размытия дало отличный результат. Отметим, что степень размытия будет зависеть от размера используемой матрицы свертки.

При применении матрицы свертки для улучшения четкости у предметов лучше прослеживается их текстура, однако в некоторых случаях на фоне возможно появление мелкого зернистого шума. У изображения, содержащего множество мелких деталей, наблюдалось ухудшение цветопередачи.

Фильтр выделения границ дает хороший результат на тестовых данных, которые не содержат большого количества мелких деталей. При наличии множества деталей программа выделяет контур каждой, что в итоге приводит к нечитабельности и неузнаваемости изображения.

Рассмотрим также задачу увеличения изображений с использованием В-сплайнов. При масштабировании необходимо будет вычислять значения новых пикселей по следующей формуле:

$$S(x, y) = \sum_{p=i-1}^{i+2} \sum_{q=j-1}^{j+2} f_{p,q} B_p(x) B_q(y),$$

где  $f_{p,q}$  — это коэффициенты, значения которых берутся из заданной матрицы пикселей, а  $B_p(x)$  и  $B_q(y)$  — это базисные кубические В-сплайны. Их значения вычисляются заранее и используются далее без повторных расчетов. Получим значения кубических В-сплайнов на отрезке  $[x_i, x_{i+1}]$  согласно формулам:

$$\begin{aligned} B_{i-1}(x) &= \frac{1}{6}(1 - t^3), \\ B_i(x) &= \frac{1}{6}[1 + 3(1 - t) + 3t(1 - t^2)], \\ B_{i+1}(x) &= \frac{1}{6}[1 + 3t + 3t^2(1 - t)], \\ B_{i+2}(x) &= \frac{1}{6}t^3, \end{aligned}$$

где параметр  $t = \frac{x - x_i}{h}$ , а  $h$  — шаг сетки [2].

Применим данные формулы для восстановления значения пикселей при увеличении тестовых изображений. Заметим, что при наличии небольшого количества деталей алгоритм дает хорошие результаты, качество изображений при увеличении практически не страдает. Однако в случае с изображением с большим количеством мелких деталей пропадает четкость. Насыщенность и яркость изображений практически не изменилась.

## Литература

1. Матричные фильтры обработки изображений:

<https://habr.com/ru/post/142818/>

2. Завьялов Ю.С., Квасов Б.И., Мирошниченко В.Л., Методы сплайн-функций, Наука, Москва (1980).

## Алгоритм Тренча в задачах аппроксимации

*Е. А. Краснова, К. Н. Чубукова (ЯрГУ им. П.Г. Демидова, Ярославль)*

*Научный руководитель: д.ф.-м.н., доцент Невский М.В.*

В ряде прикладных задач теории приближений требуется эффективно выполнять обращение ганкелевых матриц. К таким задачам относятся вычисление аппроксимаций Паде [1] и реализация быстрых методов рациональной интерполяции [2]. Для численного решения этих задач авторы применили алгоритм Тренча обращения ганкелевой матрицы, который имеет трудоемкость  $O(n^2)$  по сравнению с  $O(n^3)$  для метода Гаусса.

Интересно, что наиболее подходящей для использования оказалась схема алгоритма, приведённая в оригинальной работе Уильяма Ф. Тренча [3]. Алгоритм Тренча был реализован на компьютере и тестирован на матрицах достаточно большого порядка. Ниже даётся короткое описание этой вычислительной схемы. Мы применяем определения из монографии [4].

Тёплицевой матрицей  $T = (a_{ij}) \in M_{n+1}$  называется матрица, имеющая вид

$$T = \begin{bmatrix} a_0 & a_1 & a_2 & \dots & a_n \\ a_{-1} & a_0 & a_1 & \dots & a_{n-1} \\ a_{-2} & a_{-1} & a_0 & \dots & a_{n-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{-n} & a_{-n+1} & \dots & a_{-1} & a_0 \end{bmatrix}. \quad (1)$$

Она определяется числами  $a_{-n}, a_{-n+1}, \dots, a_{-1}, a_0, a_1, \dots, a_{n-1}, a_n$ , причем  $a_{ij} = a_{j-i}$ . Ганкелевой матрицей  $H = (a_{ij}) \in M_{n+1}$  на-

зывается матрица вида

$$H = \begin{bmatrix} a_0 & a_1 & a_2 & \dots & a_{n-1} & a_n \\ a_1 & a_2 & a_3 & \dots & a_n & a_{n+1} \\ a_2 & a_3 & a_4 & \dots & a_{n+1} & a_{n+2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_n & a_{n+1} & a_{n+2} & \dots & a_{2n-1} & a_{2n} \end{bmatrix}.$$

Она определяется последовательностью  $a_0, a_1, a_2, \dots, a_{2n-1}, a_{2n}$ , причем  $a_{ij} = a_{i+j-2}$ .

Пусть  $P$  — *переходничная матрица*, у которой элементы побочной диагонали равны 1, а остальные элементы равны 0. Нетрудно видеть, что матрица  $PT$  является ганкелевой тогда и только тогда, когда  $T$  — *тёплицева*, и матрица  $PH$  — *тёплицева* в том и только том случае, когда  $H$  ганкелева.

Пусть дана ганкелева матрица  $H$  порядка  $n$  и требуется найти обратную к ней  $H^{-1}$ . Положим  $T = PH$ . Будем считать, что для матрицы  $T$  выполняется равенство  $a_0 = 1$  (см. вид (1)).

