

16+



# XXIII Всероссийская студенческая научно-практическая конференция Нижневартовского государственного университета



Часть 4

Информационные технологии.  
Математика

Нижневартовск, 6-7 апреля 2021

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Нижевартовский государственный университет»

**XXIII Всероссийская студенческая  
научно-практическая конференция  
Нижевартовского  
государственного университета**

**Часть 4**

*ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ. МАТЕМАТИКА*

*Нижевартовск,  
6-7 апреля 2021 года*

*Под общей редакцией Д.А. Погоньшева*

Нижевартовск  
НВГУ  
2021

ББК 72я43  
Д 25

16+

Печатается по решению Ученого совета  
ФГБОУ ВО «Нижевартовский государственный университет»  
(протокол № 1 от 26.01.2020 г.)

Под общей редакцией: Д.А. Погоньшев, канд. биол. наук, Нижевартовский  
государственный университет.

Редакторы: Е.С. Овечкина, И.С. Анцева, Е.В. Вилявина, Д.В. Вилявин.

Д 25 **XXIII Всероссийская студенческая научно-практическая конференция  
Нижевартовского государственного университета (г. Нижевартовск, 6-7 апреля  
2021 г.) / Под общей ред. Д.А. Погоньшева. Ч. 4. Информационные технологии. Математика.  
Нижевартовск: Издательство НВГУ, 2021. 229 с.**

**ISBN 978-5-00047-598-0**

ББК 72я43



Тип лицензии CC, поддерживаемый журналом: Attribution 4.0 International (CC BY 4.0).

**ISBN 978-5-00047-598-0**

© НВГУ, 2021

ISBN 978-5-00047-598-0



## ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ. МАТЕМАТИКА

УДК 330.43

Севальников А.С.

Научный руководитель: Мамонов С.С., д-р физ.-мат. наук  
Рязанский государственный университет  
им. С.А. Есенина, г. Рязань, Россия

### ПРОГНОЗИРОВАНИЕ ПО ДВУХПАРАМЕТРИЧЕСКОЙ ТРЕНДОВОЙ МОДЕЛИ

Рассмотрим наблюдения временного ряда  $y_t$ , при  $t = \overline{1, n}$ . В качестве модели будет рассматриваться уравнение:

$$y_t = \theta_1 + \theta_2 \varphi(t) \quad (1)$$

Два эксперта высказали суждения: «Значения временного ряда в момент времени  $t^{(1)} > n$  и окажется не больше  $d^{(1)}$ » и «знание временного ряда  $y(t)$  в момент  $t^{(1)} > n$  периода упреждения окажется не менее  $c^{(1)}$ ». Суждения экспертов задаются неравенствами:

$$c^{(1)} \leq \theta_1 + \theta_2 \varphi(t^{(1)}) \leq d^{(1)} \quad (2)$$

Для определения  $\theta_1$  и  $\theta_2$  решают задачу нахождения:

$$\min S(\theta_1, \theta_2) = \sum_{t=1}^n (y_t - (\theta_1 + \theta_2 \varphi(t)))^2 \quad (3)$$

при ограничениях (2).

Построение модели (1) производится следующим образом. Сначала необходимо минимизировать функцию  $S(\theta_1, \theta_2)$  без учета ограничений (2), используя метод наименьших квадратов (МНК). Если построенная модель определяет значение  $(\theta_1 + \theta_2 \varphi(t^{(1)}))$ , удовлетворяющее неравенствам (2), что проверяется простой подстановкой, то это и будет являться оптимальным решением. Если для построенной модели не выполняются неравенства (2), то рассматривают два уравнения:

$$\theta_1 + \theta_2 \varphi(t^{(1)}) = c^{(1)} \quad (4)$$

$$\theta_1 + \theta_2 \varphi(t^{(1)}) = d^{(1)} \quad (5)$$

и решают две задачи минимизации  $S(\theta_1, \theta_2)$  с учетом равенства (4) и ту же задачу минимизации  $S(\theta_1, \theta_2)$  с учетом (5). Из полученных решений выбирают то, при котором значение функции  $S(\theta_1, \theta_2)$  минимально.



Задача минимизации (3) при ограничениях неравенства в виде равенства (4) относится к классу задач условный экстремум (в нашем случае минимум). Ее удастся решить, применяя метод множителей Лагранжа.

*Правило Лагранжа.* Пусть даны функция  $u = f(x, y, z)$  и уравнение связи  $\varphi(x_1 y_1 z) = 0$ , известно что в некоторой точке  $M_0(x_0, y_0, z_0)$  функция  $f$  имеет экстремум, для которого  $\varphi(x, y, z) = 0$ , тогда для функции:

$$\phi(x, y, z) = f(x, y, z) + \lambda\varphi(x, y, z), \quad (6)$$

справедливы соотношения

$$\phi'_x = \phi'_y = \phi'_z = \phi'_\lambda = 0 \quad \text{в точке } M_0. \quad (7)$$

*Замечание.* Уравнения (7) определяют наблюдение условия условного экстремума.

В нашем случае для функции  $S(\theta_1, \theta_2)$  и уравнение связи (4) составили функцию Лагранжа:

$$y(\theta_1, \theta_2, \lambda) = S(\theta_1, \theta_2) + \lambda(\theta_1 + \theta_2\varphi(t^{(1)}) - c^{(1)}) \quad (8)$$

Для (8) найдем частное произведения и приравняем к нулю.

$$\left\{ \begin{array}{l} \frac{\partial y}{\partial \theta_1} = 2 \sum_{t=1}^n (y_t - (\theta_1 + \theta_2\varphi(t^{(1)}))) \cdot (-1) + \lambda = 0 \\ \frac{\partial y}{\partial \theta_2} = 2 \sum_{t=1}^n (y_t - (\theta_1 + \theta_2\varphi(t^{(1)}))) \cdot (-\varphi(t)) + \lambda\varphi(t^{(1)}) = 0 \\ \frac{\partial y}{\partial x} = \theta_1 + \theta_2\varphi(t^{(1)}) - c^{(1)} = 0 \end{array} \right. \quad (9)$$

Систему (9) запишем в виде:

$$\left\{ \begin{array}{l} n \cdot \theta_1 + \theta_2 \sum_{t=1}^n \varphi(t) + \frac{1}{2} \lambda = \sum_{t=1}^n y_t \\ \theta_1 \cdot \sum_{t=1}^n \varphi(t) + \theta_2 \sum_{t=1}^n \varphi(t) + \frac{1}{2} \varphi(t) \cdot \lambda = \sum_{t=1}^n y_t \varphi(t) \\ \theta_1 + \theta_2\varphi(t^{(1)}) = c^{(1)} \end{array} \right. \quad (10)$$

$$\text{Пусть } A = \begin{pmatrix} n & \sum \varphi(t) & 1/2 \\ \sum \varphi(t) & \sum \varphi^2(t) & 1/2 \varphi(t^{(1)}) \\ 1 & \varphi(t^{(1)}) & 0 \end{pmatrix}, \theta_\lambda = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_1 \end{pmatrix}, b = \begin{pmatrix} \sum y_t \\ \sum y_t \varphi(t) \\ c^{(1)} \end{pmatrix} \quad (11)$$

Тогда система (10) имеет вид:

$$A \cdot \theta_\lambda = b \quad (12)$$

Решение уравнения (12) имеет вид  $\theta_\lambda = A^{-1}b \Rightarrow \theta_1, \theta_2 \Rightarrow \hat{y}_t = \theta_1 + \theta_2 \varphi(t)$

Аналогично, применяя метод множителей Лагранжа. Определяются оценки параметров  $\theta_1, \theta_2$  при ограничении (5). Далее составляя полученные решения в критерий (3), выбираем то, при котором функция (3) будет меньше. Это и будут искомые оценки параметров тренда с учетом экспертного суждения.

*Пример.* Известны наблюдения временного ряда.

$$\begin{array}{c|c|c|c|c} y_t & 1 & 2 & 3 & 4 \\ \hline t & 1 & 2 & 3 & 4 \end{array} \quad n=4. \text{ Модель (1) в виде } \hat{y}_t = a + bt \Rightarrow \theta_1 = a, \theta_2 = b \Rightarrow \varphi(t) = t$$

Предположим, что  $t^{(1)} = 5$  и суждения экспертов задаются неравенствами:

$$2 \leq \theta_1 + \theta_2 \varphi(t^{(1)}) \leq 3 \Leftrightarrow 2 \leq \theta_1 + \theta_2 \cdot 5 \leq 3 \quad (13)$$

С помощью МНК без учета неравенств (13) построим линейную регрессию

$$\hat{y}_t = a + bt \Rightarrow a = 0, b = 1 \Rightarrow \hat{y}_t = (0 + t), \theta_1 = 0, \theta_2 = 1, c^{(1)} = 2, d^{(1)} = 3$$

Неравенство (3) примет вид:

$$2 \leq 0 + 1t^{(1)} \leq 3 \Leftrightarrow 2 \leq 5 \leq 3 \Leftrightarrow \text{Ложь}$$

Для построения  $\hat{y}_t = \theta_1 + \theta_2 t$  решаем две задачи минимизации с учетом равенств (4) и (5). Для (3) с ограничением (4) получим систему (10) в следующем виде:

$$\begin{cases} 4a + 10b + 0,5\lambda = 10 \\ 10a + 30b + 2,5\lambda = 30 \\ a + 5b = 2 \end{cases} \Leftrightarrow \begin{cases} -10b + 0,5\lambda = 2 \\ -20b + 2,5\lambda = 10 \\ a + 5b = 2 \end{cases} \Leftrightarrow \begin{cases} -10b + 0,5\lambda = 2 \\ 1,5\lambda = 6 \\ a + 5b = 2 \end{cases} \Leftrightarrow \begin{cases} b = 0 \\ \lambda = 4 \\ a = 2 \end{cases} \quad (14)$$

Используя (14) получили модель  $\hat{y}_t^c = 2 + 0 = 2$ , при этом  $2 \leq \theta_1 + \theta_2 \varphi(t^{(1)}) \leq 3$ . Для  $\hat{y}_t^c = 2$ . Найдем  $S(\theta_1, \theta_2) = \sum_{t=1}^4 (yt - 2) = 1 + 0 + 1 + 4 = 6$ .

Для (3) с ограничением (5) получаем систему (10) в следующем виде:

$$\begin{cases} 4a + 10b + 0,5\lambda = 10 \\ 10a + 30b + 2,5\lambda = 30 \\ a + 5b = 3 \end{cases} \Leftrightarrow \begin{cases} -10b + 0,5\lambda = 2 \\ -20b + 2,5\lambda = 10 \\ a + 5b = 3 \end{cases} \Leftrightarrow \begin{cases} -10b + 0,5\lambda = -2 \\ 1,5\lambda = 4 \\ a + 5b = 3 \end{cases} \Leftrightarrow \begin{cases} b = \frac{1}{10}(4/3 + 2) = 1/3 \\ \lambda = 8/3 \\ a = 4/3 \end{cases} \quad (15)$$

Используя (15), получили модель  $\hat{y}_t^d = 4/3 + 1/3 t$ , при этом выполняется неравенство

$$2 \leq \theta_1 + \theta_2 \varphi(t^{(1)}) = 4/3 + 1/3 \cdot 5 = 3 \leq 3.$$

С помощью модели  $\hat{y}_t^d$  найдем теоретические значения.

(16)

$$\begin{array}{c|c|c|c|c} y_t & 5/3 & 6/3 & 7/3 & 8/3 \\ \hline t & 1 & 2 & 3 & 4 \end{array}$$

Для  $\hat{y}_t^d = 4/3 + 1/3 t$  найдем  $S(\theta_1, \theta_2) = (y_t - \hat{y}_t^d)^2 = 4/9 + 0 + 4/9 + 16/9 = 24/9 < 6$ . Лучшей модель является взаимосвязь  $\hat{y}_t^d = 4/3 + 1/3 t$ . При этом  $\hat{y}_t^d(6) = 10/3$  - прогноз временного ряда  $t = 6$ .

*Геометрическая интерпретация.* В системе координат  $\theta_1, \theta_2 = x, y$ , построим линии

$$l_1: \theta_1 + \theta_2 \cdot 5 = 3 \Rightarrow y = -x/5 + 3/5, \text{ (прямая)}$$

$$l_2: \theta_1 + \theta_2 \cdot 5 = 2 \Rightarrow y = -x/5 + 2/5, \text{ (прямая)}$$

$$L_1: S(\theta_1, \theta_2) = C_1 = 6, \text{ (эллипс)}$$

$$L_2: S(\theta_1, \theta_2) = C_2 = 24/9, \text{ (эллипс)}$$

$$L: S(\theta_1, \theta_2) = C \Leftrightarrow \sum(y_t^2 - 2y_t(\theta_1 + \theta_2 t) + (\theta_1 + \theta_2 t)^2) = \sum y_t^2 + 2\theta_1 \sum y_t - 2\theta_2 \sum y_t t + 4\theta_1^2 + 2\theta_1\theta_2 \sum t + \theta_2^2 \sum t^2 = \theta_2^2 \sum t^2 + 2\theta_2(\theta_1 \sum t - \sum y_t t) + (\sum y_t^2 - 2\theta_1 \sum y_t + 4\theta_1^2) = C$$

$$y^2 \cdot 30 + 2y(10x + 30) + (4x^2 + 20x + 30) = C$$

$$L: y = 1/30 \cdot (30 + 10x \pm \sqrt{(10x - 30)^2 - 60(2x^2 - 10x + 15 - C/2)})$$

$$C = C_1 = 6$$

$$C = C_2 = 24/9$$

*Программа 1:* Экспертные регрессии

Определение параметров системы

> restart:

Параметры

> CS:=6:

Графики

```
> plot([-x/5+3/5,-x/5+2/5,(30-10*x+sqrt((10*x-30)^2-60*(2*x^2-10*x+15-CS/2)))/30,(30-10*x-sqrt((10*x-30)^2-60*(2*x^2-10*x+15-CS/2)))/30,(30-10*x+sqrt((10*x-30)^2-60*(2*x^2-10*x+15-24/18)))/30,(30-10*x-sqrt((10*x-30)^2-60*(2*x^2-10*x+15-24/18)))/30],x=-4.0..4.0,y=-0.3..2.2,style=line,color=[black,black,red,red,blue,blue]);
```

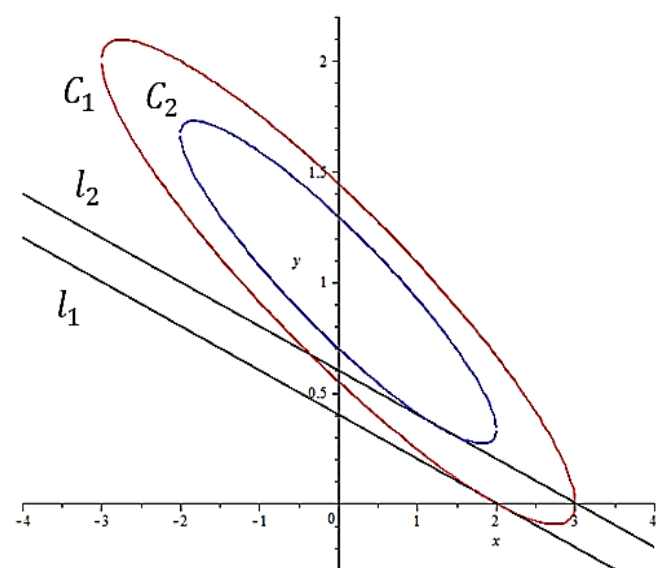


Рис. Графическая интерпретация функции  $S(\theta_1, \theta_2)$

### Литература

1. Домбровский В.В. Эконометрика. М., 2004. 342 с.
2. Орлов А.И. Организационно-экономическое моделирование. Ч. 2: Экспертные оценки. М., 2011. 486 с.
3. Данелян Т. Я. Формальные методы экспертных оценок // Экономика, статистика и информатика. Вестник УМО. 2015. №1. С. 183-187.

© Севальников А.С., Мамонов С.С., 2021

УДК 004.428.4

**Бобокало Д.С.**

*Научный руководитель:* **Кондратьева Т.Н.**, канд. техн. наук  
Донской государственной технической университет,  
г. Ростов-на-Дону, Россия

## **ФАЛЬШИВЫЕ НОВОСТИ. FAKE NEWS**

В современном мире объем информации, охваченный монетизацией неуклонно растет. Большая часть информации является результатом финансовых операций корпораций и объединений, которые борются за свои интересы и сферы влияния. Из этого следует, что таким компаниям выгодно создавать новости, которые смогут повлиять на выбор отдельных групп людей в свою пользу. Таким образом гражданам становится все труднее найти информацию, которая не была бы создана в чьих-то интересах, а также оценить степень ее достоверности.

Недобросовестные средства массовой информации могут манипулировать с помощью заведомо ложной, искаженной информацией, что может стать причиной общей дестабилизации и появлению финансовых рисков. Проблема настолько обширна, что небольшая группа лиц может полностью выдавить конкурента с рынка.

Так, например, на малый и средний бизнес конкуренты могут повлиять, оставляя негативные и необоснованные отзывы на торговых площадках. В следствии чего средняя оценка товаров и самого магазина может снизиться, что приведет к падению продаж, вплоть до закрытия магазина.

Современные новостные ресурсы имеют большую аудиторию по охвату, как можно увидеть на рисунке 1 [6].

Из исследования следует, что первые три сайта являются агрегаторами.

Такие сайты сами не производят новости, а лишь собирают их из других источников. Очевидно, что перед публикацией каждая статья должна проходить проверку, но на деле агрегаторов в первую очередь интересует лишь прибыль. Из этого следует, что в новостную выборку могут попадать фальшивые новости.

В следствии чего, по данным опроса «Pew Research», проведенного в 2016 году, 64% американцев заявили, что фальшивые новости вызвали много путаницы. Результаты опроса приведены на рисунке 2.

Приведенный выше опрос был проведен во время президентской компании в США [11]. Основной мишенью стала кандидат в президенты Хиллари Клинтон. Одним из последствий стала теория “Pizzagate”, согласно которой сторонники Хиллари связано с тайной организацией.

№	Ресурс	Среднемесячная аудитория* (тыс. чел)		Динамика	Перемещение
		янв-сент 2012	янв-сент 2013		
1	Яндекс Новости	13420,0	13914,1	3,7% ↑	0
2	Mail.ru Новости	11689,1	11902,3	1,8% ↑	0
3	Ria.ru	7704,5	9110,5	18,2% ↑	+ 2
4	Rbc.ru	9302,3	8650,1	7% ↓	- 1
5	Kp.ru	8147,3	8196,9	0,6% ↑	- 1
6	Vesti.ru	6653,0	6541,7	1,7% ↓	0
7	Lenta.ru	5227,3	6016,9	15,1% ↑	+ 1
8	Rambler Новости	6474,5	5822,1	10% ↓	- 1
9	Rg.ru	4213,1	5739,7	36,2% ↑	0
10	Gazeta.ru	3057,7	4589,8	50,1% ↑	+ 4
11	Aif.ru	3269,4	4372,8	33,7% ↑	0
12	Smi2.ru	3129,5	4226,6	35,6% ↑	+ 1
13	Utro.ru	3803,9	3672,0	3,5% ↓	- 3
14	Newsru.com	3179,9	3104,2	2,4% ↓	- 2
15	Rbcdaily.ru	2395,5	2962,3	23,6% ↑	+ 2
16	Interfax.ru	2813,5	2957,1	5,1% ↑	0
17	Itar-tass.com	1680,2	2885,2	71,7% ↑	+ 3
18	Echomsk.ru	2851,4	2860,6	0,3% ↑	- 3
19	Mk.ru	2071,2	2679,7	29,4% ↑	0
20	Kommersant.ru	2133,9	2666,8	25% ↑	- 2

Рис. 1. Рейтинг новостных ресурсов Рунета

	A great deal	Some	Not much	No
	%	%	%	%
Total	64	24	6	4
Men	61	25	8	5
Women	68	23	5	4
Ages 18-29	67	26	4	2
30-49	66	21	9	4
50-64	64	23	6	6
65+	58	29	6	5
High school or less	61	23	8	7
Some college	67	24	5	3
College+	67	26	5	1
<\$30,000	58	26	8	7
\$30,000- \$74,999	65	27	5	2
\$75,000+	73	19	5	2
White	65	25	6	3
Black	67	18	11	4
Hispanic	61	19	6	10
Republican	57	28	9	5
Democrat	64	25	4	6
Independent	69	23	6	1

Рис. 2. Влияние, которое оказывают фальшивые новости



Под понятием фальшивой новости могут попадать лживые статьи. Также могут быть представлены реальные факты, но приведена их неверная интерпретация, распространены псевдонаучные новости и мнения отдельно взятых людей, чаще всего авторов.

Так же фальшивые новости распространяются с огромной скоростью. Так как они содержат более вызывающие заголовки, тем самым собирают на себе основное внимание читателей.

Государственное регулирование ложных новостей несет проблему, с которой сталкиваются законодатели – это баланс между свободами человека и сохранением общественных интересов. Даже в тех странах, где конституция запрещает принимать закон, ограничивающий свободу слова, сегодня предпринимаются попытки урегулировать недостоверные новости, что воспринимается неоднозначно.

Проблема ложных новостей распространена и в Индонезии, где есть так называемые фабрики фальшивых новостей. Они повлияли на выборы 2012 года, после чего правительство страны усилило действующее законодательство не только введением новых положений, но и выпуском руководящих принципов, которые способствуют их реализации. Например, в 2017 году здесь было создано полицейское мультимедийное бюро для борьбы с дезинформацией. Аналогичная структура сегодня существует и в Чехии.

Некоторые государства пошли другим путем: они предпочитают применять не законодательные меры, а просто проверяют факты и противодействуют поддельным порталам. Например, в Малайзии создан веб-сайт для проверки информации, а в Катаре запустили онлайн-ресурс для борьбы с кампанией по дезинформации.

Сегодня существует несколько решений, которые могли бы помочь с фильтрацией, тем самым снизить влияние от недостоверных источников на общественное мнение. Основным являются два метода, с помощью которых можно определить тональность текста: технологии контент-анализа в компьютерной лингвистике на основе машинного обучения и метода, основанного на словарях и правилах. Возможно также использование этих методов вместе.

В методе классификации по словарям используются готовые, заранее составленные словари тональности, в соответствии с которыми, высказывание разбивается на отдельные слова, которым дается оценка [1, с. 129]. Оценка может быть позитивной, негативной и нейтральной. Итоговая тональность высказывания определяется суммой тональностей каждого слова.

Сложность данного метода состоит в создании словарей для каждой предметной области, что существенно увеличивает трудозатраты [8]. Так, например, слова, используемые для одной предметной области, могут быть для нее позитивными, в то время как для другой предметной области негативными.

Существует несколько крупных готовых словарей:

WordNet-Affect – словарь, позволяющий оценивать чувства, эмоции, отношения, ощущения. Всего содержит около двухсот предметных областей. Позволяет получить эмоциональные метки: позитивная, негативная, нейтральная и неоднозначная.

Данный продукт изначально создавался для английского языка. Позже, учеными была добавлена поддержка русского и румынского языка. Ресурс доступен для исследовательских целей.

SenticNet – проект разработан в 2010 г. Главным назначением является упрощение машинного распознавания эмоциональной информации, передаваемой с помощью естественного языка. Главным отличием от WordNet-Affect является возможность выявлять смысловую составляющую, так как анализ происходит на семантическом уровне.

Методы, на основе машинного обучения с учителем, сегодня применяются все чаще [10, с. 31]. Суть таких методов заключается в обучении классификатора на заранее подготовленных текстах с известной тональностью и последующее использование классификатора для анализа тональности новых текстов.

В результате анализа тональности, текст может быть классифицирован по нескольким категориям:

Классификация по бинарной шкале позволяет дать оценку тексту позитивная или негативная. Проблема данной классификации состоит в том, что не всегда можно однозначно определить эмоциональную окраску [2, с. 19].

Классификация объективность/субъективность – является более сложным способом и позволяет отнести текст к одному из двух классов: субъективный или объективный. Основывается на субъективности слов и фраз. Проблема данной классификации заключается в том, что субъективность может сильно зависеть от контекста. Результат оценки может сильно ухудшиться, в зависимости от объема объективных предложений [7, с. 43].

Более подробная модель данной классификации является анализом на основе функции/аспекта. Данная модель рассматривает мнения и настроения, заложенные в тексте.

Используются следующие методы обучения: обучение, основанное на правилах и словарях, обучение с учителем, обучение без учителя, обучение на теоретико-графовых моделях.

Для оценки тональности используются два параметра: точность и полнота. Полнотой является отношение количество верно определенных мнений к общему количеству мнений. Точностью является отношение верно определенных мнений к общему числу мнений, найденных алгоритмом.

Существует несколько платформ на основе машинного обучения для решения поставленной задачи [3, с. 100]:

Azure Machine Learning предоставляет инструменты, которые можно использовать в том числе и для анализа тональности текста [9]. Azure предоставляет web интерфейс для управления блоками и их параметрами, а также поддерживает скрипты на языке R. Для обучения классификатора используется заранее подготовленная выборка текстов с тональностями. Оценка определяется следующим образом: -10 (негативная), +10 (позитивная). После обучения на небольшой выборке и использовании несложных подходов можно получить точность около 70%.

Разработчик Fast.io предложил использование модели ULMFIT в 2018 г. Суть состоит в изменении способа обучения классификатора [4]. Для его обучения используется два шага: предобучающая языковая модель и дообучение модели [5, с. 8]. Языковая модель должна уметь предсказывать последующие слова в последовательности. Для обучения предобучающей модели используется набор коротких сообщений из Twitter RuTweetCorp. Набор содержит около 115000 позитивных, около 112000 негативных и около 17000000 неклассифицированных твитов.

После обучения классификатор определяет тональность текста с точностью около 80%.

Помимо фальшивых новостей, которые могут негативно сказываться на многих аспектах жизни, сегодня существуют и средства, позволяющие анализировать и фильтровать отдельные высказывания и новости. Применение таких средств поможет значительно уменьшить финансовые риски или риски, связанные с общеполитической ситуацией.

Дальнейшее развитие темы, рассмотрение других алгоритмов, а также применение одного из них будет представлено в следующих статьях.

### Литература

1. Большакова Е.И., Воронцов К.В., Ефремова Н.Э. Автоматическая обработка текстов на естественном языке и анализ данных. М., 2017. 269 с.
2. Вьюгин В.В. Математические основы машинного обучения и прогнозирования. М., 2013. 391 с.
3. Гифт Н. Прагматичный ИИ. Машинное обучение и облачные технологии. СПб., 2019. 304 с.
4. Замощин О. Анализируем тональность текстов с помощью Fast.ai. <https://clck.ru/UFpfx>
5. Миронов А. Машинное обучение. М., 2013. 83 с.
6. Николаева Ю. Sostav.ru продолжает исследовать российский рынок новостных медиа. <https://clck.ru/Tr6iR>
7. Посевкин Р. Автоматизация sentiment-анализа естественно-языкового текста. Кишинев, 2014. 76 с.
8. Посевкин Р.В., Бессмертный И.А. Применение sentiment-анализа текстов для оценки общественного мнения // Научно-технический вестник информационных технологий, механики и оптики. 2015. Т. 15. №1. С. 169-171.
9. Романов А. Анализ тональности текста с помощью Azure Machine Learning. <https://habr.com/ru/post/243705/>
10. Хасти Т., Тибширани Р., Фридман Д., Основы статистического обучения. Интеллектуальный анализ данных, логический вывод и прогнозирование. М., 2020. 755 с.
11. Barthel M., Mitchell A. Many Americans Believe Fake News Is Sowing Confusion. <https://clck.ru/Tr2wz>

© Бобокало Д.С., Кондратьева Т.Н., 2021

УДК 687.016.5

**Бражникова А.М., Хорина И.В.**, канд. экон. наук  
Самарский государственный технический университет,  
г. Самара, Россия

## **УМНЫЕ ПАЙЕТКИ - ЭЛЕКТРОМЕХАНИЧЕСКИЕ ЦВЕТОВОСПРОИЗВОДЯЩИЕ УСТРОЙСТВА ДЛЯ ДИЗАЙНЕРСКОЙ ОДЕЖДЫ**

Развитие информационных технологий и новых текстильных материалов привело к появлению умной одежды, которая способна получать информацию из окружающего мира (или собирать информацию о своем хозяине), обрабатывать ее и изменять свои свойства в зависимости от текущей ситуации

Индустрия моды - одной из первых откликается на появление новых технологий. Дизайнеры, постоянно ищущие новые идеи и концепции, быстро подхватывают новые разработки и находят им применение в своих концептах, создавая уникальные коллекции одежды.

Одно из направлений, активно поддерживаемое многими современными дизайнерами - одежда меняющая цвет. Идея создания одежды способной самостоятельно перекрашиваться в режиме реального времени озвучивалась во многих фантастических произведениях. Причем интерес к такому свойству одежды проявляют не только дизайнеры, но и военные. Костюм, способный окрашиваться в цвет травы, камня, дерева, подражая любому природному сочетанию оттенков, станет превосходным камуфляжем для солдата.

С технической точки зрения на данный момент существует два способа заставить одежду менять свой цвет: использовать специальные краски, изменяющие свой цвет при воздействии на них внешних факторов, или использовать электронный текстиль, оснащенный множеством цветовоспроизводящих элементов, управляемых микропроцессором.

Первый способ основан на изготовлении красящих веществ из полимерных материалов, изменяющих свои оптические свойства (способность отражать, поглощать и преломлять свет с разной длиной волны) под действием температуры (термохромные), влажности (гидрохромные) или магнитного поля (парамагнитные). Коллекция лондонского дизайнера Лорен Броукер, представленная в 2014 г демонстрирует все возможности данных технологий (<https://clck.ru/UFrSV>).

Второй способ использует достижения микроэлектроники и предполагает интегрирование в ткань электронных компонентов (аккумуляторов, проводников, микропроцессоров, светодиодов и электродвигателей). Управляемые от центрального процессора или действуя автономно, составные части умной ткани способны при помощи цветовоспроизводящих устройств синтезировать любой цвет и оттенок.

В качестве цветовоспроизводящих устройств могут выступать [1]:

- LED (светодиодные) матрицы;
- LCD и LED дисплеи;

- e-link (электронные чернила) экраны;
- электроактивные полимеры.

Наиболее популярными являются светодиодные матрицы и отдельные светодиоды. Благодаря простоте использования, малому энергопотреблению, миниатюрным размерам отдельных светодиодов многие дизайнеры обратили внимание именно на них.

Однако светодиодные матрицы обладают рядом недостатков. Каждый светодиод матрицы постоянно потребляет энергию. Чем выше яркость светодиода, тем больше энергии он потребляет. Проблему могут решить e-link дисплеи, но они имеют очень высокую стоимость. Авторами данной работы разработаны электромеханические цветовоспроизводящие устройства, которые получили название «Умные пайетки» (рис. 1).

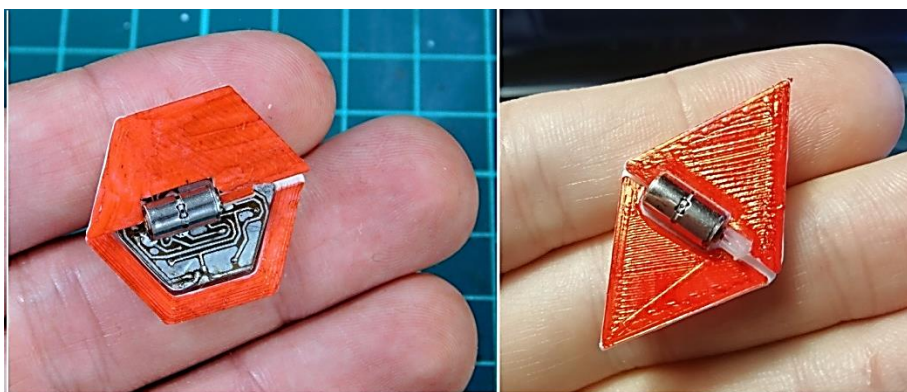


Рис. 1. Умная пайетка различной формы

Каждая пайетка состоит из основания, электронного блока управления (печатной платы), миниатюрного шагового двигателя и лепестка, окрашенного с двух сторон в различные цвета (рис. 2). Устройство имеет цифровой интерфейс и входит в состав матрицы, с неограниченным количеством элементов с управлением на уровне отдельной пайетки. Основание размещается на эластичной основе (сетчатой ткани).

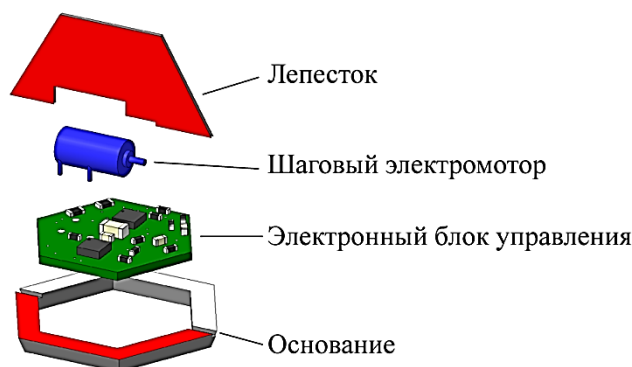


Рис. 2. Структура умной пайетки

Лепесток пайетки может быть не только окрашен в различные цвета, но и иметь различную текстуру: матовую, глянцевую, пористую. Это позволяет контролировать количество поглощаемого одеждой солнечного излучения, а также управлять процессом



испарения влаги из-под поверхности материала и, таким образом, реализовать контроль микроклимата одежды (рис. 3).

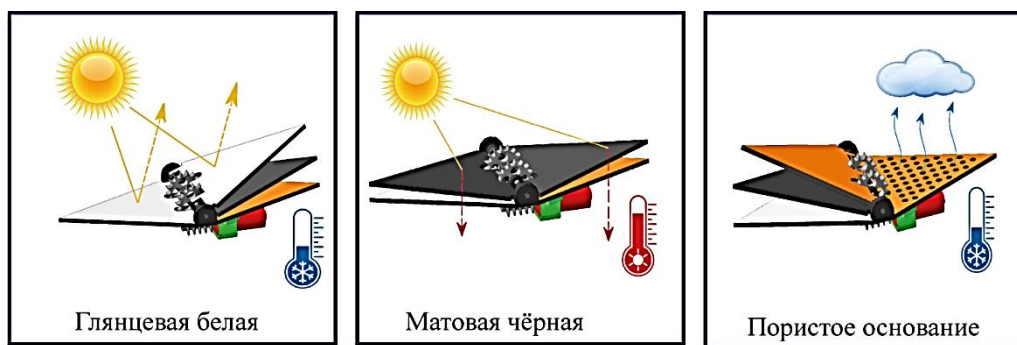


Рис. 3. Контроль микроклимата одежды

3D печать быстро проникла в сферу дизайна одежды. Простота использования современных принтеров, широкий ассортимент материалов и практически неограниченные возможности по созданию самых сложных пространственных структур привлекли молодых дизайнеров, стремящихся сделать одежду настолько личной, индивидуальной, насколько это возможно.

Необходимо отметить, что создание одежды с применением аддитивных технологий - не простая инженерная задача, так как «детали» одежды, выращенные на 3D принтере отличаются от традиционных текстильных материалов. Они имеют большую массу, меньшую гибкость, низкую пластичность и довольно жесткие на ощупь

На сегодняшний день существует несколько способов создания полотен «тканей» с применением аддитивных технологий.

1. Создание «кольчуги» из отдельных жестких элементов, шарнирно сцепленных друг с другом. Интересной является работа творческой студии Nervous System, которая в 2014 г создала платье, которое было напечатано на 3D принтере одной деталью, причем отличалось необычайной подвижностью и легкостью. Это заслуга специально разработанных шарнирных соединений и особой геометрии «кольчуги», а также уникальной технологии 4D печати, реализованные посредством селективного лазерного спекания порошкового полимера (SLS Печать) (<https://clck.ru/UFrTt>).

2. Печать эластичными пластиками (FLEX пластик) отдельных элементов одежды и последующее их соединение в изделие. В данном направлении работает дизайнер из Израиля Данит Пелег, которая в 2015 г представила коллекцию из 5 нарядов собранных из «лоскутков» FLEX пластика, напечатанных на 3D принтере. Получившиеся платья обладают значительной эластичностью, что не в последнюю очередь достигается за счет узора лоскутков. Причем для изготовления такой одежды достаточно иметь принтер начального уровня (<https://clck.ru/UFrQT>).

3. Печать «чешуи» (отдельных жестких пластиковых элементов) на ткани. Данный метод подходит для печати на FDM принтерах. Деталь формируется на специальной сетчатой основе. После завершения печати отдельные чешуйки образуют с основой единое целое (<https://clck.ru/UFrSV>).



Для размещения электромеханических цветовоспроизводящих устройств было решено использовать «чешуйки», напечатанные на сетчатой ткани. Выбор сделан в пользу данного метода так как с одной стороны «чешуйки» могут быть напечатаны из твердого PLA пластика и служить для крепления печатной платы пайетки, а с другой стороны метод достаточно прост и не требует специального оборудования. Следует отметить, что гибкость полотна определяется формой, размерами «чешуек» и расстоянием между ними. Для того чтобы определить оптимальную форму пайеток было напечатано несколько фрагментов с различными формами «чешуек» (рис. 4).

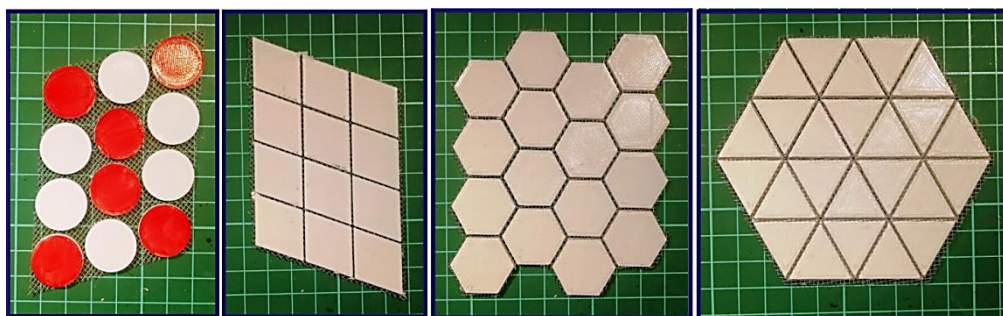


Рис. 4. Различная форма пайеток, размещенных на сетчатой ткани

Оказалось, что форма правильного шестиугольника (пчелиные соты) обеспечивает минимальный зазор между элементами полотна и максимальную гибкость во всех направлениях. Так как каждая пайетка предназначена для работы в составе матрицы стоит задача передачи информации между элементами матрицы, а также обеспечить их электрическим питанием. Для этого в сетчатую ткань вплетены нити из токопроводящего материала, обладающие высокой гибкостью и ресурсом работы (в отличие от медных проводников).

Таким образом, авторами данной статьи были разработаны электромеханические цветовоспроизводящие устройства, специально предназначенные для размещения на одежде. Несмотря на ограниченную палитру цветов, умные пайетки обладают рядом преимуществ перед светодиодными матрицами и e-link дисплеями: возможность менять не только цвет, но и текстуру материала, низкое энергопотребление (энергия расходуется только в момент переключения цвета).

Дальнейшая работа будет связана с разработкой образцов одежды с применением умных пайеток и тестированием практической применимости разработанного устройства.

### Литература

1. Бражников А.М., Бражникова А.М. Анализ технологий создания цветовоспроизводящих устройств для систем адаптивного камуфляжа // Современные инновации в науке и технике. Курск, 2020. 355 с.

© Бражникова А.М., Хорина И.В., 2021

УДК 004.85

Марунов А.А., Добровольская Н.Ю., канд. пед. наук  
Кубанский государственный университет,  
г. Краснодар, Россия

## ИСПОЛЬЗОВАНИЕ LSTM СЕТЕЙ ПРИ КЛАССИФИКАЦИИ ТЕКСТОВЫХ СООБЩЕНИЙ

В современном мире большинство предметных областей в том или ином виде занимаются обработкой, анализом данных и их классификацией. Алгоритмизация этих процессов, в частности задач классификации, несомненно, важна, так как позволяет избавиться от множества рутинной работы. С течением времени появляются различные алгоритмы для автоматизации этой задачи, например, алгоритмы, использующие в своей основе искусственные нейронные сети. Во многих предметных областях необходимо классифицировать текстовые фрагменты, определение уровня соответствия текста тому или иному классу является достаточно актуальной задачей.

Нами рассматриваются блоки новостных сообщений, автоматизированная классификация которых может быть использована в качестве отдельного программного модуля в различных приложениях, например, электронных секретарях.

Рассмотрим множество текстовых документов  $D = \{d_1, \dots, d_n\}$ , множество возможных категорий (классов)  $C = \{c_1, \dots, c_m\}$  и неизвестную целевую функцию  $\Phi : D \times C \rightarrow \{0, 1\}$ . Имеется некоторая начальная коллекция размеченных документов  $\Omega \subset C \times D$ , для элементов которой известны значения  $\Phi$ . Требуется построить классификатор  $\Phi'$ , максимально близкий к  $\Phi$  [1, с.86].

В качестве ядра программы-классификатора выбрана нейронная сеть долговременной памяти (Long Short Term Memory). Подобная сеть относится к типу рекуррентных нейронных сетей, способных к обучению долгосрочным зависимостям [2, с. 230].

Для программной реализации задачи категориальной классификации новостных сообщений выбран язык программирования Python. На вход программы подается текст новости, на выходе ее категория. Тип категорий определен ниже. На первом этапе разрабатывалась программа для обучения нейросетевой модели и создания токенизатора на предварительно подготовленных наборах данных. Токенизатор и нейросетевая модель сохранены в файлы и используются в основной программе. Для разработки этих модулей использованы следующие библиотеки. Keras – открытая нейросетевая библиотека, разработанная на языке Python. Функции библиотеки позволяют организовать работу с сетями глубинного обучения. NumPy расширение языка Python, необходимое в проекте для реализации работы больших многомерных массивов и матриц, кроме того, в нем присутствуют высокоуровневые математические функции для операций над подобными массивами. Pandas программная библиотека на языке Python для обработки и анализа данных, предоставляющая специальные структуры данных и операции для взаимодействия с

числовыми данными и временными рядами. Matplotlib – библиотека на языке программирования Python для визуализации данных двумерной графикой.

Схему обучающего приложения, предназначенного для создания токенизатора и обучения нейронной сети можно изобразить следующим образом (рис. 1):



Рис. 1. Схема обучающего приложения

Для обучения используется набор данных, извлекаемый с сайта Gazeta.Ru. Данный сайт представляет собой новостной портал, круглосуточно публикующий российские и мировые новости. Всего использовано восемь категорий: «Культура», «Бизнес», «Авто», «Спорт», «Общество», «Наука», «Технологии», «Политика». Каждый класс пронумерован от 0 до 7 соответственно. Данные для обучения загружались из CSV-файлов, в которых в каждой строке указывался номер класса и новостная статья через запятую. Сам набор данных разделен на две части (файла): тренировочный и тестовый. Тренировочный набор содержит 120000 статей, тестовый – 7600.

После загрузки данных из файла начинается процесс токенизации. Статьи токенизируются на уровне слов и в дальнейшем каждому слову в соответствие ставится некоторое положительное число, характеризующее частоту появления слова в наборе данных, чем оно меньше, тем чаще слово появляется в наборе данных. Также при создании токенизатора в качестве параметра конструктора передается целое число – количество слов в словаре (в данном случае 10000). Таким образом формируется ограниченный словарь наиболее часто встречаемых слов. В дальнейшем в каждой статье все слова заменяются соответствующими числовыми характеристиками, и так как нейронная сеть работает с последовательностями одинаковой длины, то полученным векторам задается одинаковый фиксированный размер, для этого используется «0», как специальный служебный символ-заполнитель. Метки классов, в свою очередь, кодируются разреженными векторами. После токенизации создается модель, схематично ее можно представить следующим образом (рис. 2):



Рис. 2. Архитектура НС

Слой Embedding формирует плотные векторные представления для числовых последовательностей, полученных из статей. На вход он получает статью (в формате числовой последовательности), на выходе выдает массив из векторов для каждого слова в предложении.

Изначально создаются случайные векторы для слов из словаря, которые затем постепенно конструируются (обучаются), как это происходит с весами нейронной сети. Примеры векторных представлений и частотных числовых кодов представлены на рисунке 3.

```
Слово: парламента
Код: 1681
Вектор: [ 0.04792599 -0.00845238 0.04105876 0.1035983 0.10541984 -0.06657976
0.05321523 -0.02885371 0.04272087 0.00394442 -0.03061586 -0.00028177
0.00473806 -0.07235456 -0.00539906 -0.04271324 -0.041515 0.03166841
0.08894118 0.0296997 -0.01224082 -0.10437789 -0.04398979 -0.07878703
0.09157989 0.03343471 0.08302817 -0.02461248 0.01454277 -0.09490976
0.03584468 -0.06445344]

Слово: финального
Код: 4286
Вектор: [-0.08200805 0.03897848 0.06930912 0.05614356 -0.05067663 -0.01779504
-0.00797767 -0.0144735 -0.01604407 0.01765086 -0.0329883 0.02020678
-0.1164768 0.07980028 0.02420238 0.00097068 -0.10844398 -0.05168251
0.0191892 0.00233021 -0.04089797 -0.06957781 0.0868243 0.0462868
-0.01071936 -0.04462774 0.02330311 0.00328036 0.05223774 0.00067833
-0.04083256 -0.03396053]

Слово: считается
Код: 3239
Вектор: [-0.1037939 -0.04917623 0.00109048 -0.1517656 0.00697461 0.01120408
-0.01147471 0.11275933 -0.05216758 -0.07162488 -0.03589858 0.07619257
0.0067953 0.05686698 0.0083855 0.0689083 -0.07043377 -0.03993417
-0.08371528 0.05454662 0.07568637 0.08138006 0.12298028 0.0948522
-0.10547075 -0.07431898 -0.08735485 -0.04278182 0.08010612 -0.00944316
-0.08501516 0.10117753]
```

Рис. 3. Пример векторизации и частотного кодирования

Далее данные проходят через LSTM слой, который состоит из 16 связанных ячеек, после чего подаются на полносвязный слой, имеющий на выходе в качестве функции активации Softmax. Эта функция преобразует вектор размерности  $n$  в вектор той же размерности, где каждая координата полученного вектора представлена вещественным числом в промежутке  $[0,1]$  и сумма координат равна 1. Это удобно, так как полученное на выходе число можно трактовать как вероятность принадлежности одному из классов. Таким образом на выходе сети массив из восьми чисел в диапазоне  $[0,1]$ , каждое из которых характеризует вероятность принадлежности к соответствующему классу.

Обучение проводилось в 5 эпох сначала на тренировочном, затем на тестовом наборе, корректировка весов проводилась после каждого прохождения 128 статей через нейронную сеть. Доля правильных верных ответов на обучающем и проверочном наборе составили 89% и 82% соответственно.

После подготовки токенизатора и обучения сети, следующим этапом разработана основная программа. Ее схема представлена ниже (рис. 4).



Рис. 4. Схема основной программы

Программа представляет собой консольное приложение и на входе требует ввести путь к файлу со статьей. В процессе работы приложения изначально инициализируется массив классов и загружается статья из текстового файла. Затем инициализируются нейросетевая



модель и токенизатор, предварительно сохраненные в файлы. Текст статьи проходит этапы токенизации и векторизации таким же образом, как и в обучающей программе. Полученная последовательность векторов передается дальше в скрытый и выходной слои. На выходе сети массив из восьми чисел, характеризующих вероятность принадлежности к соответствующему классу, из них берется максимальный элемент, индекс этого элемента совпадает с индексом категории в массиве классов. Данная категория выводится в консоль в качестве результата работы программы. Пример результирующего массива и соотношения вероятностей для научной новостной статьи изображен на рисунке 5.

#### Текст статьи

Астрономы обнаружили рядом со звездой HD 40307, расположенной в 42 световых годах от Земли, планету, потенциально пригодную для жизни, говорится в статье, опубликованной в журнале *Astronomy & Astrophysics*. Первую потенциально обитаемую планету за пределами Солнечной системы в 2011 году обнаружил космический телескоп «Кеплер». Однако планета Kepler-22b находится значительно дальше – в 600 световых годах от Земли. Звезда HD 40307 в созвездии Живописца – оранжевый карлик спектрального класса K, который примерно на 23% легче Солнца и настолько же тусклее его. В 2008 году ученые из Европейской южной обсерватории (Чили) с помощью спектрографа HARPS обнаружили рядом с этой звездой сразу три планеты – с массами в 4,2, 6,7 и 9,4 раза больше массы Земли. Эти планеты обращаются на очень «тесных» орбитах вокруг светила – их «год» длится 4,3, 9,6 и 20,4 земных суток соответственно. Группа под руководством Микко Туоми из британского университета Хертфордшира заново проанализировала данные со спектрографа HARPS с помощью новых более чувствительных методик, и обнаружила еще три «пропущенные» планеты. «Мы обнаружили два ясных дополнительных сигнала, с периодом в 34 и 51 день, которые соответствуют планетам-кандидатам с минимальными массами в несколько масс Земли. Шестая планета.. имеет период обращения около 200 дней.. и массу примерно в семь масс Земли», – говорится в статье. Шестая планета привлекла особое внимание ученых, поскольку она оказалась почти точно в середине «зоны жизни» вокруг своего светила.

#### Результирующий массив:

```
[0.10612486 0.03755518 0.12577616 0.021495 0.04831067 0.3776165  
0.2512002 0.03192145]
```

Класс	Вероятность принадлежности
Культура	0.106124863
Бизнес	0.037555177
Авто	0.125776157
Спорт	0.021495003
Общество	0.048310667
Наука	0.377616495
Технологии	0.251200199
Политика	0.031921446

Рис. 5. Распределение вероятностей текстовой принадлежности

Приложение решает задачу однозначной классификации, однако если требуется определение принадлежности статьи к нескольким классам, алгоритм следует модифицировать. Так как каждый элемент результирующего массива можно считать за вероятность принадлежности к соответствующему классу, то в качестве дополнительного входного параметра можно задавать некоторое достаточно малое положительное число  $\epsilon$ , такое, что все классы соответствующие элементам результирующего массива, значения которых лежат в  $\epsilon$ -окрестности максимального значения этого массива, будут определены как классы входной статьи. Но в таком случае вероятность неточной классификации возрастает.

Предлагаемый модуль позволяет достаточно точно классифицировать текущие новостные сообщения по ряду классов, задачу классификации решает нейронная сеть LSTM.

### Литература

1. Батура Т.В. Методы автоматической классификации текстов // Программные продукты и системы. 2017. №1. С. 85-99.
2. Осовский С. Нейронные сети для обработки информации. М., 2002. 344 с.

© Марунов А.А., Добровольская Н.Ю., 2021

УДК 004.85

Фурса И.И., Добровольская Н.Ю., канд. пед. наук  
Кубанский государственный университет,  
г. Краснодар, Россия

## РЕШЕНИЕ ЗАДАЧИ ПРОГНОЗИРОВАНИЯ ВОЛАТИЛЬНОСТИ С ПОМОЩЬЮ НЕЙРОННОЙ СЕТИ С ДОЛГОЙ КРАТКОСРОЧНОЙ ПАМЯТЬЮ

В настоящее время к решению экономических задач привлекаются все передовые возможности IT-сферы. Одной из базовых задач в экономике является прогнозирование, в частности, точный прогноз волатильности цены какого-либо базового актива может быть использован в различных торговых стратегиях на бирже. Особенно полезен прогноз волатильности при торговле с помощью опционных контрактов, так как цена опционного контракта зависит от того, какая волатильность цены базового актива будет через некий промежуток времени. Волатильность для расчета стоимости опциона вычисляется по специальной формуле, которая слабо коррелирует с реальным поведением цены, поэтому предсказания волатильности, точность которых будет выше, чем точность расчета по формуле, позволит получать прибыль при торговле на бирже [2, с. 241].

Наиболее популярными механизмами математического прогнозирования являются методы корреляционного и регрессионного анализа, методы экстраполяции трендов. Эти методы относятся к количественным методам и применяются для краткосрочного и среднесрочного прогнозирования. Примерами таких методов являются: скользящее среднее, экспоненциальное сглаживание, векторная авторегрессия и так далее. Однако в последнее время для решения задачи прогнозирования успешно применяются методы искусственного интеллекта. К ним относятся: искусственные нейронные сети, групповые методы обработки данных, методы опорных векторов.

Волатильность – это статистическая мера разброса доходности для данной ценной бумаги или рыночного индекса [2, с. 87]. В большинстве случаев, чем выше волатильность, тем выше риск. Волатильность часто измеряется либо как стандартное отклонение, либо как дисперсия доходности одной и той же ценной бумаги или рыночного индекса.

На рынках ценных бумаг волатильность часто связана с большими колебаниями в любом направлении. Например, когда фондовый рынок растет и падает более чем на один процент в течение длительного периода времени, это называется «волатильным» рынком. Неустойчивость актива является ключевым фактором при ценообразовании опционных контрактов.

Под волатильностью часто понимается степень неопределенности или риска, связанного с размером изменений стоимости ценной бумаги. Более высокая волатильность означает, что стоимость ценной бумаги потенциально может быть распределена по большему диапазону значений. Это означает, что цена ценной бумаги может резко измениться за короткий период



времени в любом направлении. Более низкая волатильность означает, что стоимость ценной бумаги не изменяется резко и имеет тенденцию быть более стабильной.

Один из способов измерить вариацию актива - количественно оценить ежедневную доходность (процентное изменение на ежедневной основе) актива. Историческая волатильность основана на исторических ценах и представляет собой степень изменчивости доходности актива. Это число без единицы измерения и выражается в процентах. В то время как дисперсия отражает разброс доходностей вокруг среднего значения актива в целом, волатильность является мерой этой дисперсии, ограниченной определенным периодом времени. Таким образом, можно сообщать о дневной, недельной, месячной или годовой волатильности.

Для решения задачи прогноза волатильности будем использовать сеть с долгой краткосрочной памятью [1, с. 135]. В ходе исследования попытаемся изучить возможности предсказания волатильности цены базового актива с помощью нейронной сети с долгой краткосрочной памятью и ответить на следующий вопрос: возможно ли использование предсказаний волатильности цены базового актива, сделанных с помощью нейронной сети с долгой краткосрочной памятью, в реальной торговле на бирже? LSTM – нейронные сети с долгой краткосрочной памятью являются подвидом рекуррентных нейронных сетей, способные обучаться долгосрочным зависимостям. Основное преимущество таких сетей – запоминание информации в течение длительных периодов времени.

Приложение, позволяющее спрогнозировать значение волатильности, разработано с помощью средств, предоставляемых Jupyter Notebook – веб-приложения с открытым исходным кодом. Используемый язык программирования – Python 3. В качестве основного фреймворка для машинного обучения применен Pytorch, обработка данных осуществляется с помощью библиотек pandas и sklearn, визуализация – средствами matplotlib.

Итоговый проект можно представить в виде последовательности следующих этапов (рис. 1): загрузка, предобработка и визуализация данных; моделирование; тестирование.

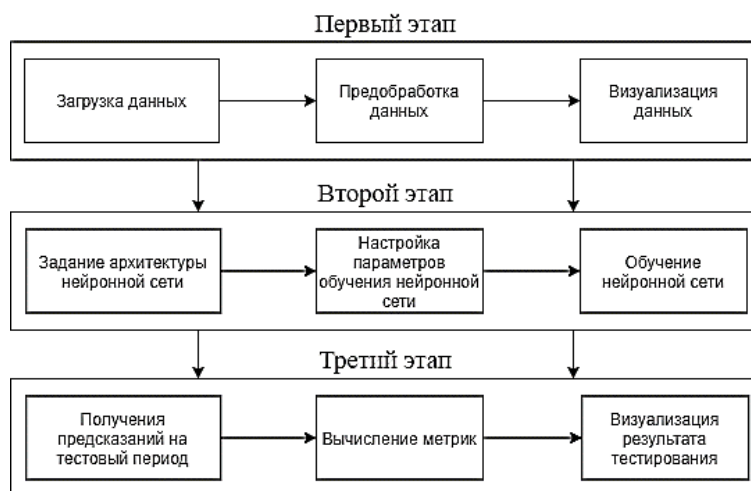


Рис. 1. Схема проекта

На первом этапе выполняется загрузка данных, затем проводится их обработка для конвертации в формат, воспринимаемой рекуррентной нейронной сетью. В работе использован индекс РТС по дням за период с 31-го августа 2017 года по 25-ое июня 2020 года. Каждый день характеризуется несколькими параметрами, но в работе задействовано только два: дата рассматриваемого дня и стоимость индекса РТС в момент закрытия Московской Биржи. Далее исходный временной ряд преобразовывается в новый временной ряд, который представляет собой изменение цены на момент закрытия некоторого торгового дня по отношению к предыдущему торговому дню в процентном соотношении – это и есть волатильность. Затем полученные данные разбиваются на две части: тренировочную и тестовую. На полученных тренировочных данных обучается специальный алгоритм шкалирования, который переводит тренировочные и, соответственно, тестовые данные в диапазон значений от -1 до 1.

На втором этапе задается структура рекуррентной нейронной сети с долгой краткосрочной памятью, а также характеристики для ее обучения и, соответственно, проводится само обучение нейронной сети на тренировочных данных. На завершающем, третьем этапе выполняется валидация нейронной сети на тестовых данных.

Промежуточные этапы обучения сети визуализируются (рис. 2, 3).

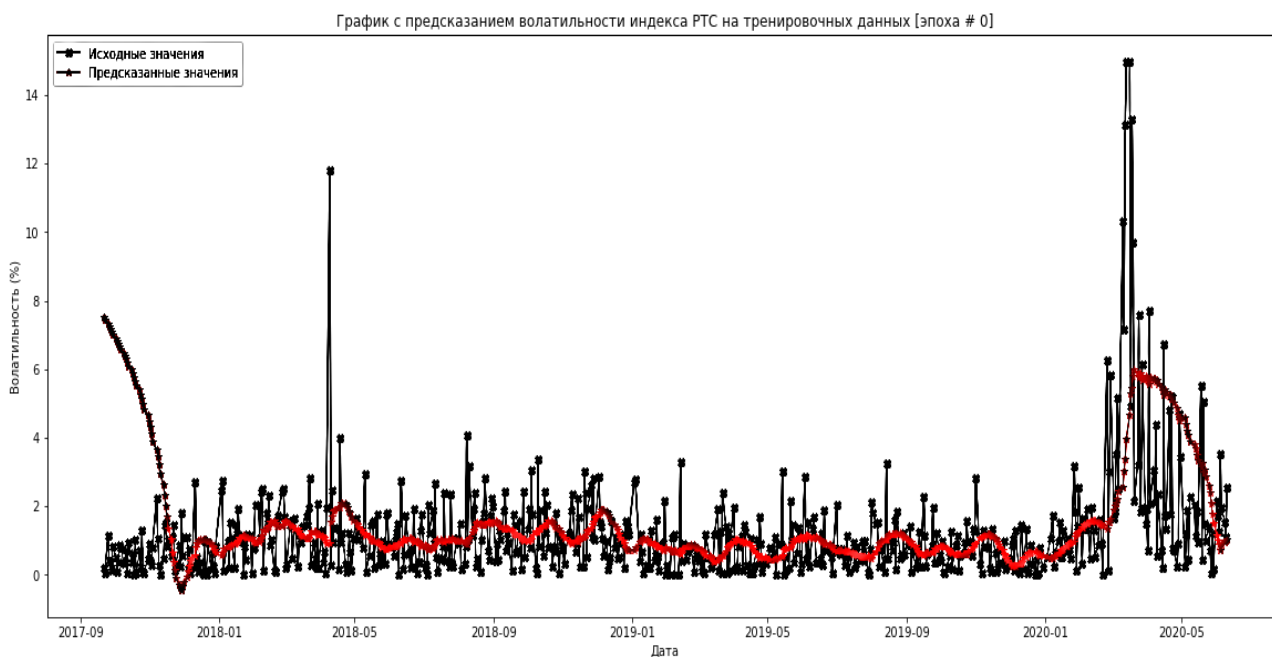


Рис. 2. Обученность нейронной сети после первой эпохи обучения на тренировочных данных

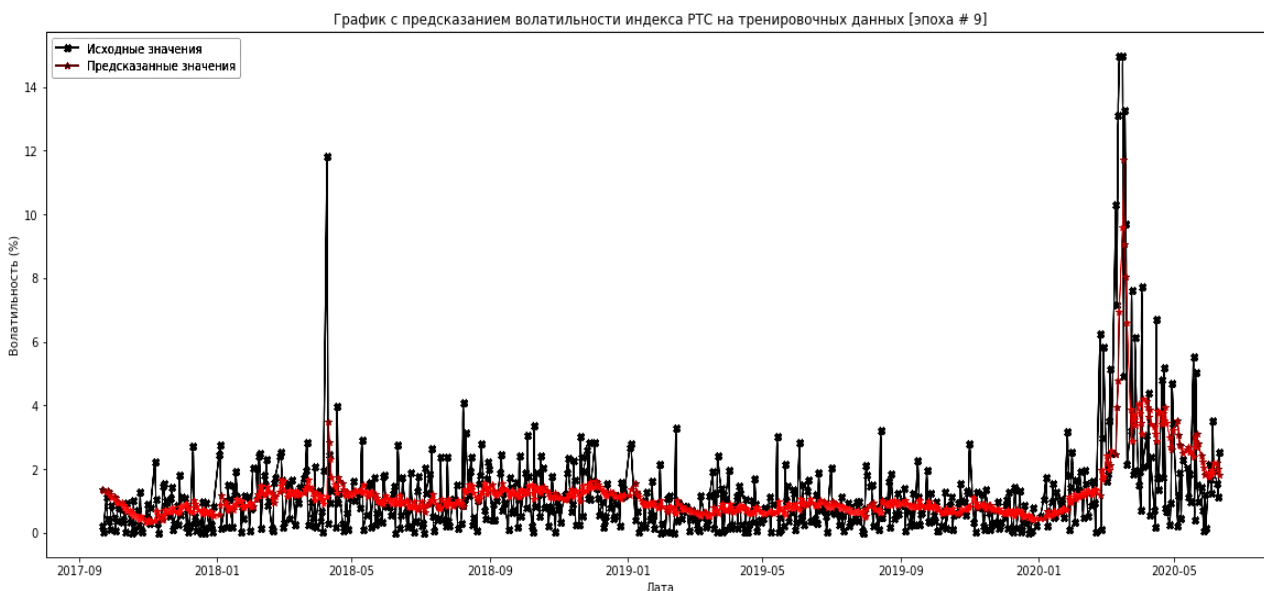


Рис. 3. Обученность нейронной сети после десятой эпохи обучения на тренировочных данных

Следующий график отражает результат валидации модели на тестовых данных (рис. 4).

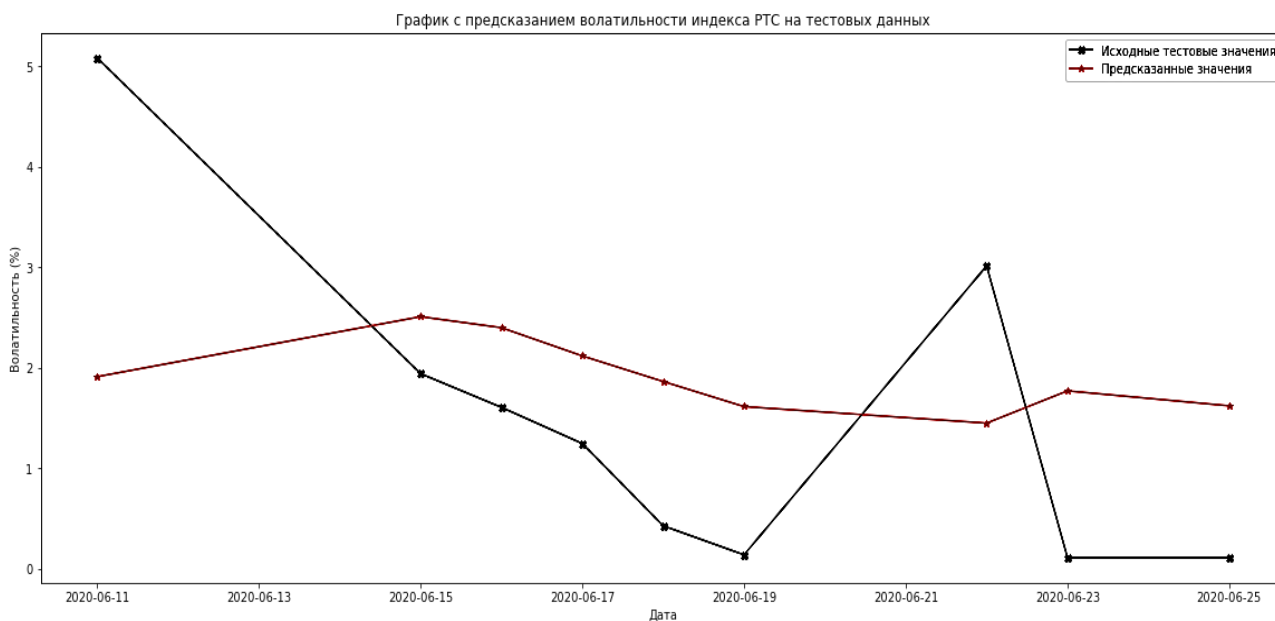


Рис. 4. График с предсказанием волатильности индекса РТС на тестовых данных

Таким образом, построили и протестировали искусственную нейронную сеть с долгой краткосрочной памятью для прогнозирования волатильности цены базового актива на день вперед на исторических данных. В ходе тестирования модели было выявлено, что при обучении на тренировочных данных нейронная сеть имеет тенденцию запоминать на каждой итерации среднее значение нескольких последних наблюдений временного ряда, не моделируя небольшие флуктуации исходных данных, то есть модель вела себя как метод скользящего среднего. Следовательно, сети с долгой краткосрочной памятью дают достаточно хороший прогноз и могут широко использоваться в экономических задачах.

Построенную модель искусственной нейронной сети можно использовать в качестве генератора торговых сигналов для трейдинга, что позволит оценивать среднее значение дневной волатильности на следующий торговый день, используя лишь исторические данные по цене базового актива, и принимать те или иные решения по покупке или его продаже. Другими словами, модель позволяет реализовывать некую торговую стратегию, учитывающую дневную волатильность цены базового актива.

### Литература

1. Гудфеллоу Я. Глубокое обучение. М., 2018. 652 с.
2. Коннолли Б. Покупка и продажа волатильности. М., 2006. 264 с.

© Фурса И.И., Добровольская Н.Ю., 2021

УДК 004.428.4

**Шипилова А.А.**

*Научный руководитель:* **Кондратьева Т.Н.**, канд. техн. наук  
Донской государственной технической университет,  
г. Ростов-на-Дону, Россия

## **АНАЛИЗ ДИНАМИКИ КУРСА ВАЛЮТЫ СРЕДСТВАМИ ПЛАТФОРМЫ METATRADER 5 НА ПРИМЕРЕ ВАЛЮТНОЙ ПАРЫ USD/EUR**

MetaTrader 5 – это мультиактивная платформа, которая позволяет торговать на финансовом рынке Forex, различными видами ценных бумаг, в том числе производными бумагами первого и второго порядка. Платформа предлагает инструменты технического и фундаментального анализа для комплексного разбора динамики цен [6, с. 158–169].

Платформа позволяет одновременно изучать около 100 графиков котировок валют и акций. Доступный 21 TimeFrames позволяет проводить исчерпывающий детальный анализ динамики цен в режиме 24/7. Более 80 технических индикаторов, осцилляторов и аналитических инструментов, включая графические объекты, позволяют детально анализировать динамику котировок [1, с. 56–60; 2].