Принимаем  $\psi_{00} = a_1$ ,  $\eta_{00} = a_{-1}$ ,  $\Delta_{-1} = 1$ . С помощью рекурсивных формул вычислим значения:

$$\begin{aligned} \Delta_{m-1} &= (1 - \psi_{m-1,m-1}\eta_{m-1,m-1})\Delta_{m-2}, \quad m = 1, \dots, n-1, \\ \psi_{mm} &= \Delta_{m-1}^{-1} \left[ \phi_{m+1} - \sum_{s=0}^{m-1} \psi_{s,m-1}\phi_{m-s} \right], \\ \eta_{mm} &= \Delta_{m-1}^{-1} \left[ \phi_{-m-1} - \sum_{s=0}^{m-1} \eta_{s,m-1}\phi_{-m+s} \right], \\ \psi_{rm} &= \psi_{r,m-1} - \psi_{mm}\eta_{m-r-1,m-1}, \quad 0 \leq r \leq m-1, \\ \eta_{rm} &= \eta_{r,m-1} - \eta_{mm}\psi_{m-r-1,m-1}, \quad 0 \leq r \leq m-1. \end{aligned}$$

Элементы матрицы  $T^{-1} = (b_{rs})$  задаются выражениями:

$$\begin{aligned} b_{00} &= \Delta_{n-1}^{-1}, \\ b_{0r} &= -\Delta_{n-1}^{-1}\psi_{r-1,n-1}, \quad 1 \leq r \leq n, \\ b_{s0} &= -\Delta_{n-1}^{-1}\eta_{s-1,n-1}, \quad 1 \leq s \leq n, \\ b_{rs} &= b_{r-1,s-1} + \Delta_{n-1}^{-1} [\psi_{r-1,n-1}\eta_{s-1,n-1} - \eta_{n-r,n-1}\psi_{n-s,n-1}], \\ &\quad 1 \leq r, s \leq n. \end{aligned}$$

В заключение полагаем  $H^{-1} = T^{-1}P$ , и задача решена.

**Литература**

1. Бейкер Дж., Грейвс-Моррис П. Аппроксимации Паде. М.: Мир, 1986. 502 с.
2. Egencioglu, O., and Koc, C. K. A fast algorithm for rational interpolation via orthogonal polynomials // Math. Comp. 1989. V. 53, no. 187. P. 249–264.
3. Trench, W.F. An algorithm for the inversion of finite Toeplitz matrices // J. Soc. Indust. Appl. Math. 1964. V. 12, no. 3. P. 515–522.
4. Хорн Р., Джонсон Ч. Матричный анализ. М: Мир, 1989. 655 с.

## СЕКЦИЯ МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

### О решении задач методами теории подобия и размерностей

*А. А. Давыдов (РГАТУ им. П.А. Соловьева, Рыбинск)*

*Научный руководитель: к.т.н., доцент Васильева А. В.*

Теорию подобия и размерностей широко применяют в моделировании различных процессов. При выявлении критериев подобия используются общие уравнения, описывающие процесс, и условия однозначности. В тех же случаях, когда рассматривается новый процесс, и аналитического описания для него нет, выявление критериев подобия может производиться методом анализа размерностей.

Любой процесс характеризуют определяющие величины и функциональная зависимость, связывающая эти величины. Определяющие величины  $x_1, x_2, x_3, \dots, x_N$  — это различные размерные или безразмерные переменные и размерные постоянные. Определяющая зависимость выражается в общем случае некоторым конечным уравнением и может быть представлена как степенной комплекс вида:  $y = x_1^{\alpha_1} x_2^{\alpha_2} x_3^{\alpha_3} \dots x_N^{\alpha_N}$ , где  $\alpha_i$  — любые числа.

Рассмотрим решение подобной задачи на примере работы бойлерной установки. Пусть труба с нагреваемой водой диаметром  $d$  омывается горячей жидкостью с постоянной температурой  $\theta$ . Температура стенки трубы поддерживается постоянной и равной  $\theta_0$ . Требуется определить количество тепла  $Q$  Дж/с, переданного в единицу времени от горячей жидкости.

Переданное тепло за единицу времени по закону Ньютона равно:

$$Q = \alpha_c \cdot (\theta - \theta_0) \cdot S.$$

Величины  $\theta$ ,  $\theta_0$  и  $S$  известны. Необходимо знать величину коэффициента теплоотдачи соприкосновением  $\alpha_c$ :

$$\alpha_c = \text{Дж/м}^2 \cdot \text{с} \cdot \text{К}.$$

Пусть на коэффициент  $\alpha_c$  оказывают значительное влияние величины процесса:

1. Диаметр  $d$ ,  $[d] = \text{м}$ ;
2. Скорость движения жидкости  $v$ ,  $[v] = \text{м/с}$ ;
3. Коэффициент теплопроводности жидкости  $\lambda_c$ ,  $[\lambda_c] = \text{Дж/м} \cdot \text{с} \cdot \text{К}$ ;
4. Коэффициент кинетической вязкости жидкости  $\nu_c$ ,  $[\nu_c] = \text{м}^2/\text{с}$ ;
5. Удельная объемная теплоемкость жидкости  $\rho$ ,  $[\rho] = \text{Дж/м}^3 \cdot \text{К}$ .

Представляем  $\alpha_c$  как степенной комплекс этих величин

$$\alpha_c = c_0 \cdot d^{y_1} \cdot v^{y_2} \cdot (\rho)^{y_3} \cdot \lambda_c^{y_4} \cdot \nu_c^{y_5}.$$

Чтобы уравнение было правильно составлено, необходимо равенство одноименных размерностей левой и правой частей.

Поэтому пять показателей степеней должны подчиняться определенным связям, вытекающим из следующих равенств для одинаковых размерностей. Для получения критериев подобия путем анализа размерностей [1, стр. 34]

$$\begin{aligned} d [=] \text{м}^{\beta_{11}} \cdot \text{с}^{\beta_{21}} \cdot \text{Дж}^{\beta_{31}} \cdot \text{К}^{\beta_{41}}, \quad v [=] \text{м}^{\beta_{12}} \cdot \text{с}^{\beta_{22}} \cdot \text{Дж}^{\beta_{32}} \cdot \text{К}^{\beta_{42}}, \\ \rho [=] \text{м}^{\beta_{13}} \cdot \text{с}^{\beta_{23}} \cdot \text{Дж}^{\beta_{33}} \cdot \text{К}^{\beta_{43}}, \quad \lambda_c [=] \text{м}^{\beta_{14}} \cdot \text{с}^{\beta_{24}} \cdot \text{Дж}^{\beta_{34}} \cdot \text{К}^{\beta_{44}}, \\ \nu_c [=] \text{м}^{\beta_{15}} \cdot \text{с}^{\beta_{25}} \cdot \text{Дж}^{\beta_{35}} \cdot \text{К}^{\beta_{45}}, \quad \alpha_c [=] \text{м}^{\beta_{16}} \cdot \text{с}^{\beta_{26}} \cdot \text{Дж}^{\beta_{36}} \cdot \text{К}^{\beta_{46}} \end{aligned}$$

размерных величин можно составить и решить систему линейных однородных алгебраических уравнений:

$$\begin{cases} 1 \cdot y_1 + 1 \cdot y_2 - 3 \cdot y_3 - 1 \cdot y_4 + 2 \cdot y_5 - 2 \cdot y_6 = 0, \\ 0 \cdot y_1 - 1 \cdot y_2 + 0 \cdot y_3 - 1 \cdot y_4 - 1 \cdot y_5 - 1 \cdot y_6 = 0, \\ 0 \cdot y_1 + 0 \cdot y_2 + 1 \cdot y_3 + 1 \cdot y_4 + 0 \cdot y_5 + 1 \cdot y_6 = 0, \\ 0 \cdot y_1 + 0 \cdot y_2 - 1 \cdot y_3 - 1 \cdot y_4 + 0 \cdot y_5 - 1 \cdot y_6 = 0. \end{cases}$$

Составляем соответствующую систему вида:

$$\begin{cases} x_1 [=] \eta_1^1 \cdot \eta_2^0 \cdot \eta_3^0 \cdot \eta_4^0, \\ x_2 [=] \eta_1^1 \cdot \eta_2^{-1} \cdot \eta_3^0 \cdot \eta_4^0, \\ x_3 [=] \eta_1^{-3} \cdot \eta_2^0 \cdot \eta_3^1 \cdot \eta_4^{-1}, \\ x_4 [=] \eta_1^{-1} \cdot \eta_2^{-1} \cdot \eta_3^1 \cdot \eta_4^{-1}, \\ x_5 [=] \eta_1^2 \cdot \eta_2^{-1} \cdot \eta_3^0 \cdot \eta_4^0, \\ x_6 [=] \eta_1^{-2} \cdot \eta_2^{-1} \cdot \eta_3^1 \cdot \eta_4^{-1}. \end{cases}$$



Преобразуем три последних уравнения системы

$$\left\{ \begin{array}{l} x_1 [=] \eta_1^1 \cdot \eta_2^0 \cdot \eta_3^0 \cdot \eta_4^0, \\ x_2 [=] \eta_1^1 \cdot \eta_2^{-1} \cdot \eta_3^0 \cdot \eta_4^0, \\ x_3 [=] \eta_1^{-3} \cdot \eta_2^0 \cdot \eta_3^1 \cdot \eta_4^{-1}, \\ x_1^{-1} x_2^{-1} x_3^{-1} x_4 [=] 1, \\ x_1^{-1} x_2^{-1} x_3^0 x_5 [=] 1, \\ x_1^0 x_2^{-1} x_3^{-1} x_6 [=] 1. \end{array} \right.$$

Составляем систему степенных комплексов из трех последних уравнений:

$$\left\{ \begin{array}{l} x_1^{-1} x_2^{-1} x_3^{-1} x_4 [=] 1, \quad y_4 \\ x_1^{-1} x_2^{-1} x_3^0 x_5 [=] 1, \quad y_5 \\ x_1^0 x_2^{-1} x_3^{-1} x_6 [=] 1, \quad y_6 \\ \hline y_1 y_2 y_3 \end{array} \right.$$

На основании соответствий получаем решение:

$$\left\{ \begin{array}{l} y_1 = -1 \cdot y_4 - 1 \cdot y_5, \\ y_2 = -1 \cdot y_4 - 1 \cdot y_5 + 1, \\ y_3 = -1 \cdot y_4 + 1. \end{array} \right.$$

Тогда составленный нами степенной комплекс для  $\alpha_c$  примет вид:

$$\alpha_c = c_0 \cdot d^{-y_4 - y_5} \cdot v^{-y_4 - y_5 + 1} \cdot (\rho)_c^{-y_4 + 1} \cdot \lambda_c^{y_4} \cdot \nu_c^{y_5}.$$

Теперь у уравнения размерность левой части должна быть равна размерности правой части. Поэтому соответствующей комбинацией величин можно получить критерий подобия и критериальное уравнение, устанавливающее связь между ними.

Учитывая физику процесса, вероятнее всего искать зависимость критерия Нуссельта от критерия Рейнольдса и Прандтля.

Преобразуем найденное нами уравнение и после выделения критериев подобия получим три критерия подобия и форму критериального уравнения:

$$Nu = c_0 \cdot Re^{(-y_4 + y_5 + 1)} \cdot (Pr)^{-y_4 + 1}.$$

### Литература

1. Силин С.С. Теория подобия в приложении к технологии машиностроения: Учебное пособие /ЯПИ. – Ярославль, 1989.