Кроме того, аналитические возможности терминала не ограничиваются встроенными инструментами [5, с. 65–71]. Пользователи могут выбирать между тысячами бесплатных индикаторов из Code Base, покупать или арендовать приложения в MetaTrader Market из более, чем 2500 готовых алгоритмических приложений, торговых стратегий, а также заказывать пользовательские индикаторы у профессиональных программистов через сервис Freelance. Благодаря специализированному языку программирования высокого уровня MetaQuotes Language 5 (MQL5) можно создать собственный индикатор со своими уникальными характеристиками. Все эти функции увеличивают аналитические возможности платформы и способность проводить наиболее подробный и точный технический анализ котировок [3, с. 56–60].

Фундаментальный анализ – еще один инструмент, который можно использовать для анализа динамики цен финансовых инструментов. Целью фундаментального анализа является постоянный мониторинг различных экономических и промышленных показателей из открытых источников. Исходные фундаментальные данные доступны прямо из MetaTrader 5.

Новостные репортажи от международных информационных агентств транслируются прямо на платформу, предоставляя важную финансовую информацию. В экономическом календаре представлены основные макроэкономические показатели разных стран, что позволяет пользователям анализировать их влияние на различные финансовые инструменты [7, р. 104–129].

Один из наиболее эффективных способов работы на платформе является алгоритмическая торговля с использованием специализированных приложений – торговых

роботов (экспертов). Роботы работают без участия трейдера – они могут анализировать котировки и совершать торговые операции по базовому алгоритму.

В качестве финансового инструмента для анализа будущих изменений цен, будем рассматривать валютную пару USD/EUR (<https://clck.ru/UFtTn>). Основная задача – выявить различные особенности и закономерности в динамике цен, которые не видны невооруженным глазом.

На рисунке 1 представлены детали обзора рынка, основные характеристики выбранного продукта и построен график по тиковым данным валютной пары USD/EUR на платформе Metatrader 5 (июнь 2020 – ноябрь 2020).

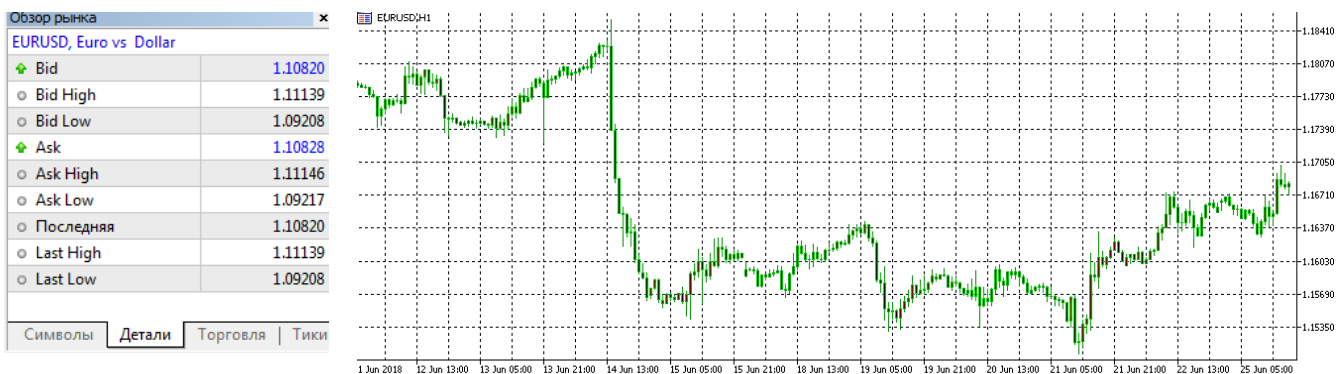


Рис. 1. График валютной пары USD/EUR

Настройка периода и выбор инструментария определен следующим образом: индикатор из раздела Трендовые (Moving Average); объекты данных TimeFrames; структуры данных объектов MqlRates, MqlTick.

В работе предлагается построить торговые стратегии инструментами платформы Metatrader 5, провести графический анализ динамики курса валют, сравнить результаты полученных стратегий, интерпретировать их.

Для анализа динамики курса валют применим трендовым индикатором Moving Average (MA), проведем настройку параметров, представим графики работ. На рисунке 2 представлена работа индикатора Moving Average с параметрами: период – 12; Метод – Simple; цены – close.



Рис. 2. Работа индикатора MA(12). Метод – Simple



За выбранный период, индикатор MA(12) с заданным периодом и соответствующим методом провел 120 стопов при 109 шагов и 60 стопов при 55 шагов со сдвигом на 6 тиков. Продолжим наше исследование, сохранив выбранный период для индикатора Moving Average, применим еще пару, тройку методов и сравним полученные результаты.

На рисунке 3 представлена работа индикатора MA, параметр – 12, Метод – Экспоненциальный (экспоненциальный).



Рис. 3. Работа индикатора MA(12). Метод – Экспоненциальный

За выбранный период, индикатор MA(12) с заданным периодом и соответствующим методом провел 120 стопов при 109 шагов и 60 стопов при 55 шагов со сдвигом на 6 тиков, что соответствует результатам, полученным ранее, применяя метод Simple.

На рисунке 4 представлена работа индикатора MA, параметр – 12, Метод – Smoothed (сглаженный).



Рис. 4. Работа индикатора MA(12). Метод – Smoothed

За выбранный период, индикатор MA(12) с заданным периодом и соответствующим методом провел 120 стопов при 109 шагов и 60 стопов при 55 шагов со сдвигом на 6 тиков, что соответствует результатам, полученным ранее методами: Simple; Экспоненциальный.

Далее, на рисунке 5 представлена работа индикатора МА, параметр – 12, Метод – Linear Wiegthed (линейно-взвешенный).



Рис. 5. Работа индикатора МА(12). Метод – Linear Wiegthed

За выбранный период, индикатор МА(12) с заданным периодом и соответствующим методом провел 120 стопов при 109 шагов и 60 стопов при 55 шагов со сдвигом на 6 тиков, что соответствует результатам, полученным ранее методами: Simple; Exсponential; Smoothed.

На рисунке 6 представлена итоговая работа индикатора Moving Average с параметрами: период – 12; цены – close по всем исследуемым методам.

Переменная	Значение	Старт	Шаг	Стоп	Шаги
<input checked="" type="checkbox"/> Maximum Risk in percentage	0.02	0.02	0.002	0.2	91
<input checked="" type="checkbox"/> Decrease factor	3	3	0.3	30.0	91
<input checked="" type="checkbox"/> Moving Average period	12	12	1	120	109
<input checked="" type="checkbox"/> Moving Average shift	6	6	1	60	55
					49644595

Рис. 6. Результаты работы индикатора МА(12)

Следует отметить, что отчеты по работе индикатора МА(12) в сочетании с выбранными методами показали одинаковые результаты, что говорит об оптимальном подборе инструментов платформы Metatrader 5.

По результатам анализа были сделаны следующие выводы. Первое – эффективность работы построенного советника нашла свое подтверждение в четырех методах.

Второе – торговый советник четко предугадывает движение цен по выбранному объекту. Данный факт отражен на всех графиках (рис. 2, 3, 4, 5). Во всех случаях индикатор делает четкие предсказания по нисходящему тренду (линия индикатора пересекает тренд снизу вверх); по восходящему тренду (линия индикатора пересекает тренд сверху вниз); по флэтовым участкам тренда (линия индикатора асимптотически сопровождает тренд либо сверху, либо снизу).

На рисунках 2-5 представлены разные участки графиков ценовой динамики валютной пары USD/EUR для лучшей визуализации всех частей графиков. Надо отметить, количество

обработанных типов, соответствует 49644595, и наблюдается по всем четырем методам в результате работы торгового индикатора (рис. 6).

Третье – при изменении параметра «период», количество зафиксированных тиков может значительно увеличиться/уменьшиться. Такого рода изменения могут привести к разному роду зависимостям между рассматриваемыми параметрами: прямой; обратной; смещенной. То есть, с увеличением TimeFrames количество зафиксированных тиков (данных по объекту) может, как увеличиться, так и уменьшиться, а вместе, с этим изменятся и основные характеристики по объекту, что повлечет за собою смену применяемых методов. Аналогичная ситуация может происходить и при уменьшении TimeFrames, в частности, появится необходимость применения скальпингового анализа на минимальных TimeFrames, что увеличит количество stop–line и количество long–line, и может привести к еще большей прибыли (доходности выбранных активов) трейдера. Формирование таких торговых стратегий требует дополнительного времени, как для построения, так и для анализа полученных результатов.

Разработка собственного модуля может повысить эффективность торговли на финансовом рынке. Построение собственной торговой стратегии следует начинать с создания библиотек. Необходимо будет организовывать простые списки объектов для анализа всех данных. Все объекты в одном списке должны иметь однотипные данные, присущие именно данному типу объекта. Для каждого конкретного объекта нужно будет создать класс, наделенный минимально–необходимым функционалом для хранения, сортировки и выдачи данных.

Надо сказать, что MetaTrader 5 включает в себя большое количество различных библиотек, построенных для различных объектов с различными инструментами с применением разного рода методов. От содержания и наполненности данных модулей зависит их стоимость, чем проще модуль, тем ниже его цена. Существуют и бесплатные версии библиотек, но надо ли говорить о надежности результатов работы динных торговых советников, учитывая тот факт, что финансовый рынок всегда имеет либо положительную, либо отрицательную динамику. Даже если рынок стагнирует, это не повод для расстройств, необходимо сменить стратегию и подобрать нужный инструментарий. При желании, можно написать код самостоятельно, решением подобного рода задач планируем заняться в следующих работах.

### Литература

1. Ананченко И., Мусаев А., Сушко А. Для торговли на FOREX: разработка системы, индикатора, программы. М., 2014. 148 с.
2. Кондратьева Т. Н. Принцип работы трендовых индикаторов // Интернет-журнал Науковедение. 2013. №3. С. 163-163.
3. Кондратьева Т.Н., Развеева И.Ф. Автоматическая торговая система в клиентском терминале MetaTrader 5 // Современные наукоемкие технологии. 2018. № 10. С. 56-60.



4. Кондратьева Т.Н., Развеева И.Ф., Лесовая А.А. Прогнозирование динамики цен фондового рынка // Патент РФ № 201866578. Патентообладатель ФГБОУ ВО «Донской государственный технический университет» (ДГТУ) (RU). 2018. Бюл. № 9.

5. Лафоре Р. Объектно-ориентированное программирование в C++. СПб., 2018. 928 с.

6. Морозов И. Forex. От простого к сложному. М., 2016. 328 с.

7. Schwager J. D. Technical analysis. John Wiley & Sons, 1995. V. 43.

© Шипилова А.А., Кондратьева Т.Н., 2021

УДК 004

Алмакаев Д.Р.

Научный руководитель: Кошкарлов А.В., канд. техн. наук  
Астраханский государственный  
университет, г. Астрахань, Россия

## АВТОМАТИЗАЦИЯ ПРОЦЕССОВ ВЗАИМОДЕЙСТВИЯ ПОЛЬЗОВАТЕЛЯ ПУТЕМ СОЗДАНИЯ СПЕЦИАЛИЗИРОВАННЫХ ИНФОРМАЦИОННО-СПРАВОЧНЫХ СИСТЕМ

В настоящее время большинство специализированных информационно-справочных систем, которые используются для поддержки пользователей в организациях, недостаточно самостоятельны и независимы. Все ключевые действия в таких системах выполняются человеком. В связи с этим актуальной задачей является автоматизация процессов взаимодействия пользователя и системы, начиная от приема заявок до успешного выполнения действий на компьютере пользователя, посредством разработки информационно-справочных систем (ИСС).

Информационно-справочная система в широком смысле – это система, в которой реализован принцип открытых компьютерных технологий, которые позволяют развивать систему, наполняя ее новыми документами, а также реализуют возможность сбора, хранения и обработки мультимедиа данных с использованием либо средств самой системы, либо внешних приложений [4, с. 329].

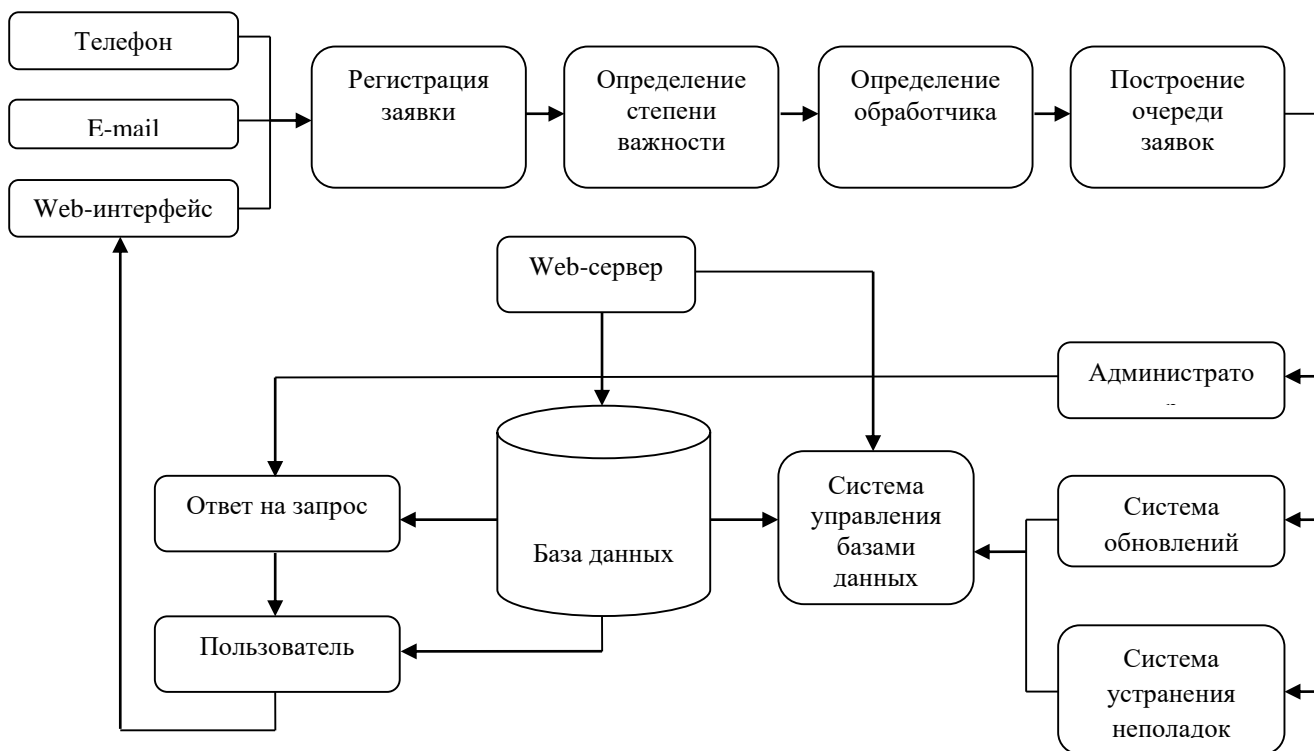


Рис. 1. Структурно-функциональная схема системы



Указанная система предусматривает не только автоматизацию деятельности локального предприятия, но и разработку универсальных алгоритмов и методов, которые при минимальных настройках можно будет применять на предприятиях любого масштаба и уровня компьютеризации.

Понятию «информационная система» уделяется значительное внимание в нормативно-правовых актах, регулирующих данную сферу деятельности. В частности, в ст. 2 ФЗ от 27 июля 2006 г №149-ФЗ «Об информации, информационных технологиях и о защите информации» рассматривается следующее определение информационной системы: это «совокупность содержащейся в базах данных информации и обеспечивающих ее обработку информационных технологий и технических средств» [1].

Как правило, техническое задание на проектирование ИСС включает в себя общие сведения, назначение и цели создания (развития) системы, характеристику объектов автоматизации, требования к системе, состав и содержание работ по созданию системы, порядок контроля и приемки системы, требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие, а также требования к документированию и источникам разработки. Такая структура закреплена документально действующими государственными стандартами в области информатизации [2].

Автоматизированная система обработки данных – комплекс взаимосвязанных методов и средств преобразования данных, необходимых пользователю. Здесь данные понимаются как информация, представленная в формализованном виде, пригодном для автоматической обработки при возможном участии человека.

Информационно-справочная система предназначена для оказания автоматизированной поддержки пользователей на предприятии. Ее преимуществами в данном случае являются существенное увеличение скорости реакции на поступившую заявку пользователей и рациональное использование людских ресурсов. Помимо этого, ИСС обеспечивает круглосуточную работу пользователей в различных регионах и часовых поясах.

Важную роль ИСС играет в решении вопросов поддержки разработки и принятия управленческих решений по сложным проблемам, требующих многовариантных расчетов, организации взаимодействия человека и компьютера. Выработка решений происходит на основании циклического алгоритма, в котором выделяют две ключевые стороны: система поддержки принятия решений в качестве вычислительного элемента и объекта управления и, непосредственно, лицо, принимающее решение, как управляющий элемент, задающий входные данные и оценивающий полученный результат компьютерных вычислений [5, с. 150-151].

Важным условием функционирования ИСС является метод формирования очереди на обработку пользовательских заявок, который позволяет классифицировать заявки по степени важности, по времени создания, а также по степени значимости отправителя.

Автоматизация процессов заключается в повышении качества работы пользователей информационно-справочной системы за счет реализации единой точки регистрации и контроля всех заявок в службу технической поддержки, механизмов самостоятельного

решения задач пользователями, средств контроля сроков выполнения заявок на базе регламентированных каталогов услуг и соглашений об уровне услуг.

В процессе разработки ИСС необходимо уделить внимание ее обучаемости, то есть каждую последующую не разрешимую для автоматизированной системы устранения неисправностей проблему администратор заносит в базу данных, которая предоставляет в последующем инструкции по диагностике неполадок в системе при возникновении аналогичной проблемы.

Для пользователя система должна быть спроектирована унифицированной и прозрачной, так как для него не представляет значимости механизм обработки заявки, а важен только результат. Основные системные принципы включают в себя:

- целостность (зависимость каждого элемента, свойства и отношения от его места и функций внутри целого);
- структурность (возможность описания системы через установление ее структуры, т. е. сетей связей и отношений системы);
- иерархичность (каждый компонент системы в свою очередь может рассматриваться как система, а исследуемая система – как компонент более широкой системы);
- множественность описания (посредством использования множества различных моделей);
- взаимозависимость системы и среды (система формирует и проявляет свои свойства в процессе взаимодействия со средой, являясь при этом активным компонентом взаимодействия) [3, с. 14].

В настоящее время проблемами разработки ИСС занимаются различные компании и корпорации, которые постоянно совершенствуют алгоритмы и возможности данных систем, для того чтобы снизить трудозатраты и повысить производительность труда. Предполагается на основании изучения специфики существующих ИСС разработать уникальную на данный момент систему, которая позволит значительно снизить время обработки заявок и принесет положительный экономический эффект в виде снижения трудозатрат администраторов.

Для достижения поставленной цели планируется использовать методы системного анализа существующих систем поддержки пользователей, их основных особенностей и принципов работы, технологической базы. Существует достаточное количество систем поддержки пользователей на предприятиях. Их общее название по-английски звучит как HelpDesk или ServiceDesk. Все они имеют похожие функции, и в тоже время каждая имеет свои особенности.

Преимущества Open View ServiceDesk заключаются в том, что этот продукт не производит самостоятельно инвентаризационный опрос персональных компьютеров и серверов и не предупреждает самостоятельно о критических ситуациях. Программа предназначена для учета и систематизации информации о состоянии и взаимосвязях компонентов инфраструктуры, контроле вносимых в конфигурацию изменений, регистрации обращений пользователей и организации работы как внутри подразделения (между работниками), так и между подразделениями. Система позволяет структурировать данные с

помощью удобных иерархических классификаторов, определять уникальные для организации взаимосвязи между элементами инфраструктуры, что впоследствии позволяет даже неподготовленному пользователю быстро обнаружить расположение, тип и влияние того или иного устройства.

Развертывание системы Docs Vision ServiceDesk производится в автоматическом режиме. В систему входит полный набор документации, которая облегчает быстрый старт работы с системой: ролевые инструкции работы пользователей, руководство по обслуживанию решения и подробное описание настроек системы. Docs Vision ServiceDesk обладает простым интерфейсом и позволяет пользователю работать как с использованием клиента DocsVision, так и через электронную почту или корпоративный портал. Рабочие места пользователей системы (в частности, диспетчера обращений или исполнителя работ) организованы максимально эргономично и предоставляют полный набор сервисов, необходимых для данной роли.

В ходе сравнения и оценки существующих программных решений в сфере информационно-справочной поддержки сотрудников предприятия, таких как Open View ServiceDesk и Docs Vision ServiceDesk, выявлены ключевые функции и характеристики, оптимальные для ИСС предприятия. Среди существующих ИСС самое большое распространение получили системы с одним администратором. Автоматизация в таких ИСС заключается в приеме заявок, все остальные действия приходится делать человеку. Приведем сравнительную таблицу, которая позволяет наглядно показать преимущества ИСС над остальными.

Таблица

Сравнение ИСС с аналогами

	Системы с одним администратором	Системы с двумя администраторами	Системы с тремя администраторами	Проектируемая ИСС
Среднее время обработки одного запроса	540	630	655	556
Коэффициент качества обработки	0,79	0,73	0,63	0,55
Количество успешно обработанных заявок в день	39	62	110	420

Согласно данным таблицы, видно, что среднее время обработки 1-го запроса у ИСС не многим больше аналогичного параметра у системы с одним обработчиком, это время будет сокращаться с ростом базы данных. Главный показатель эффективности применения ИСС – это количество успешно обработанных заявок за рабочий день.

Таким образом, проектируемая ИСС позволит предприятиям различного масштаба организовать службу технической поддержки пользователей. Система отвечает всем требованиям, за счет чего позволяет уменьшить нагрузку на администратора, что в свою

очередь позволяет экономить на его труде, используя либо меньшее количество людей, либо удаленную работу администратора. Также полностью автоматизирован процесс приема, регистрации и обработки некоторых заявок от пользователей. Система, ориентируясь по типу запроса, умеет направлять его тому обработчику, который специализируется на данных запросах. ИСС снабжена инструментами сбора статистических данных по своей работе, таких как: количество обработанных заявок за единицу времени, среднее время обработки по каждому исполнителю, а также качество обработки по каждому исполнителю. Анализ соответствующих данных позволяет оперативно корректировать функционирование системы.

Применение ИСС в организациях позволит в несколько раз увеличить количество обработанных заявок, сократить время ожидания реакции на запрос, уменьшить расходы по заработной плате. Для предприятий, работающих в разных часовых поясах и имеющих распределенную ИТ-инфраструктуру, посредством ИСС имеется возможность организовать круглосуточную техническую поддержку. Благодаря гибкости ИСС можно использовать за пределами организации и в других регионах.

Информация в современном мире превратилась в один из наиболее важных ресурсов, а информационные системы стали необходимым инструментом практически во всех сферах деятельности. Наиболее важными требованиями к ИСС являются масштабируемость, непротиворечивость и репликация, отказоустойчивость, самообучаемость и кроссплатформенность, что позволит использовать ее преимущества пользователями предприятий различных сфер деятельности.

### Литература

1. Федеральный закон от 27.07.2006 № 149-ФЗ (ред. от 30.12.2020) «Об информации, информационных технологиях и о защите информации» // <https://clck.ru/U4Mid>
2. ГОСТ 34.602–89. Информационная технология. Технические задания на создание автоматизированной системы. <https://clck.ru/UFtqi>
3. Барахнин В. Б., Федотов А. М. Информационная система: взгляд на понятие // Вестник Новосибирского государственного университета. Серия: Информационные технологии. 2007. Т. 5. №2. С. 12-19.
4. Галанкин А. В., Войцеховский С. В., Прохоров М. А. Разработка информационно-справочной системы в интересах развития информационных систем специального назначения // Известия Тульского государственного университета. Технические науки. 2018. №1. С. 327-335.
5. Мударисова А. Р., Хайруллина Л. Р. Информационные технологии в разработке и принятии управленческих решений // Экономика и бизнес: теория и практика. 2019. №5-2. С. 149-153.

© Алмакаев Д.Р., Кошкаров А.В., 2021

**Игнатов С.В., Мамедли Р.Э.**, канд. физ.-мат. наук  
Нижевартовский государственный университет,  
г. Нижневартовск, Россия

## ФУНКЦИИ ОБРАТНОГО ВЫЗОВА В AJAX

Аббревиатура AJAX означает Asynchronous JavaScript and XML. JavaScript управляет динамическим контентом и взаимодействием с пользователем на WEB-сайте. Как видно из аббревиатуры, AJAX это концепция использования JavaScript и XML для выполнения, в основном, асинхронных запросов к серверу [2]. Асинхронный метод передачи данных позволяет браузеру продолжать работать во время выполнения запроса. Эта концепция позволяет обращаться к серверу без перезагрузки страницы, что уменьшает время отклика и увеличивает интерактивность интерфейса, а также уменьшает объем передаваемых данных [1].

Под функцией обратного вызова или callback функцией в контексте модели AJAX понимается функция, которая вызывается в качестве аргумента другой функции, например такие функции могут выполняться в случае если приходит ответ на некий запрос. Распространенный пример использования – добавление товаров в корзину в онлайн-магазине: кликая на товар, который вы хотите приобрести, не обновляется вся страница целиком, но выполняется функция обратного вызова, которая обновляет содержимое корзины.

```
> function reading(book, callback) {  
    alert(`Started reading ${book}`);  
    callback();  
}  
  
reading('Dead souls', function() {  
    alert('Finished reading');  
});
```

Рис. 1. Пример функции обратного вызова

В приведенном примере функция «callback» является функцией обратного вызова, или callback-функцией. Для того чтобы ее вызвать, ее нужно определить в качестве аргумента в функцию reading. Браузер выведет один за другим два сообщения: “Started reading Dead souls” и “Finished reading”.

Рассмотрим пример, в котором функция загружает на страницу новый скрипт:

```
> function loadScript(src) {  
    let script = document.createElement('script');  
    script.src = src;  
    document.head.append(script);  
}  
  
> loadScript('/myDirectory/script.js');
```

Рис. 2. Пример загрузки скрипта



JavaScript – событийно ориентированный язык, это означает что код не дожидается выполнения отдельных функций и продолжает выполнение, в том числе вызывает другие функции. В приведенном примере код не будет дожидаться пока скрипт загрузится и работает.

Передадим callback вторым аргументом:

```
> function loadScript(src, callback) {  
  let script = document.createElement('script');  
  script.src = src;  
  
  script.onload = () => callback(script);  
  
  document.head.append(script);  
}
```

Рис. 3. Добавление параметра callback

И вызываем:

```
> loadScript('/myDirectory/script.js', function() {  
  newFunction();  
});
```

Рис. 4. Вызов callback функции

Теперь функция будет вызвана, когда скрипт загрузится.

Так же в случае, когда есть необходимость последовательного выполнения функций, могут использоваться функции обратного вызова.

```
> loadScript('/myDirectory/script.js', function(script) {  
  alert(`Script ${script.src} loaded, next one`);  
  loadScript('/myDirectory/script2.js', function(script) {  
    alert(`Loaded`);  
  });  
});
```

Рис. 5. Вызов двух скриптов последовательно

Желательно не вызывать таким способом несколько функций, поскольку такой подход может привести к путанице<sup>[3]</sup>.

Для упрощения написания кода могут применяться API, например XMLHttpRequest или jQuery. Для отправки HTTP запросов без перезагрузки страницы может быть создан экземпляр объекта XMLHttpRequest. Сначала регистрируется функция обратного вызова с помощью XMLHttpRequest и асинхронно посылается запрос. Контроль останется за браузером, а функция обратного вызова будет вызвана, когда придет ответ сервера.

На web-сервере запрос придет так же, как и любой другой HttpServletRequest. После анализа параметров запроса, сервлет выполнит все необходимые действия для приложения, сериализует его запрос и вписывает его в HttpServletResponse.

Возвращаясь к клиенту, функция обратного вызова, зарегистрированная в XMLHttpRequest, теперь выполняется для того, чтобы обработать XML-документ, возвращаемый сервером. Наконец, пользовательский интерфейс обновляется в соответствии с данными сервера, используя код JavaScript для преобразования страницы. Рассмотрим небольшой пример:

```
var request = new XMLHttpRequest();  
request.open('GET', 'http://www. http://nvsu.ru/', true);  
request.send(null);
```

Рис. 6. Обратный вызов в XMLHttpRequest

В примере в первой строке создается экземпляр класса XMLHttpRequest. Во второй строке “GET” либо “POST” это тип запроса. В первом случае происходит обращение на сервер через URL, во втором в теле запроса в виде .send (body). “true/false” означает асинхронное и синхронное использование соответственно. В третьей строке .send отправляет запрос.

У XMLHttpRequest есть состояния, которые меняются по мере выполнения запроса от 0 для «неопределенного» и до 4 для «завершенного». Текущее состояние можно посмотреть в свойстве .readyState.

В следующем примере getReadyStateHandler() - возвращает значение функции, которая проверяет, завершился ли XMLHttpRequest и передает ответ в XML-формате в оператор req (XMLHttpRequest):

```
1 function getReadyStateHandler(req, responseXmlHandler) {  
2  
3 // Возвращает неопределенную функцию, которая считывает  
4 // данные XMLHttpRequest return function () {  
5  
6 // Если требуется статус "закончен"  
7 if (req.readyState == 4) {  
8  
9 // Проверяем, пришел ли успешный ответ сервера  
10 if (req.status == 200) {  
11  
12 // Передает XML оператору  
13 responseXmlHandler(req.responseXML);  
14  
15 } else {  
16  
17 // Возникла ошибка HTTP  
18 alert("HTTP error: "+req.status);  
19 }  
20 }  
21 }  
22
```

Рис. 7. ReadyState в XMLHttpRequest

В заключение, функции обратного вызова – это то что можно назвать ядром AJAX, благодаря такому инструменту интернет смог стать таким, какой он есть сегодня – удобный и интерактивный.

### Литература

1. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. 4-е изд. СПб., 2016. 768 с.
2. Пауэрс Ш. Добавляем Ajax: Пер. с англ. СПб., 2009. 448 с.

© Игнатов С.В., Мамедли Р.Э., 2021

УДК 004.85

**Насыпова Э.А., Развеева И.Ф.**

Донской государственный технический  
университет, г. Ростов-на-Дону, Россия

## **ПРИМЕНЕНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ФОРМИРОВАНИЯ ПОРТРЕТА КЛИЕНТА ПСИХОЛОГИЧЕСКОГО ЦЕНТРА**

Для того, чтобы процесс определения типа помощи для клиента психологического центра проходил быстрее и был наиболее эффективным, стоит заранее иметь некое представление о клиенте – его общий портрет. Анамнез данных о жизни пациента является важной частью подготовки гипотезы диагноза специалистом. Сведения фиксируются в психологической карте для дальнейшего анализа [1, с. 32]. Сокращение времени систематизации собранной информации на сегодняшний день – это одно из ключевых и актуальных направлений в работе психологических центров.

Цель данной работы – создание пользовательского интерфейса с внедренной моделью машинного обучения для определения типа психологической помощи. Источником информации для исследования выступает психологический центр, расположенный в городе Ростов-на-Дону. Для достижения поставленной цели предполагается решить следующие задачи: изучить основные индикаторы состояния клиента психологического центра; сформировать анкету и провести опроса респондентов на основе ключевых метрик; изучить классификацию методов машинного обучения; решить задачу классификации для определения вида первичной психологической помощи; реализовать пользовательский интерфейс.

После определения основных индикаторов состояния клиентов необходимо провести анкетирование, оценить их удовлетворенность оказанными услугами и на основе полученной информации выделить скрытую логику, которая приводит к наиболее эффективному и быстрому результату (в данном случае – улучшение состояния обратившегося). При помощи метода анкетирования можно с наименьшими затратами за короткий срок получить высокий уровень массовости исследования [4, с. 1264].

В качестве основной информационной технологии для проведения анкетирования была выбрана платформа Google. Сервис «Google Формы» позволяет создавать тесты и опросы, отправлять их другим пользователям и получать ответы. Статистику ответов, в том числе в виде диаграммы, можно увидеть прямо в форме, а ответы респондентов – в автоматически созданной таблице Google.

На основе вопросов, определенных специалистами психологического центра, была создана анкета, отправленная респондентам по электронной почте с почтового ящика психологического центра для проведения заочного анкетирования. Также со своего гаджета или компьютера в психологическом центре респонденты могли пройти анкетирование очно.

Всего анкету прошли 1075 респондентов от 18 до 78 лет. Для дальнейшей обработки были взяты 1000 анкет, в которых респондент дал оценку выше 7 на вопрос о качестве

оказания помощи, что говорит об удовлетворенности клиента оказанной услугой. Согласно статистике, доступной в сервисе «Google Формы», оценку выше 7 (включительно) дали порядка 93% респондентов.

**Дорогие друзья!**  
\* Обязательно

Ваш возраст?  
Мой ответ \_\_\_\_\_

Количество членов в Вашей семье?  
Мой ответ \_\_\_\_\_

Укажите среднюю заработную плату за последний год.  
Мой ответ \_\_\_\_\_

Выберите способ общения со специалистом центра

- дистанционно
- очно
- дневной стационар
- на постоянной основе (стационар)

Какой тип терапии Вы предпочитаете?

- Арт-терапия
- Гештальт-терапия
- Психоанализ
- Гипноз
- НЛП

Психологические консультации какого типа оказались для Вас наилучшими?

- семейный психолог
- индивидуальные консультации
- групповые консультации
- реабилитационный центр
- психиатр

На сколько по 10-ти бальной шкале Вы оцениваете помощь от специалистов психологического центра? \*

1 2 3 4 5 6 7 8 9 10

Неудовлетворительно           Отлично

**Отправить**

Никогда не используйте формы Google для передачи паролей.  
Компания Google не имеет никакого отношения к этому контенту. [Сообщение о нарушении](#) · [Условия использования](#) · [Политика конфиденциальности](#)

Рис. 1. Google Форма

По итогам опроса был сформирован набор данных (датасет), содержащий 1000 строк.

	A	B	C	D	E	F	G	H
		salary	city	age	form_preference	type_preference	family_members	type_of_therapy
1	0	196000	Ростов-н/Д	25	Очно	Арт-терапия	1	Семейный психолог
2	1	152000	Азов	60	Дистанционно	Гештальт-терапия	1	Психиатр
3	2	83000	Таганрог	49	Дистанционно	Арт-терапия	2	Индивидуальная консультация
4	3	161000	Ростов-н/Д	41	Очно	Арт-терапия	0	Семейный психолог
5	4	146000	Ростов-н/Д	58	Дистанционно	Гештальт-терапия	1	Индивидуальная консультация
6	5	209000	Новочеркасск	55	Дистанционно	Гештальт-терапия	2	Групповая консультация
7	6	59000	Новошахтинск	53	Дневной стационар	Арт-терапия	1	Психиатр
8	7	73000	Шахты	55	Очно	Психоанализ	2	Семейный психолог
9	8	216000	Таганрог	75	Очно	Гештальт-терапия	1	Индивидуальная консультация
10	9	56000	Батайск	60	Очно	Арт-терапия	2	Семейный психолог
11	10	12000	Батайск	63	Дистанционно	Гештальт-терапия	1	Семейный психолог
12	11	82000	Ростов-н/Д	60	Дистанционно	Психоанализ	0	Психиатр
13	12	43000	Ростов-н/Д	59	Дистанционно	Гештальт-терапия	2	Реабилитационный центр
14	13	63000	Ростов-н/Д	67	Очно	Гештальт-терапия	2	Групповая консультация
15	14	46000	Батайск	56	Дистанционно	Гештальт-терапия	2	Реабилитационный центр
16	15	69000	Ростов-н/Д	45	Дистанционно	Гипноз	3	Групповая консультация
17	16	176000	Азов	26	Дневной стационар	Гипноз	2	Групповая консультация
18	17	88000	Ростов-н/Д	36	Дистанционно	Арт-терапия	0	Реабилитационный центр
19	18	158000	Сальск	56	Дистанционно	Гештальт-терапия	1	Психиатр

Рис. 2. Датасет

Для дальнейшего исследования сформированный датасет использован для обучения модели машинного обучения с целью выявления закономерностей и скрытой логики, которая

стоит за принятием решения при определении последнего ключевого столбца таблицы т.е. типа терапии. Данная задача представляет собой задачу классификации, где входные данные – это признаковое описание объекта, а классы – непересекающиеся [2, с. 336]. В машинном обучении задача классификации относится к разделу обучения с учителем.

Для реализации модели машинного обучения была применена среда Google Colaboratory, где программирование осуществляется на высокоуровневом языке Python. С помощью библиотеки Pandas был проведен первичный анализ датасета, определены основные черты клиента центра по ключевым признакам:

1. Средний доход для большинства клиентов центра – от 50 до 100 тыс.руб.
2. Средний возраст для большинства клиентов центра – от 55 до 65 лет.

Это связано с пандемией коронавируса, охватившей мир в 2020 году. Для многих взрослых людей серьезным испытанием стала социальная изоляция. Во время карантина для многих людей чувство одиночества стало выраженным и болезненным. Пожилые люди ждут внимания, сочувствия, готовности слушать их и внимать советам, ободрения и просто теплых слов. В частных домах престарелых эту проблему решают ведением групповой терапии, которую осуществляет врач. В домашних условиях все сложнее, родственники все чаще обращаются за помощью к специалистам. Психологический центр расширил свою помощь для данной категории граждан: консультации осуществляются по скидкам, учитываются льготы отдельных категорий граждан, также имеются бесплатные дистанционные программы.

3. Наибольшее количество клиентов проживают в г. Ростов-на-Дону.

4. Учитывая, что большинство клиентов центра в исследуемый период времени относятся к гражданам в возрасте 55+, наиболее предпочтительным является дистанционный формат работы. Стационар не востребован также по причине пандемии.

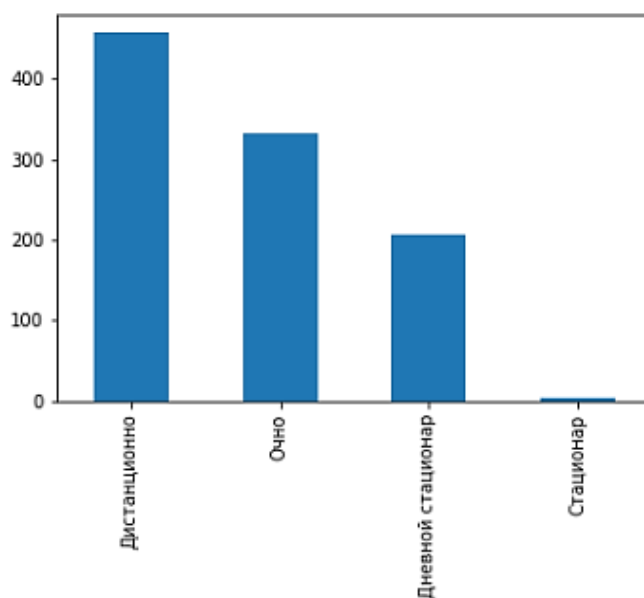


Рис. 3. Распределение по предпочитаемым формам взаимодействия

5. Сейчас уровень психологической и эмоциональной культуры в обществе постепенно повышается, никого уже не удивляет, что в перечне услуг психологического центра повышенным спросом отличается гештальт-терапия. В процессе психологической работы с



пожилыми людьми направление арт-терапии также имеет огромное значение, привлекая своим разнообразием упражнений и техник.

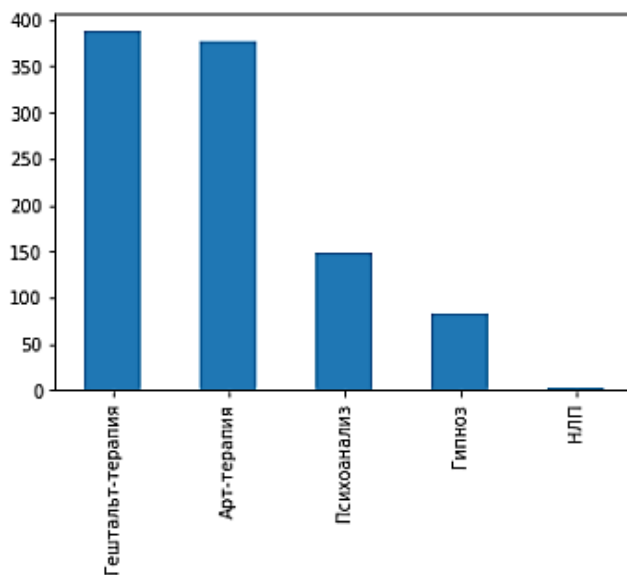


Рис. 4. Распределение по типам терапии

У специалистов по обработке и анализу данных есть множество средств для создания классификационных моделей. Один из самых популярных и надежных методов разработки таких моделей заключается в использовании алгоритма «случайный лес» (Random Forest, RF) [3, с. 15-17]. Универсальность данного метода заключается в том, что он хорош во многих задачах (в 70% из встречающихся на практике задач, если не учитывать работу с изображениями).

Реализовав классификатор случайный лес из пакета Python scikit-learn, следует оценить качество построенной модели. В рамках решения данной задачи классификации была выбрана метрика точность (accuracy), означающая долю правильных ответов алгоритма. Эта метрика полезна в задачах с равными классами.

Согласно рисунку 5 точность модели составила 0.99 на обучающих данных и 0,91 на тестовых.

```
print ('RF accuracy: TRAINING', model.score(X_train,y_train))
print ('RF accuracy: TESTING', model.score(X_test,y_test))

RF accuracy: TRAINING 0.9988888888888889
RF accuracy: TESTING 0.91
```

Рис. 5. Оценка качества модели

Для того, чтобы процесс определения типа помощи проходил быстрее, был разработан пользовательский интерфейс, представляющий собой форму на рабочем столе сотрудника психологического центра. Данная форма требует наличия операционной системы Windows, других технических требований не имеется.

В процессе разработки было создано GUI приложение с помощью кроссплатформенной событийно-ориентированной графической библиотеки на основе средств Tk - Python Tkinter.

Рис. 6. Пользовательский интерфейс

В данном исследовании обработка данных осуществлялась с точки зрения современного и актуального подхода Data Science, включающего в себя новые принципы сбора, анализа, обработки и визуализации массивов данных.

Благодаря машинному обучению программист не обязан писать инструкции, учитывающие все возможные проблемы и содержащие все решения. Вместо этого в программу закладывают алгоритм самостоятельного нахождения решений путем комплексного использования статистических данных, из которых выводятся закономерности и на основе которых делаются прогнозы.

Расширение представлений о возможностях применения методов машинного обучения в автоматизации работы психологического центра позволит вывести работу персонала на новый качественный уровень.

### Литература

1. Боттаев Н.А., Дьяченко Е.В., Самойленко Н.В., и др. Навыки эффективного общения для оказания пациент-ориентированной медицинской помощи. М., 2018. 32 с.
2. Грас Дж. Data Science. Наука о данных с нуля / Пер. с англ. Дж. Грас. СПб., 2017. 336 с.
3. Летова М.С. Реализация регрессионных и классификационных задач с помощью метода Random Forest // E-Scio. 2017. № 8 (11). С. 15-21.
4. Blinova N., Vasilyeva O., Kondratieva T. Research of social and psychological aspects of forgiveness // XVI European Congress of Psychology (ECP 2019) (Moscow, July 02–05, 2019). М., 2019. P. 1264.

© Насыпова Э.А., Развеева И.Ф., 2021

УДК 62-26

**Осмаков Н.А.**

*Научный руководитель: Жалилова А.А.*

Самарский государственный технический университет», г. Самара, Россия

## **АВТОМАТИЗИРОВАННЫЙ КОВШ СНЕГОУБОРОЧНОЙ МАШИНЫ ДЛЯ УДАЛЕНИЯ НАЛЕДИ**

В настоящее время уборка снега с тротуаров и дорог достаточно хорошо развита, однако, очищение ледяных образований не всегда удается и происходит эффективно, даже с использованием снегоуборочной техники. Довольно часто с этими образованиями справляются путем посыпания автомобильных и пешеходных дорог химическими реагентами или очищение ручным трудом. Посыпание реагентами негативно сказывается на окружающей обстановке: реагенты наносят вред природе, их испарения негативно влияют на человека, вызывая аллергию, в частности когда эти реагенты попадают в закрытые помещения вместе с грязью на обуви, а также такой метод борьбы с ледяными образованиями портит эстетический вид города. Использование ручного труда для удаления льда малоэффективно и продолжительно.

При уборке широко используются распространенные мобильные снегоуборочные машины на базе bobcat, однако, их эффективность во время перепадов температур с плюсовой на минусовую, когда образуются ледяные образования на дороге, не поддающиеся уборке основательно, можно повысить путем использования специального ковша.

Рассмотрим основные преимущества очистки снега и ледяных образований с помощью данного ковша с вибрирующей платформой:

1. Не вредит природе. При работе ковша не используются какие-либо химические реагенты
2. Легко масштабируем. Данный способ не нуждается в разработке новой узкоспециализированной техники, ковш может быть изготовлен под уже существующие и широко используемые технические средства, в частности семейств «Беларусь» и «Bobcat»
3. Эффективно убирает ледяные образования на тротуарах и дорогах, что повышает безопасность передвижения как людей, так и транспортных средств
4. Не наносит существенного ущерба дорожным покрытиям

Также, имеется ряд недостатков:

1. Дороговизна по сравнению с обычными ковшами
2. Невозможность осуществлять работы, где неизбежны большие нагрузки на ковш.

Чтобы очищать дороги с помощью данного ковша владеть какими-либо специальными навыками, которые отличались бы от навыков существующих операторов снегоуборочной техники, не требуется. Но стоит учитывать специфическую конструкцию и не перегружать ковш.

Разберем основные компоненты, входящие в состав устройства. За основу взят «двухчелюстной ковш» с официального сайта Bobcat. Такой ковш отлично справляется с задачей уборки снега и легкого строительного мусора, его конструкция довольно проста и практична. Так как ковши бывают довольно разными по размеру, возьмем приблизительные размеры 987x2030x853 (рис. 1). в будущем ковш с вибрирующей плитой можно будет адаптировать и реализовать практически под любые размеры (<https://clck.ru/U4PRr>).

Description	Item number	Shipping weight (kg)	Operating weight (kg)	Overall length (A) (mm)	Overall width (B) (mm)	Overall height (C) (mm)
Combination Bucket 158 cm	7167310	366.0	366.0	-	1580.0	-
Combination Bucket 173 cm	7167311	386.0	386.0	879.0	1730.0	719.0
Combination Bucket 1880 mm BT	7167312	407.0	407.0	879.0	1880.0	719.0
Combination Bucket 2030 mm BT	7167313	487.0	487.0	987.0	2030.0	853.0
Combination Bucket 213 cm	7167314	503.0	503.0	987.0	2130.0	853.0

Рис. 1. Характеристики двухчелюстного ковша с сайта Bobcat

Чтобы иметь визуальное представление, как выглядит ковш, приведем изображение с того же сайта производителя техники на рисунке 2:

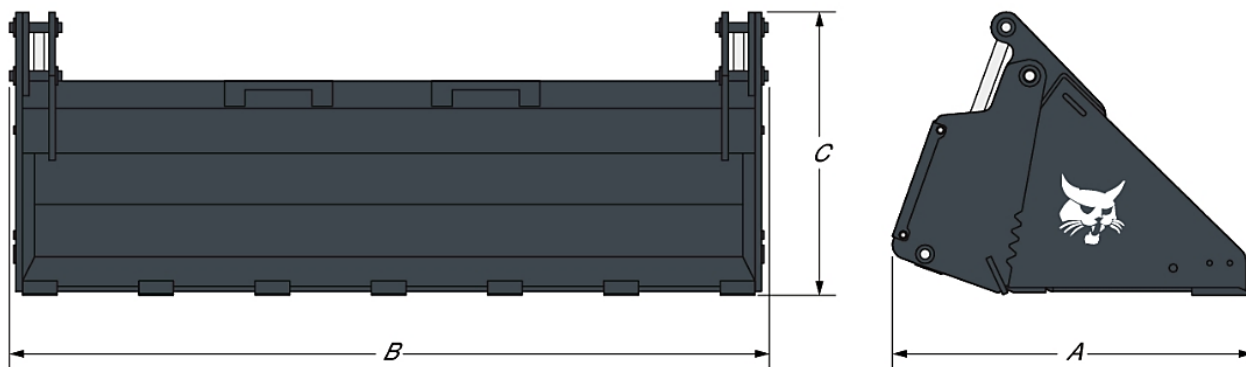


Рис. 2. Изображение ковша на сайте Bobcat

В качестве устройства для создания вибрационных воздействий взят вибромотор ВС30-0020 от производителя ООО Акетон. Данный вибромотор работает от сети постоянного тока, что позволит его подключить к аккумулятору без необходимости использования дополнительных элементов в электропитании. Характеристики данного устройства предоставлены на рисунке 3 (3 <https://clck.ru/UG2Zj>).

Код	Тип	Размер	Механические характеристики				Электрические характеристики			
			об/мин	Статический момент, кгмм	Центробежная сила, кг	Вес, кг	Потребляемая мощность, Вт	Напряжение		
								12 В	24 В	
BC30-0020	BMCC 200/3	01	3100	18,6	200	6,1	130	10,8	5,4	
BC60-0020	BMCC 200/6	01	5500	5,26	178	5,8	130	10,8	5,4	
BC30-0040	BMCC 400/3	02	3100	35,93	386		ождается			
BC60-0040	BMCC 400/6	02	5500	13,3	450		начало			
BC30-0110	BMCC 1100/3	04	3600	75,9	1100		производства			

Рис. 3. Характеристики вибромоторов

Габаритные размеры устройства, а также присоединительные размеры предоставлены на том же сайте производителя (рис. 4).

Код	Тип	Размер	Габаритные размеры, мм								Присоединительные размеры, мм							
			Рис.№	A	B	C	D	ØE	F	G	L	M	ØN	P	Q	R	S	PC
BC30-0020	BMCC 200/3	01	1	212	150	150	132	105	59	46	62+74	106	4 x M8	99	30	35	10	M16x1,5
BC60-0020	BMCC 200/6	01	1	212	150	150	132	105	59	46	62+74	106	4 x M8	99	30	35	10	M16x1,5
BC30-0040	BMCC 400/3	02	2	216	166	184	164	130	75	50	90	125	4 x M12	120	30	40	12	M20x1,5
BC60-0040	BMCC 400/6	02	2	216	166	184	164	130	75	50	90	140	4 x M10	120	30	40	12	M20x1,5
BC30-0110	BMCC 1100/3	04	3	355	210	236	216	181	101	71	120	170	4 x M16	160	40	43	20	M20x1,5

Рис. 4. Габаритные и присоединительные размеры вибромоторов

Схема вибромоторов BC30-0020 и BC60-0020 предоставлена на рисунке 5:

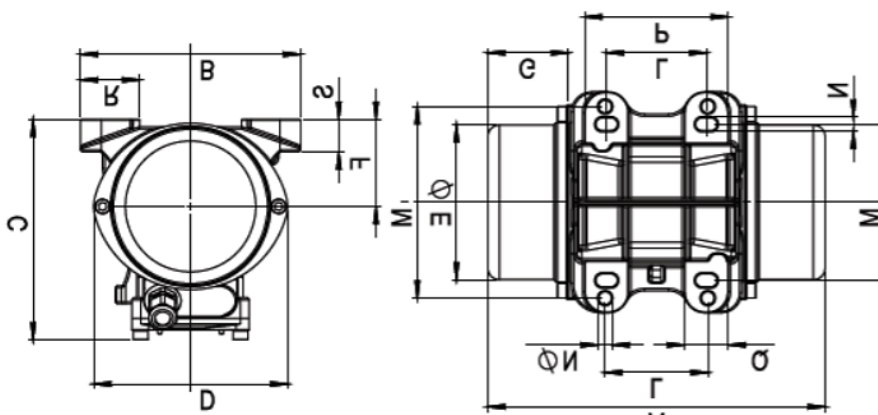


Рис. 5. Схема для вибромоторов BC30-0020 и BC60-0020

В качестве крепления плиты к ковшу взяты виброизоляторы резинометаллические K0566 от производителя ООО «Альтервиа». Их конструкция изображена на рисунке 6. Крепление плиты через гасители позволит снизить пагубное влияние вибрации на компоненты ковша, что позволит увеличить срок его службы. Негативным эффектом является снижение конструкционной прочности и исключение высоких нагрузок на плиту (<https://clck.ru/KZkvD>).



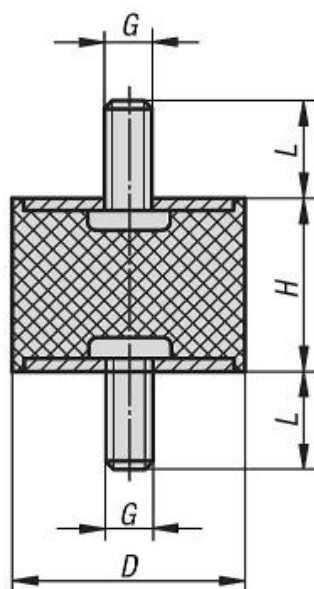


Рис. 6. Виброизолятор K056

С использованием выше описанных комплектующих, изобразим чертеж ковша с вибрирующей плитой на рисунке 7. Данная конструкция позволяет изготавливать данные ковши на основе уже существующих, что ускорит его внедрение в массовое производство.

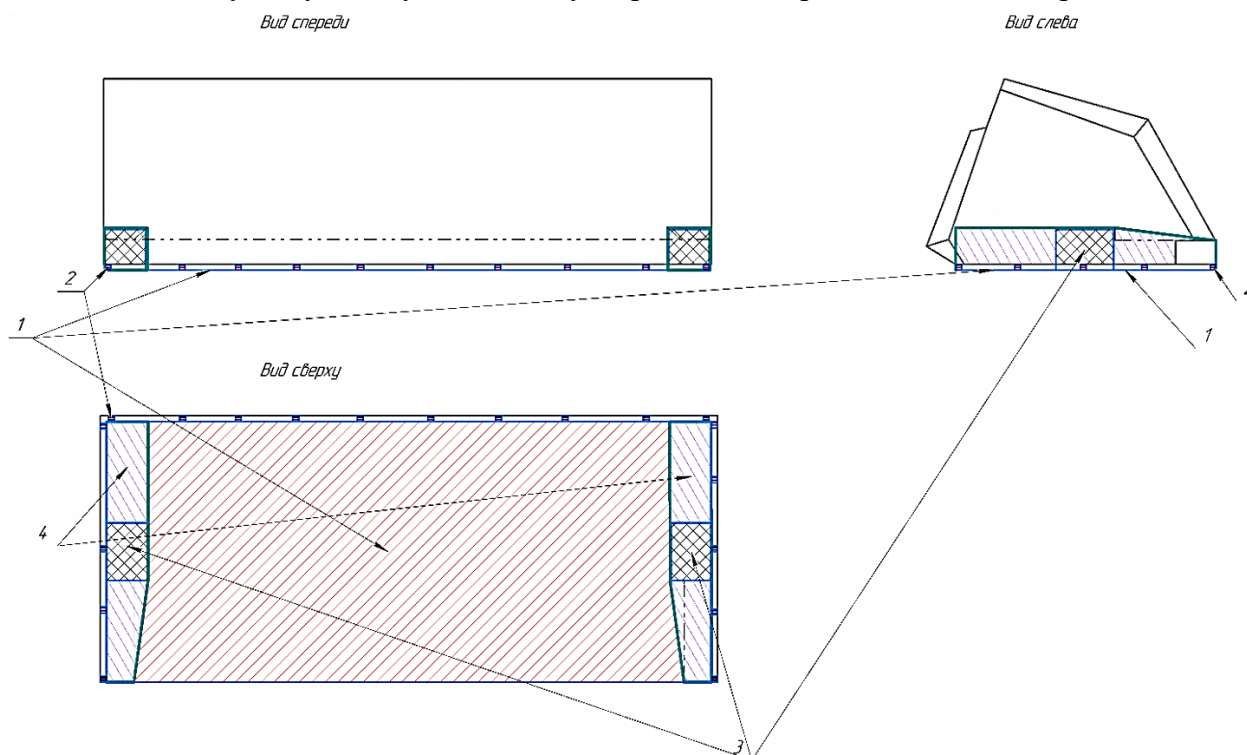


Рис. 7. Схема ковша 1 плита с продольными выступами для очистки поверхности от ледяных образований; 2 виброизоляторы; 3 Вибромоторы; 4 защитный короб

Черным цветом обведена часть контура ковша, которая берется без дополнительных доработок. Дно, заштрихованное красным под цифрой 1, выполнено из цельной металлической плиты с продольными выступами на внешней стороне. Данная платформа крепится к основной конструкции ковша посредством виброизоляторов 2, расположенных

почти по всему периметру, что обеспечивает устранение негативного влияния излишних вибрационных воздействий, но снижает конструкционную прочность к нагрузкам. Вибромоторы, обозначенные под цифрой 3 и заштрихованные сеткой, жестко крепятся к платформе 1, это обеспечит максимально полезное КПД передачи вибрационных усилий от вибромотора к платформе. Моторы и гасители закрыты специальным коробом 4, заштрихованным фиолетовым, чтобы избежать попадания в важные компоненты посторонних веществ, снега, и таким образом повысить срок службы ковша.

Описание принципа работы устройства. Ковш опускается на очищаемую поверхность, включаются вибромоторы 3. Под действием вибрационных воздействий на плиту 1, слой ледяных образований и уплотненного снега на поверхности разрушается и соскабливается ковшом. По мере заполнения ковша, снег сбрасывается в сторону путем манипуляций оператора, как при обычной очистке снега стандартным ковшом. При выключенных вибромоторах ковш также может использоваться для очистки от снега, однако следует учесть, что ковш не предназначен для высоких нагрузок подъема/опускания, а также сильных ударных и статических воздействий.

В качестве аналогов ковша можно считать все существующие ковши, а также некоторые патентные модели: ближайшие – это «Устройство для скалывания снежно-ледяных образований на дорожных покрытиях», ПМ № 177201, «Устройство для скалывания льда», ПМ № 121265. Однако первое устройство может нанести существенный вред асфальтовому покрытию, в отличие от разрабатываемого, а второе предназначено исключительно для ручной эксплуатации [1, 2].

Таким образом, данное устройство значительно автоматизирует удаление ледяных образований и снега на тротуарах и дорогах, позволяет меньше использовать химических реагентов, повышает эффективность и производительность труда при очистке улиц, не наносит существенного вреда дорожным покрытиям.

### Литература

1. Бушуев В.А, Семенова И.В., Семенов А.Н., Семенов Д.Н, Земский А.В. Устройство для скалывания льда пат. 121265 Российская Федерация, МПК E01H 5/12 (2006.01) №2012125872/13; заявл. 20.06.2012 опубл. 20.10.2012 Бюл. №29.
2. Третьяков Д.А. Устройство для скалывания снежно-ледяных образований на дорожных покрытиях пат. 177201 Российская Федерация, МПК E01H 5/12 (2006.01) №2017113323; заявл. 17.04.2017 опубл. 13.02.2018, Бюл. №5.

© Осмаков Н.А., Жалилова А.А., 2021

УДК 004.923

**Харитонов Л.С.**

*Научный руководитель: Эварт Т.Е., канд. физ.-мат. наук*  
Арзамасский политехнический институт,  
филиал Нижегородского государственного технического  
университета им. Р.Е. Алексеева, г. Арзамас, Россия

## **КОМПЬЮТЕРНАЯ ВИЗУАЛИЗАЦИЯ АНАТОМИИ И ФИЗИОЛОГИИ ГОЛОВНОГО МОЗГА ЧЕЛОВЕКА С ПОМОЩЬЮ ПРОГРАММНОГО ПАКЕТА WOLFRAM MATHEMATICA**

Визуализация (от лат. *visualis* – «зрительный») – метод представления числовой информации или физического явления (любой абстрактной информации) в виде, удобном для зрительного восприятия и анализа.

Нейровизуализация – общее название методов исследования головного мозга, позволяющих в наглядной графической форме отобразить особенности его прижизненной структуры и функционирования. Поэтому нейровизуализация включает в себя две обширные категории – структурная визуализация, которая описывает структуру головного мозга, и функциональная нейровизуализация, которая используется для исследований в области неврологии, когнитивной психологии и конструирования нейрокомпьютерных интерфейсов (НКИ). Так, функциональная нейровизуализация делает возможной, например, визуализацию обработки информации в центрах головного мозга. В медицине существует множество методов нейровизуализации – компьютерная томография (КТ), магнитно-резонансная томография (МРТ), диффузионно-тензорная томография (ДТТ или ДТ-МРТ), позитронно-эмиссионная томография (ПЭТ), метод вызванных потенциалов (МВП), однофотонная эмиссионная компьютерная томография (ОФЭКТ) и т.д. С помощью структурной визуализации исследуют, например, опухоли головного мозга и черепно-мозговые травмы, а с помощью функциональной – метаболические расстройства на ранней стадии (например, болезни Альцгеймера и Гентингтона) [3; 4].

Мозг является сложнейшим органом, поэтому задачи его изучения и нейровизуализации всегда были интересны и актуальны. Существует множество компьютерных и мобильных приложений, позволяющих осуществить структурную нейровизуализацию. Это, например, “Complete Anatomy”, “Touch Surgery”, “ZygoteBody”, “BioDigital Human”, “Anatomy Learning”, “Human Anatomy Atlas”, “Essential Anatomy 5” и др. В основном, они предназначены для студентов медицинских специальностей колледжей и вузов и служат как интерактивные анатомические атласы. Однако, несмотря на всю свою многофункциональность, визуализация реализуется недостаточно гибко. Например, зачастую невозможно окрасить ту или иную анатомическую структуру в удобный пользователю цвет, задать нужный стиль, вывести не целую систему или отдельный орган, а какую-либо конкретную анатомическую структуру, входящую в состав органа, отобразить сложные связи между системами и органами, и т. д.

Рассмотрим структурную и функциональную нейровизуализацию с помощью системы компьютерной алгебры Wolfram Mathematica 12.2. Реализованные в этом программном пакете

широчайшие возможности аналитических вычислений, функционального программирования, численных расчетов, статистической обработки и визуализации данных, моделирования сложных систем, анализа сигналов и многие другие, позволяют использовать ее для сложнейших научных и практических расчетов. Используя данную систему, можно визуализировать анатомические структуры головного мозга человека несколькими способами. Например, с помощью функций “AnatomicalStructure” и “AnatomyPlot3D” можно различными способами строить анатомические структуры и выводить их в виде проекций или трехмерных графиков в разных стилях, например, соответственно типу медицинского исследования, как рисунок в анатомическом атласе, или же в новом, пользовательском стиле.

Головной мозг – основной орган центральной нервной системы человека, управляющий многими функциями организма, включая частоту сердечных сокращений, способность к ходьбе и бегу, мышление и т. д. Визуализируем его в виде трехмерного графика в стиле “XRay”, что соответствует рентгенологическому исследованию (рис. 1). Очень важной частью головного мозга является мозжечок. Он принимает участие в регуляции произвольных движений и в сохранении равновесия. Визуализируем его трехмерную модель в стиле “Detailed”: выводится стандартный рисунок, как в анатомическом атласе, но с линейками, позволяющими уточнить размеры (рис. 2).

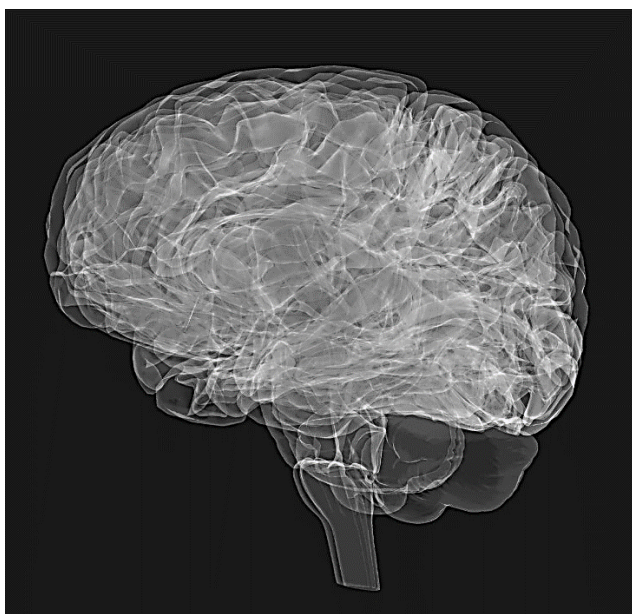


Рис. 1. 3D-визуализация головного мозга в стиле “XRay”

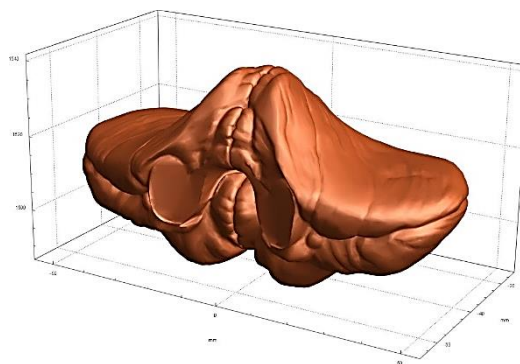


Рис. 2. 3D-визуализация мозжечка в стиле “Detailed”

Рассмотрим теперь визуализацию сложных мозговых структур. Например, внутри головного мозга расположена система сообщающихся друг с другом полостей, называемых желудочками мозга. В полушариях и стволе мозга четыре желудочка, где секретируется цереброспинальная жидкость, которая окружает головной и спинной мозг, защищая их от травм и инфекций [1, с. 20]. Мозговые желудочки имеют сложное по форме строение, о чем трудно догадаться, глядя на их двумерное изображение на срезах головного мозга. Но глядя на трехмерное изображение мозга имеем возможность лучше оценить размер и форму



желудочков, а также их положение в мозге. При визуализации можно задавать различные параметры: угол обзора, углы показа, точку обзора, направление вертикали отображаемого объекта, размер изображения, зазор для сферы, описанной вокруг ограничительной рамки, базовый стиль, масштабированные координаты на изображении, цвета, тон, фон, зеркальность, освещение, различные опции рендеринга и т.д. Отобразим четыре желудочка, а также связывающие их отверстия и водопровод в пользовательском стиле (золотистыми с эффектом зеркальности и освещением) внутри головного мозга (рис. 3). Для понятности можно их подписать, задав цвет и координаты для стрелок и самих подписей.