# Кусочно-гладкие решения одного пространственно-распределенного уравнения с кубической нелинейностью

*Д. С. Костерин (ЯрГУ им. П.Г. Демидова, Ярославль)*

*Научный руководитель: д.ф.-м.н., профессор Глызин С. Д.*

Рассматривается уравнение

$$\frac{\partial \xi}{\partial t} = \xi - (\xi^3 - M(\xi^3)), \quad (1)$$

где  $t \geq 0$ ,  $x \in [0, 1]$ ,  $\xi = \xi(t, x)$  при каждом  $t$  является кусочно-гладкой по переменной  $x$  функцией,  $M(\xi) = \int_0^1 \xi(t, x) dx$ .

Данное уравнение рассматривается с краевыми условиями

$$\xi(t, x + 1) \equiv \xi(t, x), \quad (2)$$

$$M(\xi) = 0. \quad (3)$$

Краевые задачи такого типа возникают при изучении ряда систем дифференциальных уравнений с запаздыванием [1] и сингулярно-возмущенных задач параболического типа [2].

Краевая задача (1)–(3) имеет однопараметрическое семейство решений вида

$$\xi(t, x) = \begin{cases} 0, & x = 0, \\ \pm \frac{1-\alpha}{\sqrt{1-3\alpha+3\alpha^2}}, & 0 < x < \alpha, \\ 0, & x = \alpha, \\ \mp \frac{\alpha}{\sqrt{1-3\alpha+3\alpha^2}}, & \alpha < x < 1, \\ 0, & x = 1. \end{cases} \quad (4)$$

Исследовалась устойчивость этих решений. Справедлив следующий результат.

**Теорема 1.** Если  $\frac{1}{3} < \alpha < \frac{2}{3}$ , решение вида (4) краевой задачи (1)–(3) является устойчивым.

Разобьем отрезок  $[0, 1]$  точками  $\alpha_j, j = 0, \dots, n$ , причем  $\alpha_0 = 0, \alpha_n = 1$ . Тогда функция

$$\xi(x) = \begin{cases} 0, & x = \alpha_j, j = 0, \dots, n \\ (-1)^{j-1}, & \alpha_{j-1} < x < \alpha_j, j = 1, \dots, n \end{cases} \quad (5)$$

является решением уравнения (1) и удовлетворяет краевому условию (2).

При выполнении условия

$$2\alpha_1 - 2\alpha_2 + 2\alpha_3 + \dots + 2(-1)^n \alpha_n + (-1)^{n+1} = 0 \quad (6)$$

это решение удовлетворяет краевому условию (3).

Таким образом, функция вида (5) при выполнении условия (6) является решением краевой задачи (1)–(3).

**Теорема 2.** *Решения (5) краевой задачи (1)–(3) являются устойчивыми.*

Таким образом, было доказано наличие кусочно-постоянных решений краевой задачи (1)–(3) и исследована их устойчивость.

Исследование выполнено при финансовой поддержке поддержке Российского фонда фундаментальных исследований (грант № 18-29-10055).

### Литература

1. Григорьева Е. В., Кащенко С. А., “Медленные и быстрые колебания в модели оптико-электронного осциллятора с запаздыванием.” Доклады Академии наук, **484**, No. 1, 21–25 (2019).
2. Глызин С. Д., Кащенко С. А., Толбей А. О. “Взаимодействие двух волн в модели Ферми–Паста– Улама.” Моделирование и анализ информационных систем, **23**, No. 5, 548–558 (2016).

**Особенности бифуркаций автоколебательных  
решений нелинейного параболического  
уравнения с оператором поворота  
пространственного аргумента и  
запаздыванием**

**В. А. Куликов (ЯрГУ им. П.Г. Демидова, Ярославль)**

*Научный руководитель: д.ф.-м.н., профессор Кубышкин Е. П.*

Для дифференциального уравнения с запаздывающим аргументом

$$u_t(\rho, \phi, t) + u(\rho, \phi, t) = D\Delta_{\rho\phi}u(\rho, \phi, t) + K(1 + \gamma \cos(u_\theta(\rho, \phi, t - T))) \quad (1)$$

относительно функции  $u(\rho, \phi, t + s)$ , заданной в полярных координатах  $0 \leq \rho \leq R, 0 \leq \phi \leq 2\pi$  ( $R > 0$ ) и  $t \geq 0, -T \leq s \leq 0$  ( $T > 0$ ), в котором  $\Delta_{\rho\phi}$  — оператор Лапласа,  $u_\theta(\rho, \phi, t) \equiv u(\rho, (\phi + \theta) \bmod (2\pi), t)$  ( $0 \leq \theta < 2\pi$ ) — оператор поворота пространственного аргумента,  $D, K$  — положительные постоянные,  $0 < \gamma < 1$ , в области  $\bar{K}_R \times \mathbb{R}^+$ , где круг  $\bar{K}_R = \{(\rho, \phi) : 0 \leq \rho \leq R, 0 \leq \phi \leq 2\pi\}$ ,  $\mathbb{R}^+ = \{t : 0 \leq t < \infty\}$ , рассматривается начально-краевая задача

$$u_\rho(R, \phi, t) = 0, \quad u(\rho, 0, t) = u(\rho, 2\pi, t), \quad u_\phi(\rho, 0, t) = u_\phi(\rho, 2\pi, t),$$

$$u(\rho, \phi, t + s)|_{t=0} = u_0(\rho, \phi, s) \in H_0(K_R; -T, 0),$$

где  $H_0(K_R; -T, 0)$  пространство начальных условий.

В работе исследована динамика однородных состояний равновесия и их устойчивость в зависимости от параметров уравнения (1). В плоскости основных параметров управления (коэффициента усиления  $K$  и угла поворота  $\theta$ ) с использованием метода  $D$ -разбиений построены области устойчивости (неустойчивости) однородных состояний равновесия и исследованы возможные механизмы потери устойчивости. С использованием метода центральных многообразий и теории бифуркаций исследованы возможные бифуркации пространственно неоднородных автоколебательных решений, а также их устойчивость. Изучена динамика таких решений в окрестности границы области устойчивости в плоскости управляющих параметров.