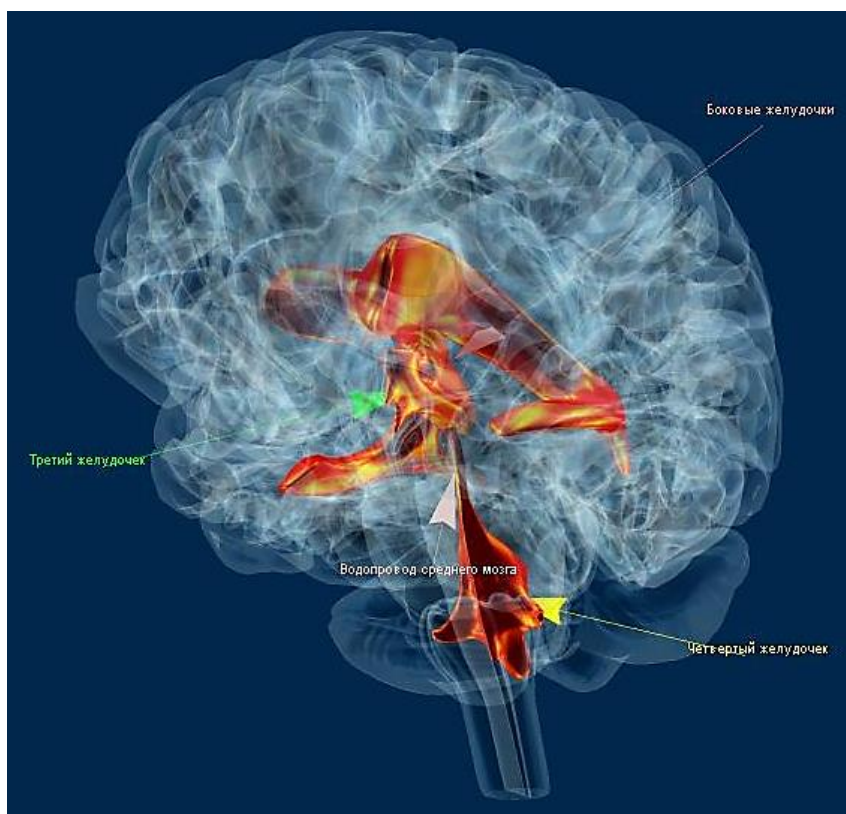


Рис. 3. Трехмерная визуализация желудочков мозга в пользовательском стиле

Одной из самых сложных систем головного мозга является так называемая лимбическая система. Она представляет из себя кольцо взаимосвязанных структур, находящееся в глубине головного мозга. Связи между этими структурами очень сложны. Лимбическая система образует связи с другими его участками и играет важную роль в процессах запоминания и в регуляции настроения, отвечает за восприятие эмоций и ответ на них [1, с. 26-27]. Важнейшей структурой лимбической системы является миндалина (амигдала, действительно миндалевидное ядро – скопление серого вещества). В мозге две миндалины – по одной в каждом полушарии. Они играют важнейшую роль в формировании и восприятии эмоций, как отрицательных, так и положительных. Визуализируем правую миндалину, окрасив ее в красный цвет и отобразив в правом полушарии мозга внутри головы (рис. 4).

Как и многие другие биологические системы, наша нервная система запрограммирована на получение положительной и отрицательной обратной связи. Обратная связь – один из ключевых аспектов обработки информации мозгом, она позволяет увеличивать или



уменьшать эффективность передачи, а также выполнять точную обработку получаемых результатов. Для отображения сложных связей миндалины можно построить интерактивный граф (при наведении на его узлы появляется название анатомической структуры) (рис. 5). Выделены цветами на нем именно обратные связи (так называемые петли).



Рис. 4. Визуализация миндалевидного ядра в окружении головного мозга и головы

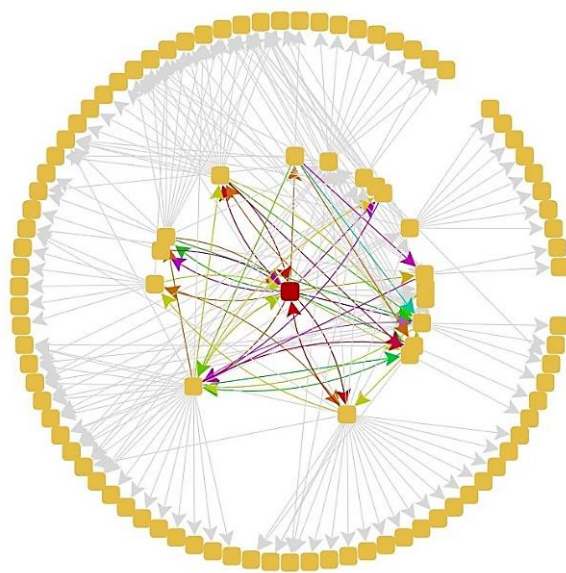


Рис. 5. Граф, отражающий обратные связи миндалевидного тела

С помощью Wolfram Mathematica можно осуществить не только структурную, но и функциональную нейровизуализацию, чего вовсе не позволяют многие другие программы. Так, используя функцию «AnatomicalFunctionalConcept» возможно визуализировать высшую нервную деятельность мозга и выводить статистические карты его активности, иллюстрирующие его когнитивные способности [2, с. 56]. Эти данные были получены с помощью функциональной МРТ (ф-МРТ). Вывод можно оформить как в виде отдельных изображений срезов мозга, так и в виде коллажа, трехмерного графика или анимации. Создадим коллаж изображений мозга, иллюстрирующий его активность при испытании эмоций, комбинируя изображение мозга спереди с изображениями его коронарных, горизонтальных и сагиттальных срезов (рис. 6).

Теперь, глядя на карты когнитивной активности мозга, можно подтвердить, что миндалевидное тело (левая и правая миндалины находятся рядом с центром мозга) активно участвует в формировании эмоций и ответной реакции на них, так как наибольшая активность наблюдается как раз в зоне их расположения. Создадим трехмерное изображение активности мозга, связанной с эмоциями, и объединим его вместе с моделью мозга, полученной ранее (рис. 4) для визуального сравнения (миндалевидное тело выделено красным цветом, правое полушарие мозга показано здесь для демонстрации) (рис. 7).

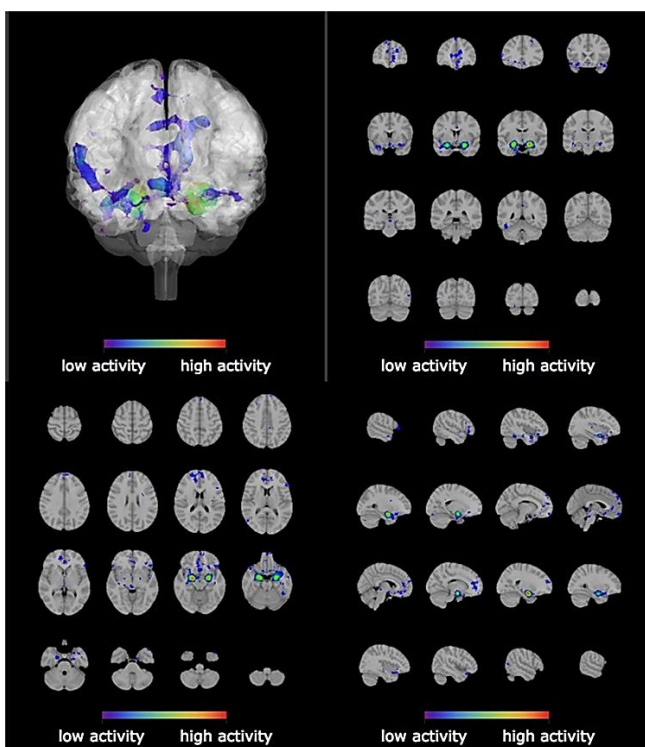


Рис. 6. Коллаж эмоциональной активности мозга

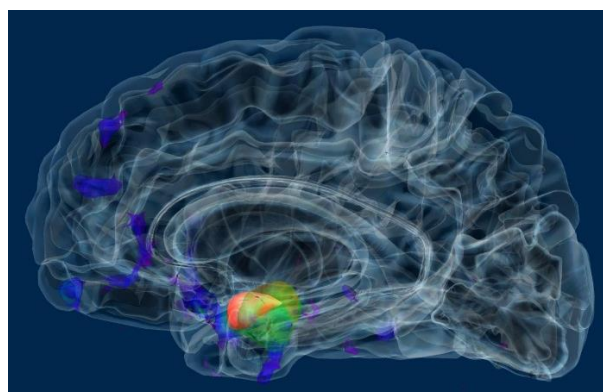


Рис. 7. Нанесение карты когнитивной активности на трехмерную модель мозга

Теперь точно заметно, что зона наибольшей мозговой активности практически совпадает с областью миндалевидного тела.

Таким образом, с помощью системы компьютерной алгебры Wolfram Mathematica можно очень гибко и подробно визуализировать анатомические (структурные) и физиологические (функциональные) особенности головного мозга (даже высшую нервную деятельность), иллюстрировать сложнейшие связи между структурами, оформлять вывод соответственно какому-то медицинскому исследованию или в пользовательском стиле, объединять структурную и функциональную нейровизуализацию для более подробного изучения структур, проявляющих активность в том или ином случае.

### Литература

1. Анатомия человека. Тело. Как это работает / Пер. с англ. А. Анваера. М., 2017. 256 с.
2. Солсо Р. Когнитивная психология. 6-е изд. СПб., 2006. 589 с.
3. Одинак М.М., Емелин А.Ю., Лобзин В.Ю., Воробьев С.В., Киселев В.Н. Современные возможности нейровизуализации в дифференциальной диагностике когнитивных нарушений // Неврология, нейропсихиатрия, психосоматика. 2012. Спецвыпуск 2. С. 51-55.
4. Пирадов М.А., Танащян М.М., Кротенкова М.В., Брюхов В.В., Кремнева Е.И., Коновалов Р.Н. Передовые технологии нейровизуализации // Анналы клинической и экспериментальной неврологии. 2015. Т. 9. №4. С. 11-18.

© Харитонов Л.С., Эварт Т.Е., 2021

УДК 51-77

**Свинцова М.А.**

*Научный руководитель: Мельников Р.А.*, канд. пед. наук  
Елецкий государственный университет  
им. И.А. Бунина, г. Елец, Россия

## **МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ В ЭКОНОМИКЕ**

Математическое моделирование экономических процессов в современном мире является важнейшей частью исследования и прогнозирования. Современная экономическая теория включает в себя математические методы и модели, как обязательный элемент. Использование данных методов при решении экономических задач делает возможным выделить и описать наиболее важные связи, помогает выбрать оптимальные варианты решений.

В современной литературе принято следующее определение: «Моделирование – это, прежде всего, один из наиболее распространенных способов изучения процессов и явлений. Моделирование основывается на принципе аналогии и позволяет изучать объект при определенных условиях и с учетом неизбежной односторонней точки зрения. Объект, труднодоступный для изучения, изучается не непосредственно, а через рассмотрение другого, подобного ему и более доступного – модели. По свойствам модели обычно оказывается возможным судить о свойствах изучаемого объекта» [1, с. 6].

«Любая математическая модель основана на упрощении, она отличается от реальной ситуации и является лишь приближенным описанием. Отсюда и вытекает некая погрешность результатов. Однако именно благодаря замене реального процесса моделью появляется возможность воспользоваться математическими методами при изучении различных процессов» [3, с. 178].

К основным свойствам модели можно отнести адекватность, экономичность, универсальность, точность, простоту, потенциальность и способность к развитию. «Адекватностью математической модели называется способность рассчитывать входные данные с относительной погрешностью, не превышающей заданного значения». Экономичность модели – это характеристика, позволяющая оценить затраты на вычислительные ресурсы. Под точностью можно понимать возможность математической модели обеспечить приемлемое совпадение реальных и найденных значений параметра.

В настоящее время существует огромное количество классификационных признаков модели. С точки зрения характера получаемой модели различают вербальное, графическое, натуральное и математическое виды моделирования. Вербальными называют те модели, которые носят простое словесное описание оригинала. К графическим относятся модели, представленные в виде изображения, например, географическая карта является моделью земной поверхности. Особенностью натурального моделирования является замена оригинала своим физическим подобием, то есть макетом. Математическое моделирование основывается на использовании какого-либо математического аппарата.

Мицель А.А. в своей работе приводит следующую классификацию моделей: «По характеру зависимости от времени: статические и динамические; по общему целевому назначению: прикладные и теоретико-аналитические; по конкретному предназначению: имитационные, оптимизационные, трендовые, балансовые и т. д.» [2].

«Построение математической модели включает в себя шесть этапов: постановка экономической проблемы и ее качественный анализ, построение математической модели; математический анализ модели; подготовка исходной информации; численное решение; анализ численных результатов и их применение» [2].

*Постановка экономической проблемы и ее качественный анализ.* «На первом этапе требуется сформулировать проблему, выявить основные черты и свойства моделируемого объекта, сформулировать гипотезы, объясняющие поведение и развитие объекта» [2].

*Построение математической модели.* «На данном этапе происходит выражение выявленной проблемы в виде конкретных математических зависимостей (неравенство, уравнение и др.). Для некоторых сложных объектов целесообразно строить несколько моделей, причем каждая из них будет выделять лишь некоторые стороны объекта, а другие стороны учитываются приближенно» [2].

*Математический анализ модели.* «На этом этапе выявляются общие свойства модели и ее решение с помощью математических приемов исследования. При аналитическом исследовании выясняется, единственно ли решение, какие переменные могут входить в решение и т. д.» [2].

*Подготовка исходной информации.* «В процессе подготовки информации используются методы теории вероятностей, математической и теоретической статистики, оценки достоверности данных. При системном экономико-математическом моделировании результаты функционирования одних моделей служат исходной информацией для других» [2].

*Численное решение.* «Этот этап включает разработку алгоритмов численного решения задачи и проведение вычислений. Обычно расчеты при решении экономических задач носят многовариантный характер. Численное решение существенно дополняет результаты аналитического исследования, что для многих моделей является единственно возможным» [2].

*Анализ численных результатов и их применение.* «На этом этапе прежде всего решается важнейший вопрос о правильности и полноте результатов моделирования и применимости их как в практической деятельности, так и в целях усовершенствования модели. Поэтому в первую очередь должна быть проведена проверка адекватности модели по тем свойствам, которые выбраны в качестве существенных. Применение численных результатов моделирования в экономике направлено на решение практических задач» [2, с. 7-8].

Этапы построения экономико-математической модели неразрывно связаны между собой. Благодаря этой связи, на этапе построения можно определить носит ли данная модель противоречивый характер или приводит к построению более сложной модели. В таком случае первоначальная постановка задачи должна быть подкорректирована.

Основной задачей различных математических методов в экономике является построение и изучение тренда на основе имеющихся данных. Тренд в экономике определяется, как



направление преимущественного движения показателей, рассматриваемого в рамках технического анализа, где подразумевают направленность движения цен или значений индексов. Среди методов оценки тренда можно выделить непараметрические – это различные методы сглаживания исходного временного ряда, применяемые как для оценки тренда, так и для прогнозирования.

Метод взвешенного скользящего среднего (WMA) является естественным расширением метода простого скользящего среднего. Суть метода определяется тем, как использование при расчетах большего веса для новых наблюдений, а меньшего – для более старых. В работе с техническим анализом данный подход является логическим и удобным с точки зрения превалирующего тренда на рынке. Формула для расчета имеет следующий вид:

$$WMA = \frac{\sum_{i=1}^n P_i \times W_i}{\sum_{i=1}^n W_i},$$

где  $P$  – цена актива в определенном периоде,  $W$  – удельный вес.

Таблица 1.

Цена актива

Дата	1 января	2 января	3 января	4 января	5 января	6 января	7 января
Цена	105	106	108	109	111	114	116

Необходимо определить значение взвешенного скользящего среднего за последние 7 периодов.

$$WMA = \frac{7 \times 116 + 6 \times 114 + 5 \times 111 + 4 \times 109 + 3 \times 108 + 2 \times 106 + 1 \times 105}{7 + 6 + 5 + 4 + 3 + 2 + 1} = 111.7$$

Для сравнения можно посчитать простую среднюю (SMA):

$$SMA = \frac{116 + 114 + 111 + 109 + 108 + 106 + 105}{7} = 109.9$$

Значение WMA в данном случае больше, и это является отражением ярко выраженного тренда к возрастанию значений.

Преимуществом данного метода является более гибкая реакция индикатора на последние тенденции изменений цены. К недостаткам можно отнести запаздывание при входе в тренд и выходе из него; сильное изменение при появлении ложного сигнала; неэффективность при позиционной торговле.

Метод экспоненциального скользящего среднего (EMA) можно определить, как частный случай взвешенного скользящего среднего. Отличительной чертой данного метода является гибкое реагирование на изменение цен по сравнению с другими типами скользящего среднего. Использование метода весьма актуально и практично, так как его главной задачей является поиск усредненного значения, позволяющего минимизировать рыночный шум в отображении графика. Стоит отметить, что метод EMA придает большое значение параметрам, которые



имеют место в последние несколько дней. Формула данного метода может иметь следующий вид:

$$EMA = \alpha \times P_t + (1 - \alpha) \times EMA_{t-1},$$

где  $\alpha = \frac{2}{N+1}$  – весовой коэффициент, отражающий скорость старения прошлых данных ( $N$  – интервал сглаживания),  $P_t$  – значение случайной величины в период времени  $t$ ,  $EMA_{t-1}$  – значение экспоненциального скользящего среднего в период времени  $t - 1$ .

Таблица 2.

Данные о цене акций компании

Дата	1 октября	2 октября	3 октября	4 октября	5 октября	6 октября
Случайная величина $P$	4.5	7.2	6.7	4.3	5.4	6.1

Необходимо определить значение экспоненциального скользящего среднего за 6 октября.

Пусть интервал сглаживания равен 5, тогда  $\alpha = \frac{2}{5+1} = 0.3$ .

В качестве первоначального значения экспоненциального скользящего среднего можно использовать простое скользящее среднее:

$$SMA = \frac{5.4 + 4.3 + 6.7 + 7.2 + 4.5}{5} = 5.6$$

Таким образом, значение экспоненциального скользящего среднего за 6 октября равно:

$$EMA = 0.3 \times 6.1 + (1 - 0.3) \times 5.6 = 5.8$$

Можно сделать вывод о том, что EMA отслеживает динамику рынка эффективнее, чем SMA и WMA, так как отдает приоритет свежим данным и не регистрирует резких колебаний в ответ на изменение старых данных.

EMA обладает плюсами и минусами, как и любой другой метод. К плюсам можно отнести простоту и понятность метода, повышенную точность, более оперативное реагирование на изменения тенденции. Минусами являются небольшие запаздывания, наличие ложных сигналов, низкий уровень эффективности на нестабильном рынке.

Основной идеей метода проецирования тренда является построение прямой, которая в среднем наименее уклоняется от массива точек, заданного временным рядом. Уравнение прямой имеет вид  $x = at + b$ , где  $a$  и  $b$  являются постоянными. Формула для нахождения значений коэффициентов  $a$  и  $b$  можно записать следующим образом:

$$\varphi(a, b) = \sum_{i=1}^m (at_i + b - x_i)^2 \rightarrow \min$$

Стоит отметить, что функция  $\varphi$  принимает минимальное значение лишь в том случае, если величины  $a$  и  $b$  удовлетворяют системе линейных уравнений:

$$\begin{cases} a \sum_{i=1}^m t_i + bn = \sum_{i=1}^n x_i \\ a \sum_{i=1}^m t_i^2 + b \sum_{i=1}^n t_i = \sum_{i=1}^n t_i x_i \end{cases}$$

Таблица 3

Расчетная таблица

$t_i$	$x_i$	$t_i \times x_i$	$t_i^2$
1	8	8	1
2	5	10	4
3	6	18	9
4	9	36	16
5	10	50	25
$\Sigma = 15$	$\Sigma = 38$	$\Sigma = 122$	$\Sigma = 55$

Необходимо определить прогнозируемое значение тренда для 6 дня.

Решение системы уравнений:

$$\begin{cases} 15a + 5b = 38 \\ 55a + 15b = 122 \end{cases}$$

позволяет определить, что  $a = 0.8$ ,  $b = 5.2$ .

Таким образом, уравнение, позволяющее определить значение тренда имеет вид:

$$x = 0.8t + 5.2$$

Так, значение составит:

$$f_6 = 0.8 \times 6 + 5.2 = 10$$

Следует отметить, что при работе с данным методом вероятность возникновения погрешности значительно меньше, чем при использовании методов скользящей средней или экспоненциального сглаживания, что делает ЕМА наиболее эффективным при построении прогнозов.

### Литература

1. Дроздова Н.В., Переломова И.Г. Экономико-математическое моделирование. Ярославль, 2010. 246 с.
2. Мицель А.А. Математическое и имитационное моделирование. Юрга, 2016. 108 с.
3. Сокольская Е.Е., Дворецкая В.И. Математическое моделирование в экономике // Современные наукоемкие технологии. 2014. №5-2. С. 177-178.

© Свинцова М.А., Мельников Р.А., 2021

УДК 004.054

**Чернявская А.Э., Ехлаков Ю.П.**, д-р техн. наук  
Томский государственный университет систем  
управления и радиоэлектроники, г. Томск, Россия

### **КРИТЕРИИ ПРИОРИТЕЗАЦИИ ТРЕБОВАНИЙ ПРИ РАЗРАБОТКЕ ПРОГРАММНОГО ПРОДУКТА**

На этапе планирования разработки программного продукта существует объективная потребность в расстановке приоритетов требований. Это связано с тем, что в первую очередь должны реализовываться наиболее значимые и ценные для бизнеса требования, что подтверждается как в стандартах на «Программную инженерию» [1], так и в ряде монографий [2; 3]. При организации процесса приоритизации сложнее всего определиться с тем, что именно считать «наиболее важным» требованием. На это могут влиять особенности контекста проекта и так же множество критериев, используемых для определения «важности». В процессе приоритизации требований могут участвовать: аналитик, менеджер и/или руководитель проекта, представитель заказчика. Оценка приоритетов происходит субъективно, и в зависимости от опыта и знаний оценщика результат может быть как удачным, так и неудачным. Сложность процедуры приоритизации требований заключается в том, что требования и их приоритеты постоянно меняются. Эти изменения могут быть обусловлены изменениями в бизнес- процессах и, как следствие, новыми пожеланиями заказчика, а также ограничены возможностями самих разработчиков.

В настоящее время в литературе описаны ряд подходов к приоритизации требований, однако все они основаны на использовании одного или нескольких критериев оценивания. Распространенным критерием при однокритериальном подходе является «ценность для бизнеса», при двух критериях – «важность и срочность», при трех – «ценность, стоимость, риск» [3], при многокритериальном подходе, помимо вышеперечисленного, – трудозатраты на реализацию, взаимосвязь между требованиями и т. д.

В таблице 1 приводится множество возможных критериев оценки относительной важности требований на основе обобщения существующих подходов по определению приоритетов требований, а также опыта ИТ-специалистов.

Таблица 1

Критерии оценки относительной важности требований

Наименование критерия	Содержательная интерпретация	Измерительная шкала
Бизнес-ценность	Ценность для бизнеса (влияние на результат) с точки зрения команды проекта и заказчика.	Порядковая или интервальная
Уровень риска	Вероятность невыполнения функциональности или качества, превышения затрат или сроков.	Интервальная
Стоимость	Суммарные финансовые затраты на реализацию требования.	Абсолютная с переводом в шкалу отношений

Наименование критерия	Содержательная интерпретация	Измерительная шкала
Волатильность требований	Вероятность изменения требования в течении очередной итерации (спринта) из-за незрелости заказчика, слабой проработки и(или) несогласованности между представителями разработчика и заказчика.	Интервальная
Трудозатраты	Усилия команды на реализацию требования в человеко-днях /часах (минимальные-максимальные).	Абсолютная с переводом в шкалу отношений
Сложность	Достаточность уровня знаний и умений для реализации требования.	Порядковая или интервальная
Срочность	Объективная необходимость - потребность реализации требования в данное время.	Порядковая или интервальная
Инновационность	Уникальность, требующая регистрации авторского и/или имущественного права.	Порядковая или интервальная
Критичность последствий от срыва сроков	Потери команды от несвоевременной реализации требования.	Порядковая или интервальная
Связность	Количество прямых зависимостей для реализации других требований.	Абсолютная с переводом в шкалу отношений
Охват	Количество пользователей, которые могут использовать результаты (количество или проценты).	Абсолютная с переводом в шкалу отношений

Очевидно, что приведенный перечень критериев не претендует на полноту и может быть дополнен. Вместе с тем, эта информация будут полезна менеджерам ИТ-проектов при приоритезации требований на очередном этапе разработки программного продукта.

К наиболее известным методам приоритезации требований можно отнести: метод экспертных оценок, метод анализа иерархий, трехуровневая шкала приоритетов, метод MoSCoW, метод 100 долларов, метод Quality Function Deployment (Структурирование функций качества), модель Кано, RICE (метод приоритезации гипотез), метод Theme scoring (оценка тем) [3; 5].

В данной работе задача определения приоритетов требования в текущем интервале планирования представлена в виде многокритериальной модели группового выбора:

$$DM = (T, K, Z, P/ S, R)$$

Пусть задано:

T – множество требований к программному продукту, которые могут быть реализованы в текущем периоде планирования (спринта);

U – множество участников процесса приоритезации требований;

K— множество критериев оценки относительной важности требований;

Z – множество измерительных шкал оценки критериев;

P – правило определения интегральной оценки приоритета требования.

Требуется определить:

S – множество оценок оценки относительной важности каждого из требований по каждому из критериев;

R – приоритет для каждого из требований.

Объектом приоритизации могут быть требования пользователей (пользовательские истории), функциональные требования, системные требования. Между требованиями могут быть технологическая, либо временная зависимости. Технологическая зависимость – когда после реализации исходного требования необходимо обязательно приступить к реализации требования зависящего от исходного. Временная зависимость - когда исполнение зависящего требования можно отложить по времени.

Для оценки относительной важности каждого из требований используется множество критериев, оценка проводится в измерительных шкалах: абсолютной, порядковой, отношений, интервалов [4]. В абсолютной шкале результатом измерения важности требований являются числа, отражающие абсолютное свойство требования по какому-либо критерию, например «трудозатраты» равны 45 человеко-часам. Как правило, эти оценки приводятся в физических величинах и при определении оценок необходимо учитывать максимальное либо минимальное значение, оценки является приоритетным.

Порядковая шкала (ранговая шкала) позволяет упорядочивать требования в зависимости от их важности для достижения результатов на очередной итерации разработки (например: высокая важность – 1, средняя важность – 2, низкая важность – 3). Если между рядом требований существуют отношения безразличия, то этим требованиям назначаются одинаковые ранги, равные среднему арифметическому рангов ( $2+3=2,5$ ). Шкала отношений позволяет сравнивать степень важностей требований друг с другом и отображать степень предпочтения одного сравниваемого объекта перед другим некоторым числом.

Перевод оценок в абсолютных шкалах в шкалу отношений производится по следующему правилу:

– если приоритетным является выбор требования с максимальным значением оценки, то

$$T_i = \frac{T_i}{T_{\max}}, \text{ где } i=1, n;$$

– если приоритетным является выбор требования с минимальным значением оценки, то

$$T_i = \frac{T_{\min}}{T_i}, \text{ где } i=1, n.$$

Пример сравнения двух требований по критерию «трудозатраты»:  $T_1=45$  чел.-час.,  $T_2=30$  чел.-час., если приоритетным является выбор требования с максимальным значением трудозатрат, тогда отношение  $\frac{T_2}{T_1} = 0.66$ , ранг  $T_1=1$ , ранг  $T_2=0.66$ .

Шкала интервалов используется для отображения различия между требованиями, каждому из требований присваивается числовые значения (баллы) по каждому из критериев в принятой системе оценок. Качественная градация шкалы может быть представлена различными вариантами: «высокая–низкая»; «высокая-средняя-низкая»; «очень высокая-высокая-средняя-низкая-очень низкая»; «несущественная-низкая-умеренная-средняя-высокая-очень высокая». Численные значения оценок выбираются в интервалах [0-1], [0-10]. Пример такой шкалы представлен в таблице 2.



Пример шкалы интервалов

Значение	Оценки				
	Очень высокая	Высокая	Средняя	Низкая	Очень низкая
Лингвистическое					
Количественное	1-0,8	0,8-0,64	0,64-0,37	0,37-0,2	0,2-0,0

В соответствии с рекомендациями метода MoSCoW [3], процедура определения интегральной оценки приоритета требования, позволяющая сгруппировать требования по важности реализации в четыре группы, может быть представлена в виде следующего правила:

– ЕСЛИ  $0,8 < P < 1$ , то требования должны быть реализованы при любых условиях.

– ЕСЛИ  $0,64 < P < 0,8$ , то требования должны быть выполнены сразу, после «очень высоких».

– ЕСЛИ  $0,37 < P < 0,64$ , то требования можно выполнить, если будет время и ресурсы.

- ЕСЛИ  $0 < P < 0,37$ , то требования можно отложить до следующего релиза.

В качестве алгоритма решения поставленной задачи использован метод экспертных оценок [4], так как он прост в реализации и понятен практическим специалистам по управлению программными проектами. Количество критериев оценки приоритетов требований, не должно быть не более пяти, поскольку взаимный анализ большего числа критериев ограничивается психологическими возможностями участников процесса приоритезации требований.

### Литература

1. ГОСТ Р ИСО/МЭК 12207-2010 «Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств». <https://clck.ru/I3vkR>

2. Халл Э., Кен Дж., Джереми Д.. Инженерия требований. М.: ДМК Пресс, 2017. 223 с.

3. Вигерс К., Джой Б. Разработка требований к программному обеспечению. <https://clck.ru/U4Q8U>

4. Евланов Л.Г. Теория и практика принятия решений. М., 1984. 176 с.

5. Саати Т. Принятие решений. Метод анализа иерархий. М., 1993. <https://clck.ru/U4Q7j>

© Чернявская А.Э., Ехлаков Ю.П., 2021

УДК 004.58

**Банников К.Д.**

*Научный руководитель: Латыпова Н.В., канд. физ.-мат. наук  
Удмуртский государственный университет, г. Ижевск, Россия*

## **ВНЕДРЕНИЕ ПРОГРАММЫ АВТОМАТИЗАЦИИ ПРОЦЕССА СКАНИРОВАНИЯ ДОКУМЕНТАЦИИ**

В настоящее время, благодаря информационным технологиям, появилась возможность облегчить ручную работу, сократить ее или вовсе автоматизировать. Множество организаций и предприятий вкладывают огромные средства, чтобы облегчить и ускорить свою работу для получения большей выручки с уменьшением затрат. На сегодняшний день процесс сканирования документов очень часто полностью зависит от человека, из-за этого в будущем при проверке могут обнаружиться ошибки. Чем больше ошибок, тем больше времени и усилий придется потратить на их исправление.

Цель работы – создание программного продукта в помощь приемной комиссии Удмуртского государственного университета для организации и оптимизации процесса сканирования документов абитуриентов. Подобная оптимизация предполагает сокращение количества задействованных в сканировании людей, которые могли бы выполнять другую работу. Разработанная программа должна будет облегчить работу человека и уменьшить его влияние на процесс, сокращая количество ошибок при формировании пакетов документов. Для достижения поставленной цели предполагается решение следующих задач:

- 1) Создание программного продукта, оптимизирующего процесс сканирования документов;
- 2) Тестирование и анализ ошибок созданной программы автоматизации процесса;
- 3) Написание инструкции пользователю программного продукта.

Данный программный продукт призван:

- 1) Ускорить и упростить процесс сканирования документов.
- 2) Уменьшить влияние человека на процесс сканирования документов.
- 3) Сократить количество ошибок при формировании пакетов документов.
- 4) Сократить число занятых сканированием работников.

Рассмотрим процесс подачи документов абитуриентом в приемную комиссию.

1. Абитуриент приходит в приемную комиссию и заполняет заявление на поступление в университет. В нем он указывает личную информацию, контакты, результаты экзаменов, а также направление, куда он хочет поступить. После заполнения заявления абитуриент отдает его вместе с требуемым пакетом документов специалисту приемной комиссии. Данные абитуриента заносят в базу интегрированной информационно-аналитической системы (ИИАС) УдГУ, присваивают ему номер, а документы собирают и отдают специалистам на дальнейшую обработку и формирование пакета в базе ИИАС УдГУ (<https://clck.ru/UG5Em>).

2. Процесс обработки документов проходит следующим образом. Специалист приемной комиссии открывает конверт со всеми необходимыми документами абитуриента и находит его

по номеру в базе данных университета. Далее специалист вносит в базу университета паспортные данные, место рождения, адрес по прописке, контактную информацию, документ об образовании, ИНН, результаты экзаменов, необходимость в предоставлении общежития, знание иностранных языков. После обработки документов идет сканирование заявления. Заявление абитуриента сканируется и помещается в папку с идентификатором (индивидуальный номер – ИД) абитуриента вместо названия. Образы заявлений всех абитуриентов называются одинаково для удобства.

3. Папки с заявлениями собираются в одном месте. В процессе работы приемной комиссии абитуриент может заполнить новое заявление и заменить им старое или дополнить старое заявление новым. После этого приходится заново сканировать документы и помещать их в папку.

4. По истечению срока приемной комиссии прошедших абитуриентов зачисляются в списки студентов различных групп.

На этом приемная комиссия заканчивает свою работу с документами абитуриента. Планировалось, что программа сможет помочь автоматизировать выгрузку документов в Федеральную информационную систему (ФИС).

Рассмотрим недостатки в работе приемной комиссии. Работа членов приемной комиссии практически полностью ручная, из-за чего на нее тратится больше сил и времени. В процессе заполнения и сортировки документов работник может допустить неосознанную ошибку, так как на сортировку документов нет ограничений, неправильное заполнение может сильно усложнить последующую работу комиссии.

Приемная кампания обычно длится с 20 июня по 27 декабря ежегодно, но основная нагрузка на приемную комиссию приходится в период с 20 июня по 31 августа.

В приемную кампанию 2019 г для сканирования и обработки документов выделялось 14 человек: 4 на сканирование и 10 на обработку документов. За время работы приемной комиссии в 2019 г подали документы 9284 человека. Время обработки документов на одного человека от 3 до 10 минут в зависимости от того, куда поступает человек: бакалавриат или магистратура, а также количества поданных документов и количества страниц заявления, которые нужно отсканировать: разница составляет от 4 до 12 листов. За 1 рабочий день летом 2019 г. обрабатывалось в среднем около 275 документов абитуриентов (<https://clck.ru/UG5Em>).

В 2019 г специалисты, работавшие в приемной комиссии, должны были выгрузить все заявления абитуриентов в ФИС, проверить правильность обработки и сканирования. Но из-за допущенных на начальном этапе ошибок, задача потребовала гораздо больше сил и времени. Учитывая статистику летнего периода приемной комиссии 2019 г, были обнаружены следующие типичные ошибки:

- 1) 50–60 человек с неправильно заполненными документами;
- 2) 662 созданные, но пустые папки.

После внедрения разработанной автором программы автоматизации процесса сканирования количество ошибок планируется свести к минимуму. Программа будет строго ограничивать действия человека инструкциями, так что он не сможет создавать папку не в том

месте, называть ее неправильно, а также называть сканируемый документ не по заложенному стандарту. Программа будет сопровождать работника на протяжении всего процесса сканирования, и давать четкие инструкции о порядке сканирования документов.

Остановимся на выборе программного обеспечения. Для разработки программы автоматизации процесса сканирования документов для приемной комиссии был выбран Oracle Forms главным образом потому, что портал ИИАС создан при помощи ПО Oracle Forms. Благодаря этому, а также графическому модулю Oracle forms, программу автоматизации удалось объединить с порталом и подключить к базе данных университета.

Достоинства:

- Внутренний язык PL/SQL;
- Уже на этапе компиляции проверяется правильность конструкций SQL и соответствие типов;
- Наличие средств буферизации.

Недостатки:

- Нет операции создания в real-time новых элементов на форме;
- Относительно сложно делать интерфейс;
- Наличие глюков.

NAPS2 – программа для сканирования документов. Она может сканировать с выбранными вами настройками или настройкой нескольких профилей для разных устройств и конфигураций. После того, как вы закончите сканирование, вы можете сохранять, отправлять по электронной почте или печатать документы. После сканирования вы можете сохранять файлы в формате PDF, TIFF, JPEG, PNG и других. NAPS2, помимо основного интерфейса, также предлагает интерфейс командной строки через программу NAPS2.Console.exe. Это позволяет автоматизировать сценарии сканирования и сохранения документов (<https://naps2.ru/>).

Код созданной программы находится на сервере, на котором хранится вся база портала ИИАС (далее «портал»). Благодаря этому, программой автоматизации может воспользоваться каждый, у кого есть соответствующий доступ к portalу. Программа автоматизации состоит из двух частей: первая (программа передачи ИД) хранится на сервере портала, а вторая (программа сканирования) на компьютере работника приемной комиссии. Первая часть – программа передачи ИД, находящаяся на сервере, связывается с компьютером работника и предоставляет доступ к portalу, а также, через консоль компьютера включает вторую часть программы и передает ИД человека. Программа сканирования: также подключается к консоли компьютера и оттуда связывается со сканирующим аппаратом через установленную на компьютер программу NAPS2, и передает ей в диалоговом режиме информацию о том, куда положить сканы документов, как их назвать, как назвать папку, в которую кладутся сканы документов.

Предварительные действия:

- Убедиться, что на вашем компьютере присутствуют программа NAPS2 и 4 файла с расширением .bat: passport.bat; portfolio.bat; zayavlenie.bat; attestat.bat.

– Последовательно выбрать в меню АБИТУРИЕНТ/СКАНИРОВАНИЕ ДОКУМЕНТОВ АБИТУРИЕНТА.

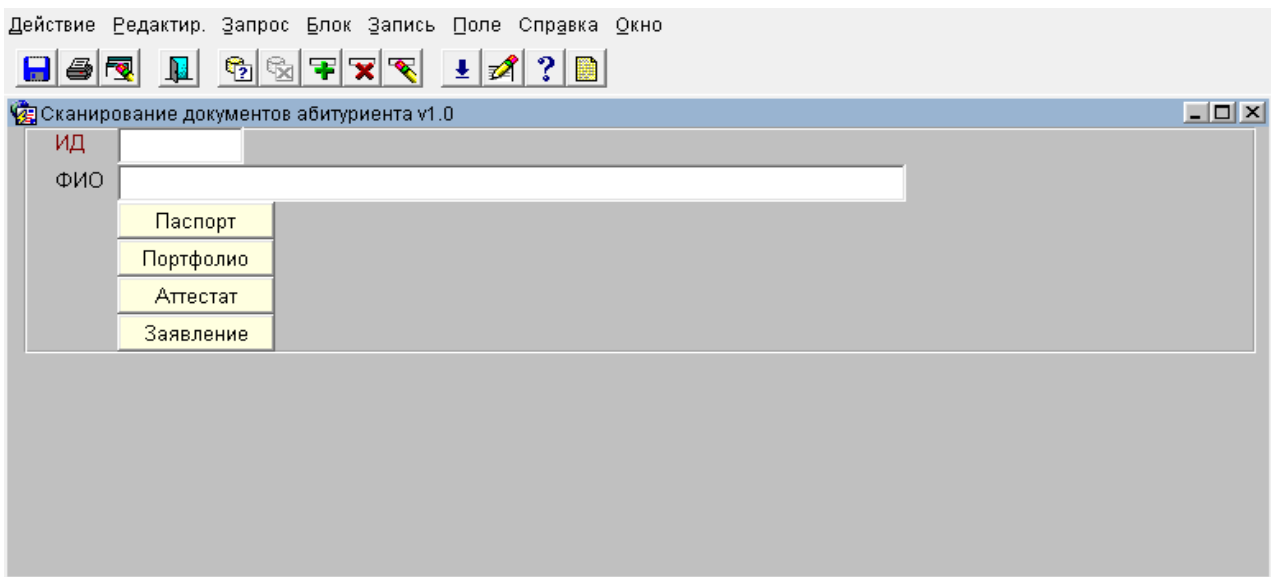


Рис. 1. Меню программного продукта

Меню предлагаемого программного продукта имеет вид, представленный на рисунке 1, и состоит из следующих полей:

Поле «ИД» – Индивидуальный номер человека в базе.

Поле «ФИО» – Фамилия, имя и отчество человека.

Кнопка «Паспорт» – нажимать при сканировании паспорта.

Остальные кнопки («Портфолио», «Аттестат», «Заявление») используются аналогично предыдущей.

Алгоритм сотрудника состоит из следующих шагов, отраженных в соответствующей инструкции пользователя.

1. Шаг:

Для начала введите Индивидуальный номер (ИД) человека в поле ИД и нажмите Enter, в поле ФИО должно отразиться ФИО человека, чье ИД вы ввели, как на рисунке 2.

2. Шаг: После этого положите нужные документы в сканирующий аппарат (например, заявление абитуриента), и нажмите на кнопку в форме, которая соответствует типу сканируемого документа (к примеру «Заявление»), после чего вы увидите сообщение «Запуск сканирования» – как на рисунке 3.

Если все было сделано правильно, то вы увидите сообщение «Команда выполнена» (рис. 4). Отсканированные файлы будут помещены в папку с ИД человека, которая в свою очередь будет размещена в указанные в настройках папки на портале.

3. Шаг: Если вы НЕ увидели сообщение «Команда выполнена», то программа не смогла выполнить свою задачу. Возможно, вы пытались поместить сканы документов в папку с ограниченным доступом, или была допущена ошибка при настройке процесса сканирования, или произошли неполадки со сканирующим аппаратом.



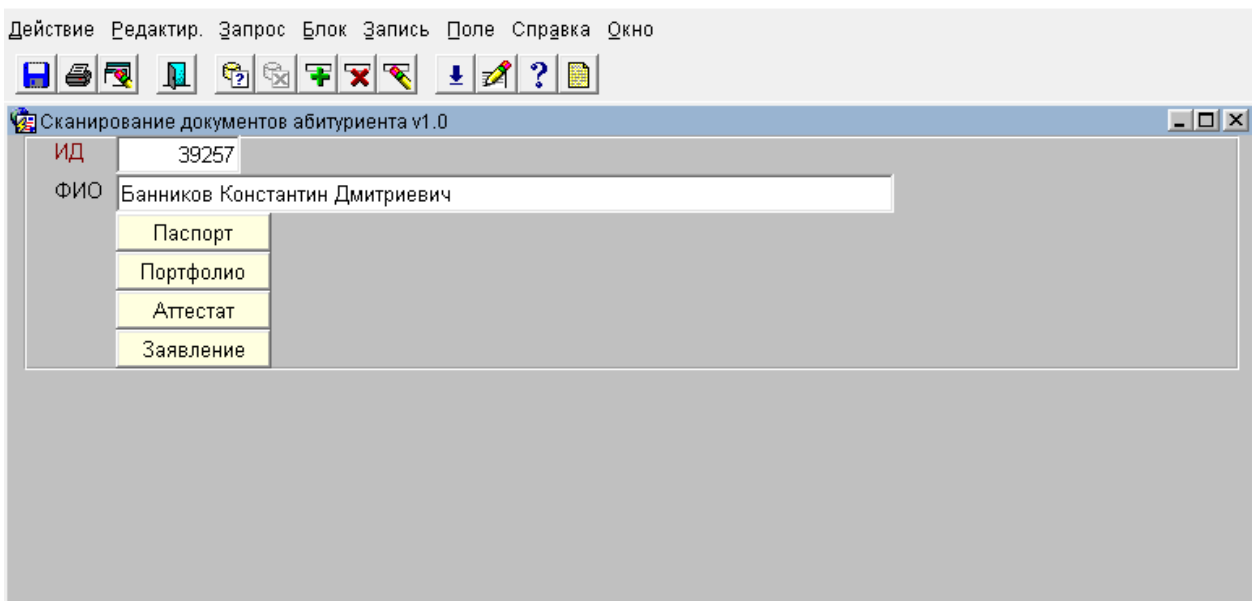


Рис. 2. Отражение ФИО абитуриента после введения ИД

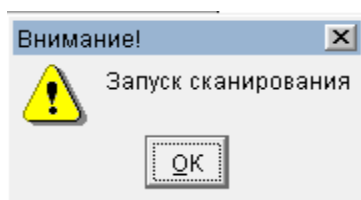


Рис. 3. Запуск сканирования

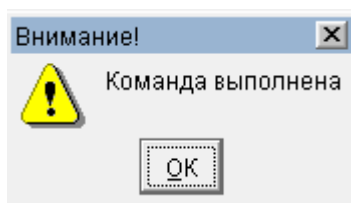


Рис. 4. Сообщение «Команда выполнена»

В связи с ситуацией, вызванной COVID-19, летом 2020 г программу автоматизации сканирования документов не удалось проверить в условиях реальной приемной кампании. Но по результатам проведенных тестов программа справилась со всеми поставленными задачами. Программа смогла ускорить процесс сканирования почти в 2 раза и уменьшила влияние человека на процесс составления пакетов документов.

Также благодаря bat файлам можно автоматизировать процесс сканирования программы NAPS2, что делает программу автоматизации полезной не только для приемной комиссии, но и для любого процесса сканирования при помощи программы NAPS2.

Программа автоматизации не ограничивается применением только в рамках Удмуртского госуниверситета, и может работать в любых организациях, где база данных построена с помощью Oracle Forms и присутствует программа NAPS2. Если у организации нет единой базы данных, или база построена не на основе Oracle Forms, то организации может



пригодиться вторая часть созданного программного продукта. Вторая часть может работать на любом компьютере, на котором есть программа NAPS2, и выполнять сканирование любых документов для различных нужд по bat файлу в два клика мыши, в нем же можно и настраивать все нюансы ее использования.

© Банников К.Д., Латыпова Н.В. 2021

УДК 004

**Попова Д.Е., Кузьмичева Е.А.**

Воронежский государственный педагогический  
университет, г. Воронеж, Россия

## ПРИМЕНЕНИЕ ИКТ В ОБРАЗОВАТЕЛЬНОМ ПРОЦЕССЕ

*Золотое правило дидактики – наглядность.*  
Ян Каменский

Воспитание и образование нового поколения осуществляется в век информационных технологий. В связи с этим самой важной задачей для педагога является идти в ногу со временем. Современный преподаватель должен разрабатывать и совершенствовать содержание образования.

Обучение школьника и его всесторонняя подготовка возможна лишь с помощью инновационных образовательных технологий, основанных на новейших достижениях в сфере информатизации. Компетентность в области информационно-коммуникационных технологий (ИКТ) является одной из первостепенных целей образования, но также является актуальной проблемой современного образования. [1]

На сегодняшний день ИКТ широко используется в сфере образования, дает оптимизацию учебного процесса. Использование информационно-коммуникационных технологий является эффективным средством обучения, его можно использовать на любом из этапов изучения отдельного предмета, оно развивает умственные способности учащихся, стимулирует их заниматься самостоятельно.

Наиболее часто используются следующие элементы ИКТ:

Таблица 1

Ресурсы ИКТ

Информация в электронном формате	Электронные учебники и пособия, демонстрируемые с помощью компьютера и мультимедийного проектора, электронные энциклопедии и справочники.
Информационные носители	DVD и CD диски с картинками и иллюстрациями, флэш-накопители.
Мультимедиа	Игровые компьютерные программы, презентации и др.
Аудиовизуальное оборудование	Интерактивные доски, видео и аудиотехника, персональный компьютер, ноутбук.
Интерактивные	Интерактивные карты и атласы, конференции и конкурсы, научно-исследовательские работы и проекты.

Существуют некоторые факторы, которые будут влиять на успешность урока с применением ИКТ: вид группы школьников; тип урока и его методическая цель; высокая степень индивидуальной работы; уровень подготовки учащихся [2].

Благодаря ИКТ, учащиеся расширяют свой кругозор, учатся ориентироваться в информационном пространстве, развивают познавательные интересы и самое главное – учатся работать с различными видами ресурсов ИКТ.

На сегодняшний день ИКТ, кроме учебных целей по предметам, решает еще и задачи по формированию информационной грамотности учащихся: получение знаний, умений и навыков, позволяющих перерабатывать и осмысливать большие потоки информации и применять ее; овладение ИКТ как инструментом профессиональной деятельности.

В зависимости от типа урока могут применяться различные информационно-коммуникационные технологии. Например, для комбинированного урока или для изучения нового материала уместно использовать проектор, CD – диск, обучающие программы, всемирную сеть Интернет. Для закрепления знаний и совершенствования умений и навыков – обучающе-контролирующие программы, а также моделирующие системы. Чтобы проконтролировать знания учащихся логично использовать программы с тестовыми системами.

Урок с применением ИКТ имеет ряд преимуществ: наглядность учебного материала; экономия времени; доступность информации; оперативность получаемой информации; возможность развития пространственного мышления.

Так же часто используются ЦОР – цифровые образовательные ресурсы. Целью их применения является укрепление умственных способностей учащихся и повышение качества обучения на каждом этапе образовательной системы. К цифровым образовательным ресурсам можно отнести: фотографии; видеофрагменты; звукозаписи; объекты виртуальной реальности; электронные учебники, атласы и т. д.; текстовые документы.

ЦОР имеет ряд задач в образовательном процессе:

Таблица 2

Задачи ЦОР

Помощь учителю при подготовке к уроку	Компоновка и моделирование урока из отдельных цифровых объектов, большое количество дополнительной и справочной информации, эффективный поиск информации, подготовка контрольных работ и творческих заданий.
Помощь учителю при проведении урока	Демонстрация подготовленных заданий, использование виртуальных лабораторий и интерактивных моделей, тестирование учащихся для оценки качества знаний.
Помощь учащемуся при подготовке домашнего задания	Автоматизированный самоконтроль учащихся, база для подготовки выступлений, докладов, рефератов и т.д., оперативное получение дополнительной информации, формирование потребности в овладении информационными технологиями и постоянной работе с ними.

Несмотря на достоинства цифровых образовательных ресурсов, у них есть свои недостатки и проблемы в применении на практике.

Использование ИКТ может иметь ряд негативных последствий. Использование информационных ресурсов, опубликованных в сети, часто приводит к отрицательным последствиям. К таким случаям можно отнести использование готовых проектных работ, индивидуальных творческих и технических заданий, что ведет к неумению правильно искать, обрабатывать и представлять информацию. Все это не способствует повышению общего и индивидуального развития учащихся.

Проблемы и пути решения применения ЦОР

Проблемы применения ЦОР	Возможные пути решения проблем
Неумение включать в учебный материал цифровые образовательные ресурсы	Заимствование план-конспектов коллег с передовым опытом, прохождение курсов повышения квалификации
Отсутствие демонстрационного центра	Данную проблему возможно решить только на уровне комплектации администрацией учебного заведения классов
Недостаточная компьютерная грамотность учителя	Посещение курсов повышения квалификации, а также подготовка кадров в рамках обучения в бакалавриате и магистратуре
Отсутствие в образовательных учреждениях специалиста по содержанию компьютерной техники и оказанию консультативной помощи при ее эксплуатации	Данная проблема решается путем открытия соответствующей ставки
Отсутствие грамотно составленных ЦОР	Использование ЦОР, предоставляемых авторами УМК

Длительная работа за компьютером и нахождение за жидкокристаллическим монитором может отрицательно сказываться на организме: эндокринной системе, иммунной системе, высшей нервной системе, на костно-мышечном аппарате и зрении человека. Так же учащиеся могут потерять способность фантазировать, появляются проблемы с визуализацией образов, трудности с обобщением и анализом информации, может снизиться функционирование некоторых видов памяти. Но, существует и обратная опасность. Компьютер может так и остаться средством для развлечения и игры, что зачастую перерастает в болезненную зависимость.

В связи с этим, к использованию информационно-коммуникационных технологий в обучении педагогам необходимо строго соблюдать санитарно-гигиенические требования и не превышать норму проведения учащихся за мониторами: 20 минут – 1–2 класс; 25 минут – 3–4 класс; 30 минут – 5–6 класс; 35 минут – 7–11 класс.

Одним из последствий последней информационной революции, повлекшей за собой повсеместное внедрение информационных технологий, стало дистанционное обучение. По ряду объективных причин, на данном этапе жизни общества, дистанционное обучение играет немаловажную, а возможно, передовую роль в образовании и воспитании [3].

Дистанционное обучение (ДО) – технология обучения на расстоянии, при которой преподаватель и обучаемые физически находятся в различных местах. Дистанционная форма обучения не заменяет, а дополняет очную и заочную формы обучения.

Система общедоступного образования призвана обеспечить равноправную возможность получения образования для всех категорий лиц без исключения. Эта возможность ценна для людей, которые физически не могут добраться до места учебы.

Гибкость условий формирования собственной образовательной программы помогает государственным служащим, инженерам, педагогам в реализации творческого и научного



потенциала, а также в повышении квалификации в более удобном формате времени и общения.

Ранее, дистанционное обучение означало заочное обучение. Однако это не совсем так. Когда речь идет о процессе дистанционного обучения, то предполагается наличие в этом процессе преподавателя и учащихся, их постоянное общение. В этом принципиальное отличие дистанционного обучения от различных форм заочного обучения, систем и программ самообразования, представленными автономными курсами на каких-либо носителях, при работе с компьютерными программами. В этом же ряду следует рассматривать и процесс самообразования на основе сетевых программ, курсов и т. д., где не предусматривается взаимодействия педагогов с учащимися. Применять в данном случае термин «дистанционный» не вполне правильно, поскольку речь идет о самостоятельной работе любого учащегося с обучающей программой, или информационно-образовательными ресурсами.

В центре процесса дистанционного обучения находится не преподавание, а учение, то есть самостоятельная познавательная деятельность обучаемого по овладению знаниями, умениями и навыками. При этом обучающийся должен не только владеть навыками работы с компьютером, но и способами работы с учебной информацией, с которой он встречается в процессе дистанционного обучения [4].

При дистанционном обучении достигается главная цель информационной революции – доступность поиска, хранения, обработки и представления информации различными способами и на разном качественном уровне.

Благодаря информатизации общества, образование стало более мобильным и доступным. Усовершенствование методов, способов и технологий позволяет педагогам, опираясь на традиционные знания, реализовывать свои идеи на более высоком технологическом и профессиональном уровне.

Каждая информационная революция вносила свой вклад в развитие современной педагогической науки, однако, наиболее ощутимый толчок для педагогики дала последняя революция, подняв индустриальное общество на новый, информационный уровень.

Главным продуктом и ценностью современного общества являются знания. Повышение качества, доступности и гибкости образования, а также, формирование информационной культуры на основе создания современной информационной образовательной среды – это основные приоритеты развития образования в современном обществе.

### Литература

1. Абрамов А.Г., Булгаков М.В., Иванников А.Д., Сигалов А.В. Федеральный портал «Информационно-коммуникационные технологии в образовании»: пять лет в образовательном Рунете // Дистанционное виртуальное обучение. 2009. №3. С. 14-30.
2. Афанасьева И. Э. Использование ИКТ в образовательном процессе // Перспективы развития науки и образования. 2012. С. 17-19.



3. Бабич И. Н. Новые образовательные технологии в век информации // Применение новых технологий в образовании: Материалы XIV Международной конференции. Троицк, 2009. С. 68-70.

4. Кузнецов А.А., Хеннер Е.К., Имакаев В.Р. и др. Информационно-коммуникационная компетентность современного учителя // Информатика и образование. 2010. №4. С. 3-11.

© Попова Д.Е., Кузьмичева Е.А., 2021

УДК 004.5

**Григорьева Д.Д., Зайнуллина Л.М.**

*Научный руководитель: Шаймиева Э.Ш.*, д-р экон. наук  
Казанский инновационный университет  
им. В.Г. Тимирязова, г. Казань, Россия

## **ПОДХОДЫ К УТОЧНЕНИЮ ПОНЯТИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ ЕГО ПРИМЕНЕНИЯ В КОНЦЕПЦИИ ЭЛЕКТРОННОГО ПРАВИТЕЛЬСТВА**

В настоящем исследовании нами изучаются развитие понятий «электронного правительства», «искусственного интеллекта», технологий на основе искусственного интеллекта<sup>1</sup> в условиях развития цифровой экономики [3; 5; 18]. Объектом исследования выступают соответствующие нормативно-правовые документы в области развития электронного правительства, информационно-коммуникационных технологий, искусственного интеллекта в РФ [6]. Предметом исследования являются вопросы взаимодействия понятийных аппаратов в области электронного правительства и технологий искусственного интеллекта [5; 13].

Концепция электронного правительства в РФ реализуется в условиях развития информационного общества в российском экономическом пространстве с 2010 г., масштабного применения информационно-коммуникационных технологий (ИКТ) в системе государственно-муниципального управления<sup>2</sup>, предоставления услуг физическим, юридическим лицам в электронном форме [7; 8; 9]. Понятие «электронного правительства» понимается в широком и узком слове, в двух пониманиях упоминается использование ИКТ в деятельности органов государственной власти, в процессе трансформации внутренних и внешних взаимодействий в системе государственного управления [5; 17].

В «Стратегии развития информационного общества...» отмечается инфраструктура электронного правительства как «...совокупность размещенных на территории Российской Федерации государственных, информационных систем, программно-аппаратных средств и сетей связи, обеспечивающих при оказании услуг и осуществлении функций в электронной форме взаимодействие органов государственной власти РФ, органов местного самоуправления, граждан и юридических лиц...» [15]. Таким образом, ИКТ используется в развитии электронного правительства на основе его архитектуры в областях: G2C<sup>3</sup>, G2G<sup>4</sup>, G2E<sup>5</sup>, G2B<sup>6</sup>, а также в трех направлениях развития электронного правительства: электронное

<sup>1</sup> Понятия «искусственного интеллекта» и «технологии на основе искусственного интеллекта» являются синонимами в настоящем исследовании.

<sup>2</sup> Под «информационным обществом» понимается «...общество в котором информация и уровень ее применения и доступности кардинальным образом влияют на экономические и социокультурные условия жизни граждан...» [15].

<sup>3</sup> Взаимодействие государственных органов с гражданами (Government to Citizen).

<sup>4</sup> Горизонтальное и вертикальное взаимодействие государственных структур (Government to Government).

<sup>5</sup> Обеспечение взаимодействия государственных служащих (Government to Employees).

<sup>6</sup> Взаимодействие государственных органов с бизнес-структурами (Government to Business).

администрирование (e-administration), электронное взаимодействие с гражданами (e-services), развитие информационного общества (e-society). ИКТ не заменяет человеческий интеллект, способствуют реализации функций в государственной и муниципальной службе [10-12; 17].

Понятие «искусственного интеллекта» (ИИ) впервые упоминается в 2009 г., где под ИИ понимается «...моделируемая (искусственно воспроизводимая) интеллектуальная деятельность мышления человека...» [6].

В «Национальной стратегии развития искусственного интеллекта на период до 2030 г.» введено понятие «искусственного интеллекта», под которым понимается «... комплекс технологических решений, позволяющий имитировать когнитивные функции человека (включая самообучение и поиск решений без заранее заданного алгоритма) и получать при выполнении конкретных задач результаты, сопоставимые, как минимум, с результатами интеллектуальной деятельности человека. Комплекс технологических решений включает в себя информационно-коммуникационную инфраструктуру, программное обеспечение (в том числе, в котором используются методы машинного обучения), процессы и сервисы по обработке данных и поиску решений...» [6]. Очевидной является эволюция понятия искусственного интеллекта с 2009 по 2021 г. в российском экономическом пространстве. Характеристикой ИИ выступают: (а) его возможность имитировать когнитивные (познавательные) функции человека, (б) возможность получения результатов на основе заданных человеческим интеллектом параметров деятельности, (в) его моделируемость со стороны человеческого интеллекта. Данные характеристики отличают ИИ от ИКТ, используемых в развитии электронного правительства в РФ с 2010 г., представленные в определении электронного правительства в широком и узком смысле слова. Отсюда возникает вопрос о необходимости уточнения областей применения ИИ в развитии информационного общества, что, в свою очередь, приведет к использованию ИИ в развитии электронного правительства.

Сфера действия принятого ФЗ от 31.07.2020 №258-ФЗ «Об экспериментальных правовых режимах в сфере цифровых инноваций в Российской Федерации» является – в том числе – предоставление государственных и муниципальных услуг и осуществление государственного контроля (надзора) и муниципального контроля с использованием цифровых технологий [16]. Однако понятия ИИ, ИИ-технологий нами не выявлено.

В этой связи научно-практический интерес представляет опыт Китая в области развития и применения ИИ-технологий (или технологий с использованием ИИ) в государственном управлении. В работе Примшиц Д., Голубева С. отмечаются достижения китайского государства в области развития ИИ-технологий, где государство КНР является лидером в глобальном масштабе по количеству патентов за период 2014-2018 гг. с показателем роста 190% [13].

Значение ИИ-технологий в шестом технологическом укладе NBIC-технологий (нано-, био-, инфо-, -когнитивных) неоспоримо. Опыт развития ИИ-технологий в управлении государством, принятии управленческих решений в различных сферах развития электронного правительства, в том числе, в антикоррупционной политике, в сфере оказания медицинских

услуг, социального обеспечения, в развитии системы оказания помощи гражданам в чрезвычайных ситуациях и др. формирует основу развития институциональных условия для внедрения ИИ-технологий в следующих направлениях: (а) условия развития ИИ-технологий на государственных, муниципальных предприятиях, концепции электронного правительства; (б) источники финансирования развития ИИ-технологий в системе ГиМУ; (в) учет и контроль ИИ-технологий как объектов интеллектуальной собственности, автором и патентообладателем которых является государство, государственные структуры; (г) обучение персонала (государственных и муниципальных служащих) ИИ-технологиям; (д) учет возникающих рисков и угроз со стороны ИИ на возможные сферы человеческой деятельности [1; 2; 14].

При развитии понятия ИИ, ИИ-технологии для концепции электронного правительства актуальность приобретает вопрос ответственности решений, принятых на основе и/или с использованием ИИ-интеллекта.

В завершении исследования можно сделать следующие выводы:

1. в современных трактовках электронного правительства в РФ понятие искусственного интеллекта не упоминается;
2. понятие «искусственного интеллекта» в нормативно-законодательную практику РФ введено в 2019 г.;
3. значение искусственного интеллекта приобретает значение интеллектуального, технологического лидерства государств в шестом технологическом укладе (на примере опыта и достижений КНР в области ИИ-технологий);
4. обоснована необходимость использования понятия ИИ-технологий для его применения в электронном правительстве в РФ.

### Литература

1. Булгакова Е.В., Булгаков В.Г., Акимов В.С. Механизмы электронного государства по противодействию коррупции // Правовая информатика. 2014. №1. С. 23-27.
2. Гумерова Г.И., Шаймиева Э.Ш. Процессная интеграция в системе электронного здравоохранения на основе Индустрии 4.0: разработка бизнес-модели для российской практики (теоретический аспект)» // Врач и информационные технологии. М., 2019. №2. С. 50-66.
3. Дятлов С.А., Марьяненко В.П., Селищева Т.А. Информационно-сетевая экономика: структура, динамика, регулирование. М., 2016. 414 с.
4. Национальный стандарт Российской Федерации. Информационное обеспечение техники и операторской деятельности. Утвержден и введен в действие приказом Федерального агентства по техническому регулированию и метрологии от 15.12.2009 №959-ст. <https://clck.ru/UG62J>
5. OECD E-Government Studies E-Government for Better Government. OECD. 2005. 206 p. <https://clck.ru/UG5zd>



6. Национальная стратегия развития искусственного интеллекта на период до 2030 г. Утверждена Указом Президента РФ от 10.01.2019 №490. <https://clck.ru/UG6LN>

7. Об обеспечении доступа к информации о деятельности Правительства Российской Федерации и федеральных органов исполнительной власти: пост. Правительства Рос. Федерации от 24 ноября 2009 г. №953; в ред. пост. Правительства Рос. Федерации от 10 июля 2013 г. №583 // Собр. законодательства Рос. Федерации. 2009. №48, ст. 5832; 2013. №30 (часть II), ст. 4107.

8. О единой системе межведомственного электронного взаимодействия: пост. Правительства Рос. Федерации от 8 сентября 2010 г. №697; в ред. пост. Правительства Рос. Федерации от 19 марта 2014 г. № 208 // Собр. законодательства Рос. Федерации. 2010. №38, ст. 4823; 2014. №12, ст. 1303.

9. Об использовании простой электронной подписи при оказании государственных и муниципальных услуг: пост. Правительства Рос. Федерации от 25 января 2013 г. №33; в ред. пост. Правительства Рос. Федерации от 9 декабря 2013 №1135 // Собр. законодательства Рос. Федерации. 2013. №5, ст. 377; 2013. №50, ст. 660.

10. О плане перехода на предоставление государственных услуг и исполнение государственных функций в электронном виде федеральными органами исполнительной власти: распоряжение Правительства Рос. Федерации от 17 октября 2009 г. №1555-р; в ред. распоряжения Правительства Рос. Федерации от 20 мая 2014 г. №869-р // Собр. законодательства Рос. Федерации. 2009. №43, ст. 5155; 2014. №21, ст. 2763.

11. Об утверждении Концепции снижения административных барьеров и повышения доступности государственных и муниципальных услуг на 2011–2013 годы и Плана мероприятий по реализации указанной Концепции: распоряжение Правительства Рос. Федерации от 10 июня 2011 г. №1021; в ред. распоряжения Правительства Рос. Федерации от 28 августа 2012 г. №1548-р // Собр. законодательства Рос. Федерации. 2011. №26, ст. 3826; 2012. № 36, ст. 4963.

12. Об утверждении Положения о пользовании официальным сайтом Российской Федерации в сети Интернет для размещения информации о размещении заказов на поставки товаров, выполнение работ, оказание услуг и о требованиях к технологическим, программным, лингвистическим, правовым и организационным средствам обеспечения пользования указанным сайтом: приказ Минэкономразвития России № 646, Казначейства РФ №21н от 14 декабря 2010 г. // Российская газета. 2011. №11.

13. Примшиц Д., Голубев С. Китайский подход к ускоренному освоению технологий искусственного интеллекта // Синергия знаний. №194. 2019. С. 43-50.

14. Смирнов Н.В. Антикоррупционная политика как фактор социально-экономического развития России и ее регионов // Труды Института системного анализа Российской академии наук. 2009. Т. 43. С. 53-69.

15. Стратегия развития информационного общества в Российской Федерации на 2017 - 2030 годы». Указ Президента РФ от 09.05.2017 г. №203 <https://clck.ru/EhPxу>

16. ФЗ от 31.07.2020 2020 N 258-ФЗ «Об экспериментальных правовых режимах в сфере цифровых инноваций в Российской Федерации». <https://clck.ru/U4ReD>

17. Чугунов А.В. Электронное правительство: базовые концепции и российская практика // Интернет и современное общество: Труды IX Всероссийской объединенной конференции (IMS-2006) (Санкт-Петербург, 14-16 ноября 2006 г.). СПб., 2006. С. 187-192

18. Шеве Г., Гумерова Г.И., Хюзиг С., Шаймиева Э. Менеджмент цифровой экономики. Менеджмент 4.0. М., 2019. 232 с.

© Григорьева Д.Д., Зайнуллина Л.М., Шаймиева Э.Ш., 2021

УДК 519.687.7

**Лазоренко Е.В.**

*Научный руководитель:* **Слива М.В.**, канд. пед. наук  
Нижевартовский государственный университет,  
г. Нижевартовск, Россия

### ЧПУ СТАНОК – ЛАЗЕРНЫЙ ГРАВЕР

Уже давно не новость, что в мире постоянно используют ЧПУ станки, и они в основном промышленные, которые дорого стоят. Но для того, чтобы использовать станок в домашних условиях, не обязательно покупать промышленные. Станки с ЧПУ (числовым программным управлением) – это высокотехнологичное оборудование, в котором исполнительные устройства (приводы) управляются автоматически (<https://clck.ru/U4Sjo>). Для того, чтобы собрать ЧПУ станок, не обязательно иметь под рукой высокоточные модули и контролеры. Достаточно использовать Arduino и специальные модули для создания станка.

Для создания основы ЧПУ станка были использованы материалы, которые доступны в любом строительном магазине. Так как для основы ЧПУ станка была использована фанера, то не было никакого труда закрепить на ней:

- управляющий блок, который состоит из Arduino Uno и CNC Shield;
- Ось X (рис. 1);
- стойки для оси Y (рис. 1).

Сами стойки были сделаны из алюминиевых уголков. На них и крепится ось Y и лазерный модуль (рис. 2). Чтобы во время работы, или от случайного соприкосновения с ЧПУ станком, лазер не сместился, нужно было укрепить стойки OY (рис. 2). Помимо этого, в случае если настройка драйвера была случайно сбита, алюминиевые крепежи могут выступать как радиатор для шаговых двигателей от перегрева.

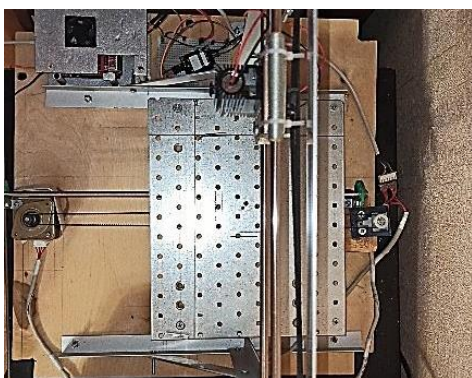


Рис. 1. Крепление осей X и Y



Рис. 2. Крепление оси Y и модуля лазера

Электронная часть ЧПУ станка состоит из следующих компонентов: Arduino; CNC Shield; Модуль управления шаговым двигателем DRV8825; Шаговый двигатель; Модуль лазера 1000 мВт; Реле.

CNC Shield v3.0 (рис. 3) – это расширяющая плата размером с Arduino UNO. Она предназначена для создания на основе контроллера Arduino UNO 3D принтеров, станков с числовым программным управлением, в том числе гравировальных, фрезерных, маркировальных, станков портальной резки, промышленных роботов. Контроллер Arduino позволяет работать станку автономно или управлять с помощью компьютера через USB-порт (<http://www.electronic52.in.ua/stanki-cnc--lazernye-i-drugie/plata-dlya-chpu-cnc-stanka>). Плата CNC Shield крепится поверх платы Arduino.

Плата имеет четыре слота, для подключения четырех драйверов двигателей. Слоты, обозначенные желтым цветом, отвечают за оси X, Y, Z, красный слот за ось A.

Ось A может дублировать одну из осей X, Y, Z с помощью дополнительного двигателя и драйвера. То есть входной сигнал приходит одинаковый, а драйверы и шаговые двигатели разные, но двигаются одинаково. Например, ось A может быть использована для двигателя экструдера в случае 3D-принтера (<http://www.electronic52.in.ua/stanki-cnc--lazernye-i-drugie/plata-dlya-chpu-cnc-stanka>). Но поскольку создавался Лазерный гравер, было использовано лишь 2 слота. Слот x и слот y.

Таблица 1

Характеристики CNC Shield

Параметр	Описание
Модель (версия)	CNC Shield version 3.0
Совместимость с ...	Arduino UNO и другие совместимые платы
Количество осей	4 (X, Y, Z, A)
Напряжение питания логической части	5 В
Напряжение питания силовой части	12 – 36 В
Совместимость с драйверами	A4988, DRV8825 и другие
Интерфейсы	UART, I2C
Прошивка	Arduino GRBL
Размеры	65 x 55 x 20 мм
Вес	32 г

Плата расширения может работать как минимум с двумя драйверами (рис. 4) двигателей A4988 или DRV8825 при помощи прошивки GRBL.

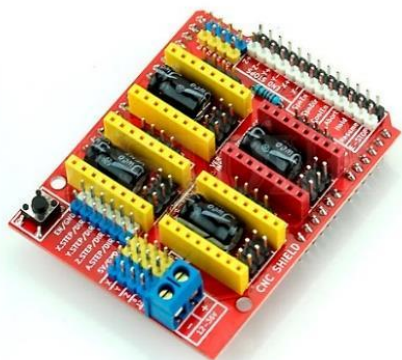


Рис. 3. CNC Shield



Рис. 4. DRV8825 и A4988



Почитав несколько электронных ресурсов, для чпу станка был выбран драйвер DRV8825. DRV8825 - это драйвер шагового двигателя, с током до 2.2А и возможностью делить шаги микрошагом вплоть до 1/32 (<https://clck.ru/U4Si2>). Драйвер пользуется популярностью из-за того, что рассчитан на хороший ток, хорошо делит шаги и его можно легко установить в плату RAMPS 1.4 и cnc shield.

Прежде чем начинать собирать электронику, необходимо выставить по табличке переключки (MODE0, MODE1, и MODE2) в соответствии с таблицей 2 (<http://www.electronic52.in.ua/stanki-cnc--lazernye-i-drugie/plata-dlya-chpu-cnc-stanka>).

Таблица 2

Таблица переключек микрошагов

MS1	MS2	MS3	Разрешение микрошага
Низкий	Низкий	Низкий	Полный шаг
Высокий	Низкий	Низкий	1/2 шага
Низкий	Высокий	Низкий	1/4 шага
Высокий	Высокий	Низкий	1/8 шага
Низкий	Низкий	Высокий	1/16 шага
Высокий	Низкий	Высокий	1/32 шага
Низкий	Высокий	Высокий	1/32 шага
Высокий	Высокий	Высокий	1/32 шага

**Основные моменты:**

- повышение значения микрошага ведет к потере крутящего момента на шаговом двигателе;
- высокие значения микрошага не ведет к кратному увеличению точности работы оборудования, из-за наличия люфта в подвижных элементах конструкции.

После настройки микрошага необходимо установить драйвера шаговых двигателей. Драйверы A4988 и DRV8825 вставляются по-разному. Далее подключаем двигатели, к колодкам, которые находятся возле слотов осей X и Y (рис. 5) и настраиваем ток.

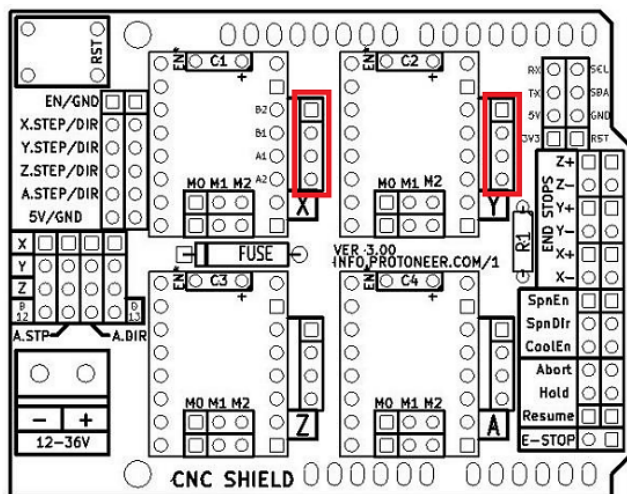


Рис. 5. Подключение шаговых двигателей к пинам X и Y



Чтобы настроить ток, нужно минусовой щуп мультиметра подсоединить к контакту GND (под номером 1 минус, он общий), плюсовым прикасаемся к корпусу подстроечного резистора на драйвере (рис. 6). На изображении он отмечен под числом 2. Крутим подстроечный резистор отверткой и замеряем расчетное напряжение  $V_{ref}$ . Таким образом выставляем правильный ток для драйвера шагового двигателя. Для каждого вида драйвера своя формула расчета  $V_{ref}$  (<https://clck.ru/U4Sir>). При настройке питание CNC Shield и Arduino должно быть подключено.

После чего высчитываем по формуле ток для драйвера шагового двигателя.  $V_{ref}$  – пин замера напряжения для установки тока по формуле. Current Limit – ток шагового двигателя.

Формула для драйвера DRV8825

$$\text{Current Limit} = V_{ref} * 2$$

$$V_{ref} = \text{Current Limit} / 2$$

Например, для 17HS4401:  $V_{ref} = 1,7 / 2 = 0,85V$  [3].

Дальше можно подключить модуль лазера. Модуль лазера подключается через специальный драйвер, который регулирует мощность лазера от 12V. Таким образом, лазер может выжигать не только черно-белые картинки, но и делать градиацию серого. Модуль лазера подключается через реле и запитывается от 5 V (рис.7). Что само по себе снижает мощность лазера. Из-за этого он может плохо выжигать (гравировать).

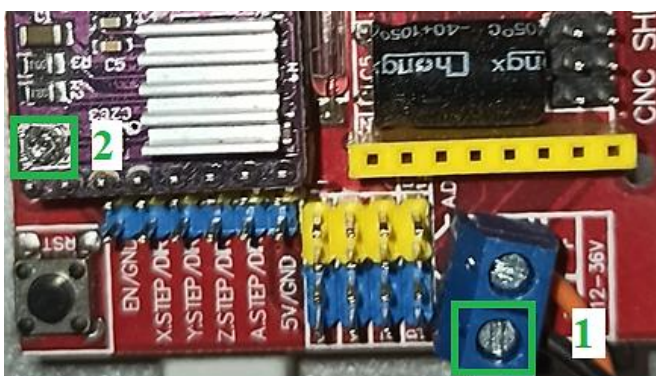


Рис. 6. Настройка тока для ШД  
(<http://www.electronic52.in.ua/stanki-cnc--lazernye-i-drugie/plata-dlya-chpu-cnc-stanka>)

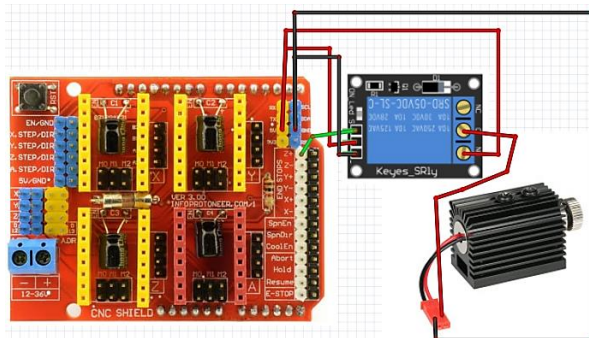


Рис. 7. Схема подключения модуля лазера  
(<http://www.electronic52.in.ua/stanki-cnc--lazernye-i-drugie/plata-dlya-chpu-cnc-stanka>)

При постоянной работе драйвера нагреваются, поэтому подключаем активное охлаждение для радиаторов драйверов шаговых двигателей.

Следующим шагом нужно загрузить прошивку GRBL в Arduino. После того как загрузили в Arduino готовую прошивку (GRBL), в ней ничего менять не надо. Все настройки производятся в консоли ПО (LaserGRBL). В самой программе для того, чтобы модуль лазера работал, нужно выставить в прошивке параметр laser mode на единицу.

Прошивка (GRBL) внутри контролера обменивается информацией с ПК при помощи G-кодов.

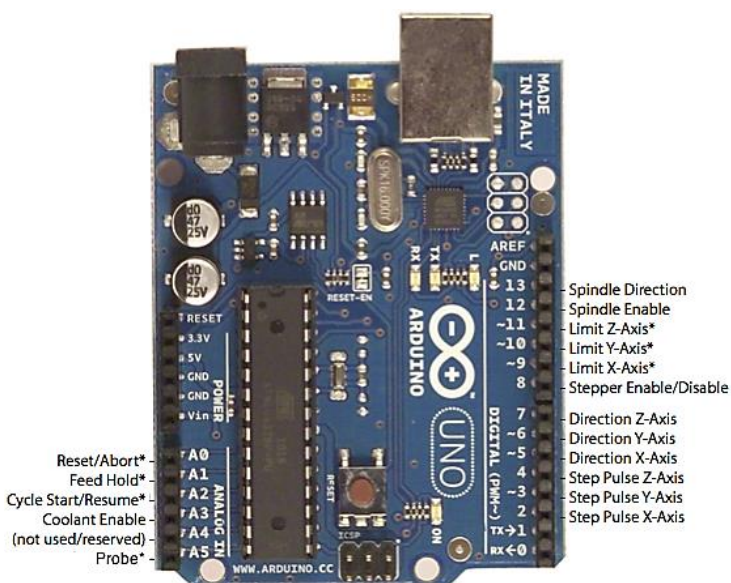
ПК из программы управления (LaserGRBL), через USB порт ПК посылает на COM порт Arduino строчки с GCODE формата G91 G1 X10 Y10 F3000. Шаговые двигатели вращаются [1].

Также помимо управления станком при помощи ПО на плате CNC Shield есть контакты (рис. 9) для физического управления станком через кнопки (<http://www.electronica52.in.ua/stanki-cnc--lazernye-i-drugie/plata-dlya-chpu-cnc-stanka>):

Таблица 3

Распиновка физических кнопок

Описание	Значение
Аварийной кнопки останова	(E-STOP)
Кнопка паузы	(Hold)
Кнопка продолжения	(Resume)
Кнопка возвращения на исходную позицию	(Abort)
Включения шпинделя	(SpnEn)
Направления шпинделя	(SpnDir)
Включения подачи охлаждения	(CoolEn)
Концевики	(X+ X- Y+ Y- Z+ Z-)



\* - Indicates input pins. Held high with internal pull-up resistors.

Рис. 8. Использование портов ардуино

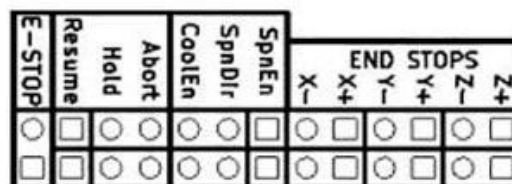


Рис. 9. Контакты физического управления

На данный момент лазерный гравер может выжигать только черно-белые изображения на пластмассе, дереве и картоне. В ближайшее время как придет драйвер, лазерный гравер сможет выжигать уже градацию серого, и можно будет уже вырезать из не толстой фанеры разные узоры и гравировать орг. стекло. В будущем лазерный гравер будет так же работать и по металлу.

© Лазоренко Е.В., Слива М.В., 2021

УДК 004.05

**Брагин К.А., Кушев В.О.**, канд. пед. наук  
Пермский государственный национальный  
исследовательский университет, г. Пермь, Россия

## **РАЗРАБОТКА ОПТИМАЛЬНОЙ МЕТОДИКИ ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

В цикле разработки программного обеспечения (далее ПО) присутствует этап его тестирования [1, с. 32]. Такой этап позволяет определить качество программно-технической системы, посредством определения соответствия требованиям, предъявленным к разработанному ПО.

При формировании методики тестирования программного обеспечения, тестировщики определяют необходимые методы тестирования, которые будут использоваться для оценки качества конкретных программ, а также определяют план, в котором отображается временная шкала тестирования. Универсальной методики тестирования для разнотипных программ не существует. Если провести анализ выбора тестировщиками типов и методов тестирования, возможно, сформировать их классификацию, которая поможет оптимизировать выбор типов тестов и методов тестирования для конкретного ПО. Оптимальная методика тестирования позволит произвести эффективное тестирование за необходимое и достаточное время.

В практике тестирования принято разделение тестирования на тестирование функционального показателя качества, называемое «функциональным тестированием», и тестирование других показателей качества, называемое «нефункциональным тестированием». Тип тестирования, используемого для определения показателя качества, отличного от функциональной пригодности, обычно называют нефункциональным типом тестирования и к нему можно отнести такие типы тестирования, как нагрузочное тестирование, стрессовое тестирование, тестирование на возможность проникновения, тестирование удобства использования и т. д. [2, с. 41].

Под функциональным тестированием обычно подразумевается проверка программного обеспечения на выполнение заявленных функций. Функциональное тестирование в основном включает тестирование «черного ящика» и не касается исходного кода приложения. Это тестирование проверяет пользовательский интерфейс, API, базу данных, безопасность, связь клиент/сервер и другие функциональные возможности тестируемого приложения. Тестирование такого типа может проводиться как вручную, так и с использованием средств автоматизации (автотестирование). В методику тестирования также должно быть включено описание степени автоматизации каждого типа тестирования.

Для того чтобы проверить насколько ПО удобно для его предполагаемого применения, необходимо произвести проверку эргономичности (юзабилити) данного обеспечения. Это понятие означает легкость и доступность применения. Более конкретное определение, означает, насколько быстро можно научиться пользоваться пользовательским интерфейсом, какова эффективность его применения, как легко на нем все запоминается, насколько

интерфейс подвержен различным ошибкам и сбоям и насколько пользователям нравится с ним работать [3, с. 14]. Важность проверки эргономичности ПО может быть меньше чем у остальных проверок качества и проводится не для всех видов программ, поэтому проводить тестирование нужно после успешного прохождения приоритетных типов тестирования.

Если разрабатываемое ПО подвержено внешнему воздействию с целью получения доступа к функциям или данным, необходимо тестирование ПО на способность противостоять к попыткам взлома системы. Тестирование безопасности проверяет фактическую реакцию защитных механизмов, встроенных в систему, на незаконное проникновение. Не все системы подвержены внешнему воздействию, а те что подвержены разделяются на уровни безопасности ПО [4]. В зависимости от уровня безопасности должен осуществляться выбор метода тестирования безопасности.

Тестирование нагрузки позволяет оценить поведение системы, когда все большее число пользователей обращаются к ней в одно и то же время. Стрессовое тестирование оценивает устойчивость приложения к нагрузкам, превышающим ожидаемую [5, с. 232]. Если программа предусматривается как индивидуальная, а не как многопользовательская, то данный тип тестирования исключается из разрабатываемой оптимальной методики тестирования ПО. Нагрузочное тестирование во многом похоже на автотестирование, только с применением большего количества сценариев. В последнее время, большие компании по разработке ПО, стараются объединить автотесты и нагрузочные тесты, с целью уменьшения трудозатрат для того или иного типа тестирования.

Мы видим, что, некоторые типы тестирования возможно произвести автоматизированным способом. Так, например, с помощью тестовых сценариев и средств автоматизированного тестирования, возможно проверить программу с высокой точностью и намного быстрее, чем эта же проверка будет произведена вручную. Несмотря на положительные стороны автотестов, есть несколько проблем в их реализации. Одной из причин отказа от автотестов является их дороговизна. Трудозатраты, потраченные на разработку автотестов в несколько раз, превышают трудозатраты, потраченные на тестирование вручную. Но в перспективе тестирования, если программный пакет требует произведения тестирования повторно, то целесообразней будет заранее произвести разработку автотестов [6]. Второй причиной отказа от автотестов является уникальность ПО. Не все виды программ возможно протестировать с помощью автотестов, так как автотесты производятся по заранее подготовленному сценарию, и даже малейшее изменение в программе несет за собой неисправно работающий автотест. Здесь наблюдается противоречие. С одной стороны, автотесты должны ускорить процесс тестирования, а с другой стороны трудозатраты на них и вероятность некорректного тестирования при изменении кода программы мотивируют к отказу от них. Поэтому выбор метода тестирования важно производить исходя из результатов анализа других методик тестирования.

Существуют и другие методы, и типы тестирования ПО, но на практике их применение не имеет высокую необходимость, так как частично затрагиваются при проверке других показателей качества. Таким образом, если произвести исследование по всем типам



тестирования, появится возможность разделить их на группы по признаку значимости. В более значимую группу войдут те типы тестирования, которые не проверяются никакими другими тестами, и имеют большое значение для получения качественного ПО.

При разработке методики тестирования следует учитывать принимаемую во внимание методологию. Выше говорилось о линейной методологии тестирования. Альтернативой для линейной методологии разработки ПО служат такие методологии, жизненный цикл в которых является замкнутым, и относятся они к группе гибких методологии разработки программного обеспечения, примером этих методологий является методология Scrum [7].

В гибкой методологии разработки ПО предусматривается, что с течением времени происходит постепенное изменение функционала в программную систему и, соответственно, коррекция ее исходного кода, что в свою очередь подразумевает возможность возникновения новых, еще незарегистрированных дефектов [7, с. 391]. Таким образом, существует необходимость в периодической проверке ПО, называемое регрессионным тестированием [8, с. 80]. Такое тестирование также подвергается анализу и по итогу должно быть также учтено в методике тестирования.

Без методики тестирования производить оценку качества программы крайне неэффективно и дорого. В зависимости от объема производимых тестов возрастают или уменьшаются трудозатраты тестировщиков, которые проводят эти тесты. Важность правильно составленной методики тестирования очень высока. Немаловажно в процессе подбора оптимальной методики тестирования не выходить за рамки требований к этапу проверки качества программ. Для этого производится специальный анализ требований, на самом раннем этапе всего жизненного цикла тестирования [9, с. 30].

В процессе разработки программных систем, в зависимости от дальнейшего использования ПО, разрабатываются разные методики тестирования. При наличии нескольких методик тестирования, которые после всех пройденных типов тестирования показали некоторую зависимость между выбранной методикой и результатом тестирования, возможно произвести расчет закономерности выбора тех или иных методов тестирования и сформировать классификацию типов тестирования в зависимости от:

- параметров, характеризующих тестируемое ПО;
- вида программного обеспечения;
- операционной системы устройства;
- аппаратного обеспечения и т. д.

Программное обеспечение подразделяется на системное, инструментальное и прикладное ПО. Большинство разрабатываемых программ относятся к прикладному ПО. Далее разделение происходит от вида устройства, на котором выполняется ПО. От вида ПО очень зависит выбор дальнейшей методики тестирования.

Для персональных компьютеров разрабатываются такие виды ПО, как веб-приложения и пакеты прикладных программ. В работе веб-приложения происходит взаимосвязь клиента с веб-сервером посредством браузера. Пакеты прикладных программ устанавливаются в память устройства и чаще всего не используют внешнее подключение к чему-либо.



Для мобильных устройств разработка ПО также разделяется на разработку веб-приложений и разработку пакетов программ. В большинстве случаев при работе с веб-сервером разница между мобильными и персональными устройствами не велика, но в то же время пользовательские интерфейсы данных веб-приложений имеют серьезные различия. При работе с помощью мобильных устройств учитывается тот факт, что форм-фактор данных устройств имеет небольшие размеры.

В таблице продемонстрирован вариант выбора типа тестирования в зависимости от вида ПО. В таблице приведена сокращенная классификация типов тестирования ПО, которую в дальнейшем необходимо расширять, включая данные из новых методик тестирования. Исходя из данных представленных в таблице, можно выделить, что пакеты программ, не нуждаются в нескольких типах тестирования. На основании этого можно сделать вывод, что, имея больше данных для составления закономерности, возможно, произвести еще более расширенную классификацию данных, и в последующей разработке оптимальной методики тестирования ПО воспользоваться этой классификацией.

Таблица

Требуемые типы тестирования в зависимости от вида ПО

Вид ПО	Типы тестирования				
	Функциональное тестирование	Тестирование производительности		Тестирование безопасности	Проверка эргономичности
		Нагрузка	Стресс-тестирование		
Веб-приложение (мобильные устройства)	+	+	+	+	+
Веб-приложение (персональные компьютеры)	+	+	+	+	+
Пакет программ (мобильные устройства)	+	-	-	-	+
Пакет программ (персональные компьютеры)	+	-	-	-	+

После анализа закономерности выбора тестов для еще большего удобства представления конечных данных, их следует преобразовать в вид, предлагающий оптимальную методику автоматизированным способом, после выбора заданных параметров.

Ключевыми параметрами при выборе методики будут:

1. Требования к тестированию ПО.
2. Вид программного обеспечения.
3. Квалификация предполагаемых пользователей.
4. Предполагаемое время тестирования системы.
5. Архитектура платформы.
6. Возможная ОС
7. Уникальность ПО.

Разработка оптимальной методики тестирования на основе исследования предыдущих тестирований, позволяет автоматизировать процесс документирования, уменьшить трудозатраты на разработку такой методики вручную, произвести тестирование за необходимое и достаточное время.

В результате, произведенного анализа зависимости выбора типов и методов тестирования от ключевых параметров, появляется возможность классифицировать полученные данные, и на основе этой классификации разрабатывать следующие оптимальные методики тестирования.

### Литература

1. Орлик С. Введение в программную инженерию и управление жизненным циклом ПО, 2005. <https://clck.ru/UG8BV>
2. ГОСТ Р 56920-2016/ISO/IEC/IEEE 29119-1:2013 Системная и программная инженерия. Тестирование программного обеспечения. Ч. 1.
3. Нильсен Я., Лоранжер Х. Web-дизайн: удобство использования Web-сайтов. М.: Вильямс, 2007. 366 с.
4. ГОСТ Р 56939-2016 Защита информации. Разработка безопасного программного обеспечения. Общие требования.
5. Кристин Л., Грегори Д. Гибкое тестирование: Практическое руководство для тестировщиков ПО и гибких команд / Пер. с англ. М., 2010. 464 с.
6. Садыков А. Мифы автоматизированного и нагрузочного тестирования // JetInfo. 2019. №5-6(297). <https://clck.ru/UG8mD>
7. Кон М. Scrum: гибкая разработка ПО / Пер. с англ. М., 2011. 576 с.
8. Канер С., Фолк Д., Кек Нгуен Е. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений / Пер. с англ. К., 2001. 544 с.
9. Куликов С.С. Тестирование программного обеспечения. Базовый курс. Минск, 2017. 312 с.

© Брагин К.А., Кушев В.О., 2021

УДК 004.4'2

**Болхудере Е.И.**

*Научный руководитель: Пономарева Ю.С., канд. пед. наук*  
Волгоградский государственный социально-педагогический  
университет, г. Волгоград, Россия

## **СРАВНЕНИЕ СИСТЕМ РАЗРАБОТКИ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ**

В настоящее время невозможно представить общество без мобильных технологий. Они являются удобными, компактными и производительными средствами коммуникации, доступа в Интернет, автоматизации обработки различных данных. Приложения для мобильных устройств, предоставляют доступ к поисковым системам, картам, позволяют строить маршруты, играть в сетевые игры и изучать иностранные языки.

Поэтому на данный момент разработка мобильных приложений является востребованной сферой. Мобильные приложения – это программное обеспечение для мобильных устройств. Их способности отчасти зависят от возможностей, заложенных разработчиком приложений, частично от возможностей мобильных устройств.

В рамках данного исследования были поставлены следующие задачи:

1. Раскрыть требования к дизайну и функционалу мобильных приложений.
2. Провести обзор наиболее распространенных сред разработки мобильных приложений.
3. Построить теоретическую модель мобильного приложения “CLC”.
4. Привести описание разработки мобильного приложения “CLC” в средах AppInventor и Thinkable X.

В связи с большой конкуренцией на рынке мобильных приложений одной из главных задач разработчика – сделать свой продукт максимально удобным и юзабельным. Поэтому дизайн интерфейса приложения должен соответствовать: во-первых, эргономическим требованиям, во-вторых, потребностям своих потенциальных потребителей.

На основании анализа различных источников были сформулированы следующие требования к разработке дизайна мобильного приложения:

- организованность составляющих интерфейса — они обязаны быть логически структурированы и взаимосвязаны;
- объединение составляющих интерфейса — объединение в группы логически связанных составляющих - меню, формы;
- выравнивание составляющих интерфейса — плохо составленный интерфейс не может быть для кого-либо удобным;
- единый стиль составляющих интерфейса — стилевое оформление играет не последнюю роль, как раз оно сохраняется в памяти пользователя
- простота в использовании.

Также немаловажным элементом конкурентоспособности является функционал мобильного приложения.

Функционал мобильного приложения – мощный инструмент для продвижения данного продукта, который не стоит недооценивать. Он выполняет сразу несколько важных функций: повышает интерес пользователей к предоставляемым товарам и услугам компании, а также привлекает новых клиентов и партнеров. Это удобно для современного пользователя, поскольку доступ к приложению возможен в любое время в режиме 24/7 в любой точке мира.

Однако следует соблюдать баланс между предлагаемым функционалом мобильного приложения и требованиями клиентов. Идеальное приложение включает в себя все необходимые опции для выполнения своего непосредственного назначения, но, в то же время, он не должен быть перегружен.

Функциональный набор варьируется от предназначения разрабатываемого приложения и области его применения.

Возможный функционал мобильного приложения представлен на рисунке 1.



Рис. 1. Возможные функционал современного мобильного приложения

Существуют различные среды разработок мобильных приложений, они отличаются по различным характеристикам, например, удобность конструктора интерфейса, наличие встроенного комплекта различных средств разработки.

В данной статье рассматриваются такие среды разработки мобильных приложений, как Android Studio, MIT AppInventor и IDE Thinkable X.

*Android Studio* — интегрированная среда разработки для работы с платформой Android, созданная компанией Google в 2013 г. Языками программирования в Android Studio являются Java и C++.

Простой и удобный интерфейс, содержание всего необходимого функционала, возможность выбора версий API и вида устройства — все это очень сильно помогает начинающим и даже опытным программистам при работе, значительно ускоряя их продуктивность [1].

*MIT AppInventor* — это облачная среда визуальной разработки приложений для платформы OS Android. Работа в ней не требует знания языка программирования Java и Android SDK, достаточно знания элементарных основ алгоритмизации.

Язык MIT AppInventor очень похож на Scratch, в его основе лежит тот же принцип перетаскивания визуальных кирпичиков и сборки программы из блоков [3].

Приложения на AppInventor умеют подстраивать приложение под размеры экрана мобильного гаджета, управлять встроенной камерой, видят, как ориентирован телефон в пространстве и многое другое.

Понятный и удобный интерфейс делает эту среду очень востребованной и подходящей для разработки мобильных приложений даже при отсутствии специальных знаний и опыта работы в данной области.

*Thunkable X* разрабатывался как альтернатива MIT AppInventor. При работе в *Thunkable X* используется визуальное программирование с использованием блоков, но при желании можно применить и текстовое программирование.

Функциональность среды разработки и возможности по созданию приложений в *Thunkable X* заметно уступают альтернативам под Android, но при помощи трех десятков визуальных и невидимых компонентов можно создать простое приложение с нативным и гибридным интерфейсом, реализовать компоновки и навигацию между экранами, работать с мультимедиа, хранилищами данных, базой данных Firebase, таблицами AirTable, синтезатором речи, камерой, Google Maps, Yandex-переводчиком и другими [2].

*Thunkable X* находится в состоянии активного развития и к ее ограничениям можно отнести:

- нестабильность работы IDE и приличное количество багов, включая безвозвратное удаление блоков из редактора блоков при невыясненных обстоятельствах. Масштаб последствий этого можно уменьшить только путем восстановления утерянного из последней копии проекта (если она была создана);
- удаление вложенных блоков без предупреждения;
- отсутствие ряда необходимых функций помимо рюкзака – экспорта и импорта приложения, невозможность переименования компонентов;
- отсутствие блоков для многих свойств компонентов;
- наличие в палитре свойств компонентов большого количества свойств с интуитивно непонятной и недокументированной функциональностью.

Таким образом, в результате теоретического исследования мною был сделан вывод, что наиболее подходящими и удобными для разработки приложения являются MIT AppInventor и IDE *Thunkable X*.

В рамках данного исследования была поставлена задача изучения данных сред разработки с помощью разработки в них мобильного приложения.

Была построена теоретическая модель мобильного приложения “CLC”.

Основные возможности функционала, а также особенности оформления и использования данного приложения кратко представлены на рисунке 2.



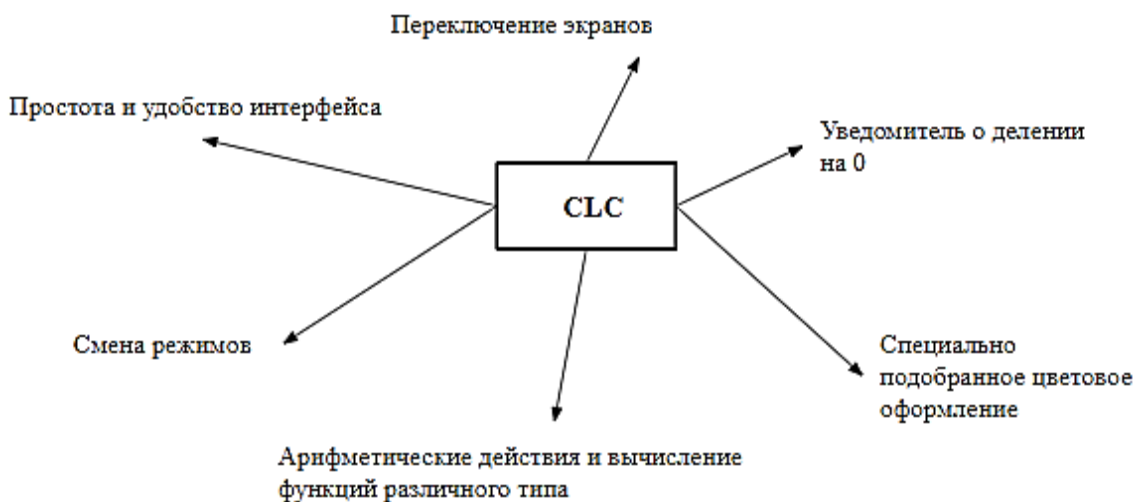


Рис. 2. Функциональные возможности и особенности интерфейса разрабатываемого мобильного приложения

Основными компонентами мобильного приложения будут:

- текст – для ввода числовых значений пользователем,
- надпись – для вывода полученных в результате вычисления значений, переключатель режима, а также кнопки – для ввода чисел и вычислений.

В качестве невидимого компонента будет уведомитель об ошибке при попытке поделить на ноль.

Схематичное расположение элементов интерфейса приложения представлено на рисунке 3.

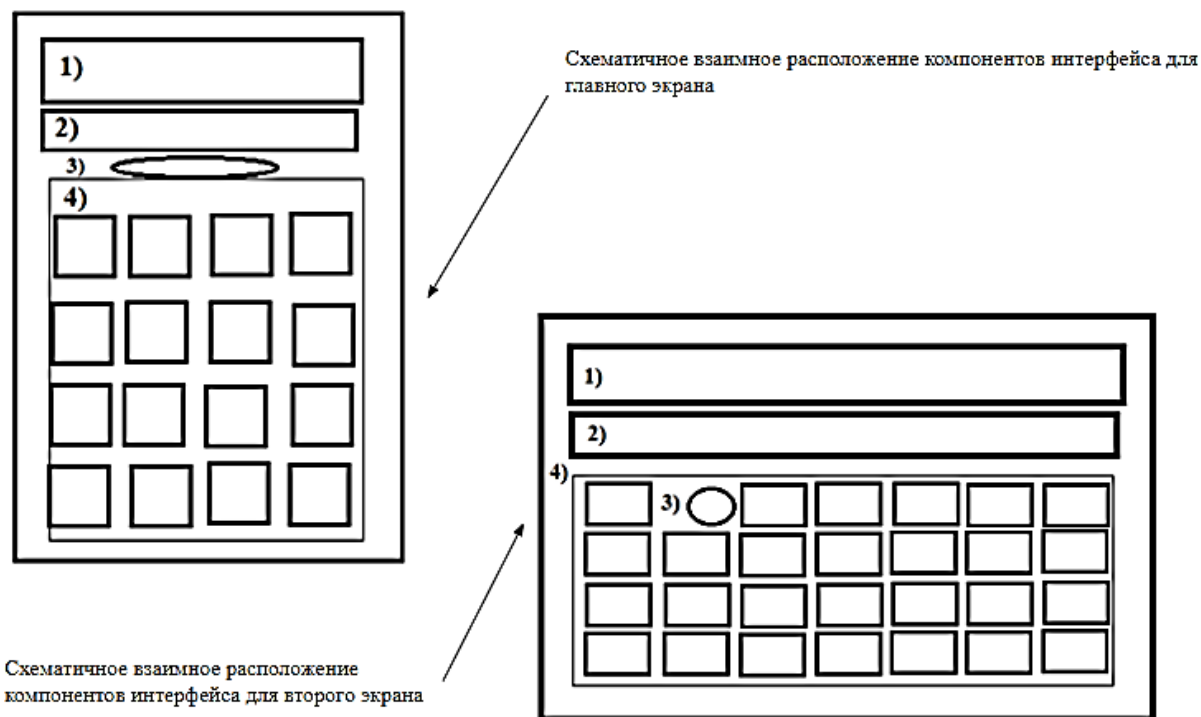


Рис. 3. Схематичное расположение элементов интерфейса приложения

В результате разработки данного приложения в MIT AppInventor, были изучены возможности интерфейса пользователя в данной среде, доступные для настройки свойства компонентов интерфейса (рис. 4).

Также была проведена работа с блочным программированием, разобрано применение обработчиков событий, использование свойств таких компонентов интерфейса как «Текст», «Надпись», «Кнопка», «Switch», «Уведомитель» различных операторов и математических действий (рис. 5).

Выявленным в ходе разработке приложения недостатками является то, что для проверки работы приложения на реальном устройстве нужно скачивать дополнительный инструмент — MIT AI2 Companion, а также отсутствие некоторых необходимых встроенных блоков. Несмотря на это, данная облачная среда визуальной разработки мобильных приложений оказалась очень удобным и понятным инструментом для реализации различных проектов.



Дизайн приложения на реальном устройстве.  
Главный экран.

Дизайн приложения на реальном устройстве.  
Второй экран.

Рис. 4. Мобильное приложение «CLC» в среде разработки MIT AppInventor



Рис. 5. Блочное программирование приложения «CLC» в среде разработки MIT AppInventor

В результате разработки приложения “CLC” в IDE Thinkable X были изучены возможности интерфейса пользователя в данной среде, доступные для настройки свойства компонентов интерфейса (рис. 6).

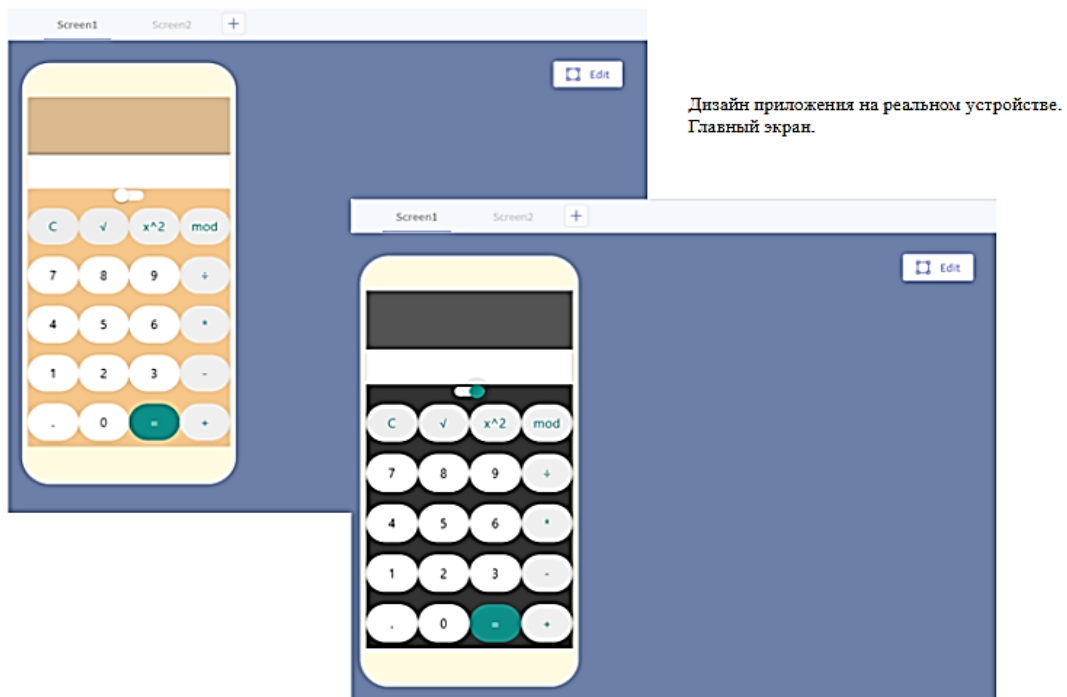


Рис. 6. Мобильное приложение “CLC” в среде разработки IDE Thinkable X

Также была проведена работа с блочным программированием, разобрано применение обработчиков событий, использование свойств таких компонентов интерфейса как “Text\_Input”, “Label”, “Button”, “Switch”, “Alert”, а также различных операторов и математических действий (рис. 7).

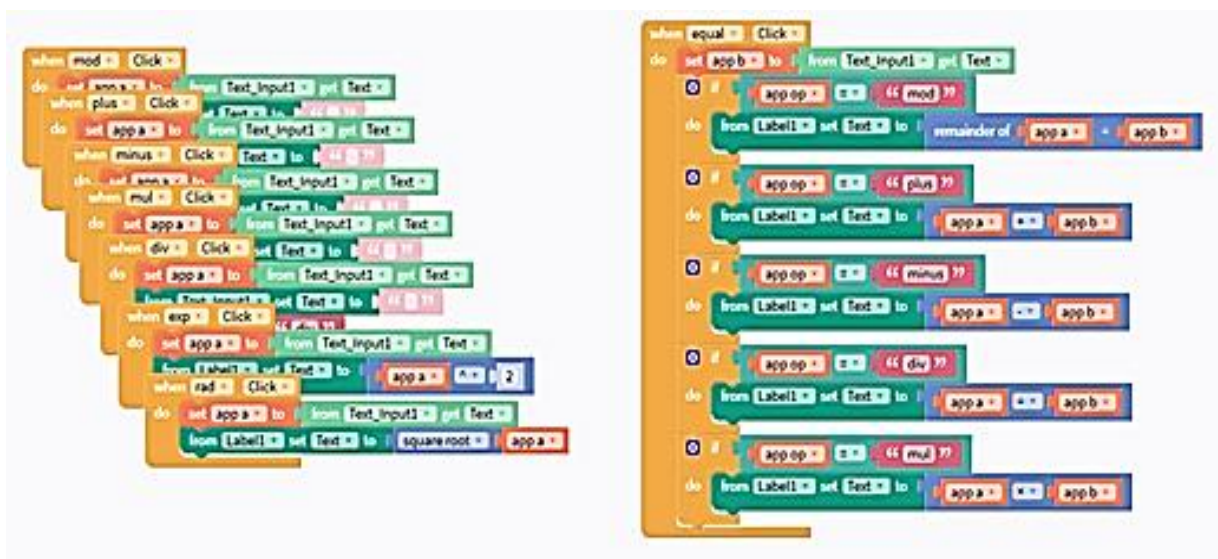


Рис. 7. Блочное программирование приложения “CLC” в среде разработки IDE Thinkable X

Отличаясь от MIT AppInventor современным интерфейсом и необходимыми дополнительными функциями, IDE Thinkable X все же имеет недоработки, которых достаточно, чтобы отдать предпочтение первой среде.

Таким образом, разработка данного приложения показала положительные и отрицательные стороны рассматриваемых сред разработки мобильных приложений и дала возможность сделать вывод, что наиболее удобной и понятной средой разработки является MIT AppInventor.

### Литература

1. Дочкин А.С. Преимущества Android Studio перед другими средами при разработке мобильных приложений // Новая наука: стратегии и векторы развития. 2016. №9. С. 106-107.
2. Дудко А.В., Пищугин М.В. Обзор и сравнительный анализ современных сред разработки мобильных приложений для операционной системы Android // Молодежная наука в XXI веке: традиции, инновации, векторы развития. 2019. С. 94-96.
3. Патюченко Ф.В., Слащев И.С., Клименко А.В., Трегубенко Л.А. App Inventor 2 для создания Android приложений // Modern Science. 2019. №7-2. С. 303-307.

© Болхудере Е.И., Пономарева Ю.С., 2021

УДК 517.962.24

Горшков П.В., Абрамов В.В., канд. физ.-мат. наук  
Рязанский государственный университет  
им. С.А. Есенина, г. Рязань, Россия

### МАЛОЕ СТАЦИОНАРНОЕ РЕШЕНИЕ НЕЛИНЕЙНОЙ СИСТЕМЫ В КОНЕЧНЫХ РАЗНОСТЯХ

Рассмотрим систему вида:

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = A \begin{pmatrix} x_n \\ y_n \end{pmatrix} + f_k(x_n, y_n, \varepsilon) \quad (1)$$

в которой  $A$  – ненулевая  $2 \times 2$ -матрица,  $\varepsilon \in R$  – малый параметр,  $f_k(x, y, \varepsilon)$  – вектор-форма от  $x, y, \varepsilon$  порядка  $k \geq 2$ ,  $k \in N$ ,  $f_k(\alpha x, \alpha y, \alpha \varepsilon) = \alpha^k f_k(x, y, \varepsilon)$ . Так как  $f_k(0_2, \varepsilon) \equiv 0_2$ , то  $\begin{pmatrix} x_n \\ y_n \end{pmatrix} \equiv 0_2$  – стационарное решение система (1) при всех  $\varepsilon$ .

*Определение 1.*  $(x_0(\varepsilon), y_0(\varepsilon))^T$  – малое стационарное решение системы (1), если  $(x_0(\varepsilon), y_0(\varepsilon))^T \neq 0_2$  при малых  $\varepsilon \neq 0$  и  $\lim_{\varepsilon \rightarrow 0} (x_0(\varepsilon), y_0(\varepsilon))^T = 0_2$ .

*Определение 2.* Малое стационарное решение системы (1) асимптотически устойчиво, если для любого  $\mu > 0$  существует  $\delta > 0$  такое, что при любых начальных значениях  $x_0, y_0$

решения и любом значении параметра  $\varepsilon$  из условия  $\left\| \begin{pmatrix} x_0(\varepsilon) \\ y_0(\varepsilon) \end{pmatrix} - \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} \right\| < \delta$  следует, что при всех

$n \in N$  справедливо неравенство  $\left\| \begin{pmatrix} x_0(\varepsilon) \\ y_0(\varepsilon) \end{pmatrix} - \begin{pmatrix} x_n \\ y_n \end{pmatrix} \right\| < \mu$  и  $\lim_{n \rightarrow +\infty} \left\| \begin{pmatrix} x_0(\varepsilon) \\ y_0(\varepsilon) \end{pmatrix} - \begin{pmatrix} x_n \\ y_n \end{pmatrix} \right\| = 0$ .

Ставится задача о поиске условий существования и устойчивости малого стационарного решения системы (1).

Из вида системы (1) следует, что ее стационарное решение следует искать как вектор, удовлетворяющий системе вида:

$$(A - E) \begin{pmatrix} x \\ y \end{pmatrix} + f_k(x, y, \varepsilon) = 0_2 \quad (2)$$

При достаточно малых ненулевых значениях переменных справедливо следующее утверждение.

*Теорема 1.* Если  $\det(A - E) \neq 0$ , то существует такое  $\delta > 0$ , что система (1) не имеет

стационарных решений  $\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$ :  $0 < \left\| \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} \right\| < \delta$  при любых  $\varepsilon$ :  $|\varepsilon| < \delta$ .



В силу теоремы 1 справедливы следующие утверждения.

*Следствие 1.* Если система (1) имеет малое стационарное решение, то:

$$\det(A - E) = 0 \quad (3)$$

Всюду далее будем предполагать, что условие (3) выполнено.

*Следствие 2.* Если  $(x_0(\varepsilon), y_0(\varepsilon))^T$  – малое стационарное решение системы (1), то при любых малых  $\varepsilon$  справедливо условие:

$$(A - E) \begin{pmatrix} x_0(\varepsilon) \\ y_0(\varepsilon) \end{pmatrix} \equiv 0_2 \quad (4)$$

*Следствие 3.* Если  $(x_0(\varepsilon), y_0(\varepsilon))^T$  – малое стационарное решение системы (1), то при любых малых  $\varepsilon$  справедливо условие:

$$f_k(x_0(\varepsilon), y_0(\varepsilon), \varepsilon) \equiv 0_2 \quad (5)$$

Данное утверждение непосредственно вытекает из следствия 2 и равенства (2).

Итак, справедливо следующее основное утверждение.

*Теорема 2.* Вектор  $(x_0(\varepsilon), y_0(\varepsilon))^T$ :  $x_0(\varepsilon) \rightarrow 0$  и  $y_0(\varepsilon) \rightarrow 0$  при  $\varepsilon \rightarrow 0$  является малым стационарным решением системы (1) тогда и только тогда, когда для него выполняются условия (4) и (5).

Значит, малое стационарное решение системы (1) вычисляется по следующей схеме.

1. Находим стационарное решение линейной системы  $\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = A \begin{pmatrix} x_n \\ y_n \end{pmatrix}$ , соответствующей системе (1). Такое стационарное решение удовлетворяет условию (4). При этом очевидно, что стационарные решения данной системы заполняет некоторую прямую  $M$ . В силу теоремы 2 малые стационарные решения системы (1), если они существуют, также лежат на  $M$ .

2. Находим  $\phi_0$  – угол, образованный прямой  $M$  и осью  $Ox$ .

3. Пусть в системе (1):  $f_k(x, y, \varepsilon) = \overline{f}_k(x, y) + \overline{\overline{f}}_k(x, y, \varepsilon)$ , где  $\overline{\overline{f}}_k(x, y, 0) \equiv 0_2$ .

4. В системе (2) выполним переход к полярным координатам при  $\phi = \phi_0$  или при  $\phi = \phi_0 + \pi$ :

$$\begin{cases} x = \rho \cos \phi_0, \\ y = \rho \sin \phi_0 \end{cases} \quad \text{или} \quad \begin{cases} x = -\rho \cos \phi_0, \\ y = -\rho \sin \phi_0. \end{cases} \quad (6)$$

При условии (6) в силу следствия 2. получим систему:

$$\rho^k \overline{f}_k(\cos \phi_0, \sin \phi_0) + \overline{\overline{f}}_k(\rho \cos \phi_0, \rho \sin \phi_0, \varepsilon) = 0_2 \quad (7)$$

5. Находим из системы (7) функцию  $\rho = \rho(\varepsilon) \neq 0$ ,  $\lim_{\varepsilon \rightarrow 0} \rho(\varepsilon) = 0$ , подставим ее в равенство (6) и получим малое стационарное решение системы (1).

Рассмотрим случай, когда в системе (1):

$$f_k(x, y, \varepsilon) = f_3(x, y, \varepsilon) = \begin{pmatrix} a_1 \varepsilon^2 x + a_2 \varepsilon^2 y + a_3 \varepsilon x^2 + a_4 \varepsilon xy + a_5 \varepsilon y^2 + a_6 x^3 + a_7 x^2 y + a_8 xy^2 + a_9 y^3 \\ b_1 \varepsilon^2 x + b_2 \varepsilon^2 y + b_3 \varepsilon x^2 + b_4 \varepsilon xy + b_5 \varepsilon y^2 + b_6 x^3 + b_7 x^2 y + b_8 xy^2 + b_9 y^3 \end{pmatrix} \quad (8)$$

При условии (6) система (8) примет вид:

$$\begin{cases} \varepsilon^2 g_1(\phi_0) + \varepsilon \rho g_2(\phi_0) + \rho^2 g_3(\phi_0) = 0, \\ \varepsilon^2 g_4(\phi_0) + \varepsilon \rho g_5(\phi_0) + \rho^2 g_6(\phi_0) = 0 \end{cases} \quad (9)$$

или:

$$\begin{cases} -\varepsilon^2 g_1(\phi_0) + \varepsilon \rho g_2(\phi_0) - \rho^2 g_3(\phi_0) = 0, \\ -\varepsilon^2 g_4(\phi_0) + \varepsilon \rho g_5(\phi_0) - \rho^2 g_6(\phi_0) = 0. \end{cases} \quad (10)$$

где  $g_1(\phi_0) = a_1 \cos \phi_0 + a_2 \sin \phi_0$ ,  $g_2(\phi_0) = a_3 \cos^2 \phi_0 + a_4 \cos \phi_0 \sin \phi_0 + a_5 \sin^2 \phi_0$ ,  $g_3(\phi_0) = a_6 \cos^3 \phi_0 + a_7 \cos^2 \phi_0 \sin \phi_0 + a_8 \cos \phi_0 \sin^2 \phi_0 + a_9 \sin^3 \phi_0$ ,  $g_4(\phi_0) = b_1 \cos \phi_0 + b_2 \sin \phi_0$ ,  $g_5(\phi_0) = b_3 \cos^2 \phi_0 + b_4 \cos \phi_0 \sin \phi_0 + b_5 \sin^2 \phi_0$ ,  $g_6(\phi_0) = b_6 \cos^3 \phi_0 + b_7 \cos^2 \phi_0 \sin \phi_0 + b_8 \cos \phi_0 \sin^2 \phi_0 + b_9 \sin^3 \phi_0$ .

*Лемма.* Пусть выполнено условие (9). Тогда, если  $g_i(\phi_0) = 0$  и  $F_i(\phi_0) = 0$ ,  $i = \overline{1, 6}$ , то целый луч  $L$ , заданный условием (6) заполнен стационарными решениями системы (1).

Очевидно, ни одно из стационарных решений на  $L$  не является малым по определению 1. Поэтому, рассмотрев частные случаи нарушения условий леммы 1, получим следующие утверждения.

*Теорема 3.* Если существует  $\phi_0$ , для которого выполняются условия  $g_3(\phi_0) = 0$ ,

$g_6(\phi_0) \neq 0$  и  $\frac{g_1(\phi_0)}{g_2(\phi_0)} > 0$   $\left( \frac{g_1(\phi_0)}{g_2(\phi_0)} < 0 \right)$ , то система (1) имеет малое стационарное решение тогда

и только тогда, когда  $\frac{g_6(\phi_0) g_1^2(\phi_0)}{g_2^2(\phi_0)} - \frac{g_5(\phi_0) g_1(\phi_0)}{g_2(\phi_0)} + g_4(\phi_0) = 0$ .

*Теорема 4.* Если существует  $\phi_0$ , при котором  $g_3(\phi_0) \neq 0$ ,  $g_6(\phi_0) = 0$  и  $\frac{g_4(\phi_0)}{g_5(\phi_0)} > 0$

$\left( \frac{g_4(\phi_0)}{g_5(\phi_0)} < 0 \right)$ , то система (1) имеет малое стационарное решение тогда и только тогда, когда

$\frac{g_3(\phi_0) g_4^2(\phi_0)}{g_5^2(\phi_0)} - \frac{g_2(\phi_0) g_4(\phi_0)}{g_5(\phi_0)} + g_1(\phi_0) = 0$ .

**Теорема 5.** Пусть существует  $\phi_0$ , для которого выполняются условия  $g_3(\phi_0) \neq 0$ ,  $g_6(\phi_0) \neq 0$ ,  $g_2^2(\phi_0) - 4g_1(\phi_0)g_3(\phi_0) \geq 0$  и  $g_5^2(\phi_0) - 4g_4(\phi_0)g_6(\phi_0) \geq 0$ . При этом система (1) имеет малое стационарное решение тогда и только тогда, когда

$$\begin{cases} \frac{-g_2(\phi_0) + \sqrt{g_2^2(\phi_0) - 4g_1(\phi_0)g_3(\phi_0)}}{g_3(\phi_0)} = \frac{-g_5(\phi_0) + \sqrt{g_5^2(\phi_0) - 4g_4(\phi_0)g_6(\phi_0)}}{g_6(\phi_0)}, \\ \frac{-g_2(\phi_0) + \sqrt{g_2^2(\phi_0) - 4g_1(\phi_0)g_3(\phi_0)}}{g_3(\phi_0)} = \frac{-g_5(\phi_0) - \sqrt{g_5^2(\phi_0) - 4g_4(\phi_0)g_6(\phi_0)}}{g_6(\phi_0)}, \\ \frac{-g_2(\phi_0) - \sqrt{g_2^2(\phi_0) - 4g_1(\phi_0)g_3(\phi_0)}}{g_3(\phi_0)} = \frac{-g_5(\phi_0) + \sqrt{g_5^2(\phi_0) - 4g_4(\phi_0)g_6(\phi_0)}}{g_6(\phi_0)}, \\ \frac{-g_2(\phi_0) - \sqrt{g_2^2(\phi_0) - 4g_1(\phi_0)g_3(\phi_0)}}{g_3(\phi_0)} = \frac{-g_5(\phi_0) - \sqrt{g_5^2(\phi_0) - 4g_4(\phi_0)g_6(\phi_0)}}{g_6(\phi_0)}. \end{cases}$$

Исследуем устойчивость малого стационарного решения системы (1).

В системе (1) выполним замену  $\begin{cases} x = u + x_0(\varepsilon), \\ y = v + y_0(\varepsilon), \end{cases}$  получим:

$$\begin{pmatrix} u_{n+1} \\ v_{n+1} \end{pmatrix} + \begin{pmatrix} x_0(\varepsilon) \\ y_0(\varepsilon) \end{pmatrix} = A \left( \begin{pmatrix} u_n \\ v_n \end{pmatrix} + \begin{pmatrix} x_0(\varepsilon) \\ y_0(\varepsilon) \end{pmatrix} \right) + f_k(u_n + x_0(\varepsilon), v_n + y_0(\varepsilon), \varepsilon) \quad (11)$$

Так как  $(x_0(\varepsilon), y_0(\varepsilon))^T$  – стационарное решение системы (1), то при всех малых  $\varepsilon$

справедливо равенство  $\begin{pmatrix} x_0(\varepsilon) \\ y_0(\varepsilon) \end{pmatrix} = A \begin{pmatrix} x_0(\varepsilon) \\ y_0(\varepsilon) \end{pmatrix} + f_k(x_0(\varepsilon), y_0(\varepsilon), \varepsilon)$ . Поэтому линейное

приближение системы (11) имеет вид  $\begin{pmatrix} u_{n+1} \\ v_{n+1} \end{pmatrix} = \left( A + \begin{pmatrix} f'_x(x_0(\varepsilon), y_0(\varepsilon), \varepsilon) \\ f'_y(x_0(\varepsilon), y_0(\varepsilon), \varepsilon) \end{pmatrix} \right) \begin{pmatrix} u_n \\ v_n \end{pmatrix}$ .

По теореме об устойчивости нулевого решения разностной системы по линейному приближению [1] справедливо следующее утверждение.

**Теорема 6.** Если при всех достаточно малых  $\varepsilon$  собственные значения матрицы  $\left( A + \begin{pmatrix} f'_x(x_0(\varepsilon), y_0(\varepsilon), \varepsilon) \\ f'_y(x_0(\varepsilon), y_0(\varepsilon), \varepsilon) \end{pmatrix} \right)$  входят в единичный круг комплексной плоскости (хотя бы одно не входит), то малое стационарное решение  $(x_0(\varepsilon), y_0(\varepsilon))^T$  системы (1) асимптотически устойчиво (неустойчиво).

**Пример.** Пусть дана система вида:

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} 3 & -2 \\ 2 & -1 \end{pmatrix} \begin{pmatrix} x_n \\ y_n \end{pmatrix} + \begin{pmatrix} 2\varepsilon^2 x - 3\varepsilon^2 y + 7\varepsilon x^2 - 9\varepsilon xy + 5\varepsilon y^2 + 5x^3 + x^2 y + 2xy^2 - 4y^3 \\ -4\varepsilon^2 x + 2\varepsilon^2 y + \varepsilon x^2 - 2\varepsilon xy + 5\varepsilon y^2 + 12x^3 + x^2 y - 3xy^2 + 6y^3 \end{pmatrix}. \quad (12)$$

Согласно следствию 2 малое стационарное решение следует искать во множестве стационарных решений  $x - y = 0$  соответствующей линейной однородной системы. Следуя изложенной выше схеме отыскания малых стационарных решений, находим  $\phi_0 = \pi / 4$ .

Нелинейную часть исследуемой системы представим в виде

$$f_3(x, y, \varepsilon) = \begin{pmatrix} 2\varepsilon^2 x - 3\varepsilon^2 y + 7\varepsilon x^2 - 9\varepsilon xy + 5\varepsilon y^2 + 5x^3 + x^2 y + 2xy^2 - 4y^3 \\ -4\varepsilon^2 x + 2\varepsilon^2 y + \varepsilon x^2 - 2\varepsilon xy + 5\varepsilon y^2 + 12x^3 + x^2 y - 3xy^2 + 6y^3 \end{pmatrix} = \\ = \begin{pmatrix} 2\varepsilon^2 x - 3\varepsilon^2 y \\ -4\varepsilon^2 x + 2\varepsilon^2 y \end{pmatrix} + \begin{pmatrix} 7\varepsilon x^2 - 9\varepsilon xy + 5\varepsilon y^2 \\ \varepsilon x^2 - 2\varepsilon xy + 5\varepsilon y^2 \end{pmatrix} + \begin{pmatrix} 5x^3 + x^2 y + 2xy^2 - 4y^3 \\ 12x^3 + x^2 y - 3xy^2 + 6y^3 \end{pmatrix}.$$

Переходим к полярным координатам при  $\phi = \phi_0$  и при  $\phi = \phi_0 + \pi$ :  $x = \pm\sqrt{2}\rho/2$ ,  $y = \pm\sqrt{2}\rho/2$ . Получим систему вида (9): 
$$\begin{cases} \pm\sqrt{2}\rho^2 \pm 3\varepsilon\rho/2 \mp \sqrt{2}\varepsilon^2/2 = 0, \\ \pm 4\sqrt{2}\rho^2 + 2\varepsilon\rho \mp \sqrt{2}\varepsilon^2 = 0. \end{cases}$$
 Эта система имеет решение  $\rho = \mp\sqrt{2}\varepsilon/4$ . Тогда система (12) имеет малое стационарное решение:  $x_0(\varepsilon) = \varepsilon/4$ ;  $y_0(\varepsilon) = \varepsilon/4$ , которое движется вдоль прямой  $x - y = 0$  при изменении  $\varepsilon$ .

Определим характер устойчивости найденного малого стационарного решения.

Составим систему вида (13): 
$$\begin{pmatrix} u_{n+1} \\ v_{n+1} \end{pmatrix} = \begin{pmatrix} 3 + 71\varepsilon^2/16 & -2 - 51\varepsilon^2/16 \\ 2 - 29\varepsilon^2/16 & -1 + 77\varepsilon^2/16 \end{pmatrix} \begin{pmatrix} u_n \\ v_n \end{pmatrix}.$$
 Характеристическое

уравнение 
$$\begin{vmatrix} 3 + 71\varepsilon^2/16 - \lambda & -2 - 51\varepsilon^2/16 \\ 2 - 29\varepsilon^2/16 & -1 + 77\varepsilon^2/16 - \lambda \end{vmatrix} = \lambda^2 - (2 + 37\varepsilon^2/4)\lambda + 1 + 51\varepsilon^2/4 + 997\varepsilon^4/64 = 0$$

этой системы имеет корни  $\lambda_{1,2} = \frac{1}{2} \left( 2 + \frac{37}{4}\varepsilon^2 \pm \sqrt{-14\varepsilon^2 + \frac{93}{4}\varepsilon^4} \right)$ . Оценив модули собственных значений, получим  $|\lambda_{1,2}| > 1$  при любом  $\varepsilon \neq 0$ . Таким образом, по теореме 6 малое стационарное решение системы (13) неустойчиво.

Предложенная процедура исследования малого стационарного решения эффективна с точки зрения реализации в прикладных пакетах компьютерных программ.

### Литература

1. Халанай А., Векслер Д. Качественная теория импульсных систем. М., 1971. 309 с.

© Горшков П.В., Абрамов В.В., 2021

УДК 517.962.24

**Горшков П.В.***Научный руководитель: Абрамов В.В., канд. физ.-мат. наук  
Рязанский государственный университет им. С.А. Есенина,  
г. Рязань, Россия***УСТОЙЧИВОСТЬ ЛИНЕЙНОЙ СИСТЕМЫ В КОНЕЧНЫХ РАЗНОСТЯХ**

Пусть дана линейная система разностных уравнений вида:

$$x_{n+1} = Ax_n + f_n \quad (1)$$

в котором  $A$  – постоянная матрица размерности  $m \times m$ , а  $f_n$  – вектор-функция от  $n$  размерности  $m$ . Определим условие асимптотической устойчивости произвольно выбранного решения системы (1).

*Определение 1.* Решение  $\bar{x}_n$  асимптотически устойчиво, если выполняются условия:

1) для любого  $\varepsilon > 0$  существует такое  $\delta(\varepsilon) > 0$ , что для любого решения  $x_n$  и для любого  $n \geq n_0$  из неравенства  $\|x_{n_0} - \bar{x}_{n_0}\| < \delta(\varepsilon)$  следует оценка  $\|x_n - \bar{x}_n\| < \varepsilon$  (то есть решение  $\bar{x}_n$  является устойчивым);

2)  $\lim_{n \rightarrow +\infty} \|x_n - \bar{x}_n\| = 0$  (то есть решение  $\bar{x}_n$  является притягивающим).

Для системы (1) соответствующая однородная система имеет вид:

$$x_{n+1} = Ax_n \quad (2)$$

Величина  $x_n - \bar{x}_n$  (возмущение решения  $\bar{x}_n$ ) из определения 1 удовлетворяет системе (2). Действительно, так как  $x_n = Ax_{n-1} + f_{n-1}$  и  $\bar{x}_n = A\bar{x}_{n-1} + f_{n-1}$  для любого  $n$  в силу (1), то вектор-функция  $y_n = x_n - \bar{x}_n$  для любого  $n$  удовлетворяет условию  $y_n = Ax_{n-1} + f_{n-1} - A\bar{x}_{n-1} - f_{n-1} = A(x_{n-1} - \bar{x}_{n-1}) = Ay_{n-1}$ , то есть является решением системы (2). Итак, исследование устойчивости любого решения системы (1) сводится к исследованию устойчивости нулевого решения системы (2).

*Определение 2.* Решение  $x_n \equiv 0_m$  системы вида (2) асимптотически устойчиво, если: 1) для любого  $\varepsilon > 0$  существует такое  $\delta(\varepsilon) > 0$ , что для любого решения  $x_n$  и для любого  $n \geq n_0$  из неравенства  $\|x_{n_0}\| < \delta(\varepsilon)$  следует оценка  $\|x_n\| < \varepsilon$ ; 2)  $\lim_{n \rightarrow +\infty} \|x_n\| = 0$ .

В случае асимптотической устойчивости нулевого решения системы (2) систему (1) естественно называть асимптотически устойчивой.

*Лемма 1* [1]. Система (2) асимптотически устойчива тогда и только тогда, когда выполняется условие:



$$|\lambda_i| < 1, i = \overline{1, m}, \quad (3)$$

в котором  $\lambda_i$  – собственные значения матрицы  $A$  (то есть спектр матрицы  $A$  лежит внутри единичного круга комплексной плоскости).

В общем случае задача о вычислении всех корней многочлена с целью их последующей оценки является нетривиальной с вычислительной точки зрения. Поэтому целесообразно организовать процедуру проверки условий (3) без непосредственного решения характеристического уравнения. Для оценки собственных значений матрицы  $A$  построим вспомогательный многочлен, для которого условие (3) можно проверить с помощью критерия Рауса – Гурвица [2]. Такой прием целесообразен для организации компьютерных расчетов. Кроме того, такой прием позволяет реализовать в пакетах прикладных программ эффективную проверку робастной устойчивости при наличии интервальных ограничений на коэффициенты системы (2).

Характеристическое уравнение системы (2) имеет вид:

$$f(\lambda) = \lambda^m + b_{m-1}\lambda^{m-1} + b_{m-2}\lambda^{m-2} + \dots + b_1\lambda + b_0 = 0, \quad (4)$$

в котором  $b_r$  – сумма всех главных миноров порядка  $r$  матрицы  $A$ , умноженных на  $(-1)^r$ , в частности,  $b_{m-1} = a_{11} + a_{22} + \dots + a_{mm} = \text{tr}A$  – след, а  $b_0 = \det A$  – определитель матрицы  $A$ . Многочлен вида (4) с условием (3) назовем устойчивым.

Выполним замену переменной в уравнении (4):

$$\lambda = \frac{d+1}{d-1} \quad (5)$$

Из уравнения (5) можно выразить  $d = \frac{1-\lambda}{1+\lambda}$  при  $\lambda \neq -1$ . Очевидно, это ограничение необходимо для условия (4). Определим геометрический смысл отображения (5).

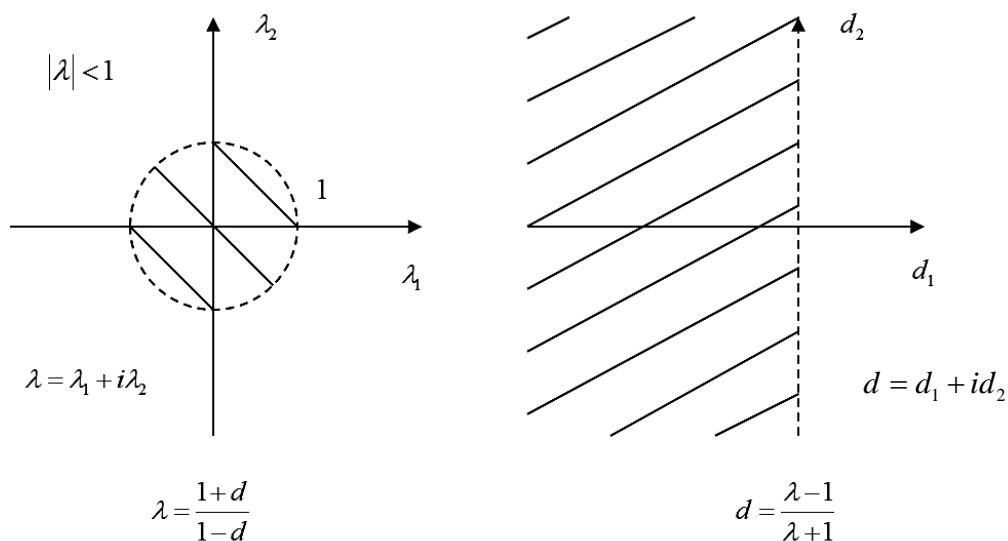


Рис. 1. Биекция, заданная равенством (5)

*Лемма 2.* Отображение (5) – это биекция внутренности круга единичного радиуса с центром в начале координат комплексного переменного  $\lambda$  в левую полуплоскость комплексного переменного  $d$ .