Работа выполнена при финансовой поддержке РФФИ (код проекта 19-31-90133).

**Сравнение методов диагностики в реальном  
времени фазовой синхронизации  
автогенераторов по их нестационарным  
временным рядам**

*А. В. Курбако, Е. И. Боровкова, Д. Д. Кульминский  
(Саратовский филиал ИРЭ им. В.А. Котельникова  
РАН, СГУ им. Н.Г. Чернышевского, Саратов)*

*А. Н. Храмков, Д. М. Ежов (СГУ им. Н.Г.  
Чернышевского, Саратов)*

*Научный руководитель: д.ф.-м.н., доцент Караваев А. С.*

Анализ фазовой синхронизации автогенераторов по их экспериментальным временным реализациям является важной задачей современной нелинейной динамики [1]. Известно множество методов, предназначенных для исследования сигналов различной природы. Однако, для анализа нестационарных сигналов сложных систем, для которых характерно чередование интервалов синхронизации и несинхронного поведения, требуется разработка специализированных методов [2]. Примерами таких систем являются объекты биологической природы, в частности, элементы сердечно-сосудистой системы (ССС) человека.

В работе [3] был предложен метод диагностики интервалов фазовой синхронизации по нестационарным сигналам ССС. Метод основан на оценке угла наклона аппроксимирующей прямой сигнала разности мгновенных фаз  $\Delta\varphi(t)$  в скользящих окнах шириной  $b_\alpha$ . Участок временного ряда диагностируется как интервал фазовой синхронизации, если его длительность превышает  $l_\alpha$  и  $|\alpha_i| \leq |\alpha_0|$ , где  $\alpha_0$  — пороговое значение. Описанный метод имеет ряд недостатков, среди которых невозможность априорной оценки значений свободных параметров, квадратичная сложность вычисления и использование при расчётах арифметики с плавающей точкой.

Предлагаемый способ диагностики развивает подход, предложенный в [4]. Метод основан на усреднении значений разностей мгновенной разности фаз  $\Delta\varphi(t)$  в скользящих окнах

шириной  $w$ . Пусть  $h_i$  — усреднённое значение для  $i$ -го окна.  $i$ -е окно диагностируется как интервал фазовой синхронизации, если  $\Delta h_i = |h_i - h_{i-1}| \geq h$ , где  $h$  — пороговое значение. Так же оцениваются длины участков синхронных и несинхронных участков: участок синхронизации должен быть длиннее  $T_s$ , а участок несинхронного поведения должен быть длиннее  $T_n$ . Предложенный подход решает ряд проблем известного метода: параметры метода могут быть оценены из априорной информации об анализируемом сигнале, сложность вычисления снижена за счёт использования сложения вместо линейной аппроксимации в окне. Также предложенный метод позволяет использовать только простые целочисленные операции вместо операции с плавающей точкой.

Для настройки и проверки сравниваемых методов использовалась модель генерации тестовых разностей мгновенных фаз  $\Delta\varphi(t)$ , аналогичная [5]. Тестовый ряд повторяет статистические свойства экспериментальных временных рядов. Статистика, необходимая модели для построения тестовых реализаций, оценивалась по выборке экспериментальных данных, полученных от 23 испытуемых мужчин без признаков патологий ССС возрастом от 25 до 35 лет. Были сгенерированы 100 тестовых реализаций разностей мгновенных фаз длительностью 10000 секунд с частотой дискретизации 5 Гц.

Описанные методы сопоставлялись в ходе ROC-анализа. При переборе значений параметров методов выявлялся набор, обеспечивающий наилучшую чувствительность при фиксированном значении специфичности. Производилось сравнение значений AUC — интегральной характеристики ROC-кривой. AUC для известного метода равен 0.911, для предложенного метода — 0.910. Также сравнивались значения чувствительности и специфичности в точках сечения ROC-кривой диагональю: для известного метода TPR=0.842, FPR=0.158, а для предложенного метода TPR=0.837, FPR=0.163. Из результатов видно, что новый подход демонстрирует близкую к известному ранее методу чувствительность.

Для сопоставления вычислительных сложностей двух методов была оценена зависимость времени анализа от длительности реализации. Длительность временной реализации изменя-

лась в диапазоне от 500000 дискретных отсчётов до 5000000 с шагом 500000. Длительность окна для обоих методов составляла 100 дискретных отсчётов, а сдвиг окна — 1. При длине реализации в 500000 отсчётов известный метод, производит анализ за 7339.2 мс, а предложенный подход за 35.9 мс. При анализе реализации длительностью 5000000 отсчётов время работы предложенного метода составило 312.2 мс, а известного метода — 76742.2 мс. Также проводилось сопоставление времени работы методов в зависимости от длительности скользящего во временной области окна при фиксированной длине реализации в 50000 дискретных отсчётов. Размер скользящего окна изменялся в диапазоне от 50 до 500 дискретных отсчётов, сдвиг окна был равен 1. При ширине окна 50 отсчётов время работы предложенного метода составило 5.9 мс, а известного метода — 3626.6 мс. При ширине окна 500 время работы предложенного метода составило 48.4 мс, а известного метода — 34992.3 мс.

Таким образом, в ходе работы было проведено сравнение двух методов диагностики фазовой синхронизации автогенераторов по их нестационарным экспериментальным временным рядам.

Работа выполнена в рамках государственного задания и частично поддерживается Российским Фондом Фундаментальных Исследований, грант №20-02-00702, грантом Президента РФ для поддержки молодых кандидатов наук № МК-2723.2021.4

### Литература

1. Pikovsky A., Rosenblum M., Kurths J., Synchronization: A universal concept in nonlinear sciences, Cambridge University Press, Cambridge (2001).
2. Безручко Б. П., Смирнов Д. А., Математическое моделирование и хаотические временные ряды, ГосУНЦ “Колледж”, Саратов (2005).
3. Karavaev A. S., Prokhorov M. D., Ponomarenko V. I., Kiselev A. R., Gridnev V. I., Ruban E. I., Bezruchko B. P., “Synchronization of low-frequency oscillations in the human cardiovascular system,” *Chaos*, **19**, 033112 (2009).
4. Боровкова Е. И., Караваев А. С., Киселев А. Р., Шварц В. А., Мионов С. А., Пономаренко В. И., Прохоров М. Д., “Метод диагностики синхронизованности 0.1-Гц ритмов вегетативной регуляции сердечно-сосудистой системы в реальном времени,” *Анналы аритмологии*, **2**, No. 11, 129–136 (2014).
5. Боровкова Е. И., Караваев А. С., Пономаренко В. И., Прохоров М. Д. “Сопоставление методов диагностики фазовой синхронизованности по те-

стовым данным, моделирующим нестационарные сигналы биологической природы,” Известия Саратовского университета Новая серия Физика, **15**, No. 3, 36–42 (2015).

## Локальная динамика уравнения второго порядка с запаздыванием в производной

**И. Н. Масленников (ЯрГУ, Ярославль)**

*Научный руководитель: д.ф.-м.н., доцент Кащенко И.С.*

Рассмотрим дифференциальное уравнение второго порядка с запаздывающей обратной связью, которое представляет собой реализацию модифицированного уравнения Икеды с задержкой по времени [1]:

$$\varepsilon \frac{d^2 y}{dt^2} + \frac{dy}{dt} + \delta y = F \left( \frac{dy}{dt}(t - \tau) \right). \quad (1)$$

Здесь  $\varepsilon$  и  $\delta$  — малые и пропорциональные параметры:  $0 < \varepsilon \ll 1$ ,  $\delta = k\varepsilon$ ,  $\tau$  — 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. Рассматриваемая задача является сингулярно возмущенной.

Отметим, что в статье [2] рассмотрена похожая модель опто-электронного осциллятора, в которой параметр  $\delta$  не является малым.

Уравнение (1) представимо в виде (2) при разложении неоднородности в ряд Тейлора:

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

где  $\beta_1, \beta_2, \beta_3$  — некоторые постоянные.

Характеристический квазиполином линеаризованной в нуле задачи имеет вид

$$\varepsilon \lambda^2 + \lambda + k\varepsilon = \lambda \beta_1 e^{-\lambda}. \quad (3)$$

Показано, что при  $|\beta_1| < 1$  нулевое состояние равновесия устойчиво, а при  $|\beta_1| > 1$  — неустойчиво. В критических случаях



$\beta_1 = \pm 1$  характеристическое уравнение имеет бесконечное количество корней, стремящихся к мнимой оси при  $\varepsilon \rightarrow 0$ . Таким образом критические случаи имеют бесконечную размерность.

Для исследования поведения решений в случае  $\beta_1 = \pm 1$  построены квазинормальные формы [3] — специальные нелинейные уравнения параболического типа, не содержащие малых параметров, решения которых дают главную часть асимптотических по невязке равномерно по  $t \geq 0$  решений уравнения (1).

Работа выполнена при поддержке РФФИ (проект 18-29-10043).

### Литература

1. Larger L., Maistrenko Y., Penkovskyi B.. // Virtual Chimera States for Delayed-Feedback Systems Physical Review Letters, 2013. Vol. 111. pp. 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. Локальная динамика уравнений с большим запаздыванием // Журнал Вычислительной Математики и Математической Физики, 2008, Т. 48, №12, с. 2141-2150

## Моделирование движения транспортного потока

**М. А. Погребняк (ЯрГУ им. П.Г. Демидова,  
Ярославль)**

*Научный руководитель: д.ф.-м.н., доцент Кащенко И. С.*

В работе строится новая математическая модель движения транспортного потока, которая описывает движение  $N \in \mathbb{N}$  автомобилей. За  $x$  обозначено положение транспортного средства, а  $\dot{x}$  и  $\ddot{x}$  скорость и ускорение соответственно. Все автомобили считаются материальными точками, поэтому их внутренняя структура и внешние габариты не учитываются.

Все движение автомобиля разделено на две фазы: ускорение и торможение, причём в конкретный момент времени автомобиль либо разгоняется, либо тормозит. Для этого введена

релейная функцию вида:

$$R(\Delta x_n(t, \tau)) = \begin{cases} 1, & \text{если } \Delta x_n(t, \tau) > S + l, \\ 0, & \text{если } \Delta x_n(t, \tau) \leq S + l, \end{cases}$$

где  $\tau$  — время реакции водителя,  $\Delta x_n(t, \tau) = x_{n-1}(t - \tau) - x_n(t)$  — расстояние между соседними автомобилями,  $l$  — безопасное расстояние между автомобилями, а  $S$  — тормозной путь. Под тормозным понимается расстояние, которое проходит транспортное средство с момента срабатывания тормозной системы до полной остановки. Для расчёта данной величины использована часто встречающаяся в литературе формула:

$$S = \frac{v^2}{2\mu g},$$

где  $v$  — текущая скорость транспортного средства,  $\mu$  — коэффициент трения,  $g$  — ускорение свободного падения.

Первая фаза движения — разгон. Для описания разгона использован принцип, при котором преследующий автомобиль подстраивает свою скорость относительно впереди идущего. Вторая фаза движения — торможение. Торможение автомобиля зависит от разности расстояний и скоростей между самим автомобилем и впереди идущим автомобилем, а также от безопасной дистанции между ними.