*Доказательство.* Допустим,  $\lambda : |\lambda| < 1$  или, что то же самое,  $|1+d| < |1-d|$  в силу (5), где  $d = d_1 + id_2$ . Неравенство  $|1+d| < |1-d|$  равносильно следующим оценкам  $(1+d_1)^2 + d_2^2 < (1-d_1)^2 + d_2^2 \Leftrightarrow |1+d_1| < |1-d_1|$ . Решив методом интервалов получим:

$$\begin{cases} d_1 < -1, \\ -1-d_1 < 1-d_1; \\ -1 \leq d_1 < 1, \\ 1+d_1 < 1-d_1; \end{cases} \Leftrightarrow \begin{cases} d_1 < -1, \\ -1 \leq d_1 < 0, \Leftrightarrow d_1 < 0 \Leftrightarrow \operatorname{Re} d < 0. \\ d_1 \in \emptyset; \end{cases}$$

$$\begin{cases} d_1 \geq 1, \\ 1+d_1 < -1+d_1; \end{cases}$$

Допустим, для переменной  $d$ :  $\operatorname{Re} d < 0$  или, что то же самое,  $\operatorname{Re} \left( \frac{\lambda-1}{\lambda+1} \right) < 0$  в силу условия (5), где  $\lambda = \lambda_1 + i\lambda_2$ . Вычислим

$$\frac{\lambda-1}{\lambda+1} = \frac{(\lambda-1) \cdot \overline{(\lambda+1)}}{(\lambda+1) \cdot \overline{(\lambda+1)}} = \frac{(\lambda-1) \cdot (\bar{\lambda}+1)}{(\lambda+1) \cdot (\bar{\lambda}+1)} = \frac{\lambda \cdot \bar{\lambda} + \lambda - \bar{\lambda} - 1}{\lambda \cdot \bar{\lambda} + \lambda + \bar{\lambda} + 1} = \frac{|\lambda|^2 + 2\lambda_2 i - 1}{|\lambda|^2 + 2\lambda_1 + 1}$$

и, выделив действительную часть, получим:  $\operatorname{Re} \left( \frac{\lambda-1}{\lambda+1} \right) = \frac{|\lambda|^2 - 1}{|\lambda|^2 + 2\lambda_1 + 1} < 0$ . Так как

$|\lambda|^2 + 2\lambda_1 + 1 > 0$ , то  $|\lambda|^2 - 1 < 0 \Leftrightarrow |\lambda| < 1$ . Лемма 2 доказана.

Иллюстрация к лемме 2 приведена на рисунке 1.

В результате замены (5) уравнение (4) примет вид

$$(d+1)^m + b_{m-1}(d+1)^{m-1}(d-1) + \dots + b_1(d+1)(d-1)^{m-1} + b_0(d-1)^m = 0$$

или:

$$g(d) = c_m d^m + c_{m-1} d^{m-1} + \dots + c_1 d + c_0 = 0 \tag{6}$$

в частности  $c_m = 1 + \sum_{s=1}^m (-1)^s b_{m-s}$ .

Многочлен вида  $g(d)$  называется гурцевым, если положительны все главные миноры

соответствующей ему матрицы Гурвица  $G = \begin{pmatrix} c_{m-1} & c_{m-3} & c_{m-5} & c_{m-7} & \dots & 0 \\ c_m & c_{m-2} & c_{m-4} & c_{m-6} & \dots & 0 \\ 0 & c_{m-1} & c_{m-3} & c_{m-5} & \dots & 0 \\ 0 & c_m & c_{m-4} & c_{m-4} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & c_0 \end{pmatrix} [2].$

В силу леммы 2 справедливо следующее утверждение.

*Лемма 3.* Многочлен (4) устойчив тогда и только тогда, когда многочлен  $g(d)$  в левой части равенства (6) гурвицев и  $c_m \neq 0$ .

Итак, решение задачи об асимптотической устойчивости системы (1) сводится к проверке критерия Рауса – Гурвица для вспомогательного многочлена из уравнения (6).

*Пример.* Рассмотрим систему вида (1):

$$x_{n+1} = \begin{pmatrix} 1/2 & 0 \\ 0 & 1/4 \end{pmatrix} x_n + \begin{pmatrix} 10 \sin(n) \\ \cos(3n) \end{pmatrix} \quad (7)$$

Проверим устойчивость системы (7).

Составим характеристическое уравнение  $\lambda^2 - \frac{3}{4}\lambda + \frac{1}{8} = 0$  для соответствующей системы вида (2). Это уравнение имеет корни  $\lambda_1 = 1/2$  и  $\lambda_2 = 1/4$ . Очевидно, выполняется условие (3). Значит, по лемме 1 система (7) асимптотически устойчива.

Теперь покажем, что такой же вывод получается по лемме 3. Выполним в характеристическом уравнении подстановку (5) и получим новое уравнение

$$\left(\frac{d+1}{d-1}\right)^2 - \frac{3}{4}\left(\frac{d+1}{d-1}\right) + \frac{1}{8} = 0 \text{ или } \frac{15}{8}d^2 + \frac{7}{4}d + \frac{3}{8} = 0 \text{ (уравнение вида (6)).}$$

Для многочлена в левой части составим матрицу Гурвица  $G = \begin{pmatrix} 7/4 & 0 \\ 15/8 & 3/8 \end{pmatrix}$ . Она имеет главные миноры

$\Delta_1 = 7/4 > 0$ ,  $\Delta_2 = \det G = 21/32 > 0$ . Итак, по лемме 3 система уравнений (7) асимптотически устойчива.

Для графической иллюстрации вывода об устойчивости изобразим с помощью пакета Maple в виде ломаных траектории решений системы (7) (рис. 2). При этом видим поэтапное уменьшение взаимных возмущений и притяжение траекторий друг к другу.

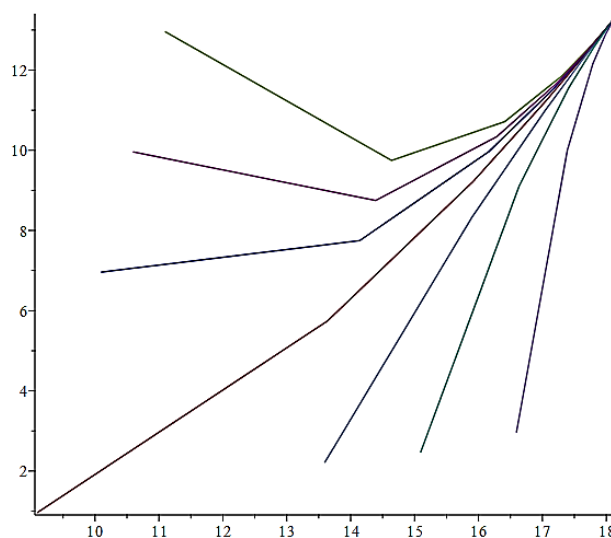


Рис. 2. Траектории решений системы (7)

Пример 2. Допустим,  $A = \begin{pmatrix} 1/4 & 1/9 & 1/3 \\ 1/8 & 1/2 & 1/3 \\ 1/2 & 1/4 & 1/3 \end{pmatrix}$  в системе вида (2).

Иследуем систему на устойчивость. Для матрицы  $A$  составим характеристическое

уравнение  $|\lambda E - A| = \begin{vmatrix} \lambda - 1/4 & -1/9 & -1/3 \\ -1/8 & \lambda - 1/2 & -1/3 \\ -1/2 & -1/4 & \lambda - 1/3 \end{vmatrix} = \lambda^3 - \frac{13}{12}\lambda^2 + \frac{1}{9}\lambda + \frac{11}{288} = 0$ . Выполним замену

(5) и получим уравнение вида (6)

$$\left(\frac{d+1}{d-1}\right)^3 - \frac{13}{12}\left(\frac{d+1}{d-1}\right)^2 + \frac{1}{9}\left(\frac{d+1}{d-1}\right) + \frac{11}{288} = 0 \Leftrightarrow g(d) = \frac{19}{288}d^3 + \frac{487}{288}d^2 + \frac{1177}{288}d + \frac{69}{32} = 0.$$

Вычислим для многочлена  $g(d)$  главные миноры соответствующей матрицы Гурвица

$$G = \begin{pmatrix} 487/288 & 69/32 & 0 \\ 19/288 & 1177/288 & 0 \\ 0 & 487/288 & 69/32 \end{pmatrix}; \quad \Delta_1 = \frac{487}{288} > 0, \quad \Delta_2 = \begin{vmatrix} 487/288 & 69/32 \\ 19/288 & 1177/288 \end{vmatrix} = \frac{70175}{10368} > 0,$$

$\Delta_3 = \det G = \frac{1614025}{110592} > 0$ . Итак, по леммам 3 и 1 рассматриваемая система асимптотически устойчива.

### Литература

1. Халанай А., Векслер Д. Качественная теория импульсных систем. М., 1971. 309 с.
2. Ланкастер П. Теория матриц. М., 1973. 280 с.

© Горшков П.В., Абрамов В.В., 2021

УДК 517.925

Земенков А.В., Абрамов В.В., канд. физ.-мат. наук  
Рязанский государственный университет  
им. С.А. Есенина г. Рязань, Россия

## УСЛОВИЯ СУЩЕСТВОВАНИЯ ПЕРИОДИЧЕСКОГО РЕШЕНИЯ СИСТЕМЫ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

Проблема существования периодического решения системы дифференциальных уравнений актуальна в качественной теории, так как в критических случаях для нелинейных систем не установлены необходимые и достаточные условия ее разрешимости. Кроме того, эта проблема имеет важное прикладное значение при изучении моделей осцилляторов. В частности, вопрос о периодическом решении неавтономной системы дифференциальных уравнений с малым параметром возникает при изучении «мягкого» режима возбуждения колебаний при наличии внешнего воздействия.

Рассмотрим систему  $n$  дифференциальных уравнений с  $\omega$ -периодической непрерывной по  $t$  правой частью:

$$\dot{x} = A(t)x + f(t, \varepsilon) + f_k(t, x, \varepsilon) + g(t, x, \varepsilon) \quad (1)$$

в которой  $A(t)$  –  $n \times n$ -матрица,  $\varepsilon \in R^m$  – малый параметр,  $f(t, \varepsilon)$  достаточно гладкая по  $\varepsilon$ ,  $f(t, 0_m) \equiv 0_n$ ,  $f(t, x, \varepsilon)$  – однородная вектор-функция от  $x$  и  $\varepsilon$ , то есть  $f_k(t, \alpha x, \alpha \varepsilon) = \alpha^k f_k(t, x, \varepsilon)$  при всех  $\alpha > 0$ ,  $\lim_{\alpha \rightarrow 0} \alpha^{-k} \|g(t, x, \varepsilon)\| \equiv 0$  (в смысле равномерной сходимости). Очевидно  $x = 0_n$  – решение системы (3.1.1) при  $\varepsilon = 0_m$ .

*Определение 1.* Решение  $x(t, x_0, \varepsilon)$ ,  $x(0, x_0, \varepsilon) = x_0$  системы (1) является малым, если  $\lim_{\varepsilon \rightarrow 0_m} \|x(t, x_0, \varepsilon)\| \equiv 0$ .

*Задача 1:* найти условия существования малого периодического решения системы (1) с периодом вида  $s\omega$ ,  $s \in N$ .

При этом целесообразно использовать первые приближения правой части системы (1). Для решения задачи 1 составим систему недифференциальных уравнений для отыскания значений  $x_0$ ,  $\varepsilon$ ,  $s$ , определяющих периодическое решение, на основе условия [1]:

$$x(s\omega, x_0, \varepsilon) = x_0. \quad (2)$$

Для проверки условия (2) найдем формулу для решения. Пусть  $x(t, x_0, \varepsilon)$  – известное решение системы (1). Составим линейную систему вида:

$$\dot{y} = A(t)y + f_1(t) + f_2(t), \quad (3)$$



в которой  $f_1(t) = f(t, \varepsilon)$ ,  $f_2(t) = (f_k(t, x(t, x_0, \varepsilon), \varepsilon) + g(t, x(t, x_0, \varepsilon), \varepsilon))$ . В силу единственности решения с заданным начальным значением справедливо следующее утверждение.

**Лемма 1.**  $x(t, x_0, \varepsilon)$  является решением системы (1) тогда и только тогда, когда  $y = x(t, x_0, \varepsilon)$  является решением системы (3).

В силу леммы 1 по методу Лагранжа и по принципу суперпозиции найдем вид решения системы (3).

Пусть  $X(t)$  – фундаментальная матрица системы  $\dot{x} = A(t)x$ ,  $X(0) = E$ ,  $V(t, \varepsilon) = X(t) \int_0^t X^{-1}(\tau) f(\tau, \varepsilon) d\tau$ ,  $z(t, x_0, \varepsilon) = X(t) \int_0^t X^{-1}(\tau) (f_k(\tau, x(\tau, x_0, \varepsilon), \varepsilon) + g(\tau, x(\tau, x_0, \varepsilon), \varepsilon)) d\tau$ .

Тогда решение системы (1) имеет вид  $x(t, x_0, \varepsilon) = X(t)x_0 + V(t, \varepsilon) + z(t, x_0, \varepsilon)$ . По формуле

Тейлора получим разложение  $U(s, \varepsilon) = X(s\omega) \int_0^{s\omega} X^{-1}(t) f(t, \varepsilon) dt = U_r(s, \varepsilon) + \tilde{U}(s, \varepsilon)$ , в котором

$U_r(s, \alpha\varepsilon) = \alpha^r U_r(s, \varepsilon)$ ,  $\lim_{\varepsilon \rightarrow 0_m} \|\varepsilon\|^{-1} \|\tilde{U}(s, \varepsilon)\| = 0$ . С учетом вида решения системы (1) условие (2)

можно представить в следующей форме:

$$F(s, x_0, \varepsilon) = B(s)x_0 + U_r(s, \varepsilon) + p_k(s, x_0, \varepsilon) + \varphi(s, \varepsilon) + \psi(s, x_0, \varepsilon) = 0_n, \quad (4)$$

где  $B(s) = X(s\omega) - E$ ;  $p_k(s, x_0, \varepsilon) = X(s\omega) \int_0^{s\omega} X^{-1}(\tau) f_k(\tau, X(\tau)x_0, \varepsilon) d\tau$  – однородная вектор-

функция порядка  $k$  от  $x_0$  и  $\varepsilon$ , то есть  $p_k(s, \alpha x_0, \alpha\varepsilon) = \alpha^k p_k(s, x_0, \varepsilon)$ ,  $p_k(s, 0_n, \varepsilon) \equiv 0_n$ ; порядок

$\varphi(s, \varepsilon)$  по  $\varepsilon$  выше  $r$ , то есть  $\lim_{\varepsilon \rightarrow 0_m} \|\varepsilon\|^{-r} \|\varphi(s, \varepsilon)\| = 0$ ; порядок  $\psi(s, x_0, \varepsilon)$  по  $x_0, \varepsilon$  выше  $k$ , то

есть  $\lim_{\alpha \rightarrow 0} \alpha^{-k} \|\psi(s, \alpha x_0, \alpha\varepsilon)\| \equiv 0$ ,  $\psi(s, 0_n, \varepsilon) \equiv 0_n$ . Теперь возникает новая задача.

**Задача 2:** Найти такие значения  $x_0 \in R^n$ ,  $\varepsilon \in R^m$ , удовлетворяющие системе (4), что  $x_0 \rightarrow 0$  при  $\varepsilon \rightarrow 0_m$ .

Задачу 2 будем решать при условии, что  $s$  выбрано предварительно. Применим схему решения задачи о периодическом решении из работ [2, 3]. Исследуем свойства первых приближений уравнения (4).

Первое приближение уравнения (4) по начальному значению имеет вид  $B(s)x_0 + U_r(s, \varepsilon) + \varphi(s, \varepsilon) = 0_n$ . Это условие периодичности для линейной системы  $\dot{x} = A(t)x + f(t, \varepsilon)$ , соответствующей системе (1). Вектор-функция  $p_k(s, x_0, \varepsilon)$  представляет собой усреднение на периоде вектор-функции  $f_k(t, x, \varepsilon)$  вдоль решений линейной системы  $\dot{x} = A(t)x$ , соответствующей системе (1) при нулевом значении параметра.

Подставим в (4) величины  $x_0 = \alpha(\bar{x}_0 + \lambda)$ ,  $\varepsilon = \alpha(\varepsilon_0 + \mu)$ , где  $\alpha > 0$ ,  $\bar{x}_0, \varepsilon_0$  – некоторые фиксированные значения,  $\lambda$  и  $\mu$  – произвольные переменные. Тогда:

$$F(s, x_0, \varepsilon) = \alpha(B(s)(\bar{x}_0 + \lambda) + \alpha^{r-1}U_r(s, \varepsilon_0 + \mu) + (\alpha^{k-1}p_k(s, \bar{x}_0 + \lambda, \varepsilon_0 + \mu) + \alpha^{-1}\varphi(s, \alpha(\varepsilon_0 + \mu)) + \alpha^{-1}\psi(s, \alpha(\bar{x}_0 + \lambda), \alpha(\varepsilon_0 + \mu))).$$

В результате оценки норм первых приближений функции  $F(s, x_0, \varepsilon)$  установлены следующие необходимые условия существования периодического решения.

*Теорема 1.* Пусть  $r=1$ . Если значения  $\bar{x}_0$  и  $\varepsilon_0$  удовлетворяют условию  $B(s)\bar{x}_0 + U_r(s, \varepsilon_0) \neq 0$ , то существуют такие  $\delta > 0$ ,  $\rho > 0$ , что  $F(s, x_0, \varepsilon) \neq 0_n$  при всех  $x_0 = \alpha(\bar{x}_0 + \lambda)$ ,  $\varepsilon = \alpha(\varepsilon_0 + \mu)$ , где  $\alpha < \rho$ ,  $\|\lambda\| \leq \delta$ ,  $\|\mu\| \leq \delta$ .

Из теоремы 1 следует, что для разрешимости задачи 2 при  $r=1$  необходимо, чтобы существовали  $\bar{x}_0 \neq 0_n$  и  $\varepsilon_0 \neq 0_m$ , удовлетворяющие условию:

$$B(s)\bar{x}_0 + U_r(s, \varepsilon_0) = 0_n, \quad (5)$$

при котором линейная система  $\dot{x} = A(t)x + f_r(t, \varepsilon_0)$ , соответствующая системе (1), имеет  $s\omega$ -периодическое решение.

*Теорема 2.* Пусть  $r > 1$ . Если значение  $\bar{x}_0 \neq 0_n$  удовлетворяет условию  $B(s)\bar{x}_0 \neq 0_n$ , то существуют такие  $\delta > 0$ ,  $\rho > 0$ ,  $\varepsilon_0 \neq 0_m$ , что  $F(s, x_0, \varepsilon) \neq 0_n$  при всех  $x_0 = \alpha(\bar{x}_0 + \lambda)$ ,  $\varepsilon = \alpha(\varepsilon_0 + \mu)$ , где  $\alpha < \rho$ ,  $\|\lambda\| \leq \delta$ ,  $\|\mu\| \leq \delta$ .

По теореме 2 для разрешимости задачи 2 при  $r > 1$  необходимо, чтобы существовало значение  $\bar{x}_0 \neq 0_n$ , удовлетворяющее условию:

$$B(s)x_0 = 0_n \Leftrightarrow \det B(s) = 0. \quad (6)$$

Условие (6) означает, что при нулевом значении параметра линейная система  $\dot{x} = A(t)x$ , соответствующая системе (1), имеет  $s\omega$ -периодическое решение.

*Теорема 3.* Пусть  $r = k$ . Если существуют значения  $\bar{x}_0 \neq 0_n$ ,  $\varepsilon_0 \neq 0_m$ , при которых выполняются условия: 1)  $B(s)\bar{x}_0 = 0_n$ , 2)  $U_r(s, \varepsilon_0) + p_k(s, \bar{x}_0, \varepsilon_0) \neq 0_n$ , то существуют такие  $\delta > 0$ ,  $\rho > 0$ , что  $F(s, x_0, \varepsilon) \neq 0_n$  при всех  $x_0 = \alpha(\bar{x}_0 + \lambda)$ ,  $\varepsilon = \alpha(\varepsilon_0 + \mu)$ , где  $\alpha < \rho$ ,  $\|\lambda\| \leq \delta$ ,  $\|\mu\| \leq \delta$ ,  $B(s)\lambda = 0_n$ .

В равенстве (4) сделаем замены переменных  $x_0 = \alpha(\bar{x}_0 + \lambda)$ ,  $\varepsilon = \alpha(\varepsilon_0 + \mu)$ , в которых  $\bar{x}_0 \neq 0_n$ ,  $\varepsilon_0 \neq 0_m$ , подчинены необходимому условию. Обозначим

$$q(\alpha, \lambda, \mu) = \alpha^{k-1}p_k(s, \bar{x}_0 + \lambda, \varepsilon_0 + \mu) + \alpha^{-1}\varphi(s, \alpha(\varepsilon_0 + \mu)) + \alpha^{-1}\psi(s, \alpha(\bar{x}_0 + \lambda), \alpha(\varepsilon_0 + \mu)).$$

Так как  $\lim_{\alpha \rightarrow 0} q(\alpha, \lambda, \mu) \equiv 0_n$ , то определим по непрерывности  $q(0, \lambda, \mu) \equiv 0_n$ . Уравнение (4) запишем в виде:

$$\bar{F}(\alpha, \lambda, \mu) = B(s)(\bar{x}_0 + \lambda) + U_r(s, \varepsilon_0 + \mu) + q(\alpha, \lambda, \mu) = 0_n. \quad (7)$$

Условия разрешимости определяющего уравнения (7) установим с помощью теоремы о неявной функции, рассмотрев частные случаи для  $r$  и  $k$ .

Пусть  $r=1$ . Тогда  $\bar{F}(s,0,\lambda,\mu) \equiv 0_n$ , следовательно  $\bar{F}(s,0,0_n,0_m) = 0_n$ . Вычислим матрицу Якоби  $\frac{\partial \bar{F}(s,0,0_n,0_m)}{\partial(\lambda,\mu)} = \frac{1}{\alpha} \frac{\partial F(s,\alpha(\bar{x}_0 + \lambda),\alpha(\varepsilon_0 + \mu))}{\partial(\lambda,\mu)} \Big|_{\substack{\alpha=0 \\ \lambda=0_n \\ \mu=0_m}} = \frac{1}{\alpha} \frac{\partial F(s,x_0,\varepsilon)}{\partial(x_0,\varepsilon)} \cdot \frac{\partial(x_0,\varepsilon)}{\partial(\lambda,\mu)} =$

$$= \begin{bmatrix} \frac{\partial F}{\partial x_0} & \frac{\partial F}{\partial \varepsilon} \end{bmatrix} = [B(S) \quad (U_r(s,\varepsilon_0))'_\varepsilon].$$

Допустим  $\text{rang} D = n = \text{rang} D_1$ , где Обозначим  $D = [B(S) \quad (U_r(s,\varepsilon_0))'_\varepsilon]$ ,  $D_1 - n \times n$ -матрица, составленная из линейно независимых столбцов матрицы  $D$ . Тогда  $\det D_1 \neq 0$ . Из оставшихся столбцов составим матрицу  $D_2$ . Тогда  $D(\lambda,\mu)^T = D_1 U + D_2 V$ , где  $U \in R^n$ ,  $V \in R^m$ .

По формуле Тейлора получим разложение  $\bar{F}(s,\alpha,\lambda,\mu) = (B(s)\bar{x}_0 + U_1(s,\varepsilon_0)) + D(\lambda,\mu)^T + \chi(\lambda,\mu) + q(\alpha,\lambda,\mu) = 0_n$ , в котором  $\lim_{\rho \rightarrow 0} \rho^{-1} \|\chi(\lambda,\mu)\| = 0$ ,  $\rho = \|(\lambda,\mu)\|$ . По необходимому условию (5) существования малого периодического решения  $B(s)\bar{x}_0 + U_r(s,\varepsilon_0) = 0_n$ . Положим  $V = 0_m$ . Тогда уравнение (7) примет вид:

$$\bar{F}(\alpha,U) = D_1 U + \chi(U) + q(\alpha,U) = 0_n. \quad (8)$$

Здесь выполняются условия: 1)  $\bar{F}(0,0_n) = 0_n$ , 2)  $\bar{F}'_U(0,0_m) = D_1$ ,  $\det D_1 \neq 0$ . Значит, уравнение (8) определяет неявную функцию  $U = U(\alpha)$ ,  $U(0) = 0_n$ . Итак, зная  $U(\alpha)$  ( $V = 0_m$ ), восстановим вектор  $(\lambda,\mu)^T = (\lambda(\alpha),\mu(\alpha))^T$ . При этом  $\lambda(0) = 0_n$ ,  $\mu(0) = 0_m$ . Тогда значения  $x_0^* = \alpha(\bar{x}_0 + \lambda(\alpha))$ ,  $\varepsilon^* = \alpha(\varepsilon_0 + \mu(\alpha))$  удовлетворяют уравнению (4) и условиям  $x_0^* \rightarrow 0_n$ ,  $\varepsilon \rightarrow 0_m$  при  $\alpha \rightarrow 0$ . Значит  $x(t,x_0^*,\varepsilon^*)$  – малое  $s\omega$ -периодическое решение системы (1). Таким образом установлено следующее утверждение.

**Теорема 4.** Пусть  $r=1$ . Если существуют значения  $\bar{x}_0 \neq 0_n$ ,  $\varepsilon_0 \neq 0_m$ , при которых выполняются условия: 1)  $B(s)\bar{x}_0 + U_r(s,\varepsilon_0) = 0_n$ , 2)  $\text{rang} D = n$ ,  $D = [B(s) \quad (U_r(s,\varepsilon_0))'_\varepsilon]$ , то система (1) имеет малое  $s\omega$ -периодическое решение.

Проведем аналогичные рассуждения для случая, когда  $1 < r < k$ .

Пусть  $B(s) = 0_m$  или  $\lambda = \bar{\lambda}$  и  $\bar{x}_0$  выбраны так, что  $B(s)(\bar{x}_0 + \bar{\lambda}) = 0_n$ . Тогда перепишем равенство (7) в виде  $\bar{F}(\alpha,\bar{\lambda},\mu) = U_r(s,\varepsilon_0 + \mu) + \tilde{q}(\alpha,\lambda,\mu) = 0_n$ , где  $\tilde{q}(\alpha,\lambda,\mu) = \alpha^{k-1} p_k(s,\bar{x}_0 + \lambda,\varepsilon_0 + \mu) + \frac{1}{\alpha} \varphi(s,\alpha(\varepsilon_0 + \mu)) + \frac{1}{\alpha} \psi(s,\alpha(\bar{x}_0 + \lambda),\alpha(\varepsilon_0 + \mu))$ . Заметим, что

выполняются условия: 1)  $\bar{F}(0,0_n,0_m) = U_r(s,\varepsilon_0) = 0_n$ , 2)  $\frac{\partial \bar{F}(0,0_n,0_m)}{\partial(\lambda,\mu)} = U_r'(s,\varepsilon_0)$ .

Рассмотрим  $n \times m$ -матрицу  $D = U_r'(s,\varepsilon_0)$ . Допустим,  $m \geq n$  и  $\text{rang} D = n = \text{rang} D_1$ , где  $D_1 - n \times n$ -матрица, составленная из линейно независимых столбцов матрицы  $D$ . Тогда

$\det D_1 \neq 0$ . Из оставшихся столбцов составим матрицу  $D_2$ . Тогда  $D(\mu) = D_1 U + D_2 V$ , где  $U \in R^n$ ,  $V \in R^{m-n}$ . По формуле Тейлора получим  $\bar{F}(\alpha, \bar{\lambda}, \mu) = D(\mu) + \chi(\bar{\lambda}, \mu) + \tilde{q}(\alpha, \bar{\lambda}, \mu) = 0_n$ ,

где  $\lim_{\rho \rightarrow 0} \frac{\|\chi(\bar{\lambda}, \mu)\|}{\rho} = 0$ ,  $\rho = \|(\bar{\lambda}, \mu)\|$ . Выберем  $V = 0_m$ . Тогда система (7) примет вид:

$$\bar{F}(\alpha, U) = D_1 U + \chi(U) + \tilde{q}(\alpha, U) = 0_n. \quad (9)$$

При этом выполняются условия: 1)  $\bar{F}(0, 0_n) = 0_n$ , 2)  $\bar{F}'_U(0, 0_m) = D_1$ ,  $\det D_1 \neq 0$ .

Применив теорему о неявной функции, получим, что система (9) определяет при малых  $\alpha$  неявную функцию  $U = U(\alpha)$ ,  $U(0) = 0_n$ . Итак, зная  $U(\alpha)$  ( $V = 0_m$ ), восстановим вектор

$\mu = \mu(\alpha)$ ,  $\mu(0) = 0_m$ . Тогда значения  $x_0^* = \alpha \bar{x}_0$ ,  $\varepsilon^* = \alpha(\varepsilon_0 + \mu(\alpha))$  удовлетворяют уравнению (4) и условиям  $x_0^* \rightarrow 0_n$ ,  $\varepsilon \rightarrow 0_m$  при  $\alpha \rightarrow 0$ . Значит,  $x(t, x_0^*, \varepsilon^*)$  – малое  $s\omega$ -периодическое решение системы (1). Таким образом, установлено следующее достаточное условие.

*Теорема 5.* Пусть  $1 < r < k$ . Если существуют значения  $\bar{x}_0 \neq 0_n$ ,  $\varepsilon_0 \neq 0_m$ , при которых выполняются условия: 1)  $B(s)\bar{x}_0 = 0_n$ , 2)  $m \geq n$ , 3)  $U_r(s, \varepsilon_0) = 0_n$ , 4)  $\text{rang} U'_r(s, \varepsilon_0) = n$ , то система (1) имеет малое  $s\omega$ -периодическое решение. По схеме рассуждений, использованных для доказательства теорем 4 и 5, устанавливается справедливость следующих утверждений.

*Теорема 6.* Пусть  $r = k$ . Если при некоторых значениях  $\bar{x}_0 \neq 0_n$ ,  $\varepsilon_0 \neq 0_m$  выполняются условия: 1)  $B(s)\bar{x}_0 = 0_n$ , 2)  $U_r(s, \varepsilon_0) + p_k(s, \bar{x}_0, \varepsilon_0) = 0_n$ , 3)  $\text{rang}(U'_r(s, \varepsilon_0) + p'_k(s, \bar{x}_0, \varepsilon_0)) = n$ , то система (1) имеет малое  $s\omega$ -периодическое решение.

*Теорема 7.* Пусть  $r > k$ . Если существуют значения  $\bar{x}_0 \neq 0_n$ ,  $\varepsilon_0 \neq 0_m$ , при которых выполняются условия: 1)  $B(s)\bar{x}_0 = 0_n$ , 2)  $p_k(s, \bar{x}_0, \varepsilon_0) = 0_n$ , 3)  $\text{rang} p'_k(s, \bar{x}_0, \varepsilon_0) = n$ , то система (1) имеет малое  $s\omega$ -периодическое решение.

### Литература

1. Красносельский М.А. Оператор сдвига по траектории дифференциальных уравнений. М., 1966.
2. Абрамов В.В. Ветвление периодического решения неавтономной системы с малым параметром // Вестник РАЕН. 2015. Т. 15. №5. С. 3-7.
3. Абрамов В.В. К задаче об устойчивости малого периодического решения // Итоги науки и техники. Серия «Современная математика и ее приложения. Тематические обзоры». 2018. Т. 148. С. 3-9.

© Земенков А.В., Абрамов В.В., 2021

УДК 519.65

Лосева А.В., Семенова А.А.,  
Покорная И.Ю., канд. физ.-мат. наук  
Воронежский государственный педагогический  
университет, г. Воронеж, Россия

## ЗНАЧИМОСТЬ СПЛАЙНА В РАЗЛИЧНЫХ СФЕРАХ ЖИЗНЕДЕЯТЕЛЬНОСТИ И ОСНОВЫ ЕГО ПОСТРОЕНИЯ

В настоящее время сплайны (*spline* – гибкое лекало) достаточно широко применяются во многих областях, однако первоначально они строились с помощью гибкой металлической линейки, универсального лекала, соединяющего заданные точки гладкой кривой.

За сплайны можно принять любые кривые, которые имеют абсолютно различную геометрическую форму: окружности, дуги, части параболы, поверхности в трехмерном пространстве и другие. Для создания огибающей поверхности используется каркас, благодаря которому появляется возможность создавать модели с высокой степенью детализации, делая поверхность более гладкой (рис. 1).

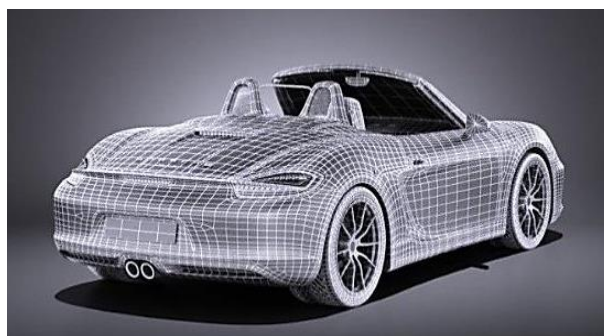


Рис. 1. Пример огибающей поверхности

На сегодняшний момент сплайны традиционно используются в системе автоматизированного проектирования. Однако сплайны имеют более широкие потенциальные возможности, чем просто описание некоторых кривых. Если рассматривать реальный мир, то с помощью сплайнов описываются многие физические процессы. Например, в термодинамике – это теплообмен в стержне, составленном из фрагментов с различной теплопередачей. В электричестве – распространение электромагнитных полей через разнородные среды. В химии – диффузия через слои различных веществ.

Внешняя оболочка современного самолета состоит из совокупности достаточно разнообразных и даже неожиданных форм. Когда происходило становление авиационной технологии, основой воспроизведения таких форм в виде готовых конструкций был так называемый плазово-шаблонный метод. Однако у этого метода было много недостатков. Поэтому для решения данной проблемы было положено начало исследованиям сплайн-



функций. Методы моделирования поверхностей сплайнами и их программная реализация на ЭВМ помогли перейти от плазово-шаблонного к независимому методу производства.

Огромный вклад в исследование сплайнов внес Юрий Семенович Завьялов – основоположник сибирской школы теории и приложений сплайн-функций [1]. Решение о моделировании сложных поверхностей сплайнами, принятое им в середине 60-х годов, оказалось пророческим. В современном мире ни одна автоматизированная система проектирования и конструирования сложных геометрических форм не обходится без сплайнов. Очень распространенными отраслями применения сплайна являются самолетостроение, судостроение, автомобилестроение, производство гидротурбин, швейная и обувная промышленности и другие. Таким образом, сплайн нельзя назвать какой-то выдуманной математической абстракцией, а, наоборот, благодаря ему можно решать даже дифференциальные уравнения, которые описывают вполне реальные физические процессы.

Сплайны являются важным инструментом в исследовании технологических процессов. Рассмотрим некоторые способы задания сплайна с помощью различных начальных точек, а также с соблюдением условий гладкости и непрерывности функции.

Рассмотрим пример построения параболического сплайна второй степени дефекта 1.

Для сплайна будем рассматривать начальные условия в точке А. Даны точки  $A(0; 8)$  и  $B(7; -3)$ , через которые проходит парабола  $S_1$ , точка склейки двух парабол  $C(4; \frac{300}{49})$ , точка  $D(8; 4)$ , через которую проходит парабола  $S_2$ , а также известно, что в начальной точке А сплайн имеет угол наклона  $\frac{\pi}{4}$ , что означает – тангенс угла наклона касательной, проведенной к сплайн-функции в точке А равен 1.

Построим данный сплайн на отрезке  $[-4; 12]$ :

$$S = \begin{cases} S_1 = a_1x^2 + b_1x + c_1 & [-4; 4] \\ S_2 = a_2x^2 + b_2x + c_2 & [4; 12] \end{cases}$$

Запишем все условия в систему, в том числе и условия непрерывности и гладкой склейки в точке С:

$$\begin{cases} S_1(4) = S_2(4) \\ S_1'(4) = S_2'(4) \\ S_1'(0) = 1 \\ S_1(0) = 8 \\ S_1(7) = -3 \\ S_2(8) = 4 \end{cases}, \text{ получим } \begin{cases} 16a_1 + 4b_1 + c_1 = 16a_2 + 4b_2 + c_2 \\ 8a_1 + b_1 = 8a_2 + b_2 \\ b_1 = 1 \\ c_1 = 8 \\ 49a_1 + 7b_1 + c_1 = -3 \\ 64a_2 + 8b_2 + c_2 = 4 \end{cases}$$

Запишем данную систему уравнений в виде матрицы и решим ее:

$$\left( \begin{array}{cccccc|c} 16 & 4 & 1 & -16 & -4 & -1 & 0 \\ 8 & 1 & 0 & -8 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 8 \\ 49 & 7 & 1 & 0 & 0 & 0 & -3 \\ 0 & 0 & 0 & 64 & 8 & 1 & 4 \end{array} \right), \quad \begin{cases} a_1 = -\frac{18}{49} \\ b_1 = 1 \\ c_1 = 8 \\ a_2 = \frac{69}{196} \\ b_2 = -\frac{233}{49} \\ c_2 = \frac{956}{49} \end{cases}$$

Таким образом, сплайн определяется следующим условием:

$$S = \begin{cases} S_1 = -\frac{18}{49}x^2 + x + 8 & [-4; 4] \\ S_2 = \frac{69}{196}x^2 - \frac{233}{49}x + \frac{956}{49} & [4; 12] \end{cases}$$

и имеет график (рис. 2), который является непрерывной и гладкой кривой.

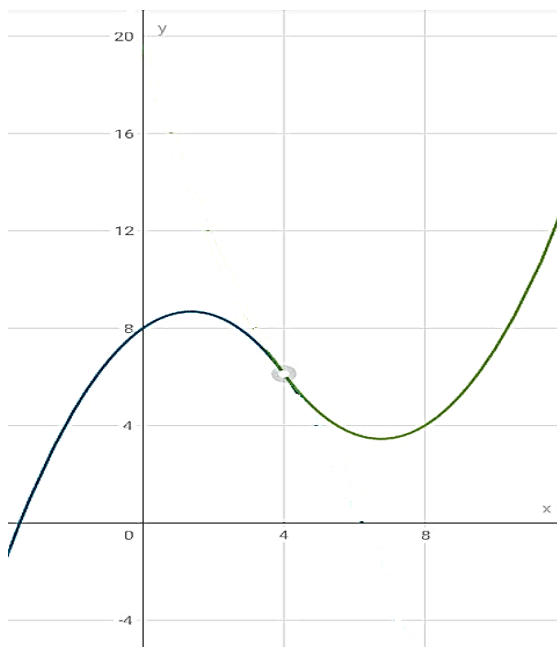


Рис. 2

Рассмотрим еще один пример построения параболического сплайна второй степени дефекта 1.

Теперь зададим краевые условия для сплайна на концах отрезка  $[-3; 7]$ .

Дана точка  $A(-3; -1)$ , через которую проходит парабола  $S_1$ , точка склейки  $B(2; 3)$ , точка  $C(7; 7)$ , через которую проходит парабола  $S_2$ , а также известно, что в начальной точке  $A(-3; -1)$  и точке  $C(7; 7)$  сплайн имеет угол наклона  $\frac{\pi}{4}$ , а это значит, что тангенс угла наклона касательных, проведенных к сплайн-функции в точках  $A$  и  $C$  равен 1.

Построим сплайн и посмотрим, что получится.

$$S = \begin{cases} S_1 = a_1x^2 + b_1x + c_1 & [-3; 2] \\ S_2 = a_2x^2 + b_2x + c_2 & [2; 7] \end{cases}$$

Запишем все условия в систему, а также условия непрерывности и гладкой склейки в точке  $B$ :

$$\begin{cases} S_1(2) = S_2(2) \\ S_1'(2) = S_2'(2) \\ S_1'(-3) = 1 \\ S_1(-3) = -1 \\ S_2(7) = 7 \\ S_2'(7) = 1 \end{cases}$$

$$\begin{cases} 4a_1 + 2b_1 + c_1 = 4a_2 + 2b_2 + c_2 \\ 4a_1 + b_1 = 4a_2 + b_2 \\ -6a_1 + b_1 = 1 \\ 9a_1 - 3b_1 + c_1 = -1 \\ 49a_2 + 7b_2 + c_2 = 7 \\ 14a_2 + b_2 = 1 \end{cases}$$

Запишем данную систему уравнений в виде матрицы и решим ее:

$$\left( \begin{array}{cccccc|c} 4 & 2 & 1 & -4 & -2 & -1 & 0 \\ 4 & 1 & 0 & -4 & -1 & 0 & 0 \\ -6 & 1 & 0 & 0 & 0 & 0 & 1 \\ 9 & -3 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 49 & 7 & 1 & 7 \\ 0 & 0 & 0 & 14 & 1 & 0 & 1 \end{array} \right), \text{ получим } \begin{cases} a_1 = -\frac{1}{25} \\ b_1 = \frac{19}{25} \\ c_1 = \frac{41}{25} \\ a_2 = \frac{1}{25} \\ b_2 = \frac{11}{25} \\ c_2 = \frac{49}{25} \end{cases}$$

Таким образом, сплайн определяется следующим условием:

$$S = \begin{cases} S_1 = -0,04x^2 + 0,76x + 1,64 & [-3; 2] \\ S_2 = 0,04x^2 + 0,44x + 1,96 & [2; 7] \end{cases}$$

и имеет график (рис. 3), который является непрерывной и гладкой кривой.

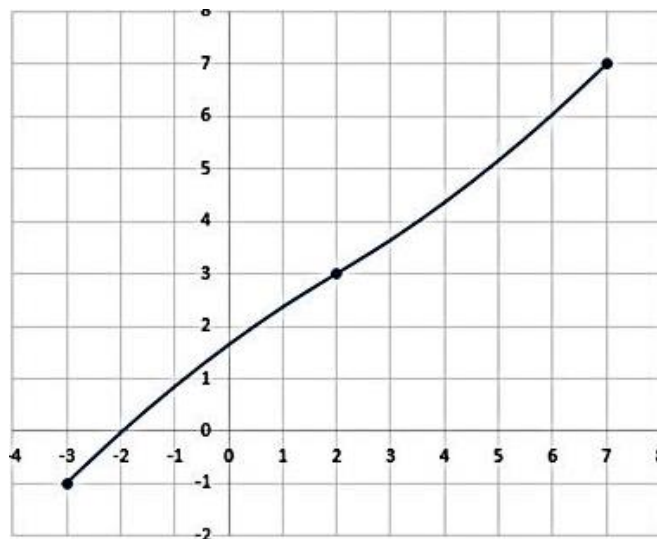


Рис. 3.

Данные примеры являются наглядными для понимания, сколькими условиями определяется сплайн и как начальные или краевые условия влияют на количество исходных заданных точек. Но, очевидно, что для построения параболического сплайна дефекта 1, склеенного из двух частей потребуется ровно 6 условий для однозначного определения всех его коэффициентов.

Безусловно, параболический сплайн дефекта 1 представляет собой кривую с достаточно большим количеством склеенных гладким образом кусочков параболы, но общая теория

складывается из самых простых примеров. Эти примеры дают возможность качественного понимания общей теории сплайнов [2, 3].

### Литература

1. Завьялов Ю.С., Квасов Б.И., Мирошниченко В.П. Методы сплайн функций. М., 1980. 352 с.
2. Покорная И.Ю., Суворова Е.А. Основы построения: сплайны и кривые Безье // Некоторые вопросы анализа, алгебры, геометрии и математического образования. Воронеж, 2017. Вып. 6. С. 149-150.
3. Покорная И.Ю., Суворова Е.А. Прикладная сфера сплайнов и их построение // Некоторые вопросы анализа, алгебры, геометрии и математического образования. Воронеж, 2016. Вып. 5. С. 139-141.

© Лосева А.В., Семенова А.А., Покорная И.Ю., 2021

УДК 519.688

**Захаров Д.М.**

*Научный руководитель:* **Слива М.В.**, канд. пед. наук  
Нижевартовский государственный университет,  
г. Нижневартовск, Россия

## **ДВИЖОК ВИЗУАЛИЗАЦИИ ДАННЫХ ROASSAL**

Представление информации через удобные для восприятия интерактивные образы является достаточно сложным технологическим процессом. При создании визуализации данных многие решения должны приниматься на основе личной интуиции или через заранее созданный и тщательно продуманный проект дизайна. Возможность быстро опробовать новую идею является ключевым аспектом в этом деле. Принципы гибкой визуализации данных основаны на уменьшении затрат, связанных с задачами программирования графического представления, и расширения границ для творческих экспериментов.

Главная проблема создания графических образов информации состоит в создании такой визуализации, которую можно было бы многократно использовать и расширять. Технологии гибких визуализаций удовлетворяют потребность в многократном использовании визуализаций и открывают возможности по использованию готовых визуализаций для данных из разных источников.

Анализ данных выполняется путем создания ряда визуализаций, многие из которых имеют очень короткий период использования, поэтому сокращение времени их создания до нескольких секунд или минут значительно увеличивает количество возможных способов решения проблемы, которые использует исследователь данных [1].

Roassal – это свободный движок для визуализации данных с открытым исходным кодом, реализованный для языков программирования Pharo и VisualWorks, являющихся диалектами языка Smalltalk. Roassal способствует созданию визуализаций путем выполнения последовательных коротких шагов – от нескольких секунд до нескольких минут. Визуализации, созданные с помощью Roassal, состоят из следующих компонентов: представления, элементы, формы, взаимодействия и анимации. Представление является контейнером для графических элементов. Каждый элемент состоит из одного или нескольких форм – объектов, описывающих различные фигуры: прямоугольник, круг, линия или надпись. Взаимодействия представляют действия пользователя, такие как события мыши и клавиатуры. Следующий пример демонстрирует визуализацию, в которой каждый узел является числом от 1 до 50.

```
v := RTView new.  
shape := RTBox new color: Color blue trans.  
elements := shape elementsOn: (1 to: 50).  
v addAll: elements.  
elements @ RTPopup.  
RTEdgeBuilder new
```



```
view: v;
objects: (1 to: 50);
connectFrom: [ :i | i // 3 ].
```

**RTTreeLayout** on: elements.

v

В данном примере представление состоит из 50 элементов, каждый из которых состоит из одной формы – прямоугольника. Каждый элемент связан со своим результатом целочисленного деления на 3. Полученное представление изображено на рисунке 1.

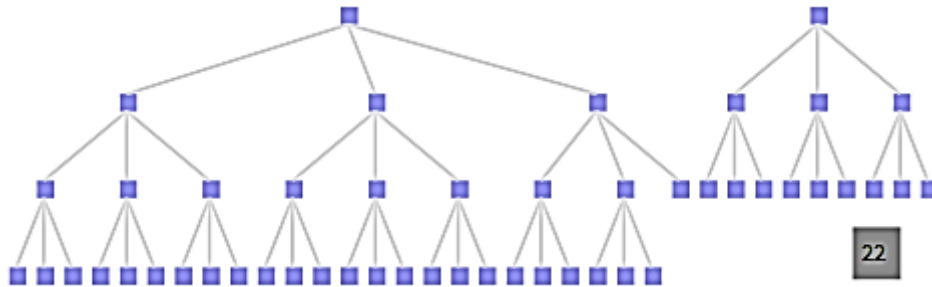


Рис. 1. Визуализация чисел

Одним из ключевых аспектов Roassal является возможность представления форм элементов через метрики или числовые значения, которые определяют их размеры. Roassal также поддерживает формат представления данных TSV (Tab Separated Values). Данный формат предполагает хранение полей таблиц в виде текста, в котором строки представляют записи с полями, разделенными символом табуляции.

```
tab := RTTabTable new input:
'id      value1  value2  category parent
1        10     20      A         1
2         5     12      B         1
3         8     17      A         1
4         9     13      D         3
5        30     30      D         3'.
tab
    removeFirstRow;
    convertColumnsAsInteger: #(1 2 3 5).
v := RTView new.
colorNormalizer := RTMultiLinearColorForIdentity new
    objects: (tab values collect: #fourth) asSet sorted;
    command: #fourth.
shape := RTBox new
    width: #second;
    height: #third;
    color: colorNormalizer.
elements := shape elementsOn: tab values.
v addAll: elements.
elements @ (RTPopup new text: [ :entry |
'id = ', entry first asString, String cr,
'value1 = ', entry second asString, String cr,
'value2 = ', entry third asString, String cr ]).
elements @ (RTLabeled new text: [ :entry | entry first ]).
```

**RTEdgeBuilder new**

```
view: v;
objects: tab values;
connectFrom: [ :entry | tab values at: entry fifth ].
```

**RTHorizontalTreeLayout new**

```
verticalGap: 30; on: elements.
```

v

Данный код является примером использования TSV формата для визуализации данных. Объекты разбиты на категории (поле category), каждой из которой автоматически присвоен определенный цвет. Результат выполнения данного кода изображен на рисунке 2.

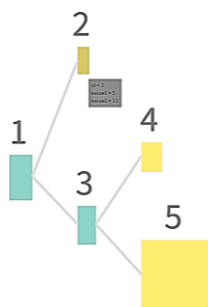


Рис. 2. Связывание элементов TSV

При наведении мыши на один из объектов отображаются его характеристики: идентификатор, первое значение и второе значение (ширина и высота).

Одной из ключевых особенностей языка Smalltalk является динамичность среды. Все классы и объекты можно изменять во время выполнения без необходимости перекомпиляции. Движок Roassal в полной мере соответствует данной идеологии и включает возможность динамического изменения визуализаций. Следующий пример демонстрирует реализацию добавления новых элементов в визуализацию при помощи кнопок.

```
v := RTView new.
v @ RTDraggableView.
stepping := RTSpringLayoutStepping new view: v.
stepping after: [ v canvas camera focusOnCenter ].
v addAnimation: stepping.
v canvas addMenu: 'Add circle' callback: [
    | el |
    el := (RTEllipse new size: 20; color: (Color blue alpha: 0.4)) element.
    el @ RTDraggable.
    el translateTo: 5 atRandom @ 5 atRandom.
    v add: el.
    stepping addNode: el.
    v signalUpdate.
].
v canvas addMenu: 'Add connected circle' callback: [
    | el edge |
    el := (RTEllipse new size: 20; color: (Color blue alpha: 0.4)) element.
    el @ RTDraggable.
    el translateTo: 5 atRandom @ 5 atRandom.
```

```
v add: el.  
    edge := RTLLine edgeFrom: el to: v elements atRandom.  
v add: edge.  
    stepping addNode: el; addEdge: edge.  
v signalUpdate.  
].  
v
```

Как показано на рисунках 3 и 4, при добавлении новых элементов визуализация обновляет свою компоновку и автоматически выстраивает объекты таким образом, чтобы их можно было легко различить, как показано на рисунках 3 и 4. Пользователь также может передвигать объекты мышью.

Add circle Add connected circle



Рис. 3. Визуализация до добавления нового объекта

Add circle Add connected circle



Рис. 4. Визуализация после добавления нового объекта

Roassal поддерживает экспортирование визуализаций в различные форматы, включая PNG, SVG и HTML5. Экспорт в HTML позволяет открывать визуализацию в браузере с сохранением интерактивности. Этот процесс создает два файла: “index.html” и “roassal.js”. Последний включает в себя реализацию функционала движка на языке JavaScript.

### Литература

1. Bergel A. Agile Visualization. 2016.

© Захаров Д.М., Слива М.В., 2021

УДК 004

**Бренер Л.И., Чеверева С.А.**, канд. пед. наук  
Самарский государственный экономический  
университет, г. Самара, Россия

## **ЭФФЕКТИВНОСТЬ ИСПОЛЬЗОВАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ПРОЕКТАМИ**

В современное время огромные объемы ежедневной информации чаще становятся проблемой для социального и экономического развития общества. В качестве решения данной проблемы выступают современные информационные технологии, которые все больше приобретают актуальность в настоящее время среди органов власти, руководства предприятий, а также населения.

Что касается органов власти, то они все активнее применяют проектные подходы для реализации проектов и программ. Проектные подходы в данном случае основаны на применении современных информационных систем управления проектами.

Рассмотрим сущность ключевых понятий информационной системы управления проектами.

Под информационной системой понимается некоторый взаимосвязанный комплекс программных, информационных, математических средств, предназначенный для сбора, обработки, хранения и передачи информации, необходимой для принятия любых управленческих решений.

Авторы статьи «Повышение эффективности информационной системы управления проектами с помощью оценки компетенций сотрудников» Е.В. Тодорица и А.К. Бардин отмечают, что информационные системы управления проектами являются инструментом, дающим возможность повысить эффективность управления предметной областью организаций, занимающихся проектированием [1, с. 430].

Системой управления проектами называют совокупность технологических и организационных инструментов и методов, связанных в общую единую структуру, с помощью которой выполняется всесторонняя поддержка и повышение эффективности управления и реализации проектов.

Некоторые ученые отмечают, что информационная система управления проектами направлена на повышение эффективности планирования деятельности организаций посредством комплекса специализированного программного обеспечения.

Информационная система управления проектами состоит из следующих элементов:

- эффективная организационная структура управления;
- комплекс взаимосвязанных процедур, осуществляющих управление проектами;
- нормативно-регламентная база;
- инструментальная система поддержки процессов управления проектами.

Целью внедрения информационной системы управления проектами является постоянный рост эффективности управления проектами в организации и поддержка процессов

всех уровней до окончания их жизненных циклов. Технологические улучшения процессов непосредственно оказывают влияние на повышение эффективности всей работы проектов.

К числу основных задач информационной системы управления проектами относят:

- составление поэтапного расписания осуществления проекта, расчет прогнозных показателей;
- составление проектного бюджета;
- определение объема ресурсов, необходимых для осуществления проекта;
- анализ рисков проекта;
- поиск резервов ресурсов и времени, в случае наступления неблагоприятного события;
- учет исполнения работ по проекту;
- ведение соответствующей отчетной документации по конкретному проекту.

В экономике выделяют большое количество информационных систем управления проектами, выбор которых зависит от специфики, технических характеристик, доступности систем для определенного проекта. В соответствии с отраслевой спецификой, современные информационные системы подразделяются на организационные, инвестиционные, инновационные, производственные системы и др.

На сегодняшний день особую популярность набирают информационные системы управления проектами направленные на автоматизацию документооборота, оперативной постановки и контроля задач проекта.

Информационные системы в первую очередь должны быть направлены на эффективное управление проектами с исключением всевозможных ошибок и проблем, возникающих на любом этапе осуществления проектных решений. Эффективное использование информационных систем в управлении проектами зависит от многих факторов, таких как процессы разработки, исследований, испытаний и оценки эффективности с разных точек зрения.

Оценка эффективности использования информационных систем в управлении проектами дается при исследовании определенного набора критериев и показателей. Как правило, выделяют количественные и качественные показатели. Количественные показатели относительно просты в расчетах, но не всегда дают полное представление о преимуществах объекта оценки. Качественные показатели позволяют оценить признаки, свойства и особенности информационных систем.

При количественной оценке эффективности проектной деятельности используется метод сравнительного анализа, который учитывает отклонения стоимости проекта, объем неразрешимых проблем, эффективность работы команды и др. Кроме того, количественную оценку проекта дает метод функционально-стоимостного анализа, позволяющий выявить потенциальные возможности и слабые стороны деятельности организаций.

Среди распространенных методов качественной оценки эффективности информационных систем является метод экспертных оценок. И. Ю. Сурыгина в своей статье «Эффективность использования информационных систем управления бизнес-проектами» отмечает, что данный метод рекомендуется использовать неоднократно на этапе выполнения



проекта. Его проводят циклически – через определенные промежутки времени, например, каждый месяц или при закрытии этапа проекта [2, с. 70].

На сегодняшний день в России существует большое количество программных продуктов как простых, доступных для индивидуального использования, так и сложных программ с помощью которых организации способны интегрировать информацию для управления проектами удаленно.

Информационные системы управления проектами делятся на три вида: локальные, серверные, информационные технологии на основе интернет-технологий.

Локальные информационные системы (Open Project, Microsoft Project Standart) достаточно доступны и недорогие по стоимости. Они предназначены для организаций, которые практически не занимаются проектной деятельностью к примеру малый бизнес.

Серверные информационные системы (HP Project, Oracle Primavera) рассчитаны на постоянное управление проектами. Данные информационные системы требуют дорогостоящую постоянную функциональную поддержку, обучение персонала.

Информационные технологии на основе интернет-технологий (COMINDWORK, IBN) отличаются от серверных лишь возможностями использования функционала какой-либо другой организации находящийся удаленно. Однако передавая информацию сторонней организации, возникают риски по утрате коммерческой тайны.

Для того что бы организации определиться какие информационные технологии в управлении проектами будут наиболее эффективны. Г. Хуснидинова в своей работе «Эффективность использования информационных систем управления проектами» советует сначала определить потребности компании и соотнести их с функциональными возможностями информационной системы управления проектами [3, с. 178].

Внедрение информационной системы управления проектами позволяет достичь следующих результатов: создание единой доступной информационной среды для участников конкретного проекта; оптимальное использование ресурсов (финансов, времени, производственных мощностей) предприятия; увеличение скорости осуществления проектов; легкость в управлении качеством работы персонала; эффективность управления ходом реализации проекта, его сроками; исключение повседневной рутинной работы, в следствии снижение загруженности персонала; отслеживание реализации сразу нескольких проектов; снижение рисков неэффективного планирования, проектных провалов.

В проектном управлении информационные системы играют первостепенную роль. Для того что бы отслеживать определенные этапы проекта организации регистрируют их в общей системе документооборота.

А.С. Иванов в работе «Методы оценки эффективности проекта» приводит основные выводы по исследованиям, согласно которым специалисты в области управления проектами и представители компаний различных отраслей США считают, что прирост эффективности составляет при использовании информационной системы управления проектами порядка 21 % по отношению к показателям компаний, не использующих подобную систему для ведения проектной деятельности [4, с. 349].

Автор статьи «Подходы к оценке эффективности использования информационной системы управления проектами в организации» Зюськин А.А. считает, что для обеспечения полной и достоверной оценки эффективности всех параметров информационной системы управления проектами в организации необходимо сформировать в штатной структуре организации специальную организационную единицу, которая будет призвана решать задачи по созданию и реализации специальных механизмов эффективного управления [5, с. 274].

Таким образом, информационная система управления проектами представляет собой комплекс программных, информационных, математических средств, предназначенных для сбора, обработки, хранения и передачи информации, необходимой для управления проектами. Эффективность информационных систем управления проектами оценивается количественными и качественными показателями. Выбор оптимальной информационной системы управления проектами зависит от специфики проекта, от возможностей организаций. Внедрение информационной системы управления проектами позволяет достичь оптимального использования ресурсов, эффективнее управлять процессом реализации проекта, создать единую доступную информационную среду для участников, выявить риски неэффективного планирования и т. д.

### Литература

1. Тодорица Е.В., Бардин А.К. Повышение эффективности информационной системы управления проектами с помощью оценки компетенций сотрудников // Цифровизация экономики: направления, методы, инструменты. Сборник материалов II всероссийской научно-практической конференции. Краснодар, 2020. С. 430-433.
2. Сурыгина И.Ю. Эффективность использования информационных систем управления бизнес-проектами // Электротехнические и информационные комплексы и системы. Уфа, 2008. Т. 4. №1. С. 66-76.
3. Хуснидинова Г. Эффективность использования информационных систем управления проектами // Современные информационные технологии: проблемы и перспективы развития: Материалы I Международной научно-практической конференции. Екатеринбург, 2017. С. 175-180.
4. Иванов А.С. Методы оценки эффективности проекта // 76-я научная конференция студентов и аспирантов белорусского государственного университета материалы конференции: В 3-х ч. Минск, 2019. С. 346-349.
5. Зюськин А.А. Подходы к оценке эффективности использования информационной системы управления проектами в организации // Роль интеллектуального капитала в экономической, социальной и правовой культуре общества XXI века. Сборник научных трудов. СПб., 2015. С. 271-275.

© Бренер Л.И., Чеверева С.А., 2021

УДК 004

**Кульмяков А.А.**

*Научный руководитель: Чеверева С.А., канд. пед. наук*  
Самарский государственный экономический  
университет, г. Самара, Россия

## **ПОСТРОЕНИЕ ЦИФРОВОГО ОБЩЕСТВА**

В настоящий момент времени происходит становление цифровизации общества, за счет внедрения новейших технологий в социально-экономические сферы деятельности человека и автоматизированные процессы внедрения бизнеса. В статье раскрывается понятие цифрового общества, технологии, определяющие построение цифрового общества в различных сферах деятельности.

Цифровое общество подразумевает под собой определенную модель жизни социума, которая функционирует на определенных показателях в настоящий момент времени. Данное общество создает качественные инновационные комплексы и системы для создания комфортных условий для жизни людей, это является так называемой «киберфизической системой». Это понятие фигурировало в научных исследованиях американца Н. Негропonte, в которых он рассмотрел достоинства этих социальных и экономических новшеств, сравнивая существующую систему и внедрение IT-технологий [1, с. 33].

Модернизирующие процессы так или иначе неизбежны, это можно видеть нарастающими изменениями национальных экономических систем, так как социально-экономические отношения и потенциал исчерпывают себя, из этого следует что необходимо внедрять новые пути развития социальной и экономической системы, механизмы их управления и анализировать факторы их изменений. На данный момент времени экономика Российской Федерации характеризуется непредсказуемостью результатов различных проектов, слабой управляемостью, зависимостью от сырья и промышленно-развитых стран. В данной системе мировой экономики Россия является слабым конкурентом.

Программа «Цифровая экономика Российской Федерации» и разработанная Стратегия развития в Российской Федерации на 2017-2030 годы, направлены на совершенствование технологий в различных отраслях деятельности [2, с. 155].

Основные цели программы построения цифрового общества:

- формирование экосистемы экономики РФ на основе цифровых технологий, где они являются ключевым фактором производственных показателей во всех социально-экономических сферах деятельности человека.
- улучшение инфраструктуры и высокотехнологичных направлений с целью формирования инновационных разработок и высокотехнологичного производства.
- увеличение конкурентоспособности РФ во всех сферах деятельности на мировом рынке [3, с. 15].

Стоит отметить, что на данный момент в цифровом обществе отмечаются три уровня во всех социально-экономических сферах деятельности человека:

- виды экономической деятельности хозяйствующих субъектов и уровень рынков.
- оснащение цифровыми платформами для экономической деятельности и их уровень.
- эффективная среда, создающая благоприятные условия для создания и совершенствования цифровых платформ, подготовка кадров, информационная безопасность и нормативная база [4, с. 17].

Так как построение цифрового общества завязано на использовании цифровых технологий и создании данных платформ, отмечают именно третий пункт.

Исходя из этого формулируются основные цели и задачи с внедрением и разработкой цифровых платформ для социально-экономической деятельности человека, включая нормативную базу, информационную безопасность, подготовку кадров:

- развитие государственных институтов для цифровой экономики (внедрение технологических платформ, подготовка специалистов, образовательная деятельность, нормативное регулирование).

- введение необходимой инфраструктуры для развития цифровой экономики, к которому можно отнести технологические платформы, программное обеспечение, цифровая инфраструктура, информационная безопасность.

Вся социально-экономическая деятельность людей переходит на новый технологический этап, в котором играют роль новые современные технологии:

1. Работа с большими данными;
2. Система распределения реестров данных;
3. Создание производственного интернета;
4. Развитие нейросетей и искусственного интеллекта;
5. Совершенствование квантовых технологий;
6. Робототехника и виртуальные предприятия;
7. Развитие беспроводной электросвязи.

Совершенствование нынешних ИКТ-средств осуществляет влияние на все сферы жизни общества, это обуславливается происходящими процессами глобализации и развитием единого информационного пространства. Примером этого является интеграция сферы экономики и научного познания [5, с. 50].

Исходя из данной тенденции развития информационного общества в современной науке произошло формирование таких понятий, как «информационная экономика» как информационное общество, инновационная экономика, информационная экономика, а также трансформацией научно-коммуникативных связей и развитием электронных и цифровых коммуникаций.

Построение цифрового общества предусматривает использование компьютерных технологий для обработки информации во всех сферах деятельности [6, с. 145]. В таких условиях информация является наиважнейшим ресурсом, который требует внедрения инновационных методов защиты.

Информационная структура напрямую связана с развитием цифровых технологий, которые получили свое название из-за специфики представления информации в дискретной

форме. Технологически, эксплуатация цифровых технологий позволяет сублимировать различные отрасли информационной системы в единое целое и предоставлять более высокие вычислительные возможности. Таким образом они являются наиболее частой формой наиболее частой формой вида информации для выполнения операций с ней [7, с. 195].

Цифровая инфраструктура в данном контексте является частью информационной инфраструктуры. Но стоит отметить, что современная инфраструктура почти вся строится на цифровых технологиях, отсюда можно отметить равнозначность понятий «цифровая инфраструктура» и «информационная инфраструктура».

Увеличение цифровой информации в социально-экономической деятельности человека наглядно показывает курс на становление цифровизации отечественного общества, что показывает данное направление как прогрессивный подход. Наглядно можно наблюдать процесс формирования нового общества (информационно-экономического) в процессах увеличения информации, глобализации экономики, транснационализация, новшества в производственных и бизнес-процессах.

Исходя из всего этого можно сделать вывод, что на данный момент с увеличением роли информации во всех сферах социально-экономической деятельности человека и развитием технологического прогресса общество находится в завершающей стадии процесса цифровизации.

#### Литература

1. Ковальчук Ю.А., Степнов И.М. Цифровая экономика: трансформация промышленных предприятий // Инновации в менеджменте. 2017. №11. С. 32-43.
2. Глухова Л.В., Митрофанова Я.С. Цифровизация экономики и особенности ее приложения в деятельность интегрированных производственных структур // Вестник Поволжского государственного университета сервиса. 2017. №3 (49). С. 155-160.
3. Давиденко Л.М. Цифровая трансформация экономики промышленного комплекса // Grand Altai Research & Education. 2018. №2 (8). С. 13-19.
4. Харченко А.А., Конюхов В.Ю. Цифровая экономика как экономика будущего // Молодежный вестник ИрГТУ. 2017. №3 (27). С. 17.
5. Колодей Ю.С., Коршунова О.С. Цифровая экономика как одно из перспективных направлений развития экономики РФ // Вестник профессионального бухгалтера. 2017. №4-6 (199-201). С. 50-56.
6. Соложенцев Е.Д. Цифровое управление государством и экономикой // Актуальные проблемы экономики и управления. 2018. №1 (17). С. 136-153.
7. Соловьев А.К. Социальные риски цифровой экономики // Вестник факультета управления СПбГЭУ. 2018. №5. С. 194-196.

© Кульмяков А.А., Чеверева С.А., 2021



УДК 004.942

**Корженко М.А., Кузьменко Т.А.**

*Научный руководитель:* **Кондратьева Т.Н.**, канд. техн. наук  
Донской государственный технический университет,  
г. Ростов-на-Дону, Россия

## **МОДЕЛИРОВАНИЕ БИЗНЕС-ПРОЦЕССОВ ИНСТРУМЕНТАМИ РАЗЛИЧНЫХ КОМПЬЮТЕРНЫХ СРЕДСТВ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Моделирование бизнес-процессов – это графическое представление бизнес-процессов или рабочих процессов компании в качестве средства выявления потенциальных улучшений [1, с. 37-38]. Обычно это делается с помощью различных графических методов, таких как блок-схема, диаграмма потока данных и т. д.

Моделирование бизнес-процессов – это высокоэффективный метод, который предоставляет организациям широкий спектр преимуществ:

1. Нахождение областей, в которых необходимо произвести улучшения. Основное использование моделирования бизнес-процессов заключается в том, чтобы предоставить заинтересованным сторонам лучшее понимание того, как процесс работает с целью внедрения улучшений.

2. Прозрачность. Модель бизнес-процесса показывает, как предполагается выполнять задачи, кто за них отвечает и как процесс способствует достижению бизнес-цели. Это играет важную роль в повышении доверия и подотчетности во всей организации.

3. Ловкость и гибкость. Бизнес-цели и стратегии могут измениться в одно мгновение – под влиянием различных факторов, например, COVID-19. С помощью моделирования бизнес-процессов заинтересованные стороны могут сразу же определить и внедрить улучшения, соответствующие новым целям.

4. Стандартизация по всем отделам. Многие процессы в организации, особенно в более крупных, включают в себя аналогичные шаги и задачи. Например, запросы на заказ на покупку часто подаются на уровне единицы, а не на уровне предприятия. Используя модели процессов, заинтересованные стороны могут выявлять передовую практику во всех подразделениях для внедрения эффективных процедур в рамках всей организации.

Существует множество способов моделирования бизнес-процессов, более 12 различных техник. Рассмотрим наиболее используемые из них:

1. Business Process Model and Notation (BPMN) – это открытый отраслевой стандарт для моделирования бизнес-процессов, разработанный компанией BPMI (Business Process Management Initiative). По сути, это стандартизированный метод отображения блок-схем – используемые объекты определяются методологией, а не человеком, выполняющим отображение [1, с. 8-9].

BPMN может выполняться точно так же, как и блок-схемы, с той лишь разницей, что элементы используются внутри графика, определенные методологией BPMN, а не по собственным предпочтениям [7].

Модель процессов позволяет пользователям проектировать бизнес-процессы с помощью интуитивно понятной и простой в использовании функции перетаскивания. Пользователи могут перетаскивать задачи, а также точки принятия решений непосредственно на холст моделирования. Также можно добавить дополнительные элементы, такие как формы, пользователи и соединители данных. Также важно, чтобы пользователи могли протестировать свои новые технологические проекты. Это достигается с помощью механизма проверки процесса. Механизмы проверки процессов позволяют пользователям тестировать свои новые проекты процессов, чтобы убедиться, что они работают до развертывания.

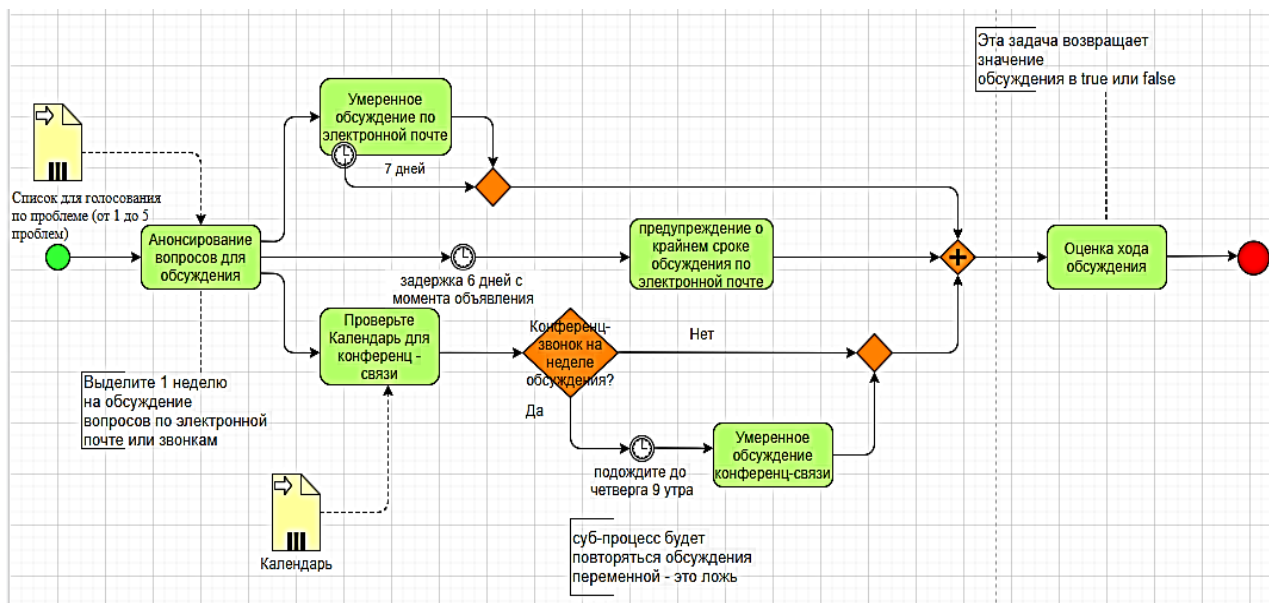


Рис. 1. Business Process Model and Notation

2. Camunda BPM – это платформа автоматизации рабочих процессов и принятия решений с открытым исходным кодом [2, с. 199].

Платформа разработана на Java и выпущена в виде программного обеспечения с открытым исходным кодом на условиях лицензии Apache.

Платформа предоставляет модель бизнес-процессов и нотацию (BPMN), совместимую со стандартом workflow engine, а также модель принятия решений и нотацию (DMN), совместимую со стандартом decision engine, которые могут быть встроены в приложения Java и с другими языками через REST.

Приложение Camunda Modeler позволяет разработчикам создавать и редактировать диаграммы процессов BPMN и таблицы решений DMN, созданные файлы развертываются в движках Camunda, которые используют анализатор BPMN для преобразования XML-файлов BPMN 2.0 и XML-файлов DMN в объекты Java и реализуют конструкции BPMN 2.0 с набором реализаций поведения BPMN.

Работа платформы Camunda на примере процесса «Обработки платежей» представлена на рисунке 2.

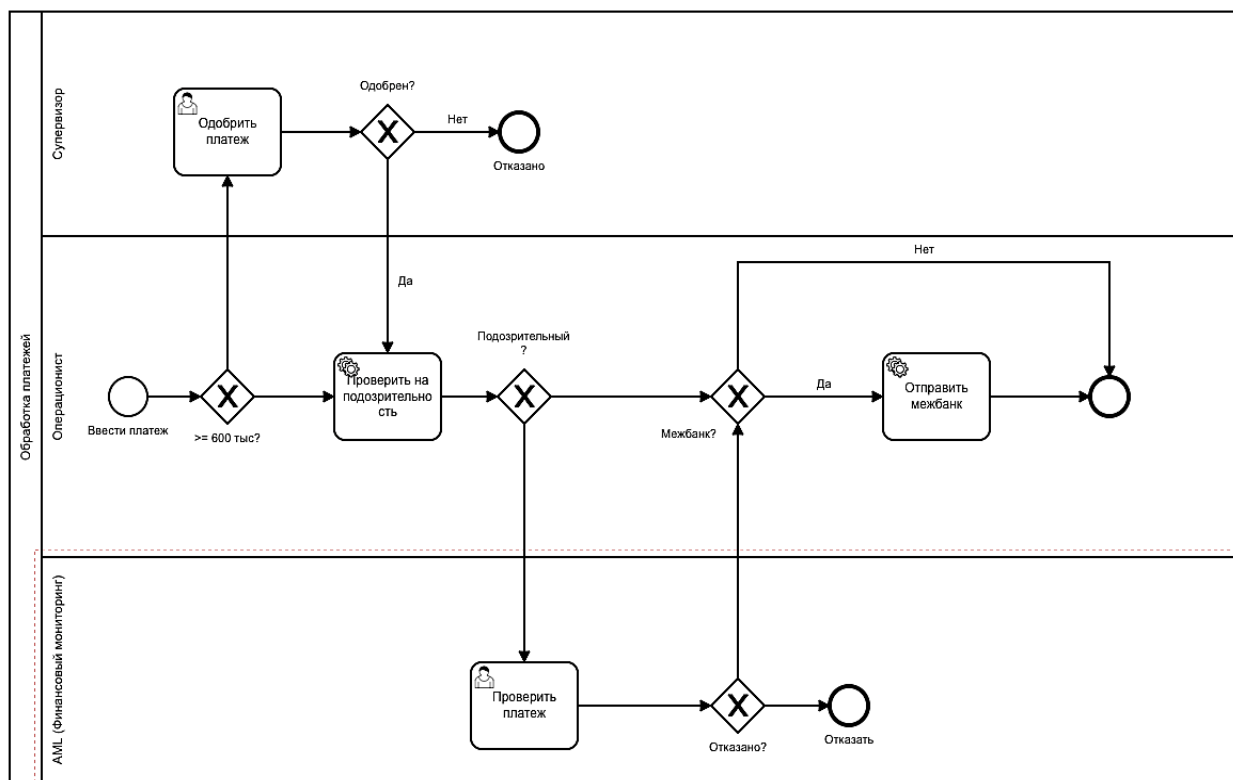


Рис. 2. Процесс «Обработка платежей»

3. ELMA реализует концепцию BPM (Business Process Management), позволяющую разрабатывать адаптируемые информационные системы, способные быстро изменяться вместе с бизнес-процессами компании.

ELMA Community Edition – позволяет создавать диаграммы процессов и преобразовывать их в персонализированные приложения BPM. Он также функционирует как виртуальный канал для пользователей и экспертов, где они могут общаться и обсуждать друг с другом идеи, такие как формулирование лучших практик BPM [6].

Она включает в себя модули CRM и ECM, которые позволяют пользователям проектировать и запускать бизнес-процессы в настраиваемых удобных интерфейсах. Они также могут делегировать и реализовывать задачи и учитывать оплачиваемые часы работы. Пользователи также могут управлять базой данных подрядчиков, планировать встречи, звонки и события, а также выполнять планирование продаж. ELMA Community Edition также облегчает организацию электронного документооборота, а также автоматизацию документооборота.

Прозрачность легко достигается с помощью ELMA Community Edition. Вы находитесь на вершине всего и можете легко видеть, какие задачи выполнены, а какие еще не выполнены. Если вы хотите проверить, на какой стадии исполнения находится бизнес-процесс в данный момент, все, что вам нужно сделать, – это взглянуть. Подотчетность также усиливается,

поскольку вы знаете, кто отвечает за задачу бизнес-процесса. Выявив узкие места, вы можете легко решить эту проблему и еще больше усовершенствовать процесс.

Дизайнер ELMA представлен на рисунке 3.

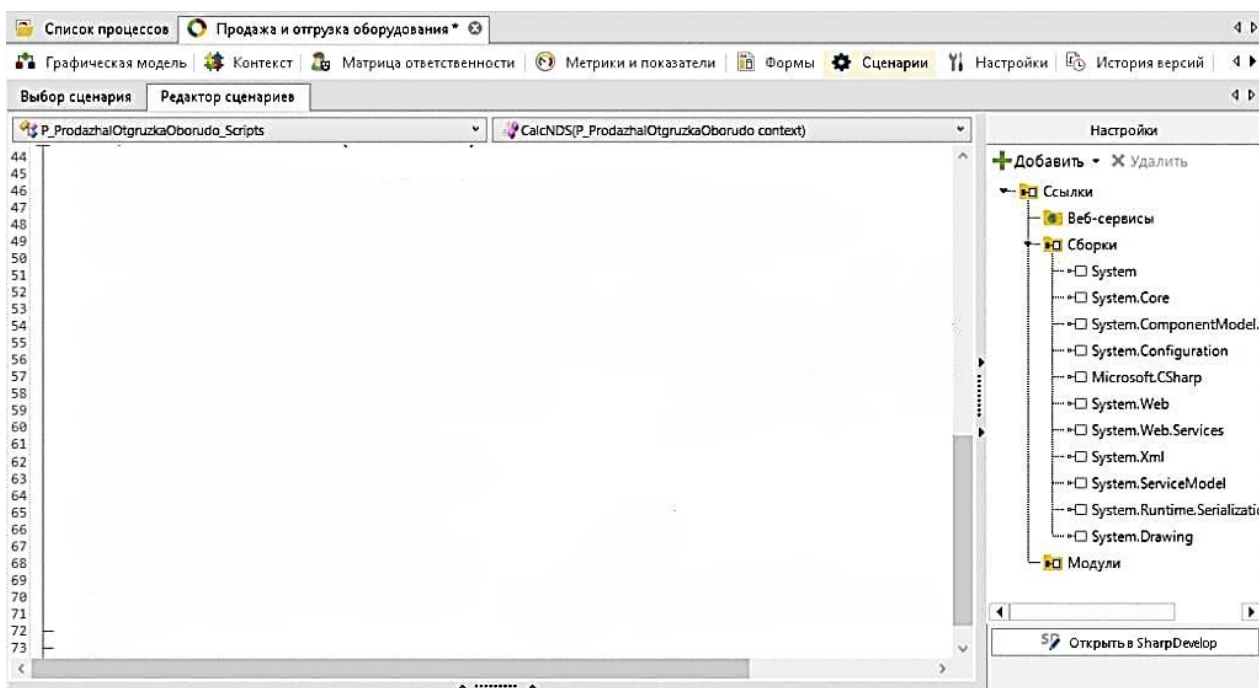


Рис. 3. Дизайнер ELMA

Карточка процесса включает соответствующие сегменты [6]:

Графичная форма – включает явное графичное понимание хода по согласованию с эталоном BPMN.

Контекст – предметы также сведения, применяемые в рамках исполнения хода.

Матрица ответственности – представляет собой список всех участников проекта и их роли.

Метрики и показатели – измеримые характеристики хода либо его экземпляров.

Формы – основанные с поддержкой конструктора фигуры, применяемые в рамках экземпляров нынешнего проекта.

Сценарии – программные коды, разрешающие автоматизировать определенные рубежи проекта.

История версий – перечень предшествующих версий проекта, которые возможно использовать также как действующий процесс.

Деятельность со справочниками «1С: Предприятие» в системе ELMA выполняется прямо для пользователя, в веб-интерфейсе концепции они смотрятся четко, таким образом, как справочники концепции ELMA. Абонент способен смотреть их содержание, расширять новейшими записями и редактировать ранее имеющиеся. Все без исключения перемены уже после выполнения подобных трудов станут отражаться в системе «1С: Предприятие».

Библиотеки интеграции с различными ERP-концепции, которые разработаны для ELMA BPM: 1C, SAP, Oracle, MS Axapta.



ELMA способен функционировать в содействии с самыми разными приложениями. Связь может осуществляться через веб-сервисы, через API либо другими методами. Интеграция ELMA BPM с системами «1С: Предприятие» обладает особенное значимость и отдельную реализацию. С Целью комфортного и эффективного взаимодействия концепции ELMA и «1С: Предприятие» применяется в отдельности лицензируемый вспомогательный узел «Интеграция с 1С».

Объединение концепции ELMA с системой «1С: Предприятие» дает возможность:

1. Функционировать с справочниками и документами концепции «1С: Предприятие».
2. Включать функции концепции «1С: Предприятие».
3. Прибегать к объектной модификации концепции «1С: Предприятие».

Кроме того, вследствие способностей ELMA возможно реализовывать пуск предпринимательских действий и осуществлять отсылку уведомлений в систему ELMA из системы «1С: Предприятие».

Скрипты можно писать привычным для 1С способом - пример написания скрипта представлен на рисунке 4.

```
44 {
45
46     var service = Locator.GetServiceNotNull<Integration1CService>();
47     ComObject connector = service.GetComConnector("ConfigName");
48     dynamic bookRef = connector.GetFunctionValue("NewObject", "CatalogManager.Контрагенты");
49     var newObject = bookRef.СоздатьЭлемент();
50     newObject.Наименование = "Название контрагента";
51     newObject.Записать();
52
53     //+++++++ Юридический адрес контрагента ++++++++
54     dynamic РегистрЮКМ = connector.GetFunctionValue("NewObject", "InformationRegisterManager.КонтактнаяИнфор
55     var РегистрЮКМ = РегистрЮКМ.СоздатьМенеджерЗаписи();
56     РегистрЮК.Объект = newObject.Ссылка;
57     dynamic Тип = connector.GetFunctionValue("NewObject", "EnumManager.ТипыКонтактнойИнформации"); //перечислени
58     РегистрЮК.Тип = Тип.Адрес;
59     dynamic Вид = connector.GetFunctionValue("NewObject", "CatalogManager.ВидыКонтактнойИнформации");
60     РегистрЮК.Вид = Вид.ЮрАдресКонтрагента;
61     РегистрЮК.Поле1 = "123456"; //Индекс
62     //РегистрЮК.Поле2 = ""; //Регион
63     РегистрЮК.Поле3 = "ур"; //Район
64     РегистрЮК.Поле4 = "Ижевск"; //Город
65     //РегистрЮК.Поле5 = ""; //Населенный пункт
66     РегистрЮК.Поле6 = "Удмуртская"; //Улица
67     РегистрЮК.Поле7 = "304"; //Дом
68     //РегистрЮК.Поле8 = ""; //Корпус
69     РегистрЮК.Поле9 = context.Kvartira.ToString(); //Квартира
70     РегистрЮК.Представление = РегистрЮК.Поле4 + ", " + " " + РегистрЮК.Поле6 + ", " + " " + РегистрЮК.Поле7;
71     РегистрЮК.Записать();
72 }
73 }
```

Рис. 4. Написание скрипта при интеграции 1с и Elma

Таким образом, программное обеспечение для моделирования бизнес-процессов создает документацию и графическую карту того, как бизнес работает сегодня, с подробным описанием задач, ответственных сторон и элементов данных, участвующих в процессах, а также взаимодействий, которые происходят в системах, процедурах и организационных иерархиях [4, с. 70].



Понимание системного взаимодействия, бизнес-процессов и организационных иерархий обеспечивает согласованность, при которой все движутся в одном направлении, и поддерживает принятие обоснованных решений для достижения оптимальных результатов и постоянного улучшения.

### Литература

1. Джестон Д., Нелис Й. Управление бизнес-процессами. Практическое руководство по успешной реализации проектов. М., 2015. 512 с.
2. Долганова О.И. Виноградова Е.В., Лобанова А.М. Моделирование бизнес-процессов: Учебник и практикум для академического бакалавриата. Люберцы, 2016. 289 с.
3. Громов А.И., Фляйшман А., Шмидт В. Управление бизнес-процессами: современные методы. Люберцы, 2016. 367 с.
4. Репин В.В., Елиферов В.Г. Процессный подход к управлению. Моделирование бизнес-процессов. М., 2013. 544 с.
5. Шеер А.-В. Бизнес-процессы. Основные понятия. Теория. Методы. М., 2019. 2-е изд., перераб. и доп. 152 с.
6. ELMA – система управления бизнес-процессами и эффективностью. <https://www.elma-bpm.ru/>
7. Терехов А.Н., Платонова М.В. Моделирование бизнес-процессов в цифровую эпоху // Российский журнал менеджмента. №17(4). С. 487-498 СПб., 2019. <https://doi.org/10.21638/spbu18.2019.404>

© Корженко М.А., Кузьменко Т.А., Кондратьева Т.Н., 2021

УДК 51-77

Закутняя Л.А., Антипова Е.С.

Государственный университет управления,  
г. Москва, Россия

## МОДЕЛЬ КОНКУРЕНТНОЙ БОРЬБЫ ДВУХ ИДЕОЛОГИЙ

Рассмотрена модель конкуренции двух непримиримых идеологий. Модель предполагает существование в обществе одновременно двух взаимоисключающих идеологий и приводит к уравнениям Лотки-Вольтерры. Проведен анализ модели и показано, что в зависимости от параметров рассматриваемой модели возможны различные состояния рассматриваемой системы: вырождение и исчезновение одновременно обеих идеологий или их сосуществование. В последнем случае реализуется либо асимптотически устойчивое стационарное состояние системы (аттрактор), когда число сторонников обеих идеологий не изменяется, либо колебательное изменение сторонников обеих идеологий, происходящее в противофазе, которому соответствует устойчивый предельный цикл. Рассмотрена интерпретация различных решений рассмотренной модели и приведены примеры, соответствующие разным состояниям системы.

В настоящее время для описания динамических процессов, происходящих в больших сообществах, широко используется модель Лотки-Вольтерры и ее обобщения. Это относится к моделированию различных процессов в биологии, экономике, социальных и политических науках, а также в других междисциплинарных направлениях [1, с. 76]. Так, например, уравнения Лотки-Вольтерры лежат в основе различных экологических моделей, моделей классовой борьбы, моделей распространения вирусных инфекций и тому подобное [2, с. 69]. Известны обобщения уравнений Лотки-Вольтерры с учетом миграции, когда рассматриваются два конкурирующих сообщества (например, «хищник-жертва»), заселяющие соседние регионы, между которыми возможна миграция хищника [3, с. 5]. Нам показалось интересным, применить модель Лотки-Вольтерры для описания конкуренции между двумя идеологиями.

Имеются две непримиримые и конкурирующие идеологии, так, что приверженцы одной идеологии не могут одновременно быть приверженцами другой. Под идеологией будем понимать некоторую систему в сообществе с большим количеством людей, объединенных общей идеей. Примерами непримиримых идеологий могут служить, фашизм и коммунизм, коммунизм и капитализм, теория Дарвина и божественное происхождение.

Пусть в рассматриваемом сообществе существует две идеологии. Имеется  $x$  приверженцев первой идеологии и  $y$  приверженцев второй идеологии. Кроме того, будем считать, что в данном сообществе имеется большое количество людей, которые не разделяют ни одну из этих идеологий, при этом своей идеологии у них нет. Обычно таких людей в обществе большинство. Будем считать, что людей, не разделяющих ни одну из этих идеологий, бесконечное множество (на практике их много больше, чем  $x + y$ ). Множество людей, не разделяющих никакой идеологии, будем называть континуумом. Из континуума

люди могут переходить в сообщество с первой или второй идеологией, то есть становятся приверженцами одной из существующих в сообществе идеологий.

В таком сообществе будет происходить конкурентная борьба между существующими идеологиями, в которой представители разных идеологий будут стремиться привлечь на свою сторону новых приверженцев (адептов).

Процесс происходит следующим образом: представители разных идеологий ведут пропагандистскую работу среди представителей противоположной идеологии и среди континуума. В результате этой работы представители противоположной идеологии могут изменить свое мнение и отказаться от своей идеологии. В этом случае они либо переходят в стан представителей противоположной идеологии, либо, вообще отказываются от этих идеологий и переходят в континуум. Кроме того, представители любой идеологии могут спонтанно (даже без общения с представителями противоположной идеологии) отказаться от своих убеждений и либо перейти в стан противоположной идеологии, либо перейти в континуум. Примером описанного выше процесса могут служить предвыборные кампании политических партий, участвующих в выборах и призывающие голосовать за них. В зависимости от цели, агитации, лидера кампании люди могут либо присоединиться к конкретной партии, либо, наоборот, выйти из нее, причем необязательно перейти в противоположную партию, а просто решить вообще ни за кого не голосовать на грядущих выборах или даже не идти на них. Принять такое решение могут не только те, кто общается с представителями политической партии или следит за ее работой. В силу определенных обстоятельств – загруженность работой, некогда интересоваться политикой, поменялись интересы, против всех партий одновременно – происходит спонтанный переход из одной идеологии в другую или выход из всех идеологий, то есть переход в континуум.

Эти процессы условно показаны на рисунке 1.

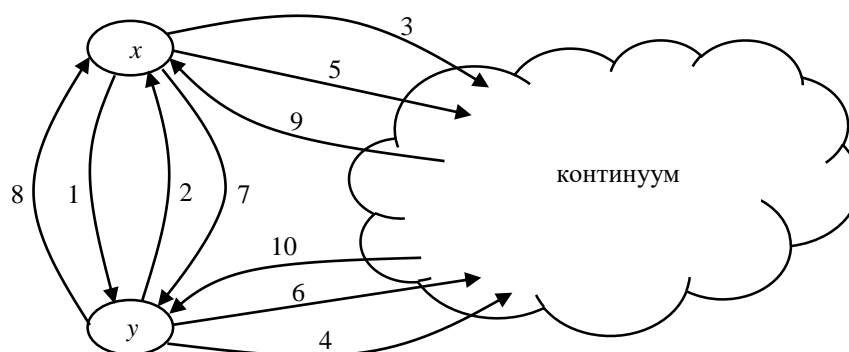


Рис. 1. Условная схема конкурентной борьбы двух идеологий. Цифрами обозначены различные процессы, поясненные далее в тексте

Рассмотрим каждый из процессов, изображенных на рисунке 1.

В процессе общения представителей идеологий  $x$  и  $y$  каждый из них будет убеждать противоположную сторону в ложности его идеологии, в том, что только его идеология является единственно правильной и верной. В результате этого некоторые представители идеологий  $x$  и  $y$  откажутся от своей идеологии. В результате произойдут процессы (1) – (4).

(1) Некоторые представители идеологии  $x$  перейдут в идеологию  $y$ . Скорость, с которой представители идеологии  $x$  в результате этого процесса будут переходить в идеологию  $y$ , равна  $\beta_x xy$ , где  $\beta_x > 0$  – некоторый параметр, который характеризует частоту встреч представителей разных идеологий и вероятность того, что в результате этих встреч представитель идеологии  $x$  перейдет в идеологию  $y$ .

(2) Некоторые представители идеологии  $y$  перейдут в идеологию  $x$ . Скорость, с которой представители идеологии  $y$  в результате этого процесса будут переходить в идеологию  $x$ , равна  $\beta_y xy$ , где  $\beta_y > 0$  – некоторый параметр, который характеризует частоту встреч представителей разных идеологий и вероятность того, что в результате этих встреч представитель идеологии  $y$  перейдет в идеологию  $x$ .

(3) Некоторые представители идеологии  $x$  откажутся от любых идеологий и перейдут в континуум. Скорость, с которой представители идеологии  $x$  в результате этого процесса будут переходить в континуум, равна  $\sigma_x xy$ , где  $\sigma_x > 0$  – некоторый параметр, который характеризует частоту встреч представителей разных идеологий и вероятность того, что в результате этих встреч представитель идеологии  $x$  перейдет в континуум.

(4) Некоторые представители идеологии  $y$  перейдут в континуум. Скорость, с которой представители идеологии  $y$  в результате этого процесса будут переходить в континуум, равна  $\sigma_y xy$ , где  $\sigma_y > 0$  – некоторый параметр, который характеризует частоту встреч представителей разных идеологий и вероятность того, что в результате этих встреч представитель идеологии  $y$  перейдет в континуум.

Представители разных идеологий могут спонтанно (самопроизвольно, без воздействия представителей противоположной идеологии) отказаться от любой идеологии и перейти в континуум. В результате произойдут процессы (5) – (10).

(5) Спонтанный переход представителя идеологии  $x$  в континуум. Скорость этого процесса равна  $\alpha_x x$ , где  $\alpha_x > 0$  – некоторый параметр, равный вероятности того, что представитель идеологии  $x$  за единицу времени спонтанно перейдет в континуум.

(6) Спонтанный переход представителя идеологии  $y$  в континуум. Скорость этого процесса равна  $\alpha_y y$ , где  $\alpha_y > 0$  – некоторый параметр, равный вероятности того, что представитель идеологии  $y$  за единицу времени спонтанно перейдет в континуум.

(7) Спонтанный переход представителя идеологии  $x$  в идеологию  $y$ . Скорость этого процесса равна  $\gamma_x x$ , где  $\gamma_x > 0$  – некоторый параметр, равный вероятности того, что представитель идеологии  $x$  за единицу времени спонтанно перейдет в идеологию  $y$ .

(8) Спонтанный переход представителя идеологии  $y$  в идеологию  $x$ . Скорость этого процесса равна  $\gamma_y y$ , где  $\gamma_y > 0$  – некоторый параметр, равный вероятности того, что представитель идеологии  $y$  за единицу времени спонтанно перейдет в идеологию  $x$ .

(9) Спонтанный переход из континуума в идеологию  $x$ . Скорость этого процесса равна  $\mu_x x$ , где  $\mu_x > 0$  – некоторый параметр, равный вероятности того, что субъект из континуума за единицу времени спонтанно перейдет в идеологию  $x$ .

(10) Спонтанный переход из континуума в идеологию  $y$ . Скорость этого процесса равна  $\mu_y y$ , где  $\mu_y > 0$  – некоторый параметр, равный вероятности того, что субъект из континуума за единицу времени спонтанно перейдет в идеологию  $y$ .

Это позволяет записать уравнения для  $x$  и  $y$ :

$$\frac{dx}{dt} = (\mu_x - \alpha_x - \gamma_x)x + \gamma_y y - (\beta_x - \beta_y + \sigma_x)xy \quad (1)$$

$$\frac{dy}{dt} = \gamma_x x + (\mu_y - \alpha_y - \gamma_y)y + (\beta_x - \beta_y - \sigma_y)xy \quad (2)$$

Обозначая

$$a = (\mu_x - \alpha_x - \gamma_x), b = (\beta_x - \beta_y + \sigma_x), g = (\mu_y - \alpha_y - \gamma_y), r = (\beta_x - \beta_y - \sigma_y), \lambda = \gamma_y,$$

$$v = \gamma_x \quad (3)$$

запишем уравнения (1) и (2) в виде

$$\frac{dx}{dt} = ax + \lambda y - bxy \quad (4)$$

$$\frac{dy}{dt} = vx + gy + rxy \quad (5)$$

В общем случае параметры  $a, b, g$  и  $r$  могут быть, как положительными, так и отрицательными. Параметры  $\lambda, v \geq 0$  по условию.

Мы получили систему уравнений (4) и (5), описывающую конкурентную борьбу двух идеологий. Система

$$\begin{cases} \frac{dx}{dt} = ax + \lambda y - bxy \\ \frac{dy}{dt} = vx + gy + rxy \end{cases} \quad (6)$$

относится к типу уравнений Вольтерра.

При  $\lambda = v = 0$ ,  $a, b, r > 0$  и  $g < 0$  уравнения (4) и (5) переходят в обычные уравнения Лотки-Вольтерры для системы хищник-жертва [4, с. 59; 5, с. 9].

Рассмотрим параметры  $a, b, g, r \in \mathbb{R}$ ,  $\lambda, v \geq 0$ . В зависимости от параметров  $a, b, g, r, \lambda$  и  $v$  система уравнений (6) может иметь принципиально разные решения. Система уравнений (6) имеет две стационарные точки:

$$(x; y) = (0; 0) \quad (7)$$

$$(x; y) = \left( \frac{\lambda v - ga}{ar + bv}; \frac{ga - \lambda v}{r\lambda + gb} \right) \text{ при } ar + bv \neq 0, r\lambda + gb \neq 0, \quad (8)$$

которые при определенных параметрах  $a, b, g, r, \lambda, v$  являются устойчивыми.

Так как  $x, y > 0$  то получим условия:



$$\frac{\lambda v - ga}{ar + bv} > 0; \frac{ga - \lambda v}{r\lambda + gb} > 0 \quad (9)$$

Рассмотрим каждый из случаев (7), (8).

Стационарная точка  $(x; y) = (0; 0)$  является устойчивой при условиях:

$$a < 0, g < 0, ag \geq \lambda v, \lambda > 0, v > 0 \quad (10)$$

Условия (10) показывают, что большее число субъектов уходит из идеологий в континуум, чем приходит в данные идеологии. В этом случае получаем исчезновение (вымирание) обеих идеологий.

На рисунке 2 показано, как распадаются идеологии. Появившиеся идеологии не выживают, а новые идеологии не возникают, остается только континуум.

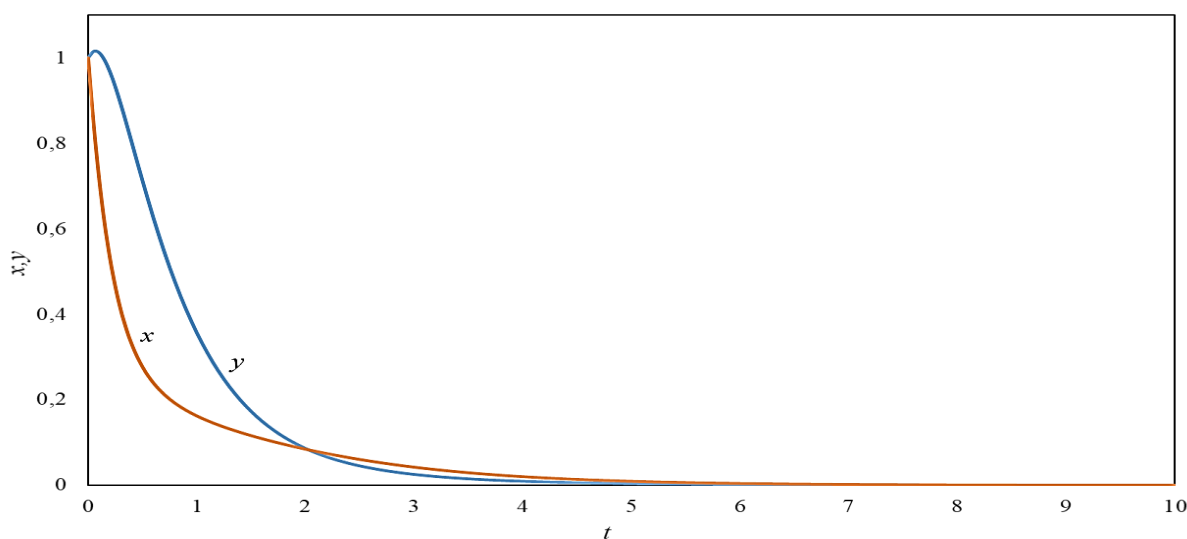


Рис. 2. Распад идеологий при  $a = -1; g = -2; r = 2; v = 0,5; b = 3; \lambda = 0,6$

Примером распада идеологий могут служить секты, субкультуры и т.п., которые не смогли выжить в конкурентной борьбе друг с другом и с континуумом.

Рассмотрим вторую стационарную точку (8).

При выполнении условий (9), (11) или (9), (12)

$$\left\{ \begin{array}{l} ag < 0 \\ \left(\lambda v - \frac{v}{\varphi}\right)^2 \geq -4ag \\ \frac{\lambda v^2}{\varphi} > ag \\ 0 < \lambda < -\frac{1}{\varphi} \\ \varphi = \frac{ar + bv}{\lambda r + bg} < 0 \\ v > 0 \end{array} \right. \quad (11)$$

или

$$\left\{ \begin{array}{l} ag < 0 \\ \left(\lambda v - \frac{v}{\varphi}\right)^2 < -4ag \\ 0 < \lambda < -\frac{1}{\varphi} \\ \varphi = \frac{ar + bv}{\lambda r + bg} < 0 \\ v > 0 \end{array} \right. \quad (12)$$

обе идеологии устойчиво существуют и представляют собой аттрактор рассматриваемой системы.

При выполнении условий (11) получаем собственные значения  $\alpha_{1,2} \in \mathbb{R}$  и  $\alpha_{1,2} < 0$ . В этом случае аттрактор является устойчивым узлом. При выполнении условий (12) собственные значения  $\alpha_{1,2} \in \mathbb{C}$  и  $\text{Re}\alpha_i < 0, i = 1, 2$ , аттрактор – устойчивый фокус.

При устойчивом стационарном состоянии в каждую из идеологий будет приходиться и уходить приблизительно одинаковое число людей – приверженцев идеологий. На рисунке 3 можно видеть возрастание, а затем выравнивание (стабилизацию) состояний двух идеологий. В начале, когда идеология является новой, она вызывает интерес, число сторонников возникшей идеологии быстро растет - всплеск на графике, но с течением времени происходит спад. Те, кто является яркими сторонниками своей идеологии остаются, другая часть уходит в континуум и происходит стабилизация.

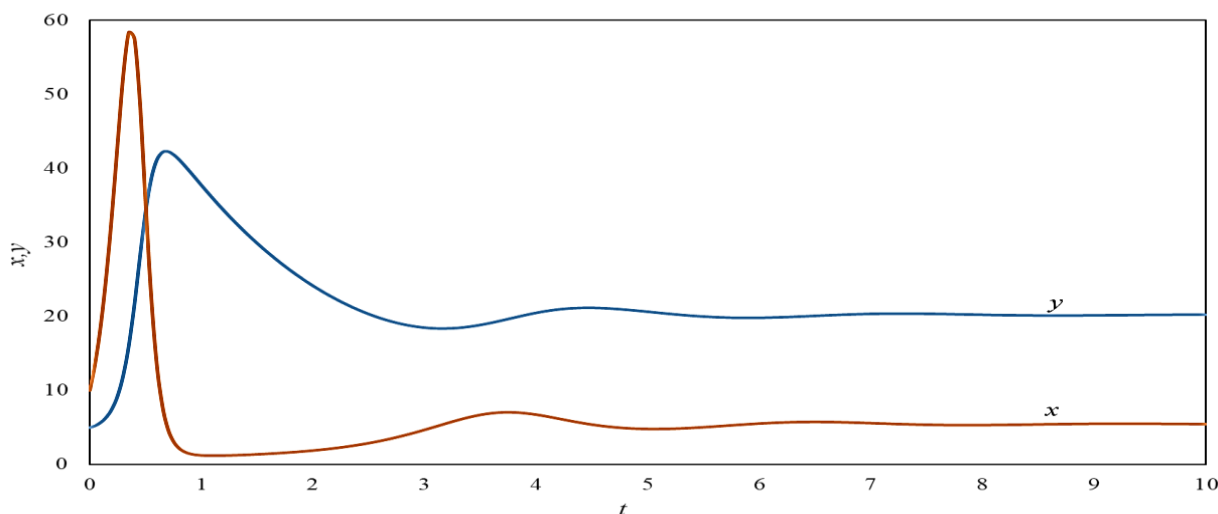


Рис. 3. Переход идеологий в стационарное состояние  
при  $a = 9; g = -0,6; r = 0,1; v = 0,2; b = 0,5; \lambda = 0,3$

Примером идеологий, находящихся в устойчивом состоянии, служат политические партии, число сторонников которых не меняется в течение длительного времени.

При параметрическом исследовании получено нестационарное решение (8) с двумя сосуществующими идеологиями, которое возникает только, когда параметры  $v = 0, \lambda = 0$ . В этом случае возникают колебания – рост числа приверженцев идеологии  $x$  и уменьшение числа сторонников идеологии  $y$ , затем, наоборот, рост числа приверженцев идеологии  $y$  и уменьшение числа сторонников идеологии  $x$  (рис. 4).

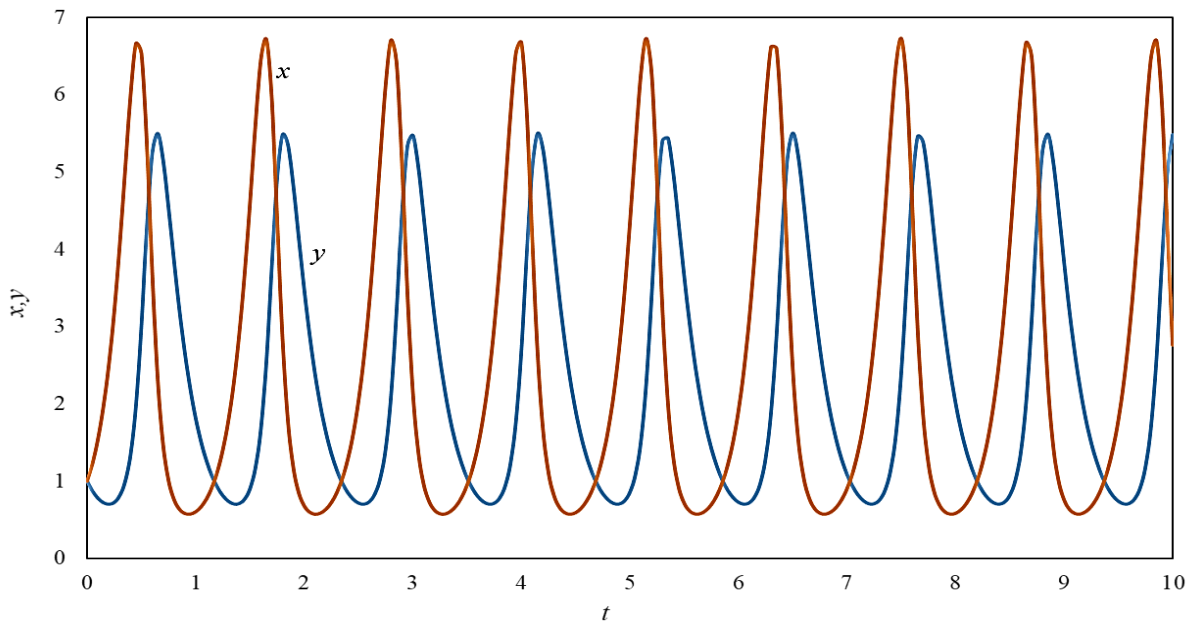


Рис. 4. Колебательный процесс при  $a = 7$ ;  $g = -5$ ;  $r = 2$ ;  $v = 0$ ;  $b = 3$ ;  $\lambda = 0$

Условия  $v = 0, \lambda = 0$  означают, что в полученной модели не существует спонтанных переходов из одной идеологии в другую. Спонтанные переходы в идеологии могут осуществляться только из континуума.

Примером конфликта идеологий является конкурентная борьба лидеров США Дональда Трампа и Джо Байдена на президентских выборах в 2020 году, когда каждый из лидеров пытался всеми средствами склонить избирателей на свою сторону – привлечь новых приверженцев из тех, кто еще не примкнул ни к одной из идеологий.

Таким образом, построена модель конкурентной борьбы двух идеологий. Конкурентная борьба идеологий существует всегда. Даже, когда идеологии находится в устойчивом состоянии, продолжается борьба за адептов. Уравнения, полученные в модели, подобны уравнениям Лотки-Вольтерры. В результате исследования полученных уравнений были сделаны выводы: при определенных параметрах две идеологии могут сосуществовать одновременно, при этом могут находиться в стационарном состоянии либо в режиме колебаний. В последнем случае колебания числа сторонников идеологий происходят в противофазе, причем в полученной модели это возможно, если спонтанные переходы в идеологии осуществляются только из континуума. При других параметрах обе возникшие идеологии могут вымирать, в этом случае в рассмотренной модели остается только континуум.

### Литература

1. Малков С.Ю. Математическое моделирование исторической динамики: подходы и модели // Моделирование социальной и экономической динамики. 2004. С. 76-188.
2. Трубецков Д.И. Феномен математической модели Лотки-Вольтерры и сходных с ней // Известия вузов «ПНД». 2011. Т. 19. №2. С. 69-88.

3. Курилова Е.В., Кулаков М.П. Условия синхронизации численности двух взаимосвязанных сообществ (на основе модели Лотки-Вольтерра) // Региональные проблемы. 2014. Т. 17. №1. С. 5-8.

4. Robert M. May Stability in Multispecies Community Models // Mathematical biosciences. 1971. №12. P. 59-79.

5. Mira-Cristiana Anisiu Lotka, Volterra and their model // Didactica mathematica. 2014. №32. P. 9–17.

© Закутняя Л.А., Антипова Е.С., 2021

УДК 004.75

**Домнина М.В.**

*Научный руководитель:* **Меджидов З.У.**, канд. экон. наук  
Дагестанский государственный университет народного хозяйства, г. Махачкала, Россия

## РЕАЛИЗАЦИЯ МНОГОПОТОЧНОГО СЕРВЕРНОГО ПРИЛОЖЕНИЯ НА ПЛАТФОРМЕ .NET CORE

В статье рассматривается способ реализации многопоточного серверного приложения на платформе .net core по протоколу передачи данных tcp с использованием базы данных mysql [1].

В бизнес-сегменте все чаще требуется создания серверных решений. Логика обработки запросов не должна находиться у пользователя, так как это может нанести удар по безопасности и раскрыть конфиденциальную информацию компании.

В данной статье код написан на языке программирования с# и используется платформа .net core, которая является преемником технологии .net framework и базируется на ней. Основное ее преимущество – это кроссплатформенность, что позволяет писать приложения для различных ОС [2].

Так как в статье рассматривается сервер, то кроссплатформенность играет существенную роль в выборе технологии для разработки. Большой сегмент операционных систем для серверов занимает Linux в виду открытого исходного кода и бесплатного коммерческого использования [3].

Реализация серверной части. Структура сервера состоит из четырех классов, а именно:

1. Класс “Server.cs”.
2. Класс “User.cs”.
3. Класс “Commands.cs”.
4. Класс “SQL.cs”

Server.cs. Основной класс, является точкой входа в приложение. В задачи входит: установка соединения с базой данных, с пользователем, обработка запросов и создание отдельного потока для нового пользователя. Отдельный поток необходим для выполнения кода без необходимости приостановки основного потока [4].

Для работы класса используются: System.Net и System.Net.Sockets для обеспечения сетевой работы, и System.Threading для работы с многопоточностью.

Сервер обрабатывает подключения по определенному IP адресу и порту, в нашем случае адрес прослушивания является адресом самого сервера, то есть локальный IP, а именно «127.0.0.1», порт выберем “8888”.66

```
1. const int port = 8888;  
2. const string ip = «127.0.0.1»;
```



Далее объявляется экземпляр класса TcpListener с параметрами IP и порта, и начинается прослушивание входящих запросов на подключение.

```
1. listener = new TcpListener(IPAddress.Parse(ip), port);  
2. listener.Start();
```

Далее в бесконечном цикле принимаются входящие подключения пользователей, создается новый экземпляр класса User, который обрабатывает уже подключившегося пользователя и заносит его в новый поток, попутно начиная его работу.

```
1. while (true)  
2. {  
3.     TcpClient client = listener.AcceptTcpClient();  
4.     User clientObject = new User(client);  
5.     Console.WriteLine(«Подключился: « + ((IPEnd-  
6.         Point)client.Client.RemoteEndPoint).Address.ToString());  
7.     //Создаем поток для обслуживания нового клиента  
8.     Thread clientThread = new Thread(new Thread-  
9.         Start(clientObject.UserStart));  
10.    clientThread.Start();  
11. }
```

User.cs. В отличие от предыдущего класса работает уже с конкретным пользователем. Основные задачи: прием запросов от пользователя и передача его на обработку, и закрытие соединения с пользователем.

Необходимо задать структуру UserData, которая содержит свойства подключившегося пользователя. Для примера она содержит единственное свойство isAuth, содержащее информацию о том, авторизовался ли пользователь или нет.

```
1. public struct UserData  
2. {  
3.     public bool isAuth;  
4. }
```

Чтобы обрабатывать запросы конкретного пользователя требуется бесконечное прослушивание запросов клиента к серверу. Далее из класса Commands вызывается статическая функция UserRequest, в которую передается поток и ссылка на структуру.

```
1. while (true)  
2.     Commands.UserRequest(stream, ref userInfo);
```

Commands.cs. Данный класс служит для обработки всех запросов пользователя, кроме sql запросов, для них из него вызываются методы из класса “SQL”, а также ответа пользователю.

Функция UserRequest обрабатывает ответ пользователя. Так как передача TCP пакетов происходит в байтовом представлении, то поток считывает в заранее объявленный байт массив, буфер, размером 512 байт. Это выполняет функция stream.Read, которая считывает полученные от клиента байты в буфер и возвращает длину полученного сообщения, которую записывают в переменную bytes типа int для последующего отсекания лишних нулевых байт [5].

Заранее известно, что клиент первых байтом пошлет номер команды, а дальнейшие будут сообщением к команде. Команда и сообщение преобразуются из массива полученных байт. Функция `UserRequest` может пропустить для дальнейшей обработки команды без авторизации подключенного клиента только запрос на авторизацию или регистрацию, так как команда 1 это запрос на авторизацию, а команда 2 на регистрацию, при выполнении этих условий команда и сообщение посылаются на дальнейшую обработку в функцию `ExecuteCommand`.

```
1. public static void UserRequest(NetworkStream stream, ref User.UserData
   userData)
2. {
3.     byte[] buffer = new byte[512];
4.
5.     int bytes = stream.Read(buffer, 0, buffer.Length);
6.     Array.Resize(ref buffer, bytes);
7.
8.     int command = Convert.ToInt16(buffer[0]);
9.     string message = Encoding.Unicode.GetString(buffer, 1, bytes - 1);
10.
11.    if (command == 1 || command == 2 || userData.isAuth)
12.        ExecuteCommand(stream, command, message, ref userData);
13.
14.    if (message != «»)
15.    {
16.        Console.WriteLine(«command: « + command + « | message: « + message);
17.    }
18. }
```

Функция `SendData` отправляет пользователю обратно массив байт, причем в отличие от клиента отправляется только сообщение, без команды.

```
1. static void SendData(NetworkStream stream, byte[] data)
2. {
3.     stream.Write(data, 0, data.Length);
4. }
```

Функция `ExecuteCommand` обрабатывает команды. Между клиентом и сервером заранее известно о значении команд, а именно:

1. Команда «1» – авторизация на сервере.
2. Команда «2» – регистрация на сервере.
3. Команда «3» – получить время.

После получения на вход команды эта функция вызывает другую функцию, соответствующую назначению команды.

```
1. public static void ExecuteCommand(NetworkStream stream, int command,
   string message, ref User.UserData userData)
2. {
3.     if (command == 1)
4.         Authorization(stream, message, ref userData);
5.     else if (command == 2)
6.         Registration(stream, message);
7.     else if (command == 3)
8.         GetTimestamp(stream);
9. }
```

Функция Authorization получает на вход поток, сообщение от пользователя и ссылку на его структуру, чтобы в случае успешной авторизации изменить значение isAuth на true, это требуется для защиты от нежелательных команд от неавторизированных пользователей и позволит в дальнейшем выполнять запросы от пользователя кроме авторизации и регистрации, в нашем случае это запрос получения времени. Эта функция посылает запрос на создание нового пользователя в класс SQL в функцию SqlAuthorization на авторизацию и в случае успеха возвращает и передает обратно пользователю сообщение “Success” и меняет isAuth на true, в противном случае посылает пользователю «Failure» и isAuth не присваивает ничего.

```
1. public static void Authorization(NetworkStream stream, string message, ref
   User.UserData userData)
2. {
3.     string[] userAuthData = message.Split(«:»);
4.
5.     string passwordFromSql = SQL.SqlAuthorization(userAuthData[0]);
6.
7.     if(passwordFromSql != «» && userAuthData[1] == passwordFromSql)
8.     {
9.         userData.isAuth = true;
10.        SendData(stream, Encoding.Unicode.GetBytes(«Success»));
11.    }
12.    else
13.        SendData(stream, Encoding.Unicode.GetBytes(«Failure»));
14. }
```

Функция Registration регистрирует нового пользователя. В случае успеха возвращает и передает обратно пользователю сообщение “Success”, в противном случае посылает пользователю “Failure”.

```
1. public static void Registration(NetworkStream stream, string message)
2. {
3.     string[] userAuthData = message.Split(«:»);
4.
5.     bool state = SQL.SqlRegistration(userAuthData[0], userAuthData[1]);
6.
7.     if (state)
8.         SendData(stream, Encoding.Unicode.GetBytes(«Success»));
9.     else
10.        SendData(stream, Encoding.Unicode.GetBytes(«Failure»));
11. }
```

Функция GetTimestamp доступна только авторизированным пользователям и возвращает время в формате unix timestamp.

```
1. public static void GetTimestamp(NetworkStream stream)
2. {
3.     SendData(stream, BitConverter.GetBytes(DateTimeOffset.Now.ToUnixTimeSeconds()));
4. }
```

SQL.cs. Класс, содержащий функции, производящие sql запросы к базе данных. Для работы с базой данных используется библиотека MySql.Data.

Для функционирования sql модуля, необходимо иметь базу данных с полями:

1. id – тип int, уникальное поле, не равное нулю, auto increment
2. login – тип varchar

### 3. password – тип varchar

Перед тем как открыть подключение, необходимо указать параметры для установки соединения и передать их в MySqlConnection.

```
1. static string connection = «serv-  
er=localhost;user=root;database=server;password=1234»;  
2. static MySqlConnection sqlConnection = new MySqlConnection(connection);
```

Функция ConnectionToSql выполняет подключение к базе данных или разрыв соединения с ней.

```
1. public static string ConnectionToSql(string action)  
2. {  
3. try  
4. {  
5. if (action == «on»)
```

```
6. sqlConnection.Open();  
7. else if (action == «off»)  
8. sqlConnection.Close();  
9.  
10. return «Подключение к SQL серверу успешно»;  
11. }  
12. catch  
13. {  
14. return «Ошибка при подключении к SQL серверу»;  
15. }  
16. }
```

Функция SqlAuthorization проверяет есть ли пользователь с таким логином в базе данных, в случае успеха возвращает пароль.

```
1. public static string SqlAuthorization(string login)  
2. {  
3. string sqlQuery = «SELECT password FROM users WHERE login = '« + login +  
«'«»;  
4. MySqlCommand command = new MySqlCommand(sqlQuery, sqlConnection);  
5.  
6. var userPassword = command.ExecuteScalar();  
7.  
8. if (userPassword != null)  
9. userPassword = userPassword.ToString();  
10. else  
11. userPassword = «»;  
12.  
13. return userPassword.ToString();  
14. }
```

Функция SqlRegistration проверяет существует ли пользователь с таким логином в базе данных, в случае если записей нет, то создает новую, куда вносит логин и пароль.

```
1. public static bool SqlRegistration(string login, string password)
2. {
3.     string sqlQuery = «SELECT id FROM users WHERE login = '« + login + «'»;
4.     MySqlCommand command = new MySqlCommand(sqlQuery, sqlConnection);
5.
6.     var id = command.ExecuteScalar();
7.
8.     if (id == null)
9.     {
10.         sqlQuery = «INSERT INTO users (login, password) VALUES ('« + login +
11.             «', '« + password + «')»;
12.         command = new MySqlCommand(sqlQuery, sqlConnection);
13.         int count = command.ExecuteNonQuery();
14.         return true;
15.     }
16.     else
17.         return false;
18. }
```

Программное решение показало свою работоспособность и может являться основой для более крупных проектов. Серверные проекты являются перспективным направлением в области разработок ИТ продукции из-за увеличивающегося спроса на приложения, в частности мобильные, требующие обработку данных на сервере.

### Литература

1. Албахари Дж., Албахари Б. С# 7.0. Справочник. Полное описание языка. М., 2018. 1024 с.
2. Рихтер Д. CLR via C#. Programming on the Microsoft .NET Framework 4.5 in C#. 4th ed. М., 2017. 896 с.
3. Тахагхогхи С., Вильямс Х.Е. Руководство по MySQL. М., 2007. 544 с.
4. Троелсен Э., Джепикс Ф. Язык программирования С# 7 и платформы .NET и .NET Core. 2018. М.; СПб., 1328 с.
5. Шилдт Г. С# 4.0. Полное руководство. М., 2019. 1056 с.

© Домнина М.В., Меджидов З.У., 2021



УДК 004.75

**Алиева А.Э.**

*Научный руководитель: Меджидов З.У., канд. экон. наук  
Дагестанский государственный университет народного  
хозяйства», г. Махачкала, Россия*

## РЕАЛИЗАЦИЯ КЛИЕНТСКОГО ПРИЛОЖЕНИЯ НА ОСНОВЕ ТЕХНОЛОГИИ WPF

Клиент написан на платформе .net core и технологии WPF, которая позволяет создавать оконные приложения, но в свою очередь накладывает ограничение на использование клиента только в ОС Windows. Клиентская часть только отражает работу в связке клиента и сервера, но так как протоколом передачи данных является протокол TCP, то технологии реализации клиента и сервера могут быть различными, то есть написаны на разных языках программирования.

Клиентом является программа, которая взаимодействует с сервером, отправляет запросы и получает ответы. Для удобства использована технология wpf (.net core), позволяющая создавать оконные приложения для ОС Windows.

Основные функции клиента: отправка запроса на авторизацию и регистрацию, а также получение времени от сервера.

На рисунке 1 представлен экран авторизации и регистрации, имеются 2 поля для ввода логина и пароля, а также 3 кнопки: регистрация, авторизация и «Время», которая добавлена для наглядного представления что сервер не ответит на запрос от неавторизованного пользователя.

На рисунке 2 показан экран для авторизованного пользователя, который имеет кнопку «Получить серверное время». В отличие от подобной кнопки на экране авторизации, при нажатии на эту сервер ответит, и клиент выделит время, так как знает, что авторизация прошла успешно и имеет у себя соответствующую отметку.

На рисунке 3 показано окно, которое выводится, если сервер прислал ответ на запрос о получении времени.

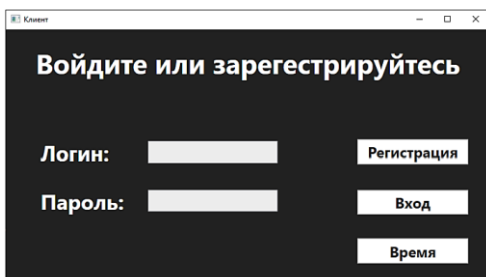


Рис. 1. Экран авторизации и регистрации

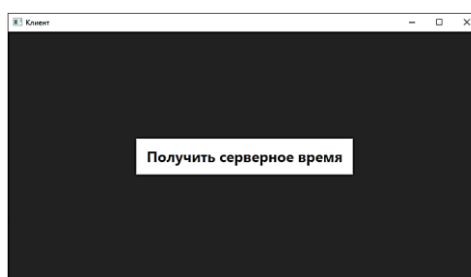


Рис. 2. Экран для авторизовавшихся пользователей

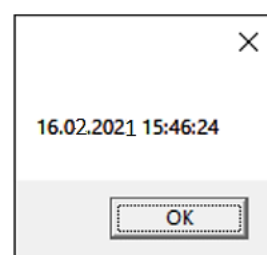


Рис. 3. Результат ответа на запрос

Ниже представлен код, который срабатывает при нажатии на кнопку авторизации на рисунке 1. В переменные login и password считываются данные, который ввел пользователь в соответствующие поля для ввода. Если пользователь не ввел в какое-либо поле ничего, то ему будет выдана ошибка, иначе логин и пароль будут отправлены на дальнейшую обработку с целью передачи серверу информации об авторизации. При ответе сервера об успешной авторизации меняет экран входа и регистрации на другой.

```
1. private void Button_Authorization(object sender, RoutedEventArgs e)
2. {
3.     string login = LoginText.Text;
4.     string password = PasswordText.Password;
5.
6.     if (login != «» && password != «»)
7.         if (ClientLogic.AuthorizationRegistration(login, password, «Autho-
            rization»))
8.             {
9.                 Auth.Visibility = Visibility.Hidden;
10.                MainScreen.Visibility = Visibility.Visible;
11.            }
12.        else
13.            MessageBox.Show(«Неправильный логин или пароль!»);
14.
15.        else
16.            MessageBox.Show(«Заполните все поля!»);
17.    }
```

Код кнопки регистрации схож с кнопкой авторизации, только вызывается функция для регистрации и при ответе сервера о успешной регистрации выдает сообщение «Регистрация прошла успешно!».

```
1. private void Button_Registration(object sender, RoutedEventArgs e)
2. {
3.     string login = LoginText.Text;
4.     string password = PasswordText.Password;
5.
6.     if (login != «» && password != «»)
7.         if (ClientLogic.AuthorizationRegistration(login, password, «Registra-
            tion»))
8.             MessageBox.Show(«Регистрация прошла успешно!»);
9.        else
10.            MessageBox.Show(«Логин занят, придумайте другой!»);
11.
12.        else
13.            MessageBox.Show(«Заполните все поля!»);
14.    }
```

Код кнопки получения серверного времени вызывается при соответствующем нажатии кнопки на рисунке 1 и 2, единственное отличие, что при нажатии на экране авторизации сервер ничего не ответит, так как пользователь еще не вошел под своими данными.

```
1. private void Button_GetServerTime(object sender, RoutedEventArgs e)
2. {
3.     MessageBox.Show(ClientLogic.GetServerTime().ToString());
4. }
```

Функции из кнопок вызываются в классе ClientLogic, в котором выполняются основные взаимодействия с сервером.

Для подключения к серверу необходимо прописать его IP-адрес, а также порт и вызвать соответствующую функцию.

```
1. const int port = 8888;  
2. const string address = «127.0.0.1»;  
3. server = new TcpClient(address, port);  
4. stream = server.GetStream();
```

Функция AuthorizationRegistration формирует запрос к серверу для авторизации или регистрации, прежде чем отправить на сервер пароль, его в целях безопасности необходимо взять хэш функцию, в нашем случае это MD5.

Затем формируется сообщение для сервера формата login + «:» + password, задается команда в зависимости от переданного в функцию параметра mode, если он равен “Authorization”, то команда равна единице, если “Registration”, то двойке, сервер знает команды заранее.

```
1. public static bool AuthorizationRegistration(string login, string password, string mode)  
2. {  
3.     using (System.Security.Cryptography.MD5 md5 = System.Security.Cryptography.MD5.Create())  
4.     {  
5.         byte[] inputBytes = Encoding.Unicode.GetBytes(password);  
6.         byte[] hashBytes = md5.ComputeHash(inputBytes);  
7.  
8.         password = BitConverter.ToString(hashBytes).Replace(«-», «»);  
9.     }  
10.  
11.     string message = login + «:» + password;  
12.  
13.     int command = 0;  
14.     if (mode == «Authorization»)  
15.         command = 1;  
16.     else if (mode == «Registration»)  
17.         command = 2;  
18.  
19.     SendData(command, message);  
20.  
21.     string serverResponse = Encoding.Unicode.GetString(ServerResponse());  
22.  
23.     if (serverResponse == «Success»)  
24.         return true;  
25.     else  
26.         return false;  
27. }
```

С помощью функции SendData отправляется запрос к серверу.

```
1. public static void SendData(int command, string message)  
2. {  
3.     List<byte> tcpMessage = new List<byte>();  
4.     tcpMessage.Add(Convert.ToByte(command));  
5.     tcpMessage.AddRange(Encoding.Unicode.GetBytes(message));  
6.     byte[] data = tcpMessage.ToArray();  
7.  
8.     stream.Write(data, 0, data.Length);  
9. }
```

Функция `ServerResponse`, ожидает ответ от сервера и затем возвращает полученные данные в байт массиве.

```
1. public static byte[] ServerResponse()  
2. {  
3.     byte[] data = new byte[512];  
4.     int bytes = 0;  
5.  
6.     while (!stream.DataAvailable) { }  
7.  
8.     bytes = stream.Read(data, 0, data.Length);  
9.     Array.Resize(ref data, bytes);  
10.  
11.     return data;  
12. }
```

Функция `GetServerTime` запрашивает время у сервера, и в случае ответа конвертирует и возвращает время в формате `DateTime`, так как сервер отвечает на этот запрос в формате `unix time`.

```
1. public static DateTime GetServerTime()  
2. {  
3.     SendData(3, «»);  
4.     int timestamp = BitConverter.ToInt32(ServerResponse());  
5.     DateTime serverDate = (new DateTime(1970, 1, 1, 0, 0, 0,  
6.         0)).AddSeconds(timestamp);  
7.     return serverDate;  
8. }
```

Изложенное выше решение доказало свою работоспособность на практике. Оно может стать фундаментом для создания клиентских приложений с большим функционалом.

### Литература

1. Албахари Дж., Албахари Б. *C# 7.0. Справочник. Полное описание языка*. М., 2018. 1024 с.
2. Рихтер Д. *CLR via C#. Programming on the Microsoft .NET Framework 4.5 in C#*. 4th ed. М., 2017. 896 с.
3. Тахагхогхи С., Вильямс Х.Е. *Руководство по MySQL*. М., 2007. 544 с.
4. Троелсен Э., Джепикс Ф. *Язык программирования C# 7 и платформы .NET и .NET Core*. 2018. М.; СПб., 1328 с.
5. Шилдт Г. *C# 4.0. Полное руководство*. М., 2019. 1056 с.

© Алиева А.Э., Меджидов З.У., 2021

УДК 519.688

**Левкин Л.В.**

*Научный руководитель:* **Слива М.В.**, канд. пед. наук  
Низневартовский государственный университет,  
г. Нижневартовск, Россия

## РАЗРАБОТКА ЛИЧНОГО КАБИНЕТА СТУДЕНТА НА ПЛАТФОРМЕ ANDROID

В настоящее время в сфере образования преобладает большое количество решений с использованием информационных технологий. Они призваны не только повысить качество получаемых знаний, но также избавить от рутинной работы и сэкономить время. А время является ценнейшим и невозполнимым ресурсом, поэтому на разных этапах учебного процесса внедряются информационные технологии. Одним из важнейших этапов учебного процесса является отслеживание успеваемости и уровня усвоения пройденного материала, а также доступ к просмотру учебного плана, списка приказов, дисциплин по выбору, портфолио. Немаловажную роль в образовательном процессе играет удобство и скорость доступа к вышеперечисленным средствам. Большинство опрошенных студентов признаются, что очень редко посещают личный кабинет университета по причине непостоянного доступа к персональному компьютеру, а макет сайта не адаптирован под мобильные устройства, из-за чего у студентов возникают трудности при работе в личном кабинете.

В большинстве университетов, в том числе и за рубежом, личный кабинет вузов базируется не только на платформе настольных компьютеров, но и мобильных платформах, благодаря чему студенты, единожды авторизовавшиеся в приложении, находятся там постоянно и получают быстрый, удобный доступ к функциям личного кабинета.

Веб версия личного кабинета состоит из навигационной панели, располагаемой сверху, и переключаемого контейнера, расположенного по середине (рис. 1).

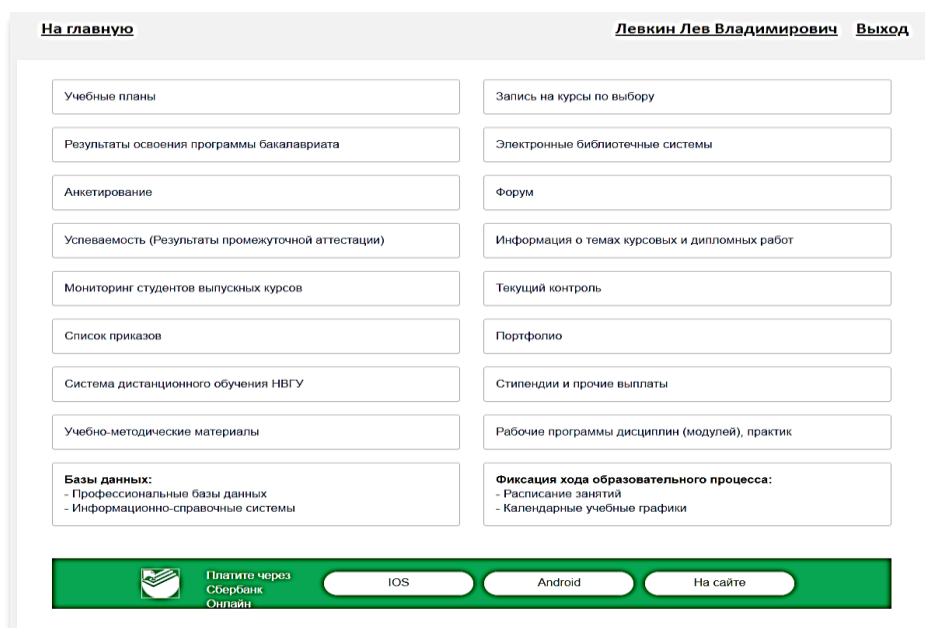


Рис. 1. Веб версия личного кабинета



Каждый пункт меню содержит свой фрагмент, заменяемый в контейнере. Навигационная панель остается на прежнем месте. Например, фрагмент элемента меню – «Успеваемость (Результаты промежуточной аттестации)» выглядит следующим образом (рис. 2).