Таким образом, на основе выше приведённых утверждений, была построена новая математическая модель движения транспортного потока, которая имеет вид системы дифференциальных уравнений с запаздыванием:

$$\begin{cases} \ddot{x}_n(t) = R(\Delta x_n(t, \tau)) \left[ a(\dot{x}_{n-1}(t - \tau) - \dot{x}_n(t)) \right] + \\ + (1 - R(\Delta x_n(t, \tau))) \left[ q \frac{\dot{x}_n(t) [\dot{x}_{n-1}(t - \tau) - \dot{x}_n(t)]}{x_{n-1}(t - \tau) - x_n(t) - l + k} \right], \\ x_n(t) = \lambda_0 - (n - 1)\lambda, \quad \dot{x}_n(t) = v_n, \\ \text{при } t \in [-\tau, 0] \text{ и } n \geq 2, \end{cases} \quad (1)$$

где  $\tau$  — время реакции водителя,  $\Delta x_n(t, \tau) = x_{n-1}(t - \tau) - x_n(t)$  — расстояние между соседними автомобилями,  $a > 0$  — коэффициент ускорения,  $q > 0$  — коэффициент торможения,  $l > 0$  —

безопасное расстояние между автомобилями,  $k > 0$  — корректирующий коэффициент,  $\lambda_0$  — положение первого автомобиля в начальный момент времени,  $\lambda$  — начальное расстояние между соседними автомобилями,  $v_n$  — начальная скорость автомобилей.

Полученная модель (1) описывает все автомобили потока. Для описания первого автомобиля ( $n = 1$ ) модель дополнена значениями  $x_0(t)$  и  $\dot{x}_0(t)$ . В качестве  $x_0(t)$  используется расстояние, которое должен проехать первый автомобиль  $x_0 = L$ . В качестве  $\dot{x}_0(t)$  в первом слагаемом используется максимальная желаемая скорость  $\dot{x}_0 = v_{max}$ , а во втором — минимальная желаемая скорость, то есть скорость до которой нужно сбросить свою текущую скорость  $\dot{x}_0 = v_{min}$ .

На рисунке 1 изображены графики изменения скорости и расстояния для нескольких автомобилей, двигающихся согласно модели (1).

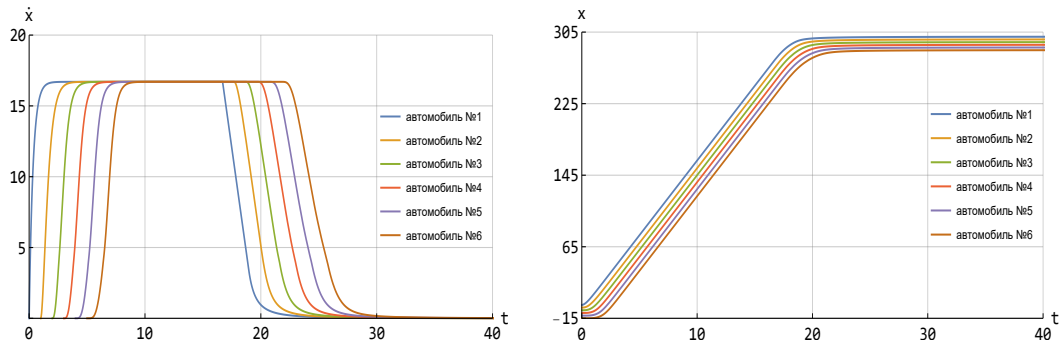


Рис. 1: Графики изменения скорости (слева) и расстояния (справа) для модели (1) с параметрами:  $a = 3$ ,  $q = 2$ ,  $v_{max} = 16.7$ ,  $v_{min} = 0$ ,  $\lambda = 3$ ,  $l = 3$ ,  $L = 300$ ,  $x_0 = 0$ ,  $v_n = 0$ ,  $g = 9.8$ ,  $\mu = 0.6$ .

Для модели (1) был проведён анализ устойчивости режима, при котором  $n$  автомобилей двигаются со скоростью  $v_{max}$  на расстоянии  $\Delta c_n = c_n - c_{n-1}$ , где  $c_n$  — положение  $n$ -ого автомобиля. Устойчивость такого режима зависит от знаков выражений:

$$d_n = -\tau v_{max} + c_n - c_{n-1} - l + k,$$

а также справедлива следующая теорема:

**Теорема 1.** Если для  $\forall n$  выполняется неравенство  $d_n > 0$ , то режим устойчив. Если хотя бы при одном каком-то  $i$  выполняется неравенство  $d_i \leq 0$ , то режим неустойчив.

Из теоремы 1 следует, что если все автомобили потока движутся на довольно большом расстоянии друг от друга, то такой режим движения устойчив. Устойчивость теряется при увеличении скорости  $v_{max}$ , времени реакции водителя  $\tau$ , безопасного расстояния между автомобилями  $l$ , или при сокращении расстояния между двумя соседними автомобилями  $\Delta c_n$ .

## Нормальная форма дифференциального уравнения с запаздыванием и непостоянными коэффициентами

*Е. М. Татанова (ЯрГУ им. П.Г. Демидова,  
Ярославль)*

*Научный руководитель: д.ф.-м.н., доцент Кащенко И. С.*

Рассмотрим дифференциальное уравнение с запаздыванием

$$\dot{x} + x = ax(t - T) + f(x), \quad (1)$$

где  $a$  — произвольный параметр,  $f(x)$  — функция, имеющая в нуле порядок выше первого, а параметр  $T$ , характеризующий запаздывание, является достаточно большим, т.е.

$$T \gg 1.$$

Поставим задачу исследовать решения уравнения (1) в окрестности  $x = 0$  и построить их асимптотические решения при достаточно большом  $T$ .