Первый семестр	Второй семестр	Третий семестр	Четвертый семестр	Пятый семестр	Шестой семестр	Седьмой семестр	Восьмой семестр	
#	Предмет				Вид контроля	Оценка	Часы	Дата
1	Инженерная графика				Зачет	Зачтено	108	25.12.2018
2	Иностранный язык				Зачет	Зачтено	108	25.12.2018
3	Информатика				Экзамен	Отлично	216	25.12.2018
4	Информационно-библиотечное обеспечение образования				Зачет	Зачтено	72	25.12.2018
5	Математика				Экзамен	Отлично	216	25.12.2018
6	Программирование				Экзамен	Отлично	216	25.12.2018
7	Русский язык и культура речи				Зачет	Зачтено	108	25.12.2018
8	Элективные дисциплины (модули) по физической культуре и спорту				Зачет	Зачтено	54	25.12.2018

Рис. 2. Элемент меню «Успеваемость (Результаты промежуточной аттестации)»

С точки зрения проектирования дружелюбного UX<sup>1</sup> некоторые пункты меню личного кабинета можно собрать в группы, уменьшая навигационную путаницу на главной странице и выделив наиболее приоритетные.

Навигация в мобильном приложении состоит из трех фрагментов, комбинирующих элементы меню.

В главном фрагменте отображается базовая информация о студенте, элементы управления профилем, сгруппированные разделы, информация о курсовых и дипломных работах, а также полезные ссылки на внешние источники (рис. 3).

В фрагменте «успеваемость» отображаются результаты промежуточной аттестации, разбитые по семестрам. При нажатии на элемент, содержащий информацию о дисциплине, откроется нижнее полноэкранное диалоговое окно, в котором отображен учебный план выбранного предмета (рис. 4).

Фрагмент портфолио отображает пункты учебной деятельности и позволяет просматривать, редактировать, удалять объекты портфолио, скачивать и загружать закрепленные файлы, а также оставлять комментарии (рис. 5).

UI<sup>2</sup> мобильного приложения однороден и наследует цветовую схему веб версии личного кабинета.

<sup>1</sup> User Experience (UX) – опыт, который получает пользователь при взаимодействии с сайтом или приложением

<sup>2</sup> User Interface (UI) – пользовательский интерфейс

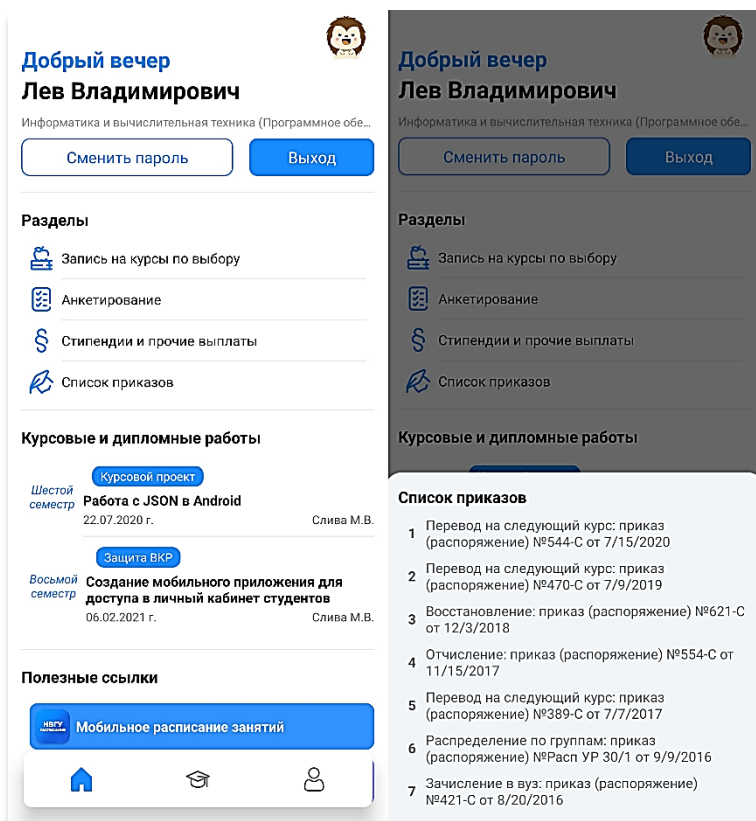


Рис. 3. Главный (домашний) фрагмент

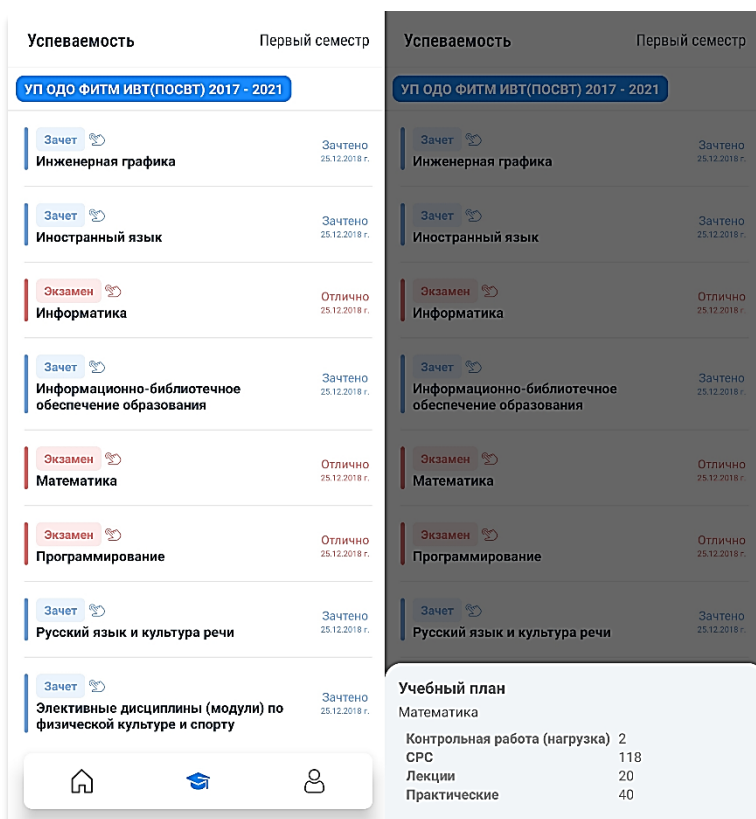


Рис. 4. Фрагмент «Успеваемость»

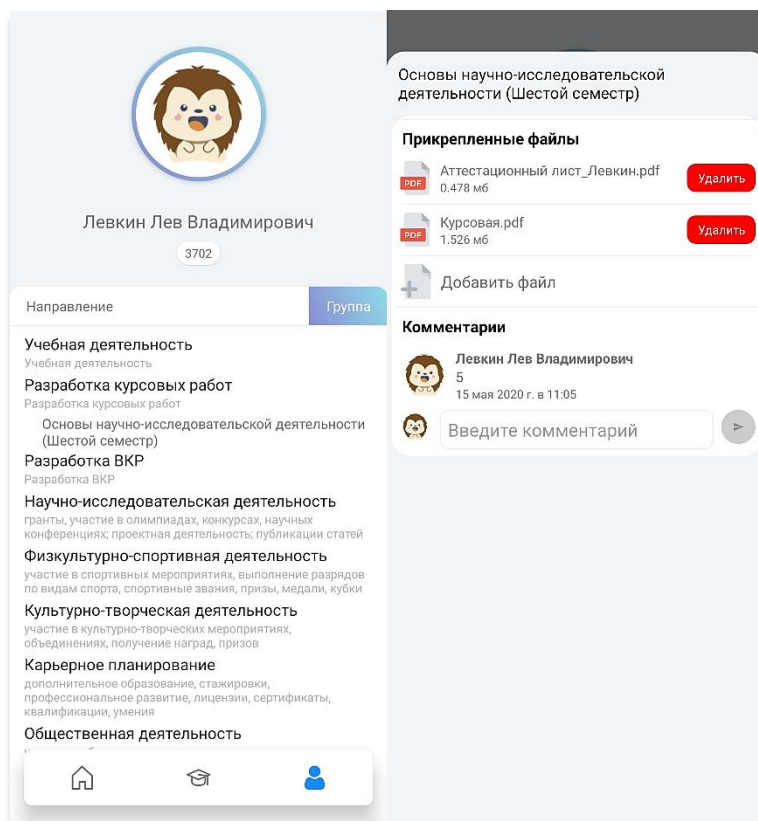


Рис. 5. Фрагмент «Портфолио»

В мобильной платформе Android визуальные компоненты представления сильно зависят от состояния приложения и регулируются его жизненным циклом (рис. 6), который в данный момент представлен (виден) пользователю.

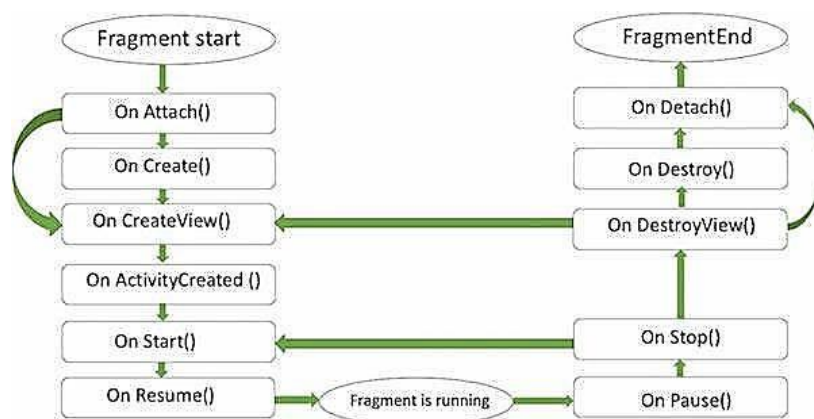


Рис. 6. Жизненный цикл фрагмента

При смене фрагментов система вызывает метод `onDestroy()`<sup>3</sup> у отображаемого представления и `onCreate()`<sup>4</sup> следующего. Состояние фрагментов не сохраняется, так как при переключении каждый раз создается новый экземпляр. Как следствие вся информация

<sup>3</sup> `onDestroy()` – метод, вызываемый перед разрушением фрагмента.

<sup>4</sup> `onCreate()` – метод, вызываемый перед созданием фрагмента.

теряется, а разделы загружаются заново, что приводит к лишней сетевой, процессорной нагрузке и ускоренному разряду батареи.

Во избежание подобных проблем было принято решение о разграничении интерфейса и бизнес логики. MVVM (Model-View-ViewModel) – самый распространенный архитектурный паттерн, позволяющий инкапсулировать части кода и переиспользовать его неограниченное количество раз. MVVM состоит из трех компонентов: модели (Model), модели представления (ViewModel) и представления (View) (рис. 7).

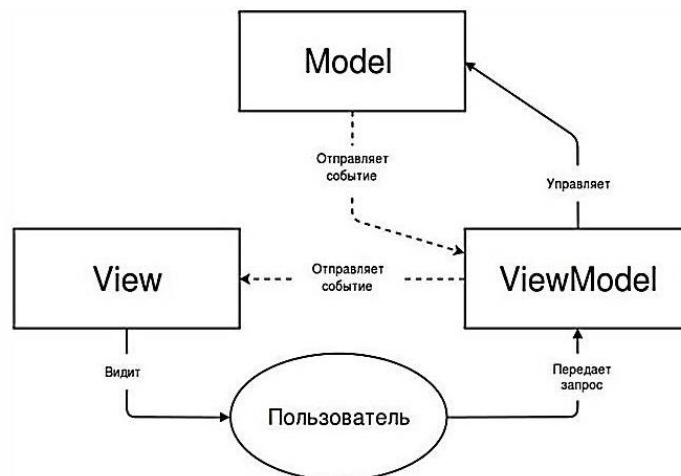


Рис. 7. Структура паттерна MVVM

Model описывает используемые в приложении данные. Модели могут содержать логику, непосредственно связанную этими данными, например, логику валидации свойств модели. В то же время модель не должна содержать никакой логики, связанной с отображением данных и взаимодействием с визуальными элементами управления [1].

View или представление определяет визуальный интерфейс, через который пользователь взаимодействует с приложением. Представление не обрабатывает события за редким исключением, а выполняет действия в основном посредством команд [1].

ViewModel или модель представления связывает модель и представление через механизм привязки данных. Определяет логику по обновлению данных в модели, а также содержит логику по получению данных из модели, которые потом передаются в представление [1].

Поскольку элементы представления, то есть визуальные компоненты типа кнопок, не используют события, то представление взаимодействует с ViewModel посредством команд.

Итогом применения паттерна MVVM является функциональное разделение приложения на три компонента, которые проще разрабатывать и тестировать, а также в дальнейшем модифицировать и поддерживать.

### Литература

1. Автор неизвестен. Паттерн MVVM // Определение паттерна MVVM. 2016. <https://metanit.com/sharp/wpf/22.1.php>

© Левкин Л.В., Слива М.В., 2021

УДК 519.8

Магомедшарипов А.М., Цыпурдиенко А.С.,  
Гробер Т.А., канд. физ.-мат. наук  
Донской государственный технический  
университет, г. Ростов-на-Дону, Россия

### РЕШЕНИЕ ЗАДАЧИ О КОНКУРЕНТНОМ ПРЕИМУЩЕСТВЕ СРЕДСТВАМИ СОВРЕМЕННЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Теория, в которой рассматриваются задачи принятия решений в условиях неопределенности при наличии противника, известна как теория игр. Теория игр довольно молодая наука, ей около 70 лет.

Ее основы были заложены в работе математика Джона фон Неймана и экономиста Оскара Моргенштерна [1].

В своей работе они смоделировали более простую версию покера и проанализировали оптимальные стратегии, используемые игроками. А спустя годы люди сочли их идеи полезными для исследований в биологии, экономике и других отраслях науки. Теория игр сейчас активно развивается, в том числе с применением информационных технологий. Отличным примером может послужить работа [2].

Теория игр наиболее широко применяется в экономике и социальных науках. Некоторые важные аспекты ее применения были рассмотрены в статье Гуровой Т.И. и Конева И.П. [3]. Рассмотрим следующую задачу о конкурентной борьбе двух производителей.

Компании «Крок» и «Компьюлинк» создают ИТ-продукцию определенного назначения и доставляют ее на рынок информационных технологий региона. В данном регионе они являются единственными подрядчиками и полностью определяют рынок выпускаемой ими продукции.

Обе компании производят ИТ-продукцию с применением одной из трех различных технологий. В зависимости от функционала ИТ-продукции, которая производится по каждой технологии, цену на продукции компания может установить на уровне 4, 6 и 10 денежных единиц соответственно. Затраты на производство ИТ-продукции приведены в таблице 1.

Таблица 1

Данные о себестоимости и цене реализации ИТ-продукции

Технология	Цена на реализацию единицы ИТ-продукции, д.е.	Полная себестоимость единицы ИТ-продукции, д.е.	
		Компания «Крок»	Компания «Компьюлинк»
1	4	1	3
2	6	3	4
3	10	8	7

В результате исследования данного рынка региона была определена функции спроса на ИТ-продукцию:

$$Y = 6 - 0,4x,$$



где  $Y$  – количество ИТ-продукции, которое купят потребители (тыс. ед.),  $x$  – средняя цена продукции производителя (ден. ед.).

Доли купленных населением ИТ-продукции компании «Крок» зависят от соотношения цен на ИТ-продукцию компании «Крок» и «Компьюлинк». В результате исследования получены данные, представленные в таблице 2.

Таблица 2

Данные о доле продукции компании «Крок», приобретенной населением,  
при использовании различных технологий

Цена реализации 1 ед. ИТ-продукции, д.е.		Доля ИТ-продукции компании «Крок», приобретенное населением
Компания «Крок»	Компания «Компьюлинк»	
10	10	0,61
10	6	0,30
10	4	0,33
6	10	0,40
6	6	0,60
6	4	0,45
4	10	0,91
4	6	0,83
4	4	0,33

Одной из важных задач любой компании является максимизация прибыли посредством реализации продукции. Но в рассматриваемой задаче более важной является конкурентная борьба. В данном конфликте выгода будет определяться не размером прибыли каждой компании, а ее разностью между прибылями этих компаний.

Составим платежную матрицу игры, элементами которой являются числа, определяющие доминирование компании «Крок» над конкурентом. Определим стратегии игроков. Обозначим через  $A_1, A_2, A_3$  – компания «Крок» выбирает технологии 1, 2 и 3 соответственно.  $B_1, B_2, B_3$  – компания «Компьюлинк» выбирает технологии 1, 2 и 3 соответственно

Проведя все расчеты, получаем платежную матрицу, представленную в виде таблицы 3.

Таблица 3

Представление платежной матрицы игры

	$B_1$	$B_2$	$B_3$
$A_1$	0,100	-2,240	-0,032
$A_2$	-1,680	3,600	3,200
$A_3$	7,872	8,600	1,408

Данная задача не имеет решения в чистых стратегиях, поэтому необходимо искать решение в смешанных стратегиях.

Требуется найти цену игры и оптимальные стратегии для обеих компаний. Оказывается, что рассмотренная задача может быть сведена к задаче линейного программирования.

Применение современных информационных технологий существенно упрощает решение подобных задач.

Чтобы не считать все вручную, воспользуемся языком программирования Python и создадим код.

Для начала импортируем необходимые функции из библиотек, как показано на рисунке 1.

```
from scipy.optimize import linprog
from numpy import array, amin, set_printoptions
```

Рис. 1. Подключение библиотек

Сделаем вывод массивов решения в процентах (рис. 2).

```
set_printoptions(formatter={"all": lambda x: "%-3.2f"% (x*100) + "%"})
```

Рис. 2. Определение формата результата

Затем запишем функцию, определяющую решение задачи (рис. 3).

```
def solve(arr, players = ["A", "B"]):
    m_arr = array(arr)
    for i in range(2):
        a = m_arr.copy()
        if i == 0: a = a.T *-1
        n = abs(amin(a))
        a[:] += n
        result = linprog([-1]*len(a[0]), a, [1]*len(a), method='simplex')
        if result["success"]:
            z = result["fun"]*-1
            x = result["x"]/z
            if i == 0: print("Цена игры: %.3f" % ((1/z-n)*-1))
            print("Оптимальная стратегия \"+players[i]+\"\":\n", x)
```

Рис. 3. Создание функции пользователя

После чего остается вызвать функцию, передав туда данные платежной матрицы и названия компаний (рис. 4).

```
arr = [[ 0.1, -2.24, -0.032],
        [-1.68, 3.6, 3.2],
        [7.872, 8.6, 1.408]]
names = "Крок", "Компьюлинк"
solve(arr, names)
```

Рис. 4. Ввод данных в программу

После запуска программы получим оптимальные смешанные стратегии для каждого игрока, представленные в процентном соотношении, как показано на рисунке 5.

```
Цена игры: 2.429
Оптимальная стратегия "Крок":
| [0.00% 56.98% 43.02%]
Оптимальная стратегия "Компьюлинк":
| [15.80% 0.00% 84.20%]
>>>
```

Рис. 5. Результат работы программы

Делаем вывод что компания «Крок» должна распределить производство продукции следующим образом (рис. 6).

$A_1$	$A_2$	$A_3$
0%	56,98%	43,02%

Рис. 6. Распределение производства продукции по технологиям компании «Крок»

А компания «Компьюлинк» – как показано на рис. 7.

$B_1$	$B_2$	$B_3$
15,8%	0%	84,2%

Рис. 7. Распределение производства продукции по технологиям компании «Компьюлинк»

Следует отметить, что полученный код является наиболее компактной реализацией (среди встречающихся на просторах сети Интернет) задачи линейного программирования. Полученное решение можно применять для множества задач, подобных этой.

Исходя из рассмотренных примеров, можно сделать вывод, что теория игр развивает логическое мышление, расширяет кругозор, а также показывает, что наша интуиция не всегда верна. А в заключении хочется отметить, что с помощью теории игр можно спрогнозировать развитие ситуации и найти наибольшую выгоду для каждого участника.

### Литература

1. Нейман Дж., Моргенштерн О. Теория игр и экономическое поведение. М., 1970. 707 с.
2. Гробер Т.А., Колотиенко М.А. Имитационное моделирование задачи о дуэли трех лиц. // Инженерный вестник Дона. 2017. №4. <https://clck.ru/UHGhp>
3. Гурова Т.И., Конев И.П. Некоторые аспекты применения теории игр в экономике // Вестник РМАТ. 2019. №1. С. 38-43.

© Магомедшарипов А.М., Цыпурдиенко А.С., Гробер Т.А., 2021

УДК 519.688

**Сорокин А.О.**

*Научный руководитель:* **Слива М.В.**, канд. пед. наук  
Нижевартовский государственный университет,  
г. Нижневартовск, Россия

## СОЗДАНИЕ ПРИЛОЖЕНИЯ ДЛЯ ПРОСМОТРА IPTV НА ПЛАТФОРМЕ ANDROID

Еще несколько десятков лет назад не каждый гражданин мог свободно приобретать все то, что сегодня составляет неотъемлемую часть рядового быта. Так, привычная нам вещь – телевизор – для многих оставалась мечтой. Этот электроприбор не только развлекал и информировал. Телевизор в доме напрямую свидетельствовал о достатке и удачливости владельца. Сейчас любое устройство с экраном может заменить привычный многим телевизор.

IPTV (IP Телевидение, Интерактивное Телевидение) – это технология передачи телевизионного сигнала по каналам Интернет-связи (по протоколу IP), применяемая операторами цифрового телевидения и провайдерами Интернет-услуг для доставки телевизионных каналов посредством широкополосного доступа в Интернет.

Для использования IPTV не требуется антенна или прокладка особенных кабелей. Все что нужно, чтобы перейти на IP-телевидение – это стабильное высокоскоростное подключение к сети интернет (IPTV – что это и как смотреть на любом устройстве <https://clck.ru/UHJLy>).

Для разработки приложения использовались:

– Android Studio – среда разработки под популярную операционную систему Андроид с использованием языка Kotlin.

– libVLC – основной движок и интерфейс к мультимедийной структуре, на которой основан медиаплеер VLC.

– RxJava – это реализация реактивных расширений Java VM: библиотека для создания асинхронных программ и программ, основанных на событиях, с использованием наблюдаемых последовательностей (RxJava: Reactive Extensions for the JVM. <https://clck.ru/UHJMz>).

– Retrofit – типизированный http клиент для Android. Библиотекой удобно пользоваться для запроса к различным веб-сервисам с командами GET, POST, PUT, DELETE. Может работать в асинхронном режиме, что избавляет от лишнего кода. Библиотека может работать с GSON и XML, используя специальные конвертеры. Для работы с ним необходимо реализовать 3 класса:

1. POJO (Plain Old Java Object) или Model Class — json-ответ от сервера нужно реализовать как модель.

2. Retrofit — класс для обработки результатов. Ему нужно указать базовый адрес в методе baseUrl().

3. Interface — интерфейс для управления адресом, используя команды GET, POST и т.д. (Retrofit. <https://clck.ru/UHJMU>).

Model class выглядит следующим образом:

```
data class Channel(  
    @SerializedName("id") val id: Int,  
    @SerializedName("name") val name: String,  
    @SerializedName("num") val channelNum: Int,  
    @SerializedName("url") val url: String,  
    @SerializedName("timeshift") val timeshift: String)
```

SerializedName - аннотация, указывающая, что этот член должен быть сериализован в JSON с указанным значением имени в качестве имени поля.

Реализация класса Retrofit:

```
fun retrofit(): Retrofit =  
    Retrofit.Builder()  
        .baseUrl(BASE_URL)  
        .addConverterFactory(ScalarsConverterFactory.create())  
        .addConverterFactory(GsonConverterFactory.create())  
        .addCallAdapterFactory(RxJava3CallAdapterFactory.create())  
        .build()
```

RxJava3CallAdapterFactory – добавление этого класса в Retrofit позволяет возвращать Observable, Single или Completable из методов службы.

ScalarsConverterFactory – конвертер строк в текст.

GsonConverterFactory – конвертер, который использует GSON для сериализации в JSON и обратно.

Реализация класса Interface для получения списка каналов:

```
@GET("/player/php/channels.php")  
fun getChannels(@Query("platform") platform: String = "android"):  
Single<List<Channel>>
```

Чтобы начать работать с полученными данными их необходимо добавить в LinkedList (является представителем двунаправленного списка, где каждый элемент структуры содержит указатели на предыдущий и следующий элементы). Реализация выглядит следующим образом:

```
val channelListsQuery = videoInteractorImpl.getChannelList()  
compositeDisposable = CompositeDisposable()  
compositeDisposable.add(  
    channelListsQuery  
        .subscribe({ channelsList ->  
            channelsList.let {  
                channels.addAll(channelsList)  
            }  
        })
```



```
}, {  
    Log.d("debugTag", it.localizedMessage)  
})
```

После выполнения вышеописанного кода все каналы с их информацией будут храниться в channels. Для группировки каналов по категориям используется фильтр: channels.filter { it.id in category.channels } где it.id - это id канала, category.channels – список id каналов выбранной категории.

Для того чтобы вывести список каналов на экран необходимо добавить их в ListAdapter

```
val adapter: Any? = context?.let {  
    ArrayAdapter<Any?>(  
        it,  
        R.layout.list_item_view, channels as List<Any?>  
    )  
}  
channelListView.adapter = adapter as ListAdapter
```

Шаблон интерфейса созданного приложения показан на рисунке.



Рис. Внешний вид программы

В первом столбце отображается список категорий, такие как: Все каналы, Федеральные, Кино и сериалы, Познавательные и тд. Во втором столбце – Название канала, номер, логотип и телепередача, которая идет в данный момент. В последнем столбце – программа телепередач за определенный промежуток времени, с возможностью перейти в режим timeshift.

Timeshift – функциональная возможность цифрового телевидения, при которой пользователь может просматривать телепрограммы, используя функции «Пауза» и «Перемотка».

Для воспроизведения выбранного канала используется libvlc.

```
mLibVLC = LibVLC(context, args)
```

```
mMediaPlayer = MediaPlayer(mLibVLC)
```

args – параметры отображения видео.

mMediaPlayer.play(Uri.parse(channel)) – функция которая воспроизводит поток по ссылке.

```
mMediaPlayer.attachViews(video_layout, null, ENABLE_SUBTITLES,  
USE_TEXTURE_VIEW) – прикрепление видео макета к плееру.
```

После выбора канала вызывается функция, которая по его номеру находит информацию о нем в channels и возвращает строку, которая содержит в себе ссылку, необходимую для воспроизведения потока. Потом вызывается функция, которая принимает ссылку потока и воспроизводит его.

Чтобы приложение могло использовать выход в интернет и работать с sd-картой, необходимо в manifest объявить следующее: `<uses-permission android:name="android.permission.INTERNET" />` и `<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />`.

Для наибольшего удобства реализована функция запоминания последних выбранных канала и категории. Реализация выполняется следующими участками кода:

– функция сохранения id последнего канала:

```
val preferences: SharedPreferences =  
PreferenceManager.getDefaultSharedPreferences(context)  
val editor = preferences.edit()  
editor.putInt("id_channel", data)  
editor.apply()
```

– функция получения id последнего канала:

```
val preferences: SharedPreferences =  
PreferenceManager.getDefaultSharedPreferences(context)  
val id = preferences.getInt("id_channel", 1)
```

© Сорокин А.О., Слива М.В., 2021

УДК 336

**Силантьева И.А.**

*Научный руководитель:* **Коненков А.Н.**, д-р физ.-мат. наук  
Рязанский государственный университет им. С.А. Есенина,  
г. Рязань, Россия

## МОДЕЛИРОВАНИЕ ДИНАМИКИ БИЗНЕС-ПРОЦЕССОВ С ПРИМЕНЕНИЕМ ЛАГОВЫХ МОДЕЛЕЙ С ИСПОЛЬЗОВАНИЕМ СТАТИСТИКИ ПОИСКОВЫХ ЗАПРОСОВ СЕРВИСА GOOGLETRENDS

В данной работе рассматривается моделирование динамики бизнес-процессов с использованием построения модели с распределенным лагом на примере данных по кредитованию населения.

Для построения модели с сайта Центрального банка Российской Федерации были собраны данные о кредитах, предоставленных физическим лицам-резидентам (сведения о размещенных и привлеченных средствах // Федеральная служба государственной статистики: официальный сайт. <https://clck.ru/UHKVw>), а также данные, показывающие уровень интереса к запросам по ключевому слову «Кредит» с сервиса GoogleTrends (в баллах) (кредиты – Анализ – GoogleTrends // GoogleTrends. <https://clck.ru/UHKVK>). Данные были взяты за период с 2011 г. по 2018 г. в регионах Центрального федерального округа Российской Федерации.

Построение указанной модели было выполнено по следующему алгоритму:

1. Определение максимальной величины лага  $l$ .
2. Определение степени полинома  $k$ , описывающего структуру лага.
3. Перерасчет переменных  $z_i, i = \overline{0; k}$ .
4. Оценка коэффициентов  $c_i, i = \overline{0; k}$ .
5. Вычисление коэффициентов  $b_j, j = \overline{0; l}$ .
6. Оценка статистических характеристик путем вычисления предсказанных значений, остатков и характеристик ( $R^2, DW, A$ ).

При построении моделей использовались метод Алмон и метод Койка. Всего было построено 7 моделей – с линейной, квадратичной, кубической структурой лага; лаговые модели 4-й, 5-й и 6-й степени; модель с бесконечным лагом.

Модель с распределенным лагом для конечного значения  $l$  описывается следующим уравнением:

$$y_t = a + \sum_{j=0}^l b_j x_{t-j} + \varepsilon_t \quad (1)$$

Для построения модели с линейной структурой лага были введены вспомогательные переменные  $z_0$  и  $z_1$ :

$$z_0 = \sum_{j=0}^6 x_{t-j}, z_1 = \sum_{j=0}^6 j x_{t-j}.$$

С помощью инструмента «Регрессия» были вычислены значения коэффициентов  $a$ ,  $c_0$  и  $c_1$ :  $a = -12802647,37$ ,  $c_0 = 13832,15$ ,  $c_1 = 9591,78$ .

Уравнение (1) можно переписать в виде:

$$y_t = c_0 \sum_{j=0}^6 x_{t-j} + c_1 \sum_{j=0}^6 jx_{t-j} + \varepsilon_t \quad (2)$$

Для найденных значений  $c_0$  и  $c_1$  получим:

$$y_t = 13832,15 \sum_{j=0}^6 x_{t-j} + 9591,78 \sum_{j=0}^6 jx_{t-j} + \varepsilon_t.$$

После вычисления значений  $c_0$  и  $c_1$  были найдены коэффициенты  $b_j$  по формуле  $b_j = c_0 + c_1j$ ,  $j = \overline{0;6}$  (табл. 1).

Таблица 1

Модель с линейной структурой лага

Коэффициент $b_j$	Значение коэффициента $b_j$
$b_0 = c_0$	13832,15
$b_1 = c_0 + c_1$	23423,93
$b_2 = c_0 + 2c_1$	33015,71
$b_3 = c_0 + 3c_1$	42607,49
$b_4 = c_0 + 4c_1$	52199,27
$b_5 = c_0 + 5c_1$	61791,05
$b_6 = c_0 + 6c_1$	71382,83

С учетом найденных значений уравнение (1) можно записать в виде:

$$y_t = -12802647,37 + 13832,15x_t + 23423,93x_{t-1} + 33015,71x_{t-2} + 42607,49x_{t-3} + 52199,27x_{t-4} + 61791,05x_{t-5} + 71382,83x_{t-6}.$$

Аналогично для модели с квадратичной структурой лага:

$$z_0 = \sum_{j=0}^6 x_{t-j}, \quad z_1 = \sum_{j=0}^6 jx_{t-j}, \quad z_2 = \sum_{j=0}^6 j^2x_{t-j}.$$

Получим, что  $a = -13365209,97$ ,  $c_0 = 51897,14$ ,  $c_1 = -29828,52$ ,  $c_2 = 6093,97$ .

Уравнение (1):

$$y_t = c_0 \sum_{j=0}^6 x_{t-j} + c_1 \sum_{j=0}^6 jx_{t-j} + c_2 \sum_{j=0}^6 j^2x_{t-j} + \varepsilon_t \quad (3)$$

$$y_t = 51897,14 \sum_{j=0}^6 x_{t-j} - 29828,52 \sum_{j=0}^6 jx_{t-j} + 6093,97 \sum_{j=0}^6 j^2x_{t-j} + \varepsilon_t.$$

Коэффициенты  $b_j$  находим по формуле  $b_j = c_0 + c_1j + c_2j^2$ ,  $j = \overline{0;6}$  (табл. 2).

Таблица 2

Модель с квадратичной структурой лага

Коэффициент	Значение коэффициента
$b_0 = c_0$	51897,14
$b_1 = c_0 + c_1 + c_2$	28162,59
$b_2 = c_0 + 2c_1 + 4c_2$	16615,99
$b_3 = c_0 + 3c_1 + 9c_2$	17257,33
$b_4 = c_0 + 4c_1 + 16c_2$	30086,62
$b_5 = c_0 + 5c_1 + 25c_2$	55103,85
$b_6 = c_0 + 6c_1 + 36c_2$	92309,02

$$y_t = -13365209,97 + 51897,14x_t + 28162,59x_{t-1} + 16615,99x_{t-2} + 17257,33x_{t-3} + 30086,62x_{t-4} + 55103,85x_{t-5} + 92309,02x_{t-6}.$$

Модель с кубической структурой лага:

$$z_0 = \sum_{j=0}^6 x_{t-j}, z_1 = \sum_{j=0}^6 jx_{t-j}, z_2 = \sum_{j=0}^6 j^2x_{t-j}, z_3 = \sum_{j=0}^6 j^3x_{t-j}.$$

$$y_t = c_0 \sum_{j=0}^6 x_{t-j} + c_1 \sum_{j=0}^6 jx_{t-j} + c_2 \sum_{j=0}^6 j^2x_{t-j} + c_3 \sum_{j=0}^6 j^3x_{t-j} + \varepsilon_t \quad (4)$$

$$y_t = 51495,55 \sum_{j=0}^6 x_{t-j} - 28572,34 \sum_{j=0}^6 jx_{t-j} + 5534,97 \sum_{j=0}^6 j^2x_{t-j} + 61,88 \sum_{j=0}^6 j^3x_{t-j} + \varepsilon_t.$$

$$b_j = c_0 + c_1j + c_2j^2 + c_3j^3, j = \overline{0; 6} \text{ (табл. 3).}$$

Таблица 3

Модель с кубической структурой лага

Коэффициент	Значение коэффициента
$b_0 = c_0$	51495,54
$b_1 = c_0 + c_1 + c_2 + c_3$	28520,05
$b_2 = c_0 + 2c_1 + 4c_2 + 8c_3$	16985,80
$b_3 = c_0 + 3c_1 + 9c_2 + 27c_3$	17264,09
$b_4 = c_0 + 4c_1 + 16c_2 + 64c_3$	29726,21
$b_5 = c_0 + 5c_1 + 25c_2 + 125c_3$	54743,48
$b_6 = c_0 + 6c_1 + 36c_2 + 216c_3$	92687,17

$$y_t = -13364281,07 + 51495,54x_t + 28520,05x_{t-1} + 16985,80x_{t-2} + 17264,09x_{t-3} + 29726,21x_{t-4} + 54743,48x_{t-5} + 92687,17x_{t-6}.$$

Модель со структурой лага четвертой степени (табл. 4).

Таблица 4

Модель со структурой лага четвертой степени

Коэффициент	Значение коэффициента
$b_0 = c_0$	55792,909
$b_1 = c_0 + c_1 + c_2 + c_3 + c_4$	21042,607
$b_2 = c_0 + 2c_1 + 4c_2 + 8c_3 + 16c_4$	17455,306
$b_3 = c_0 + 3c_1 + 9c_2 + 27c_3 + 81c_4$	22955,544
$b_4 = c_0 + 4c_1 + 16c_2 + 64c_3 + 256c_4$	30351,807
$b_5 = c_0 + 5c_1 + 25c_2 + 125c_3 + 625c_4$	47336,526



$b_6 = c_0 + 6c_1 + 36c_2 + 216c_3 + 1296c_4$	96486,079
---	-----------

$$y_t = -13363783,21 + 55792,909x_t + 21042,607x_{t-1} + 17455,306x_{t-2} + 22955,544x_{t-3} + 30351,807x_{t-4} + 47336,526x_{t-5} + 96486,079x_{t-6}.$$

Модель со структурой лага пятой степени (табл. 5).

Таблица 5

Модель со структурой лага пятой степени

Коэффициент	Значение коэффициента
$b_0 = c_0$	48762,96
$b_1 = c_0 + c_1 + c_2 + c_3 + c_4 + c_5$	33817,6
$b_2 = c_0 + 2c_1 + 4c_2 + 8c_3 + 16c_4 + 32c_5$	8172,754
$b_3 = c_0 + 3c_1 + 9c_2 + 27c_3 + 81c_4 + 243c_5$	22818,86
$b_4 = c_0 + 4c_1 + 16c_2 + 64c_3 + 256c_4 + 1024c_5$	39623,26
$b_5 = c_0 + 5c_1 + 25c_2 + 125c_3 + 625c_4 + 3125c_5$	34920,64
$b_6 = c_0 + 6c_1 + 36c_2 + 216c_3 + 1296c_4 + 7776c_5$	103103,6

$$y_t = -13344626,44 + 48762,96x_t + 33817,6x_{t-1} + 8172,754x_{t-2} + 22818,86x_{t-3} + 39623,26x_{t-4} + 34920,64x_{t-5} + 103103,6x_{t-6}.$$

Модель со структурой лага шестой степени (табл. 6).

Таблица 6

Модель со структурой лага шестой степени

Коэффициент	Значение коэффициента
$b_0 = c_0$	49665,474
$b_1 = c_0 + c_1 + c_2 + c_3 + c_4 + c_5 + c_6$	38271,791
$b_2 = c_0 + 2c_1 + 4c_2 + 8c_3 + 16c_4 + 32c_5 + 64c_6$	-4490,449
$b_3 = c_0 + 3c_1 + 9c_2 + 27c_3 + 81c_4 + 243c_5 + 729c_6$	37795,401
$b_4 = c_0 + 4c_1 + 16c_2 + 64c_3 + 256c_4 + 1024c_5 + 4096c_6$	27443,546
$b_5 = c_0 + 5c_1 + 25c_2 + 125c_3 + 625c_4 + 3125c_5 + 15625c_6$	39126,195
$b_6 = c_0 + 6c_1 + 36c_2 + 216c_3 + 1296c_4 + 7776c_5 + 46656c_6$	103543,81

$$y_t = -13344626,44 + 48762,96x_t + 33817,6x_{t-1} + 8172,754x_{t-2} + 22818,86x_{t-3} + 39623,26x_{t-4} + 34920,64x_{t-5} + 103103,6x_{t-6}.$$

*Модель с бесконечной структурой лага*

Для построения модели с бесконечной структурой лага степени были вычислены значения коэффициентов  $a$  и  $h$  с использованием инструмента регрессии на панели «Анализ данных».

Были получены следующие значения:

$$a = -379581,74, h = 1 - (1 - h) = 1 - 0,973 = 0,027.$$

$$b_j = b_0 h^j == b_{j-1} h, j = \overline{0; 6} \text{ (табл. 7).}$$

Таблица 7

Модель с бесконечным лагом

Коэффициент	Значение коэффициента
$b_0$	9357,45
$b_1 = b_0h$	260,6596852
$b_2 = b_0h^2 = b_1h$	7,260896918
$b_3 = b_0h^3 = b_2h$	0,202258451
$b_4 = b_0h^4 = b_3h$	0,005634081
$b_5 = b_0h^5 = b_4h$	0,000156942
$b_6 = b_0h^6 = b_5h$	4,37176E-06
$b_7 = b_0h^7 = b_6h$	1,21779E-07
$b_8 = b_0h^8 = b_7h$	3,39226E-09

Уравнение для модели с бесконечным лагом:

$$y_t = -379581,74 + 9357,45x_t + 260,6596852x_{t-1} + 7,260896918x_{t-2} + 0,202258451x_{t-3}.$$

Результаты, полученные при построении указанных выше лаговых моделей, представлены в таблице и на графиках (табл. 8, рис. 1 и рис. 2).

Таблица 8

Сравнение моделей с различной структурой лага

	$R^2$	DW	A	$a_0$	$c_0$	$c_1$	$c_2$	$c_3$	$c_4$	$c_5$	$c_6$
Линейный лаг	0,65	0,06	0,14	+	-	+					
Квадратичный лаг	0,85	0,21	0,09	+	+	-	+				
Кубический лаг	0,85	0,21	0,09	+	+	-	-	-			
Лаг 4-й степени	0,85	0,23	0,09	+	+	-	-	-	-		
Лаг 5-й степени	0,85	0,23	0,09	+	+	-	-	-	-	-	
Лаг 6-й степени	0,85	0,21	0,09	+	+	-	-	-	-	-	-

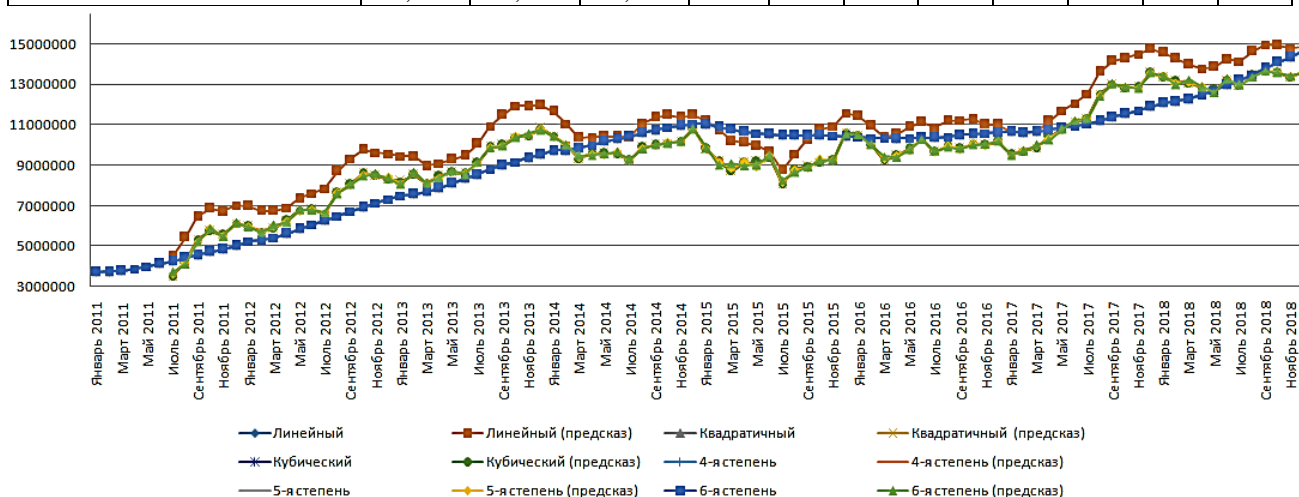


Рис. 1. Сравнение наблюдаемых и предсказанных значений

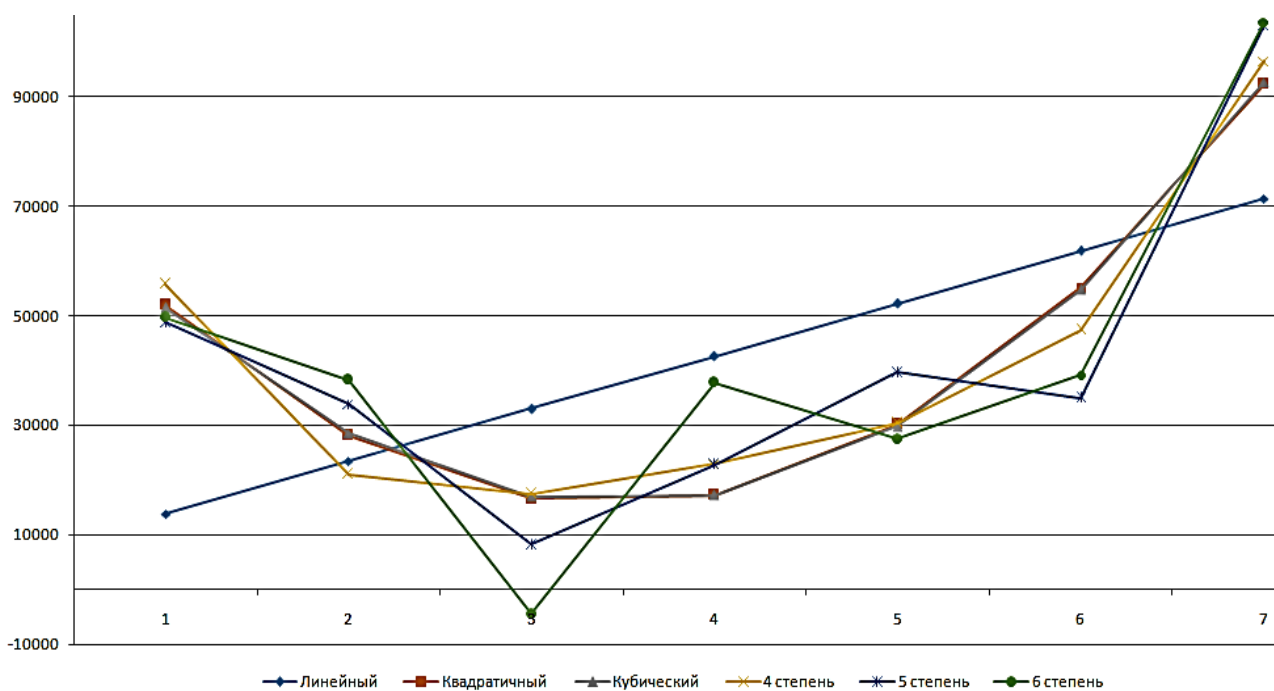


Рис. 2. Структуры лагов

Из таблицы 8 видно, что с учетом полученных значений статистических характеристик ( $R^2$ , ошибки аппроксимации  $A$ , коэффициентов  $a, c_i$ ) наиболее подходящей для исследования с дальнейшей доработкой (значение  $c_1$ ) является модель с квадратичным лагом.

© Силантьева И.А., Коненков А.Н., 2021

УДК 51-7

**Долгов А.Е., Крошилов А.Ф.**

*Научный руководитель: Клочков А.А., канд. физ.-мат. наук*  
Нижевартовский государственный университет,  
г. Нижневартовск, Россия

## **МАТЕМАТИЧЕСКОЕ ОБОСНОВАНИЕ ПРИМЕНЕНИЯ БЕРЕЖЛИВЫХ ТЕХНОЛОГИЙ**

Термин «бережливое производство» был введен Юрием Павловичем Адлером [1, с. 18] в качестве перевода оригинального понятия “Lean Production” [3, с. 407], предложенного Джоном Крафчиком [1, с. 200] для обозначения основных принципов организации производства в японской корпорации Тойота. Зародился этот термин в процессе исследования мирового автомобильного рынка в рамках Международной программы «Автомобили». Данная система опирается, прежде всего, на систему организации производственного процесса.

Бережливое производство [1, с. 236] – это концепция управления производственным предприятием, которая основана на постоянном стремлении предприятия к устранению всех видов потерь.

Кайдзен [3, с. 408] – японская практика, которая направлена на непрерывном совершенствовании процессов производства, разработки, вспомогательных бизнес-процессов и управления.

Простое понимание идеи минимизации потерь основывается на таких концепциях как Кайдзен. Основные виды потерь были сформулированы Тайити Оно, одним из основоположников бережливого производства [3, с. 408]:

1. потери из-за перепроизводства;
2. потери времени из-за ожидания;
3. потери при ненужной транспортировке;
4. потери из-за лишних этапов обработки;
5. потери из-за лишних запасов;
6. потери из-за ненужных перемещений;
7. потери из-за выпуска дефектной продукции.

Рассмотрим применение бережливых технологий в поликлинике. Их цель в данном случае – улучшить медицинское обслуживание. Будут применены следующие условия: информирование граждан о медицинском обслуживании в поликлинике (стенды, электронные табло, навигация по поликлинике), разделение потоков пациентов на условно «больных» и условно «здоровых», распределение потоков по цветовой гамме, увеличение доли наличия карт с 30% до идеального состояния в 100% в электронной базе.

Руководство городской больницы №1 г. Серова представило основные проблемы, влияющие на временные потери в сфере медицинского обслуживания (<https://clck.ru/UHQ7>).

В таблице отражены эти проблемы и их соответствующие решения, которые руководство больницы реализовало в нескольких своих отделениях.

Таблица.

Таблица проблем и решений

Проблемы	Решение
Низкий процент лиц, записывающихся на прием через ЕИЦ 30%.	Организация PR компании (бегущая строка, СМИ) об удобстве записи через ЕИЦ, Call-центр.
Отстаивание очереди в регистратуру растягивается до 30 мин.	Организация здоровой и лечебной регистратур с целью разведения потоков больных и условно здоровых пациентов.
Сложный для восприятия интерфейс электронного табло.	На электронное табло установить удобное программное обеспечение, с минимальным количеством символов.
Документы (амбулаторная карта, явка) на прием к врачу несут пациенты.	Подбор и оформление медицинской документации (амбулаторная карта, явка) регистраторами и передача непосредственно в кабинет врача (без участия пациента).
Отвлечение регистраторов от основной работы на поиск подбор карт.	Организация регистратуры, распределение обязанностей организаторов.
Терапевты ходят в регистратуру для получения информации о поступивших вызовах.	Внедрение электронного журнала вызовов врача на дом.

Самым важным ресурсом на любом производстве является время [4, с. 158]. Ожидание материала, ожидание из-за простоя машин, ожидание контроля качества, ожидание предшествующих или последующих технологических процессов, ожидание информации – эти и множество других причин потери времени сказываются на эффективности производственного процесса.

В качестве яркого примера временных затрат, рассмотрим проблему малого потока записей на прием к врачу через ЕИЦ. Запись через ЕИЦ дает экономию времени, но из-за малой осведомленности людей процент использования составляет всего 30%. В результате этого увеличивается нагрузка на отдел регистратуры в медицинском учреждении. К примеру, человек, записывающейся через регистратуру, тратит около 10-20 мин из-за ожидания в очереди. А запись через ЕИЦ занимает около 5-10 мин. Так же, сложное для восприятия интерфейс электронного табло. В среднем на него уходит от 2 до 5 мин. При внедрении оптимального ПО время поиска информации на табло сократится до 0,5-1 мин. Так же, документы, а именно амбулаторная карту на прием к врачу несут пациенты. На это уходит много времени. Около 15-20 мин в очереди за картой. При внедрении бережливых технологий, а именно подбор и оформление медицинской документации (амбулаторная карта, явка) регистраторами и передача непосредственно в кабинет врача (без участия пациента). Это полностью исключит ожидание пациентов, так как карта будет уже у доктора в кабинете.

Одним из показателей эффективности внедрения бережливых технологий в различные производственные сферы является показатель разницы во времени, отпущенного на тот или иной процесс. В данном случае будет рассматриваться изменение количества затраченного времени по описанным выше проблемам в медицинских учреждениях до внедрения бережливых технологий и после непосредственного их внедрения.

Для решения задачи с такими условиями необходимо следствие из теории непрерывных случайных величин о том, что вероятность того, что непрерывная случайная величина  $X$  примет какое-либо значение из интервала  $[a; b]$  равна определенному интегралу от ее плотности вероятности в пределах от  $a$  до  $b$  [2, с. 584]:

$$\int_a^b f(x)dx = P(a \leq X \leq b). \quad (1)$$

Формула математического ожидания НСВ имеет вид [2, с. 335]:

$$MX = \int_{-\infty}^{\infty} xf(x) dx. \quad (2)$$

Говоря об объявленных временных рамках, как о пределах интегрирования, и о том, что случайная величина, отражающая количество времени в данном случае, примет значение в рамках этих пределов с вероятностью 100%, зададим переменные  $y$ ,  $u$  и  $v$ . Эти три переменные будут использоваться для нахождения функций распределения и плотности вероятностей. Подставляя пределы интегрирования и, беря во внимание, что  $P(a \leq X \leq b) = 1$ , получим:

$$\int_{10}^{20} f(y)dy = 1, \int_2^5 f(u)du = 1 \text{ и } \int_{15}^{20} f(v)dv = 1.$$

Для нахождения искомого количества времени по каждому из трех случаев временных затрат, воспользуемся формулой математического ожидания НСВ (2) и обозначим каждое искомое количество времени переменными  $t_1$ ,  $t_2$  и  $t_3$  в соответствии с переменными  $y$ ,  $u$  и  $v$ .

$$t_1 = \int_{10}^{20} yf(y)dy, t_2 = \int_2^5 uf(u)du \text{ и } t_3 = \int_{15}^{20} vf(v)dv.$$

Для начала необходимо найти функции  $f(y)$ ,  $f(u)$  и  $f(v)$ . Учитывая, что определенные интегралы от них принимают значение 1, получаем, что для функций:

$$f(y): F(20) - F(10) = 1, \text{ откуда } \frac{20}{10} - \frac{10}{10} = 1 \text{ и } F(y) = \frac{y}{10};$$

$$f(u): F(5) - F(2) = 1, \text{ откуда } \frac{5}{3} - \frac{2}{3} = 1 \text{ и } F(u) = \frac{u}{3};$$

$$f(v): F(20) - F(15) = 1, \text{ откуда } \frac{20}{5} - \frac{15}{5} = 1 \text{ и } F(v) = \frac{v}{5}.$$

$$f(y) = F'(y) = \left(\frac{y}{20}\right)' = \begin{cases} 0 & \text{при } y < 10 \\ \frac{1}{10} & \text{при } 10 \leq y \leq 20 \\ 0 & \text{при } y > 20 \end{cases}$$



$$f(u) = F'(u) = \left(\frac{u}{3}\right)' = \begin{cases} 0 & \text{при } u < 2 \\ \frac{1}{3} & \text{при } 2 \leq u \leq 5 \\ 0 & \text{при } u > 5 \end{cases}$$

$$f(v) = F'(v) = \left(\frac{v}{5}\right)' = \begin{cases} 0 & \text{при } v < 15 \\ \frac{1}{5} & \text{при } 15 \leq v \leq 20 \\ 0 & \text{при } v > 20 \end{cases}$$

Получив значения функций  $f(y)$ ,  $f(u)$  и  $f(v)$ , найдем значения переменных  $t_1$ ,  $t_2$  и  $t_3$ :

$$t_1 = \int_{10}^{20} yf(y)dy = \int_{10}^{20} y \frac{1}{10} dy = \frac{400}{20} - \frac{100}{20} = 15;$$

$$t_2 = \int_2^5 uf(u)du = \int_2^5 u \frac{1}{3} du = \frac{25}{6} - \frac{4}{6} = 3,5;$$

$$t_3 = \int_{15}^{20} vf(v)dv = \int_{15}^{20} v \frac{1}{5} dv = \frac{400}{10} - \frac{225}{10} = \frac{175}{10} = 17,5.$$

В результате вычислений получили среднее значение временных затрат затронутых проблем. Для наглядности и дальнейшего рассмотрения разницы можно посчитать их сумму. Она составит 36 минут. Так как математическое обоснование внедрения решений в рассматриваемые проблемы предусматривает выявление разниц между затраченным временем до применения бережливых технологий и после, обозначим новые временные параметры переменными  $n_1$  и  $n_2$ , отражающие время процесса записи человека через ЕИЦ и время его взаимодействия с усовершенствованным электронным табло соответственно. И для дальнейшей работы с этими параметрами обозначим новые переменные  $w$  и  $s$  с функциями  $f(w)$  и  $f(s)$  соответственно. Первую очевидную временную разницу можно увидеть в исключении одного из временных параметров, так как у пациента отпадает необходимость ходить за картой в регистратуру. Воспользовавшись формулой математического ожидания (2) и новыми временными пределами, получим следующие два определенных интеграла:

$$n_1 = \int_5^{10} wf(w)dw \text{ и } n_2 = \int_{0,5}^1 sf(s)ds.$$

Аналогично нахождению функций  $f(y)$ ,  $f(u)$  и  $f(v)$  находим  $f(w)$  и  $f(s)$ :

$$f(w) \quad F(10) - F(5) = 1, \text{ откуда } \frac{10}{5} - \frac{5}{5} = 1 \text{ и } F(y) = \frac{y}{5};$$

$$f(s) \quad F(0,5) - F(1) = 1, \text{ откуда } 1 * 2 - 0,5 * 2 = 1 \text{ и } F(u) = 2s.$$

$$f(w) = F'(w) = \left(\frac{w}{5}\right)' = \begin{cases} 0 & \text{при } y < 5 \\ \frac{1}{5} & \text{при } 5 \leq y \leq 10 \\ 0 & \text{при } y > 10 \end{cases}$$

$$f(s) = F'(s) = (2s)' = \begin{cases} 0 & \text{при } u < 0,5 \\ 2 & \text{при } 0,5 \leq u \leq 1 \\ 0 & \text{при } u > 1 \end{cases}$$

Получив значения функций  $f(w)$  и  $f(s)$ , найдем значения переменных  $n_1, n_2$ :

$$n_1 = \int_5^{10} wf(w)dw = \int_5^{10} w \frac{1}{5} dw = \frac{100}{10} - \frac{25}{10} = 7,5;$$

$$n_2 = \int_{0,5}^1 sf(s)ds = \int_{0,5}^1 2sdu = 1 - 0,25 = 0,75.$$

Средние значения временных затрат составили 7,5 и 0,75 минут. Сумма составит 8,25 минут, то есть 8 минут и 15 секунд. Значение параметра  $n_1$  вдвое меньше, чем значение  $t_1$ , а разница между  $t_2$  и  $n_2$  составляет 2,75 минут. Общая разница между суммой  $t_1, t_2, t_3$  и суммой  $n_1, n_2$  составляет 27 минут и 45 секунд. Это говорит о том, что рассмотренные из предпринятых мер руководства городской больницы действительно оказывают положительный эффект на экономию времени в обслуживании пациентов.

Кроме того, существенный вклад в эту экономию дополнительно может внести организация рабочего места и условий труда, которые так же являются одним из инструментов бережливых технологий. К примеру, руководство детской поликлиники №7 г. Нижневартовска в рамках организации распределения потока пациентов применила в дизайне интерьера раскраску стен, окружающих определенную часть общей площади в разные цвета в зависимости от того, какого направления специалисты работают в прилежащих кабинетах. А также для экономии времени гардероб был установлен через смежную стенку от регистратуры.

Таким образом, пришли к пониманию выгоды применения бережливых технологий через математическое обоснование, применяя теорию непрерывных случайных величин и можно утверждать, что философия бережливого производства в ряде случаев имеет свое математическое начало. Существующие методы расчета инвестиций, затраченного времени и сырья позволяют крупным мировым компаниям существенно снижать затраты на производство и повышать его производительность.

### Литература

1. Джонс Д.Т., Вумек Д.П. Бережливое производство. М., 2020. 720 с.
2. Прохоров Ю.В. Вероятность и математическая статистика. М., 1999. 584 с.
3. Чернова В.А., Агеев И.Т. Концепция бережливого производства: неуклонное сокращение потерь // Молодой ученый. 2016. №26. С. 407-410.
4. Ермаков И.А. Экономические законы социализма и их использование. М., 1960. 426 с.

©Долгов А.Е., Крошилов А.Ф., Ключков А.А., 2021

УДК 519.687.7

**Прохоренков А.О.**

Научный руководитель: **Слива М.В.**, канд. пед. наук  
Низневартовский государственный университет,  
г. Нижневартовск, Россия

## СОЗДАНИЕ КОДОВОГО УСТРОЙСТВА НА ПЛАТФОРМЕ ARDUINO

В данной статье рассматривается создание кодового устройства на платформе Arduino, использующего двустрочный LCD-дисплей и 2 светодиода для вывода информации, матричную клавиатуру 4×4 для ввода и сервопривод, использующийся как блокирующее устройство.

Матричная клавиатура 4×4 (рис. 1) состоит из 16 кнопок, 4 ряда по 4 столбца. Изготовлена по пленочно-мембранной технологии. Для подключения к контроллеру на Arduino на клавиатуре имеется разъем, состоящий из 8 выводов типа «мама», с шагом 2,54 мм. Дорожки покрытия одного слоя нанесены горизонтально (выводы 5–8) (рис. 2), а другого вертикально (выводы 1–4) (рис. 2). Нажатие на кнопку приводит к соединению дорожки одного слоя с дорожкой другого и замыканию одного из выводов 1–4 с одним из выводов 5–8 (Матричная клавиатура 4×4 кнопки <https://clck.ru/Ndhim>; Матричная клавиатура 4×4 <https://clck.ru/UHQ9G>).

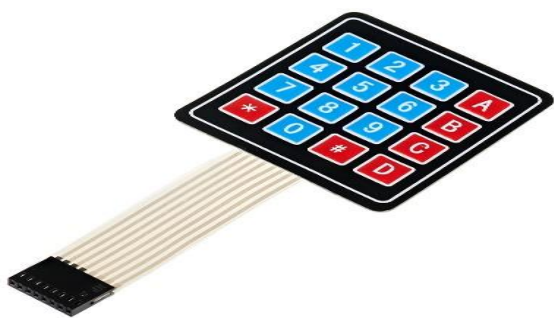


Рис. 1. Клавиатура 4x4

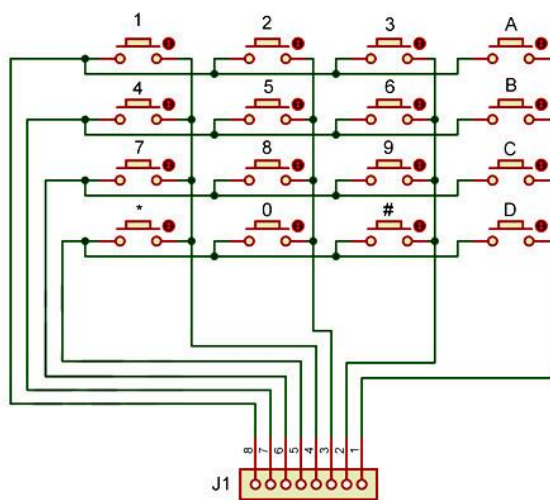


Рис. 2. Схема дорожек покрытия

Arduino LCD 1602 I2C дисплей (рис. 3) – жидкокристаллический, текстовый, двух строчный, 16 знакомест в каждой строке, цифровой I2C индикатор с подсветкой. Представляет из себя модуль Arduino. Каждое знакоместо имеет разрешение 8×5 точек. Общее количество точек экрана 1280 пикселей. Подсветка дисплея белая, светодиодная. Цвет жидких кристаллов темно-синий. Получается красивый белый текст на синем фоне. Дисплей основан на контроллере HD44780 и предназначен для отображения любой текстовой информации, в совокупности с Ардуино или другими контроллерами. Благодаря дополнительно

установленному I2C модулю (рис. 4) расширения портов на микросхеме PCF85741, дисплей стало очень просто подключить к любому микроконтроллеру. Схему подключения смотрите на рис. 5 (Дисплей LCD 1602 I2C. <https://clck.ru/UHQDC>; Текстовый экран 16×2 / I2C. <https://clck.ru/NdhmW>).



Рис. 3. LCD 1602 I2C дисплей

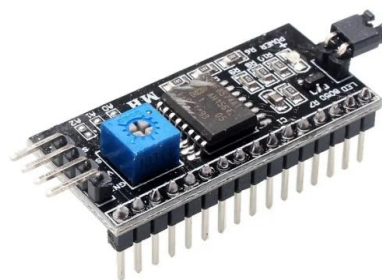


Рис. 4. Модуль I2C

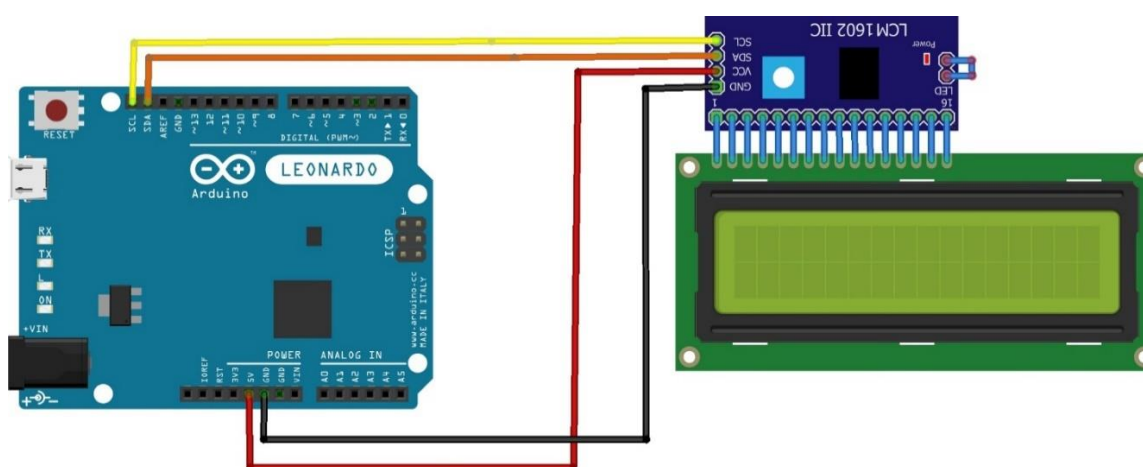


Рис. 5. Схема подключения LCD-дисплея с i2c к плате Arduino

Сервопривод (рис. 6) – это привод, вал которого может встать в заданное положение или поддерживать заданную скорость вращения. При включении электромотора запускается вращение выходного вала. К нему можно подключить или присоединить то, чем в дальнейшем планируется управлять. Сервопривод получает заданное значение, после этого сравнивает данное значение со значением на своем датчике. В случае расхождения блок управления стремится достичь и поддерживать заданное значение, чтобы оно по возможности совпадало со значением, которое поступает с датчика (что такое сервопривод (сервомотор) и как им управлять. <https://clck.ru/UHQBG>).



Рис. 6. Сервопривод

Для проекта нам понадобятся:

- Двухстрочный LCD-дисплей с i2c-переходником;
- Матричная клавиатура 4×4;
- Сервопривод;
- Красный и зеленый светодиод;
- Плата Arduino Leonardo;
- Провода.

Кодовое устройство может использоваться в различных целях, как для личных нужд, так и в развлекательных целях. Принцип работы устройства следующий: пользователь вводит пароль с матричной клавиатуры, если пароль верный, то загорается зеленый светодиод, показывающий пользователю правильность введенного им кода, на дисплее появляется соответствующее этому сообщение, а сервопривод делает поворот, тем самым открывая доступ. Если же введенный пароль оказался не верным, то загорается красный светодиод, а на экране появляется сообщение, что пользователю следует повторить попытку, затем экран очищается, а светодиод гаснет.

Опишем создание кодового устройства.

В первую очередь для работы с матричной клавиатурой, LCD-дисплеем и сервоприводом нам понадобятся специальные библиотеки – `<Keypad.h>`, `<LiquidCrystal_I2C.h>` и `<Servo.h>` соответственно.

Следующим шагом создадим переменную, хранящую наш пароль. Далее нам нужно задать значение каждой клавише клавиатуры (необязательно клавишам присваивать их предписанное значение), инициализировать дисплей и сервопривод.

После этого можно приступить к описанию проекта. Создадим пару служебных клавиш. Первая – для очистки экрана и сброса пароля, например, ‘C’, а вторая – для ввода (замена Enter), например, ‘D’. Затем создадим цикл, в котором будем последовательно считывать символы с клавиатуры и передавать их на дисплей, пока не будет нажата одна из служебных клавиш или не кончится свободное место на экране.