Через  $\varepsilon$  обозначим малый параметр  $\varepsilon = T^{-1}$ . Тогда

$$0 < \varepsilon \ll 1.$$

С помощью замены времени приходим к эквивалентному уравнению

$$\varepsilon \dot{x} + x = ax(t - 1) + f(x). \quad (2)$$

При достаточно малом  $\varepsilon$  будем исследовать решения уравнения (2) в окрестности  $x = 0$  и построим их асимптотические приближения.

Характеристический квазиполином линеаризованного на нулевом состоянии уравнения имеет вид:

$$\varepsilon\lambda + 1 = ae^{-\lambda T}. \quad (3)$$

Будем исследовать случай, когда существуют корни (3) с нулевой вещественной частью и нет корней с положительной. Это происходит при  $a = \pm 1$  [1].

Разложим функцию  $f(x)$  в ряд Тейлора в окрестности нуля:

$$\varepsilon\dot{x} + x = ax(t-1) + f_2x^2 + f_3x^3 + o(x^3). \quad (4)$$

Изучим динамику уравнения (4) в случае, когда значение параметра  $a$  близко к критическому, т.е.

$$a = 1 + \varepsilon^2 a_1(t).$$

Будем рассматривать  $a_1$  как почти периодическую функцию.

При  $a = 1$  характеристическое уравнение имеет бесконечное количество корней вида  $2\pi ki + o(1)$ .

Положим в (4)

$$x = \varepsilon^2 u(\tau, r) + \varepsilon^4 x_1(\tau, r) + \dots, \quad (5)$$

где  $\tau = \varepsilon^2 t$ ,  $r = (1 - \varepsilon + \varepsilon^2)t$ , а функция  $x_1(\tau, r)$  предполагается почти периодической по второму аргументу.

Пусть  $a_{1*}(r)$  — функция, содержащая в себе те слагаемые  $a_1(r)$ , частоты которых кратны  $2\pi$ . Тогда для определения  $u(\tau, r)$  получаем параболическое уравнение вида

$$\frac{\partial u}{\partial \tau} = \frac{1}{2} \frac{\partial^2 u}{\partial r^2} + a_{1*}(r) \cdot u(\tau, r) + f_2 u^2, \quad (6)$$

с периодическими краевыми условиями

$$u(\tau, r) \equiv u(\tau, r + 1). \quad (7)$$

Задача (6), (7) является квазинормальной формой для исходного уравнения (1). Справедлива следующая теорема.

**Теорема 1.** Пусть  $u(\tau, r)$  — ограниченное при  $\tau \rightarrow \infty$  решение задачи (6), (7), тогда исходное уравнение (4) имеет асимптотическое по невязке с точностью до  $O(\varepsilon^4)$  равномерно по  $t \geq 0$  решение вида

$$x = \varepsilon^2 u(\varepsilon^2 t, (1 - \varepsilon + \varepsilon^2)t) + \dots,$$

Изучим динамику уравнения (4) в случае, когда значение параметра  $a$  близко к критическому значению  $-1$ , т.е.

$$a = -1 + \varepsilon^2 a_1(t).$$

Аналогично рассмотрим  $a_1$  как почти периодическую функцию (тригонометрический полином).

Здесь при  $a = -1$  характеристическое уравнение имеет корни вида  $\pi(2k + 1)i + o(1)$ .

Положим в (4)

$$x = \varepsilon u(\tau, r) + \varepsilon^2 x_2(\tau, r) + \varepsilon^3 x_3(\tau, r) + \dots, \quad (8)$$

где  $\tau = \varepsilon^2 t$ ,  $r = (1 - \varepsilon + \varepsilon^2)t$ , а функции  $x_2(\tau, r)$  и  $x_3(\tau, r)$  предполагаются почти периодическими по второму аргументу.

Приравниваем коэффициенты при одинаковых степенях малого параметра  $\varepsilon$ , из равенства при  $\varepsilon^2$  находим

$$x_2 = \frac{f_2}{2} u^2.$$

Из разрешимости уравнения при  $\varepsilon^3$  получаем для определения  $u(\tau, r)$  уравнение параболического типа

$$\frac{\partial u}{\partial \tau} = \frac{1}{2} \frac{\partial^2 u}{\partial r^2} - a_{1*}(r) \cdot u(\tau, r) + (f_2^2 - f_3) u^3 \quad (9)$$

с антипериодическими краевыми условиями

$$u(\tau, r) = -u(\tau, r + 1). \quad (10)$$

Краевая задача (9), (10) является квазинормальной формой для исходного уравнения (4) при  $a$ , близком к  $-1$ .

**Теорема 2.** Пусть  $u(\tau, r)$  — ограниченное при  $\tau \rightarrow \infty$  решение задачи (9), (10). Тогда исходное уравнение (4) имеет асимптотическое по невязке с точностью до  $O(\varepsilon^3)$  равномерно по  $t \geq 0$  решение вида

$$x = \varepsilon u(\varepsilon^2 t, (1 - \varepsilon + \varepsilon^2)t) + \varepsilon^2 x_2(\varepsilon^2 t, (1 - \varepsilon + \varepsilon^2)t).$$

### Литература

1. Кащенко И.С. “Локальная динамика уравнений с большим запаздыванием”, Журнал Вычислительной Математики и Математической Физики, **48**, No. 12., 2141-2150 (2008).

Научное издание

**ПУТЬ В НАУКУ. МАТЕМАТИКА**

**Тезисы докладов конференции**

Всероссийская молодежная конференция  
11 – 16 мая 2021 г., Ярославль

Компьютерный набор – авторы  
Компьютерная верстка – А. А. Кащенко

Подписано в печать 14.09.2021. Формат 60×84/16.  
Усл. печ. л. 5,1. Уч.-изд. л. 3.0.  
Тираж 22 экз. Заказ

Оригинал-макет подготовлен на математическом  
факультете ЯрГУ

Ярославский государственный университет им. П. Г. Демидова  
Математический факультет  
150008, Ярославль, ул. Союзная, 144