Теперь, если была нажата клавиша ‘C’, можно воспользоваться двумя функциями библиотеки `<LiquidCrystal_I2C.h>`:

`lcd.clear()` – для очистки экрана,

`lcd.setCursor(0,0)` – для установки курсора в начало первой строки, а также нам следует обнулить переменную, в которую считывалась информация с клавиатуры.

Если же нажата клавиша ‘D’, то необходимо выполнить проверку заданного изначально пароля и введенной пользователем последовательности, в случае ошибки вывести сообщение об этом, включить красный светодиод и вызвать срабатывание клавиши сброса, иначе выводится сообщение об успешном вводе пароля, инструкция к дальнейшим действиям, включается зеленый светодиод и сервопривод делает поворот на необходимую для открытия замка позицию.



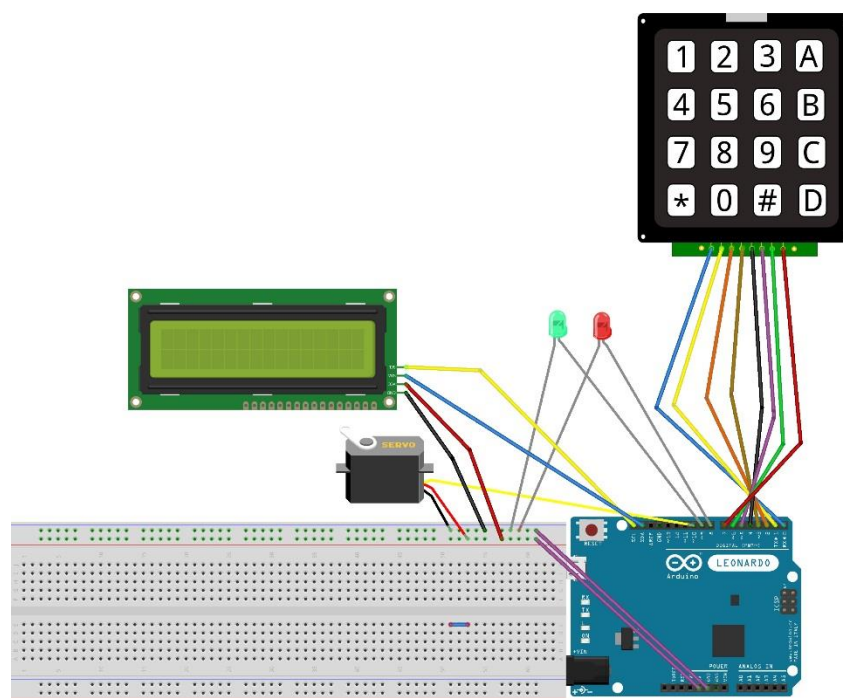


Рис. 7. Схема кодового устройства

Таким образом, рассмотрели создание кодового устройства на платформе Arduino, на основе LCD-дисплея с i2c-переходником, матричной клавиатуры 4×4, с сервоприводом и двумя светодиодами (рис. 7).

© Прохоренков А.О., Слива М.В., 2021



УДК 51-74

Семенова А.А., Лосева А.В.,  
Покорная И.Ю., канд. физ.-мат. наук  
Воронежский государственный педагогический  
университет, г. Воронеж, Россия

## ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ МАТРИЦ ПОВОРОТА В РОБОТОТЕХНИКЕ

Матричная алгебра имеет огромное значение в теории и практике математического моделирования. Большинство из нас знакомы с матрицами. Но далеко не всем известно их практическое применение. Так, например, матрицы широко используются в робототехнике. Так называемые матрицы поворота описывают положение робота или его детали в пространстве. Матричную запись применяют при систематическом и обобщенном описании расположения звеньев манипулятора, т. е. исполнительных механизмов робота, относительно заданной абсолютной системы координат [1; 2].

Звенья робота могут совершать движения нескольких видов: поступательные и вращательные. Говоря о движении последнего вида, стоит заметить, что для их описания необходима матрица поворота размерностью  $3 \times 3$ . Данная матрица переводит координаты вектора в повернутой системе координат  $OUVW$  в абсолютную систему координат  $OXYZ$ . (рис. 1).

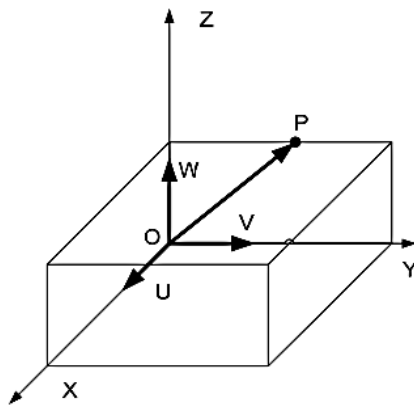


Рис. 1. Система координат OXYZ

Абсолютная система координат задана осями  $OX$ ,  $OY$ ,  $OZ$ . В свою очередь связанная система координат расположена на осях  $OU$ ,  $OV$ ,  $OW$ . Абсолютная система рассматривается как фиксированная, поэтому вращение происходит у связанной системы координат. Говоря о связанной системе координат очевидно, что она связана с каким-либо телом пространства и движется вместе с ним.

Для осей обеих систем координат  $OUVW$  и  $OXYZ$  рассмотрим единичные векторы  $(i_x, j_y, k_z)$  и  $(i_u, j_v, k_w)$  соответственно.

Выберем произвольную точку пространства  $P$ . Ее можно задать координатами любой из указанных систем. Пусть эта точка  $P$  фиксированна и неподвижна относительно системы

OUVW. Тогда в известных нам системах координат она будет иметь следующие координаты:  $P_{uvw} = (p_u, p_v, p_w)^T$  и  $P_{xyz} = (p_x, p_y, p_z)^T$ , где T указывает на то, что данные вектора транспонированы.

После того, как система координат OUVW будет повернута переход к системе OXYZ будет происходить по формуле  $P_{xyz} = R \times P_{uvw}$ , где R матрица перехода от одной системы координат к другой. Таким образом, чтобы знать как зависят координаты системы OXYZ от системы OUVW необходимо знать матрицу перехода.

Любой вектор системы можно представить в виде его линейной комбинации. Из этого получаем, что:

$$P_{uvw} = p_u * i_u + p_v * j_v + p_w * k_w \quad (1)$$

где  $p_u, p_v$  и  $p_w$  являются проекцией вектора P на оси системы координат.

Для нахождения проекций нашего вектора в системе координат OXYZ используется равенство (1). Так получаются следующие равенства:

$$p_x = i_x * p = i_x * i_u * p_u + i_x * j_v * p_v + i_x * k_w * p_w$$

$$p_y = j_y * p = j_y * i_u * p_u + j_y * j_v * p_v + j_y * k_w * p_w$$

$$p_z = k_z * p = k_z * i_u * p_u + k_z * j_v * p_v + k_z * k_w * p_w$$

Перепишем полученные равенства в матричной форме:

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} i_x * i_u & i_x * j_v & i_x * k_w \\ j_y * i_u & j_y * j_v & j_y * k_w \\ k_z * i_u & k_z * j_v & k_z * k_w \end{bmatrix} * \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix}$$

Из полученного матричного равенства видно, что матрица R принимает следующий вид:

$$R = \begin{bmatrix} i_x * i_u & i_x * j_v & i_x * k_w \\ j_y * i_u & j_y * j_v & j_y * k_w \\ k_z * i_u & k_z * j_v & k_z * k_w \end{bmatrix}$$

Аналогичным приемом пользуются для того, чтобы перейти от координат системы OXYZ к координатам системы OUVW:  $P_{uvw} = Q \times P_{xyz}$

$$\begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} = \begin{bmatrix} i_x * i_u & i_x * j_v & i_x * k_w \\ j_y * i_u & j_y * j_v & j_y * k_w \\ k_z * i_u & k_z * j_v & k_z * k_w \end{bmatrix} * \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$$

В силу коммутативности скалярного произведения получаем, что  $Q = R^{-1} = R^T$ . Стоит отметить, что при умножении матрицы R на матрицу Q получается единичная матрица.

Таким образом, выявлена зависимость между координатами одной системы и другой.

Наиболее простыми в своем понимании являются матрицы элементарных поворотов – такие матрицы, когда поворот системы координат OUVW происходит только по одной оси.

На рисунке 2 представлена ситуация, когда произошел поворот вокруг оси OX на угол  $\alpha=90^\circ$ .

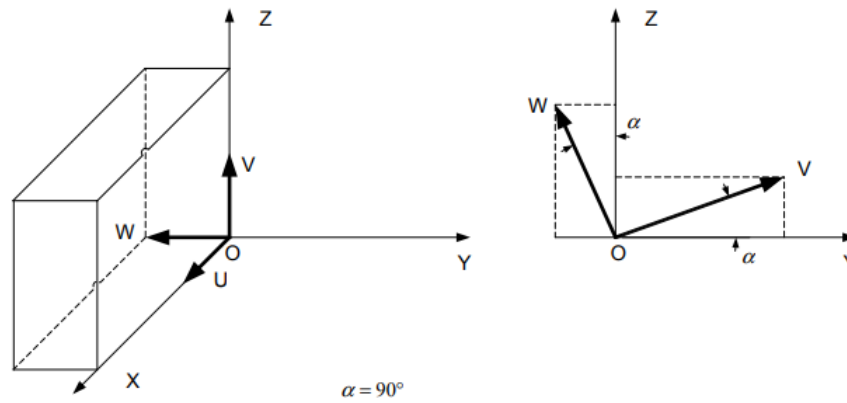


Рис. 2. Поворот вокруг оси OX на угол  $\alpha=90^\circ$

Рассмотрим эту ситуацию подробнее. Запишем уравнение перехода от одной системы координат к другой для матрицы  $R_{x,\alpha}$ :  $P_{xyz} = R_{x,\alpha} * P_{uvw}$ . Заметим, что  $i_x = i_u$ , тогда матрица  $R_{x,\alpha}$  примет следующий вид:

$$R_{x,\alpha} = \begin{bmatrix} i_x * i_u & i_x * j_v & i_x * k_w \\ j_y * i_u & j_y * j_v & j_y * k_w \\ k_z * i_u & k_z * j_v & k_z * k_w \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix}$$

Аналогично, для поворота вокруг оси OY на угол  $\varphi$ :

$$R_{y,\varphi} = \begin{bmatrix} \cos \varphi & 0 & \sin \varphi \\ 0 & 1 & 0 \\ -\sin \varphi & 0 & \cos \varphi \end{bmatrix}$$

А для поворота вокруг оси OZ на угол  $\theta$ :

$$R_{z,\theta} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Так как матрицы  $R_{x,\alpha}$ ,  $R_{y,\varphi}$  и  $R_{z,\theta}$  являются матрицами элементарных поворотов с их помощью можно получить матрицы конечных поворотов.

Рассмотрим примеры перехода из системы координат OXYZ в систему координат OUVW.

*Пример 1.* Пусть дан  $\overline{P_{xyz}} = (0; 1; 1)$ . Известно, что поворот происходит только вокруг оси OX, где угол поворота  $\alpha=45^\circ$ . Найти координаты  $\overline{P}$  в системе координат OUVW.

Так как переход из системы координат OXYZ в систему координат OUVW происходит по формуле  $P_{uvw} = Q \times P_{xyz}$ , то в нашем случае  $\overline{P_{uvw}}$  будет иметь следующие координаты:

$$\begin{aligned} \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} &= \begin{bmatrix} i_x * i_u & i_x * j_v & i_x * k_w \\ j_y * i_u & j_y * j_v & j_y * k_w \\ k_z * i_u & k_z * j_v & k_z * k_w \end{bmatrix} * \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} * \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 45^\circ & -\sin 45^\circ \\ 0 & \sin 45^\circ & \cos 45^\circ \end{bmatrix} * \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 1 & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ 0 & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} * \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \sqrt{2} \end{bmatrix} \end{aligned}$$

Таким образом, проекция вектора  $P$  в новых координатах на оси будет следующая: на ось  $OU$  равно  $0$ , на ось  $OV$  также равна  $0$ , на ось  $OW$  равна  $\sqrt{2}$ . Графическое решение данного задания приведено на рисунке 3.

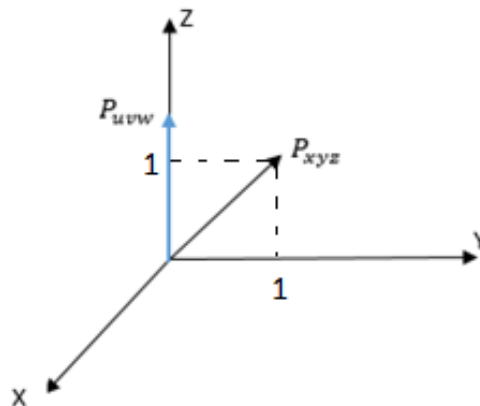


Рис. 3. Графическое решение примера 1

*Пример 2.* Пусть дан  $\overline{P_{xyz}} = \left(\frac{1}{2}; 0; \frac{\sqrt{3}}{2}\right)$ . Известно, что поворот происходит только вокруг оси  $OY$ , где угол поворота  $\varphi = 60^\circ$ . Найти координаты  $\bar{P}$  в системе координат  $OUVW$ .

Аналогично предыдущему примеру, найдем координаты вектора в новой системе координат.

$$\begin{aligned} \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} &= \begin{bmatrix} \cos \varphi & 0 & \sin \varphi \\ 0 & 1 & 0 \\ -\sin \varphi & 0 & \cos \varphi \end{bmatrix} * \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \\ &= \begin{bmatrix} \cos 60^\circ & 0 & \sin 60^\circ \\ 0 & 1 & 0 \\ -\sin 60^\circ & 0 & \cos 60^\circ \end{bmatrix} * \begin{bmatrix} \frac{1}{2} \\ 0 \\ \frac{\sqrt{3}}{2} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & 0 & \frac{\sqrt{3}}{2} \\ 0 & 1 & 0 \\ -\frac{\sqrt{3}}{2} & 0 & \frac{1}{2} \end{bmatrix} * \begin{bmatrix} \frac{1}{2} \\ 0 \\ \frac{\sqrt{3}}{2} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \end{aligned}$$

Таким образом, искомые координаты данного вектора в новом базисе  $\overline{P_{uvw}} = (1; 0; 0)$ . Из полученного ответа видно, что наш вектор будет расположен на оси  $OX$ .

Итак, в ходе проделанной работы, убедились, что матрицы поворота имеют огромное значение в сфере робототехники, ведь именно с помощью них можно рассчитать в каком положении необходимо находиться определенному звену робота. Получается, что с математической стороны движение робота определено именно матрицами поворота. На сегодняшний момент сфера робототехники активно развивается, поэтому данная тема является актуальной и перспективной.

### Литература

1. Фу К., Гонсалес Р., Ли К. Робототехника / Пер. с англ. М., 1989. 624 с.
2. Бортникова И.С., Покорная И.Ю. Матрица сложных поворотов и движения роботов / Некоторые вопросы анализа, алгебры, геометрии и математического образования. 2016. №4. С. 29-31.

© Семенова А.А., Лосева А.В., Покорная И.Ю., 2021

УДК 004.8

**Андреева А.Ю.**

*Научный руководитель: Государев И.Б., канд. пед. наук  
Национальный исследовательский университет ИТМО,  
г. Санкт-Петербург, Россия*

## **ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ: ОСНОВНЫЕ ПОНЯТИЯ, ПРОБЛЕМЫ И ПРОГНОЗЫ РАЗВИТИЯ**

В настоящее время происходит стремительное развитие IT-сферы, появляются новые современные проекты и открытия. Наша жизнь все больше совершенствуется, модернизируется и автоматизируется. Именно поэтому тема искусственного интеллекта очень актуальна на сегодняшний день.

Термин «искусственный интеллект» появился в 1956 году, но настоящей популярности технология ИИ достигла лишь недавно в связи с увеличением объемов данных, усовершенствованием алгоритмов, оптимизацией вычислительных мощностей и средств хранения данных.

В 50-х годах прошлого века начались первые исследования в области ИИ. Они были направлены на решение проблем и разработку систем символьных вычислений. В конце 50-х годов американскими учеными были созданы первые нейросети. Эта система моделировала человеческий глаз и его взаимодействие с мозгом. В середине 70-х годов произошел существенный прорыв в практических приложениях ИИ, когда на смену поискам универсального алгоритма мышления пришла идея моделировать конкретные знания специалистов-экспертов. В США появились первые экспертные системы. Пришел новый подход к решению задач искусственного интеллекта – представление знаний [1, с. 146].

Искусственный интеллект (ИИ) можно определить двумя разными способами. Первый способ определения ИИ приводится в словаре Мерриам-Вебстера: «это раздел информатики, связанный с моделированием интеллектуального поведения компьютеров» [2]. Второе определение, которое приводится в словаре, это «способность машины имитировать разумное поведение человека» [3, с. 1]. Еще одно определение, которое можно дать ИИ – это «новый элемент, который используется в бизнесе, чтобы упростить повседневные процессы и сделать их менее трудоемкими». ИИ может влиять по-разному на деятельность тех или иных компаний. Благодаря ИИ могут поддерживаться потребности бизнеса за счет автоматизации бизнес-процессов, анализа данных и взаимодействия с клиентами и сотрудниками.

Выделяют следующие типы ИИ:

- Автоматизированный (для решения рутинных задач);
- Вспомогательный (оптимизирует человеческие решения);
- Расширенный (направлен на поддержку человеческого мышления в экстремальных ситуациях);

- Автономный (осуществляет самостоятельную деятельность, подобную человеческому мышлению).

- По типу восприятия окружающей среды ИИ-системы можно разделить на:

- Реагирующие (лишь анализируют окружающую обстановку и формируют ответную реакцию);

- С ограниченной памятью (корректирует свое поведение с учетом предыдущего «опыта», как, например, беспилотные автомобили);

- С элементами разума (способны распознавать мысли и эмоции);

- «Сильный ИИ» (сопоставимый по мыслительным способностям с человеком).

В настоящее время выделяют такие подразделы искусственного интеллекта, как, машинное обучение, обработка естественного языка, нейронные сети, глубокое обучение, компьютерное зрение и когнитивные вычисления, при этом все они рассматриваются как составляющие элементы семейства ИИ. Часто, термин «искусственный интеллект» вместе с названиями данных подразделов считаются взаимозаменяемыми, но на самом деле все они помещаются под «большим куполом» ИИ.

В таком типе машинного обучения, как нейронная сеть, взаимосвязанные блоки являются обработчиками информации с учетом поступления внешних данных, а также, данные блоки обмениваются между собой информацией, по сути, как нейроны, откуда и происходит вышеуказанное название.

Первым разработчиком нейрокомпьютера является доктор Роберт Хехт-Нильсен, которому удалось определить нейронную сеть как вычислительную систему, которая состоит из простых, глубоко взаимосвязанных обрабатывающих элементов, обрабатывающих информацию с помощью динамического реагирования на внешние входные сигналы.

Сама идея вычислительной системы в виде нейронных сетей не нова, но только за последние несколько лет появилось достаточно вычислительных ресурсов, для того, чтобы продвинуть возможность разработки таких сетей.

Термины «искусственный интеллект» и «машинное обучение» иногда взаимозаменяются, что только подчеркивает важность второго определения для первого. Вместе с тем, это не совсем одно и то же, так как золотым стандартом искусственного интеллекта является именно машинное обучение, к тому же оно и наиболее активно развивается.

Под машинным обучением подразумевается использование алгоритмов для изучения данных, их анализа с последующим предсказанием или определением чего-либо. Так, если в программах, написанных в ручную, приводится четкий набор инструкций для того, чтобы выполнить конкретную задачу, то машина «обучается», используя большой объем данных и нужных алгоритмов, с помощью которых у нее появляется возможность научиться выполнять определенную задачу [4, с. 186].

Глубокое обучение использует мощные вычислительные ресурсы нейронных сетей и вычислительных устройств различных уровней для поиска закономерностей в больших массивах данных (например, для того, чтобы обнаруживать изображения). Эти модели,



которые относятся к подразделу машинного обучения, учатся на предоставляемых им данных, поэтому, чтобы добиться эффективности и точности, данных должно быть много. Когда вы используете обратный поиск Google Image или задаете вопрос Amazon Alexa, то ответ вы получаете благодаря глубокому обучению.

Глубокое обучение позволило создать множество практических приложений в области машинного обучения: восстановление изображений (DeepFace), распознавание контекста изображений (Yelp, Pinterest), боты для игр (DeepMind AlphaZero и AlphaGo, OpenAI), распознавание речи (Deep Voice), диагностика рака (IBM Watson) [4, с. 187]. Глубокое обучение разбивает задачи таким образом, что все виды машинных задач кажутся возможными.

Обработка естественного языка обозначает обработку языка, который используется людьми, например английский, а не языка программирования компьютера. В процессе обработки естественного языка такой язык используется для связи с программируемой системой искусственного интеллекта.

Обработка естественного языка включает две составляющие: понимание и генерацию. В процессе понимания сопоставляется ввод на естественном языке с полезными представлениями, при этом происходит анализ различных аспектов естественных языков. Под генерацией понимается процесс формирования значащего вывода в виде слов и предложений, она состоит в планировании текста и предложений, а также реализации текста.

В процессе когнитивных вычислений используется смесь машинного обучения, искусственного интеллекта, обработки естественного языка, нейронных сетей, анализа настроений и понимания контекста для решения повседневных проблем, как и люди. IBM определяет когнитивные вычисления как продвинутую систему, которая учится в масштабе, целеустремленно и взаимодействует с людьми в естественной форме [5, с. 207].

Когнитивные вычисления подражают процессу мышления человека. Их отличие от систем искусственного интеллекта, которые просто заботятся о данной проблеме, состоит в том, что они обучаются с помощью паттернов и предлагают людям предпринимать соответствующие действия, основанные на их понимании. В случае искусственного интеллекта система берет на себя полный контроль над процессом и предпринимает шаги для выполнения задачи или предотвращения сценария с использованием заранее определенного алгоритма [5, с. 207]. Очевидно, наиболее известный пример когнитивных вычислений - Watson, суперкомпьютер компании IBM.

В сфере искусственного интеллекта существует еще одно междисциплинарное поле, называемое компьютерное зрение. С его помощью компьютеры могут учиться на видеофильмах и изображениях, происходит автоматизация задач, для выполнения которых человек использует зрение, например, взглянуть на фотографию и определить, что на ней изображено. При этом нужно собрать, обработать, проанализировать и понять цифровые изображения и данные, которые в них содержатся.

Основными достоинствами искусственного интеллекта являются:

1. Извлечение с его помощью максимальной пользы из данных, которые сами становятся объектами интеллектуальной собственности при появлении самообучающихся алгоритмов. То есть, в определенном наборе данных уже есть все необходимые ответы, их нужно лишь найти с помощью технологий ИИ. В настоящее время важность информации играет существенную роль в обеспечении конкурентных преимуществ, при этом лидерство остается за тем, кто предоставляет наиболее точные данные при использовании одинаковых технологий.

2. Интеллектуализация существующих продуктов, которая заключается в том, что технологии искусственного интеллекта не являются отдельным независимым приложением, а встраиваются в существующее ПО, усовершенствуют его, как например, добавление технологии Siri в устройства Apple нового поколения. Таким образом, улучшение различных технологий, использующихся как в офисе, так и дома, начиная системами анализа данных о безопасности и заканчивая инструментами инвестиционного анализа, происходит за счет автоматизации, платформ для общения, «умных» компьютеров и ботов в сочетании с большими объемами данных.

3. Возможность автоматизации повторяющихся процессов обучения и поиска за счет использования данных. При этом, основное отличие искусственного интеллекта от роботизации, основанной на применении аппаратных средств, состоит в непрерывном и надежном выполнении глобальных многочисленных задач компьютеризации. Чтобы автоматизировать данный процесс, необходимо участие человека для правильной постановки вопроса и первоначальной настройке системы.

В настоящее время технологии искусственного интеллекта востребованы во многих отраслях человеческой деятельности. Особенно широко они применяются при оказании правовой помощи, в поиске патентов, в вопросно-ответных системах, при оповещении о рисках, а также в медицинских исследованиях.

Функционал ИИ часто применяется при расшифровке медицинских рентгеновских снимков в персонализированной медицине. О выполнении необходимых физических упражнений, переходе на более здоровый режим питания, о своевременном принятии лекарственных препаратов все чаще напоминают пользователям персональные медицинские помощники.

В промышленности искусственный интеллект позволяет производить анализ данных интернета вещей (IoT) от производства, получаемых с подключенного оборудования, а также прогноз загрузки и спроса, благодаря рекуррентным сетям, которые представляют собой особые сети глубокого обучения, использующиеся для работы с последовательными данными.

С помощью ИИ можно совершать онлайн покупки с индивидуальными рекомендациями, а также обсуждать условия покупок между клиентом и продавцом. Кроме того, технологии ИИ могут оптимизировать процессы управления товарными запасами и размещения товаров.

ИИ также помогает и в сфере спорта. Оптимизировать стратегию игры, расстановку игроков, и в целом, улучшить организацию игры помогают отчеты с камер и показатели датчиков, предоставляемые тренерам.

Несмотря на все возможности и достоинства использования ИИ, существуют некоторые проблемы в сфере применения искусственного интеллекта. Главное ограничение ИИ заключается в том, что обучение возможно только на основе данных, другими способами - невозможно. Это означает, что любые неточности в данных отразятся на результатах, при этом необходимо отдельно добавлять новые уровни анализа и прогнозирования.

Кроме того, без достаточного количества структурированных данных не получится создать полезное на практике решение с искусственным интеллектом. Многие отрасли, особенно госсектор, до сих пор полагаются на бумажные архивы, полная оцифровка которых займет время. Для бизнеса это означает, что недостаточно разработать софт для AI – сначала нужно получить доступ к данным. Исследование McKinsey Global Institute показало, что лидерами по внедрению искусственного интеллекта стали финтех и телеком, а среди отстающих – строительство, образование и туризм, в которых не хватает оцифрованных данных.

Современные системы ИИ призваны выполнять четко поставленные задачи. Если система предназначена для игры в покер, то она сможет разложить пасьянс или сыграть в шахматы. Функционал, настроенный на выявление мошеннических схем, не сможет предоставить правовую помощь или водить машину. И тем более, если система искусственного интеллекта имеет предназначение для выявления махинаций с налогами или претензиями по гарантиям, то она не сможет с той же степенью точности выявить мошенничество в сфере здравоохранения [6, с. 86].

Таким образом, данные системы обладают узко специализированным характером и в основном предназначены для выполнения одной определенной задачи, в отличие от многозадачности человеческого менталитета.

Искусственный интеллект пока не способен отделять правду от вымысла и бороться с дезинформацией. Хотя компания OpenAI уже создала искусственный интеллект для генерации убедительных «фейк ньюс», алгоритмы до сих пор распознают фейки хуже людей. Другое ограничение искусственного интеллекта: его неспособность распознавать эмоции в соцсетях. Этот недостаток мешает, в частности, эффективно решать проблему кибербуллинга.

Также, в настоящее время, самообучающиеся системы - не автономны. Образы технологий искусственного интеллекта, доступные человеку в кинотеатрах или на экранах телевизоров, все также представляют собой элементы фантастики. Но вместе с тем, все чаще встречается ПО, способное анализировать информацию повышенной сложности как для совершенствования конкретных навыков, так и для освоения новых.

И, пожалуй, самая большая проблема ИИ это то, что умные системы постепенно вытесняют людей с рынка труда. Машины приобретают навыки, которые когда-то казались присущими только человеку. В результате автоматизации общее число рабочих мест сократится на треть в течение 10 лет, человеку. В результате автоматизации общее число рабочих мест сократится на треть в течение 10 лет, прогнозирует исследовательская компания Gartner. Поэтому люди боятся развития ИИ. Но ИИ призван помогать людям,

автоматизировать деятельность, в какой-то мере упрощать, но никак не завладеть всем рынком труда.

Сразу в нескольких статьях предсказываются различные эффекты от развития «крупнейшей технологии 2020-х годов» - искусственного интеллекта (ИИ). Едва ли не самый неожиданный прогноз: все более широкое внедрение этой технологии породит повышенный спрос на выпускников философских и лингвистических факультетов. Причина: «удивительная слабость ИИ при принятии этических решений».

В DB Research убеждены, что «самой сложной проблемой при дальнейшем развитии ИИ станет задача давать ответы на нравственные вопросы, на которые невозможно ответить, и при этом учитывать различные моральные и культурные нормы».

К 2030 году ИИ проникнет во все сферы жизни, активно влияя на взаимодействие потребителей с брендами, технологиями, «умными» домами и даже близкими. ИИ также будет использоваться на всех корпоративных платформах и инструментах, помогая лучше понять клиентов и обеспечить более персонализированный опыт [7, с. 869].

До 2023 года искусственный интеллект и методы глубокого обучения заменят традиционное машинное обучение, как наиболее распространенный подход. Бренды, которые используют эти возможности лучше, чем их конкуренты, получают конкурентное преимущество.

По мере того, как все больше людей обращаются к голосовым устройствам “Siri”, “Alexa” и “Google Home” за необходимой информацией, обработка естественного языка (NLP) станет нормой для ее получения. И аналитика - не исключение. Обработка естественного языка не только обнаруживает новые идеи, но и упрощает наше взаимодействие с аналитическими технологиями.

Больше людей получают информацию, необходимую для создания убедительного жизненного опыта. Вместе с использованием машинного обучения и искусственным интеллектом обработка естественного языка является одной из наиболее перспективных разработок, улучшающих понимание поведения пользователей [7, с. 870].

Gartner прогнозирует, что искусственный интеллект, встроенный в аналитику и прочее маркетинговое программное обеспечение, к 2022 году освободит более трети сотрудников аналитических отделов в маркетинговых организациях. Это позволит им сосредоточиться на бизнес-приоритетах, а не тратить время на ручные процессы, такие как персонализация, оценка потенциальных клиентов, выявление аномалий, управление эффективностью маркетинга и отчетность [7, с. 870].

Подводя итог, хочется сказать, что, согласно мнению экспертов, ИИ достигнет своего пика уже к 2030 году. Мнения специалистов разделяются: одни считают, что подлинный ИИ – большая опасность для человечества, а другие считают, что это – спасение. Но одно известно точно: ИИ не замена человеку, и в обозримом будущем таковой не станет.

### Литература

1. Прицепов М.Ю. Искусственный интеллект. История развития // Наука и образование: прошлое, настоящее и будущее. 2019. С. 145-147.
2. Словарь Мерриам-Вебстера. <https://clck.ru/DSxuV>
3. Yates S., Walker A., Van Meter K. Artificial Intelligence. <https://clck.ru/UHTCY>
4. Чаткин В.В. Различия между искусственным интеллектом, машинным обучением и глубоким обучением // Материалы XXII Всероссийского экономического форума молодых ученых и студентов. 2019. С. 185-187.
5. Сухомлин Л.Р., Павлютенкова О.А. Анализ перспектив применения когнитивных вычислений // Инновационная стратегия развития современного предпринимательства. Материалы Международной научно-практической конференции. 2019. С. 207-210.
6. Трофимов Е.А. Проблемы искусственного интеллекта // Заметки ученого. 2020. №9. С. 83-90.
7. Кривоногова А.Е., Зарипова Р.С. Проблемы и перспективы развития индустрии искусственного интеллекта // Аллея науки. 2018. №1 (17). С. 869-871.

© Андреева А.Ю., Государев И.Б., 2021

УДК 004.023

**Бобко В.А.**

*Научный руководитель: Мамедли Р.Э., канд. техн. наук*  
Нижевартовский государственный университет,  
г. Нижневартовск, Россия

## МЕТОДЫ ГЕНЕТИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Генетическое программирование – это автоматическое создание программ с использованием генетических алгоритмов. В основе генетического программирования находится представление программы на высоком уровне абстракции. Это позволяет учитывать структуру создаваемой программы. В генетических алгоритмах через большое количество итераций достигается результат заданной точности.

В генетических алгоритмах роль основных строительных блоков играют строки фиксированной длины, тогда как в генетическом программировании эти строки разворачиваются в деревья, столь знакомые специалистам в области трансляции [2, с. 37].

Эволюционный поиск, с точки зрения преобразования информации – это последовательное преобразование одного конечного нечеткого множества промежуточных решений в другое. Само преобразование можно назвать алгоритмом поиска, или генетическим алгоритмом [1, с. 75]. Генетические алгоритмы эффективно используют информацию, накопленную в процессе эволюции.

Цель генетических алгоритмов:

- абстрактно и формально объяснить адаптацию процессов в естественной системе и интеллектуальной исследовательской системе;
- моделировать естественные эволюционные процессы для эффективного решения оптимизационных задач науки и техники.

Отличие генетических алгоритмов от других оптимизационных и поисковых процедур:

- работают в основном не с параметрами задачи, а с закодированным множеством параметров;
- осуществляют поиск не путем улучшения одного решения, а путем использования сразу нескольких альтернатив на заданном множестве решений;
- используют целевую функцию, а не ее различные приращения для оценки качества принятия решений;
- применяют не детерминированные, а вероятностные правила анализа оптимизационных задач.

Этапы генетического алгоритма схожи с этапами генетического программирования:

- начальная популяция создается случайно, но способы создания популяции могут быть специфичны для определенной задачи.
- наиболее распространенными методами отбора являются метод рулетки, ранговый метод, элитизм и турнирный метод. Метод рулетки состоит в том, что для каждой особи шанс



выжить прямо пропорционален значению ее функции приспособленности. При использовании рангового метода, шанс выжить прямо пропорционален рангу особи – ее порядку в отсортированном по значению фитнес-функции списку особей. Метод элитизма заключается в том, что отбрасываются все особи, кроме заданной доли наиболее приспособленных. Наконец, при использовании турнирного метода, выбираются две случайных особи. После этого с вероятностью  $p$  выживает более приспособленная особь, а с вероятностью  $1 - p$  – менее приспособленная. Процесс повторяется до тех пор, пока не останется заданное количество особей.

– скрещивание (англ. crossover) происходит следующим образом, берутся две особи, находится случайная хромосома и начиная с этой хромосомы у первого все хромосомы будут заменены на хромосомы второго и наоборот. Таким образом достигается получение двух совершенно новых особей.

– мутации – случайная замена одной хромосомы на случайную.

Основные исследования в области генетического программирования проведены Д. Коза. Генетическое программирование представляет собой одно из направлений эволюционного поиска и ориентировано в основном на решение задач автоматического синтеза программ. Хромосомы или структуры, которые автоматически генерируются с помощью генетических операторов, являются компьютерными программами различной величины и сложности.

В исследованиях по генетическому программированию применяется язык LISP, обладающий необходимыми для синтеза структур свойствами:

– LISP является синтаксически простым функциональным языком, программа на котором представляет собой рекурсивную функцию символьных выражений, состоящих из элементарных функций, условных операторов и операторов суперпозиции;

– обработка данных в LISP-программе сводится к объединению, делению и перегруппировке информации;

– LISP-выражения представляются древовидной структурой, форма и величина которой может динамически изменяться.

В генетическом программировании для представления программы используются различные структуры. В зависимости от структуры будут изменяться порядок выполнения программы, распределение и использование памяти.

В настоящее время используют следующие структуры:

– древовидная структура;

– линейная структура;

– графоподобная структура.

Чаще всего используется древовидная структура, представляющая программу в виде дерева в вершинах, которого находятся различные функции, переменные и константы.

При представлении программ в древовидной структуре, элементы дерева, которые не содержат потомков имеют значения переменных или констант и называются листьями. Значения остальных элементов принимают различные функции или математические операции и называются не листьями.

Например программа, представленная формулой  $a * b - c / (d + e)$ , будет иметь следующий вид:

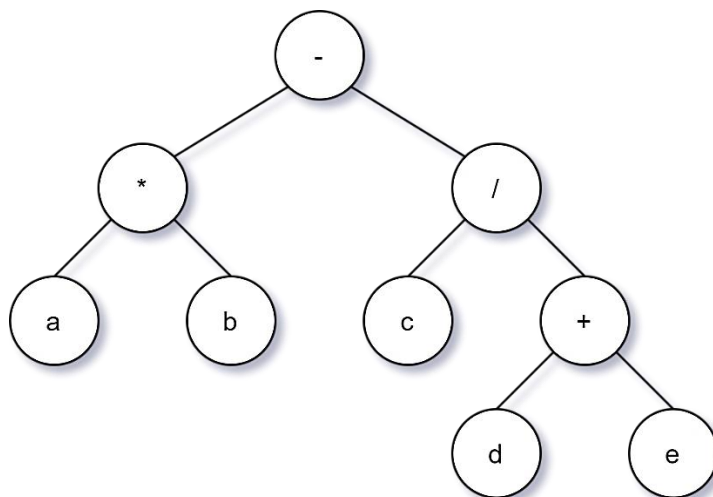


Рис. 1. Древоподобная структура

Для создания начальной популяции используется алгоритм ramped half-and-half. Он создает случайные деревья с ветвями случайной длины. Длины ветвей обычно ограничиваются.

На стадии отбора используется метод рулетки, который позволяет случайно выбирать кандидатов в зависимости от их оценки фитнес функцией.

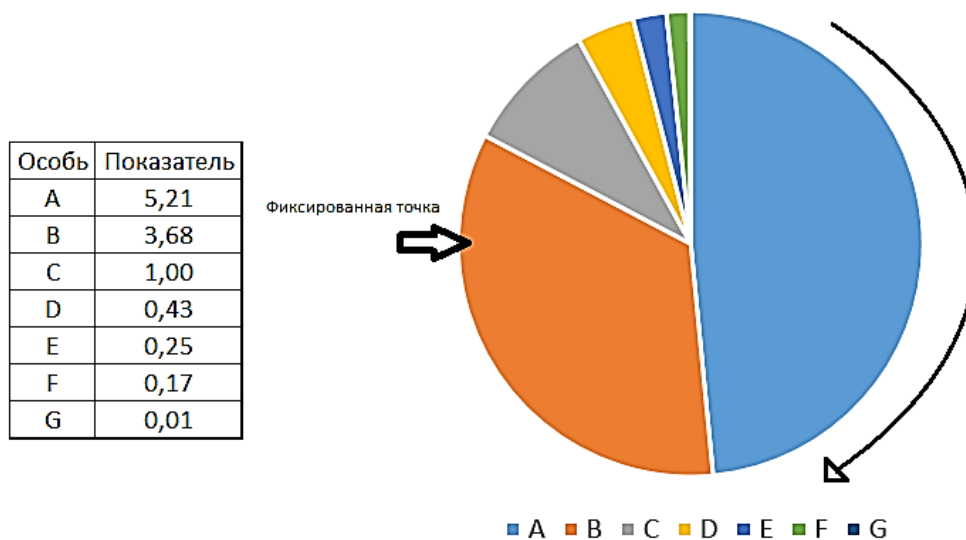


Рис. 2. Метод рулетки

При скрещивании деревья кандидаты обмениваются случайными узлами и всеми ветвями этих узлов.

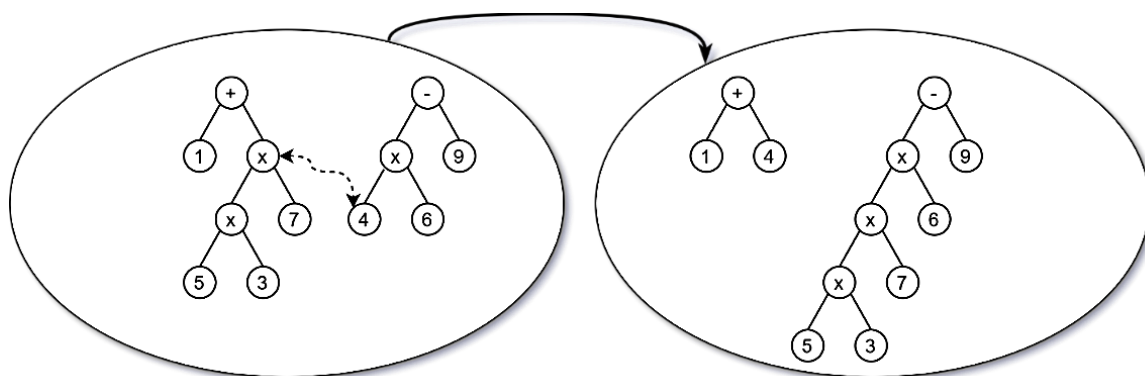


Рис. 3. Скрещивание бинарных деревьев

Мутация подразумевает создание случайных узлов, случайной длины из узла кандидата.

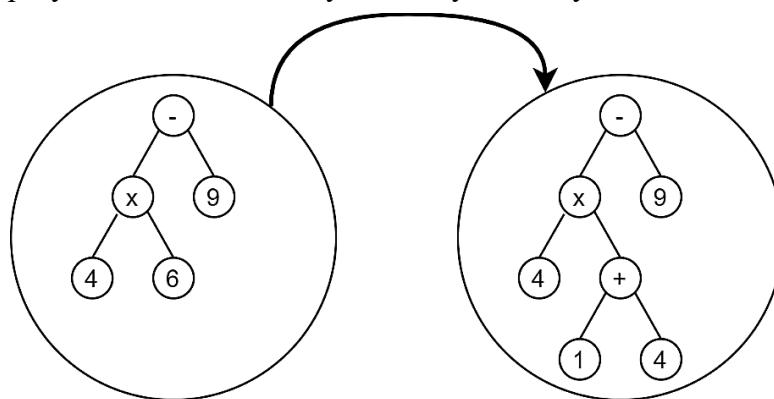


Рис. 4. Мутация бинарных деревьев

Цикл продолжается до тех пор, пока значение ошибки не достигнет заданного или произойдет заданное максимальное количество поколений.

### Литература

1. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы. М., 2010. 368 с.
2. Остроух А.В., Николаев А.Б. Интеллектуальные информационные системы и технологии. СПб., 2019. 308 с.

© Бобко В.А., Мамедли Р.Э., 2021

УДК 51-7

**Аджиева А.А., Чекалина Л.А.**

*Научный руководитель: Тихонова О.В.,* канд. физ.-мат. наук  
Рязанский институт (филиал) Московского политехнического  
университета, г. Рязань, Россия

## **РОЛЬ ИНТЕРЕНТ-ВЕЩЕЙ В РАЗВИТИИ ТРАНСПОРТНОЙ ЛОГИСТИКИ**

Современный этап развития общества характеризуется стремительной цифровизацией всех сфер жизнедеятельности человека, в том числе и производственной отрасли. Цифровизация – это процесс принятия новой формы подачи данных, рост использования цифровых или компьютерных технологий на отдельном предприятии, в отрасли или в целом по стране. Цифровизация создает условия для прорыва в научном, технологическом, социальном и экономическом развитии общества, и в то же время влечет за собой ряд неоднозначных изменений в сложившемся жизненном укладе. Применение принципа концепции «Интернет-вещей» позволит повысить эффективность организации взаимодействия между участниками цепочки ценностей. Особенностью предполагаемой концепции является организация эффективной информационной среды для обмена данными между различными системами предприятия.

Цифровая революция оказывает значительное влияние на уровень развития экономики и повышение качества жизни, порождает массовые изменения в общественной, управленческой и производственных сферах. Основными инструментами здесь являются: Big Data, машинное обучение [1, с. 453], нейронные сети, искусственный интеллект, человеко-машинные интерфейсы, виртуальная реальность, интернет вещей и роботизация.

Процессы цифровой трансформации проникают и в сферу логистики, оказывая влияние на скорость, качество и прозрачность логистических процессов, изменяя архитектуру логистических цепочек за счет уменьшения количества звеньев. Следует отметить, что цифровизация сферы логистики отстает в развитии по сравнению со сферами телекоммуникаций, средств массовой информации, банковских услуг и розничной торговли, несмотря на широкое внедрение математических и инструментальных методов в решение задач транспортной логистики [2, с. 1044].

Цифровая логистика – управление людскими, материальными, информационными и финансовыми потоками на основе их оптимизации для решения задачи минимизации затрат с применением современных информационных технологий.

Одной из основных задач логистики является мониторинг автопарка и товаров во всей продолжительности перевозки. IoT-технология, основанная на применении RFID-меток, обеспечивает такую возможность. При установке частной метки на каждый товар открывается возможность дистанционно проводить сканирование огромного количества меток, мгновенно совершать опись склада, или товара, находящегося в таре; применяя единственную RFID-метку на всем пути перевозки от производителя до продавцов.

На построение логистических объектов цифрового типа влияют: появление новых рынков, товаров, работ и услуг; внедрение новых способов управления; разработка новых нормативных актов, стандартов и регламентов информационной и коммуникационной цифровой среды; процесс роботизации; использование искусственного интеллекта, облачных хранилищ данных и др.

Учитывая всеобщее увлечение цифровыми технологиями, не стоит забывать, что технологии – это всего лишь инструмент, способ реализации конкретных целей, и нельзя ограничиваться только ими. Необходимо накапливать и научно обобщать опыт практического применения цифровых гаджетов и приложений, изучать цели, для которых они предназначены, учитывать характер изменений, которые они несут в нашу жизнь.

В цифровой экономике данные в цифровой форме становятся ключевым фактором производства, что повлияло и на формирование логистических процессов.

В эпоху цифровой трансформации стремительное развитие получили технологии, представленные на рисунке 1. Возникает вопрос: каким из современных цифровых технологий будет уделяться ключевое внимание в сфере логистики? Так, развитие цифровой логистики напрямую связывают с внедрением в практику хозяйственной жизни облачных вычислений (cloud computing) и связанных с ними технологий. Главной функцией облачных технологий является удовлетворение потребностей пользователей, нуждающихся в удаленной обработке данных (<https://clck.ru/Ak7gi>).

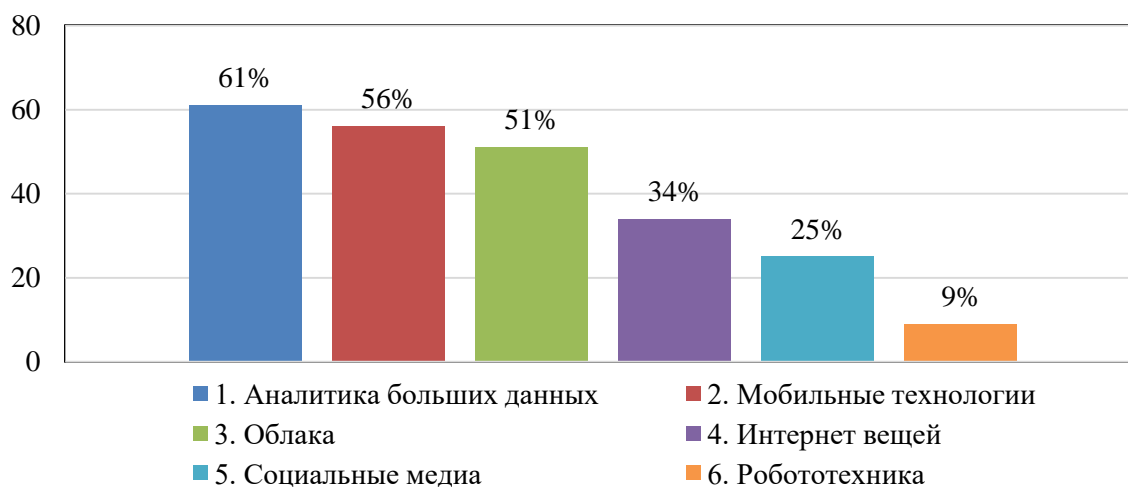


Рис. 1. Развитие технологий в ходе цифровой трансформации

Основными стратегическими направлениями развития цифровой экономики являются:

- связанность территорий РФ и интеграция в международные транспортные системы как ключевой фактор цифровой экономики;
- управление цепями поставок как ключевой фактор и драйвер для перестройки производства под требования цифровой экономики;
- формирование транспортных и логистических систем как ключевых факторов в экономической конкурентоспособности страны (<https://clck.ru/UQbUu>).

В рамках указанных направлений развития экономики определяются цели цифровой трансформации логистики:

- повышение транспортно-транзитного потенциала и внешнеторгового оборота РФ;
- внедрение и развитие смешанных (мультимодальных) перевозок;
- обеспечение доступности и качества транспортно-логистических услуг грузовых перевозок;
- обеспечение доступности и качества транспортных услуг для населения в соответствии с социальными стандартами;
- повышение эффективности управления транспортной инфраструктурой;
- создание платформы тарифного регулирования перевозок и оплаты проезда.

Цифровизация транспортного комплекса, до недавнего времени воспринимавшаяся как нечто далекое от реальности, сегодня уже является частью современных бизнес-процессов, на практике доказывая свою эффективность. «Интернет-вещей» предполагает, что каждое цифровое устройство имеет доступ в Интернет для передачи и получения данных. И, что самое главное, показывает, что за цифровой экономикой – будущее. Деятельность российского транспортного комплекса во всех сегментах имеет практические примеры перехода «на цифру», а процесс цифровизации ведется на базе общей платформы (<http://transportrussia.ru/item/4045>).

Цифровизация активно интегрируется в различные сферы жизни: экономику, телекоммуникации, культуру, средства массовой информации, логистику. Новые технологии меняют информационно-коммуникативное пространство страны, регионов и городов. Одной из последних тенденций на рынке транспортно-логистических услуг является применение самоуправляемых (автоматически управляемых) автомобилей. Технология автопилота активно внедряется в электрические грузовые машины Tesla Semi в рамках проекта Founders Edition, а в 2019 г начато серийное производство электрического грузового автомобиля с автопилотом.

В условиях динамично развивающихся цифровых технологий особенно актуальным является вопрос реформирования транспортного комплекса, неспособного при отсутствии интеграции всех видов транспорта предоставлять услуги высшего качества. Низкий уровень развитости дорожной инфраструктуры, значительная автомобилизация мегаполисов и нагрузка на улично-дорожную сеть, неразвитость пешеходной и велосипедной инфраструктуры, недостаток парковочных зон приводят к дисбалансу в городской транспортной системе. Несмотря на интенсификацию возможностей современного транспорта, на данный момент все еще актуальны такие проблемы, как загруженность маршрутов, нерегулярность рейсов, малая информированность пассажиров и, как следствие, сложность навигации, а также наличие конфликтных ситуаций в салоне во время следования транспорта.

Сегодня существует множество прикладных программных решений для совершенствования и интеллектуализации логистических процессов. Внедрение в отрасль новых цифровых технологий позволит создать систему информирования и коммуникации с



пассажирами различного вида транспорта не только с помощью традиционных каналов, таких как радио, телевидение, средства массовой информации, но и с помощью интернет технологий, а именно мобильных сайтов и приложений. Примером служит ориентированная на пассажиров метро служба «одного окна», куда можно направить жалобу или просьбу в адрес транспортных предприятий через форму обратной связи [3, с. 40].

Принцип концепции «интернет-вещей» позволяет улучшить процесс решения общей проблемы, перевести весь обмен данными в режим «онлайн», т.е. технологии обработки данных, в которых компьютерные ресурсы представляются интернет-пользователю как онлайн-сервис. Взаимодействие таких подсистем, как «умное производство», «умный склад» и «умный транспорт» могут вывести цифровую систему на принципиально новый уровень. На рисунке 2 наглядно изображена концептуальная схема интеллектуально интегрированной системы [4, с. 67].



Рис. 2. Интеллектуальная интегрированная система

С целью уменьшения холостых пробегов и повышения эффективности загрузки подвижного состава в транспортной логистике в последнее время активно применяется модель уберизации. Такое наименование является производным от названия компании Uber, разработавшей мобильное приложение, с помощью которого потенциальные клиенты могут устанавливать непосредственные контакты с поставщиками услуг в рамках пиринговых сделок.

Не менее важным вопросом является вариативность способов оплаты проезда в городском транспорте. Применение таких технологий, как Apple pay, Samsung Pay, Ali Pay, NFC, позволят ускорить и облегчить процесс оплаты, а создание универсальных проездных карт предоставит возможность беспрепятственного пользования мультимодальными перевозками.

Следует отметить, что ведущими компаниями транспортной отрасли в России было подписано соглашение о создании ассоциации «Цифровой транспорт и логистики» (ЦТЛ), в состав которой уже вошли РЖД, «Автодор», «Аэрофлот», «РТ-Инвест Транспортные Системы», «ЗащитаИнфоТранс», «Глосав» и «Деловые линии» (<https://clck.ru/UQbWF>). Целью ЦТЛ является создание и развитие единого мульти модального цифрового транспортного и

логистического пространства, основанного на применение IT-систем. Единая цифровая платформа на основе развитой отраслевой инфраструктуры позволит предоставить пользователям всех видов транспорта уникальные сервисы, повышающие безопасность, доступность и удобство.

Говоря о безопасности перевозок, в том числе городского транспорта, необходимо рассмотреть интеллектуальную транспортную систему (ИТС), появившуюся с приходом проблемы массовых скоплений транспорта на дороге вследствие увеличивающейся автомобилизации и плотности заселения территорий. ИТС – система управления, совмещающая современные информационные и телематические технологии с целью автоматизированного поиска и принятия эффективных решений управления транспортно-дорожным комплексом, а также предоставляющая конечным потребителям необходимую информацию и обеспечивающая безопасность, повышая взаимодействие участников движения. ИТС бывают простые (системы навигации, регулирования светофоров, оповестительных знаков, распознавания номеров) и сложные (системы, интегрирующие информационные потоки из различных источников). Российская интеллектуальная транспортная система на данный момент позволяет обеспечить:

- информирование водителей о нарушении ими правил дорожного движения;
- автоматическую фиксацию нарушений правил дорожного движения;
- создание условий для сокращения времени поездок пассажирами всех видов наземного транспорта;
- увеличение пропускной способности дорог;
- оптимизацию маршрутов движения.

Все это особенно актуально в условиях динамично развивающейся системы городского пассажирского транспорта. Единая цифровая платформа должна развиваться не только в масштабах страны, но и на уровне регионов и городов. Сейчас у правительства РФ есть цель – сформировать в России единое цифровое транспортное пространство, сделать пассажирские и грузовые перевозки более безопасными, удобными, доступными для людей и бизнеса, снизить издержки, расширить экспортные и транзитные возможности. Для решения этих задач предусмотрен ведомственный проект «Цифровой транспорт и логистика», который разрабатывается в рамках государственной программы по развитию транспорта.

Проблема цифровизации транспортной логистики особенно актуальна в наши дни, когда во всем мире идет борьба с пандемией коронавируса. В сложившейся кризисной ситуации с одной стороны произошло снижение объемов грузоперевозок в связи с сокращением объемов производства, с другой – вырос спрос на доставку интернет-покупок. Коронавирус является катализатором для реализации уже существующих проектов цифровой трансформации: автоматизации операций и внедрения электронного документооборота в сфере транспортных услуг. Бумажные билеты, наличная оплата проезда, накладные уступают свое место цифровым аналогам [5, с. 418].

Тенденции последних лет убедительно доказывают целесообразность внедрения цифровых технологий в транспортную систему. Цифровизация транспортных логистических

услуг влияет на повышение конкурентоспособности компаний, оперирующих на рынке. Цифровые технологии меняют каналы движения товаров, форматы поставки и процессы управления, позволяя добиться значительных успехов в повышении уровня логистического сервиса. Ожидаемый экономический эффект от внедрения интернета вещей, может быть отмечен через выгоду применения систем, которые позволяют наблюдать за активами, повышая управляемость, обеспечивая прозрачность на всем протяжении перевозки, а также выявляет причины потерь грузов и простои при их перевозке. Прямая экономия в логистике заключается в сокращении затрат на страхование товаров, а косвенная в перспективе уменьшения компаниями запасов.

Следует отметить, что внедрение цифровых технологий в транспортную сферу не является исключительной задачей для IT-специалистов. Цифровизация предполагает переход на новый уровень взаимодействия между контрагентами цепи поставок, создание принципиально новой системы функционирования бизнеса.

### Литература

1. Мамушина С.П., Гретчина П.А., Тихонова О.В. Практическое применение технологий машинного обучения // Новые технологии в учебном процессе и производства. Рязань, 2018. С. 453-455.
2. Миронова Е.И., Тихонова О.В., Сивиркина А.С., Азизян И.А. Математические и инструментальные методы оптимизации транспортной логистики // Экономика и предпринимательство. 2019. №2 (103). С. 1043-1050.
3. Рубас А.В. Системная модернизация информационно-коммуникативного пространства транспортного комплекса Москвы // МедиаАльманах. 2018. №2. С. 36-45.
4. Макарова И.В., Гиниятуллин И.А. Николаев Т.А. Влияние концепции интернет вещей на развитие логистических процессов // Социально-экономические и технические системы: исследование, проектирование, оптимизация. 2019. №3(82). С. 61-69.
5. Аджиева А.А., Чекалина Л.А., Тихонова О.В. Основные направления цифровизации транспортной логистики // Архитектурно-строительный и дорожно-транспортный комплексы: проблемы, перспективы, инновации. Омск, 2021. С. 715-719.

© Аджиева А.А., Чекалина Л.А., Тихонова О.В., 2021

УДК 519.683.8

**Кривошеков Д.С.**

*Научный руководитель:* **Слива М.В.**, канд. пед. наук  
Нижевартовский государственный университет,  
г. Нижевартовск, Россия

## ИСПОЛЬЗОВАНИЕ HIBERNATE ДЛЯ ХРАНЕНИЯ ЗАПИСЕЙ

Зачастую при написании приложения очень важной задачей является хранение данных в ORM-представлении. Эту задачу решает библиотека Hibernate, разработанная компанией Red Hat в 2001 году для кроссплатформенного языка Java. Данная библиотека является решением задач объектно-реляционного отображения, а также самой популярной реализацией спецификации Java Persistence API. JPA – это спецификация Java EE и Java SE, она предоставляет возможность сохранять Java-объекты в базе данных в удобном виде. Сама Java не содержит ее реализации, однако существует множество реализаций JPA от разных компаний, в том числе и Hibernate от компании Red Hat. Hibernate – это некий слой доступа к данным, который позволяет сократить количество кода для обращения к базе данных, автоматически обновляет и генерирует наборы таблиц и записей в ней. Тем самым уменьшает время разработки приложения, исключая написание JDBC-кода и запросов на языке SQL (<https://clck.ru/UJN2Y>).

Вкратце процесс настройки проекта состоит из следующих шагов:

1. Создание проекта с помощью Maven сборщика;
2. Подключение зависимости в файле «pom.xml»;
3. Создание и настройка конфигурационного файла «hibernate.cfg.xml»;
4. Создание схемы работы с базой данных.

Чтобы начать использовать Hibernate, создадим проект в любой среде разработки с помощью Maven сборщика и в файле «pom.xml» укажем зависимость для подключения данной библиотеки. Таким образом, сборщик скачает нужную библиотеку и автоматически подключит ее. Обновляем конфигурационный файл сборщика.

```
<dependency>  
  <groupId>org.hibernate.orm</groupId>  
  <artifactId>hibernate-core</artifactId>  
  <version>6.0.0.Alpha6</version>  
</dependency>
```

Следующим этапом будет создание и настройка конфигурационного файла «hibernate.cfg.xml». Конфигурация написана на расширяемой разметке XML, что ощутимо облегчает читаемость и написание конфигурации для библиотеки (<https://clck.ru/UJN33>). Создаем конфигурационный файл в папке resources и используем данный пример, изменив некоторые параметры под себя.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
    "http://hibernate.org/dtd/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
    <session-factory>
        <property name="hibernate.hbm2ddl.auto">update</property>
        <property
name="hibernate.connection.driver_class">com.mysql.cj.jdbc.Driver</property>
        <property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/messenger?serverTimezone
=Europe/Moscow</property>
        <property name="hibernate.connection.username">root</property>
        <property name="hibernate.connection.password">root</property>
        <property name="hibernate.connection.pool_size">10</property>
        <property name="">thread</property>
        <property name="hibernate.show_sql">>true</property>
        <property
name="hibernate.dialect">org.hibernate.dialect.MySQL8Dialect</property>
    </session-factory>
</hibernate-configuration>
```

Далее рассмотрим основные свойства конфигурационного файла:

- `hibernate.connection.driver_class` – свойство указывает драйвер для «общения» с базой данных;
- `hibernate.connection.url` – свойство указывает url, по которому идет обращение к базе данных;
- `hibernate.connection.username` – свойство указывает логин для подключения к базе данных;
- `hibernate.connection.password` – свойство указывает пароль для подключения к базе данных;
- `hibernate.connection.pool_size` – свойство указывает максимальное количество подключений к базе данных через библиотеку («пул» подключений);
- `hibernate.show_sql` – свойство указывает, показывать ли в окне вывода SQL запросы, которые генерируются автоматически;
- `hibernate.dialect` – свойство указывает диалект общения с базой данных.

Теперь отдельно рассмотрим свойство `hibernate.hbm2ddl.auto`. Данное свойство отвечает за способ работы с схемой подключения к базе данных, и тут можно указать один из представленных ниже параметров:

- `validate` – проверяет схему и запрещает вносить какие-либо изменения в базу данных;
- `update` – изменяет схему подключения к базе данных, позволяет вносить изменения в базу данных (создавать, изменять, удалять таблицы);
- `create` – создает схему и удаляет предыдущую;
- `create-drop` – создает схему и удаляет ее при явном закрытии класса `SessionFactory`, как правило при закрытии приложения;



- none – ничего не делает со схемой и не вносит изменения в базу данных.

После настройки библиотеки следует создать прототип класса, который будем хранить в виде записи в таблице базы данных. Для класса, хранимого в базе данных, есть только одно строгое правило - у него должен быть конструктор по умолчанию [2]. Также нужные поля будем указывать с помощью аннотаций. Аннотации – это что-то вроде меток в коде, которые описывают метаданные для класса, функции, пакета и т.д. Это позволяет указать библиотеке как работать с данным прототипом класса и хранить сущность в базе данных. Для облегчения понимания работы Hibernate с классом приведем простой пример сущности «User», написанной на языке Java, которую будем хранить в базе данных, а в дальнейшем напишем для нее элементарные CRUD операции:

```
@Entity
@Table(name = "User")
public class User {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "id", nullable = false)
    private long id;

    @Column(name = "login", nullable = false)
    public String login;

    @Column(name = "password", nullable = false)
    public String password;

    public User() {}

    public User(String login, String password) {
        this.login = login;
        this.password = password;
    }
}
```

Каждая аннотация над полем или классом имеет свое предназначение:

- @Entity – указывает, что данный класс является сущностью;
- @Table – указывает на имя таблицы, которая будет находиться в базе данных;
- @Id – указывает, что данное поле является идентификатором для записи в таблице;
- @GeneratedValue – указывает, что данное поле будет генерироваться автоматически по указанной стратегии (strategy – стратегия генерации значения);
- @Column – указывает на имя колонки для этой сущности, которое будет отображаться в таблице (name – имя, nullable – может ли поле быть пустым).

В Hibernate принцип работы основан на так называемых сессиях, и чтобы их создать используется шаблон проектирования «фабрика». Данную особенность покажем на конкретном примере:



```
public class HibernateSessionFactory {  
  
    private static SessionFactory sessionFactory;  
  
    public static SessionFactory getSessionFactory() {  
        if (sessionFactory == null) {  
            try {  
                Configuration configuration = new Configuration().configure();  
                configuration.addAnnotatedClass(User.class);  
                StandardServiceRegistryBuilder builder = new  
StandardServiceRegistryBuilder();  
                builder.applySettings(configuration.getProperties());  
                sessionFactory = configuration.buildSessionFactory(builder.build());  
            } catch (Exception e) {  
                e.printStackTrace();  
            }  
        }  
        return sessionFactory;  
    }  
}
```

Зачастую для создания сессий используют отдельные классы - так называемые «фабрики», в примере можем наблюдать, что сначала создаем конфигуратор, который читает наш файл конфигурации «hibernate.cfg.xml», и добавляем в его конфигурацию ранее созданный класс «User». Далее создается класс-строитель, который получает свойства, прочитанные конфигуратором, и на основе этих свойств создает «фабрику» сессий, которая в дальнейшем будет по нужному нам вызову возвращать сессию для работы с базой данных. Полная схема архитектуры Hibernate изображена на рисунке.

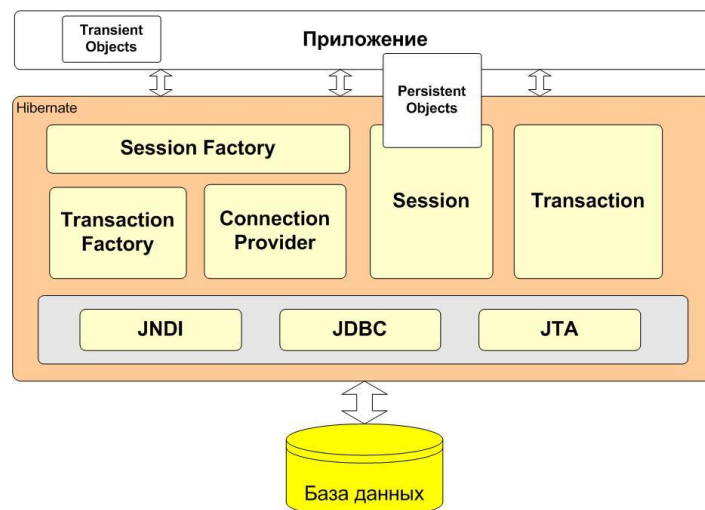


Рис. Схема архитектуры Hibernate

Также сессии используют транзакции для операции над записями в базе данных. Этот способ безопаснее, но операция займет намного больше времени. В следующих примерах, в которых будут показаны базовые CRUD операции над записями, использованы транзакции:

```
User user = new User("root", "root");
Session session = HibernateSessionFactory.getSessionFactory().openSession();
session.beginTransaction();
session.save(user);
session.getTransaction().commit();

user.login = "qwerty";
session.beginTransaction();
session.update(user);
session.getTransaction().commit();
session.close();
```

Данный код создает пользователя с определенными полями и записывает их в базу данных, но в процессе работы программы одно из его полей изменяется, и это изменение тоже фиксируется с помощью метода «update(user)». Также обязательно нужно выполнять метод «commit()», который сохраняет изменения, проведенные над записями.

Следующий пример продемонстрирует чтение записи из базы данных по полю «id»:

```
Session session = HibernateSessionFactory.getSessionFactory().openSession();
session.beginTransaction();
User user = (User) session.get(User.class, 0);
session.getTransaction().commit();
session.close();
```

Последний пример продемонстрирует удаление ранее созданной записи из базы данных:

```
Session session = HibernateSessionFactory.getSessionFactory().openSession();
session.beginTransaction();
session.delete(user);
session.getTransaction().commit();
session.close();
```

Таким образом можно увидеть, что в коде нет никаких SQL-запросов, а время, затраченное на создание, удаление, чтение и обновления записи в базе данных сведено к минимуму. В данной статье описаны простейшие примеры работы с Hibernate, но если приложение является крупным коммерческим проектом, то польза от библиотеки будет колоссальная. Всю работу с низкоуровневым кодом Hibernate берет на себя и не занимает дорогостоящее время программиста.

© Кривошеков Д.С., Слива М.В., 2021

УДК 004

**Пикуль М.А., Бощенко Е.С.**

*Научный руководитель: Матолыгин А.А.*

Томский университет систем управления  
и радиоэлектроники, г. Томск, Россия

## **СПОСОБЫ КОММУНИКАЦИИ ПРИ УПРАВЛЕНИИ ВНЕШНИМИ УСТРОЙСТВАМИ**

На текущий момент управление внешними устройствами не осуществимо без управляющего программного обеспечения, которое позволяет определять действия, передаваемые от пользователя к аппаратной части системы. В свою очередь, действия несут некоторую информацию для обработки данных и дальнейшей работы системы. Способы коммуникации являются первым вопросом при реализации передачи данных.

Коммуникация между машиной и человеком может происходить как по кабельным (проводным) каналам связи, так и по радиоканалам (беспроводной путь радиоволн). Для первого варианта плюсом является ограниченность доступа из внешней среды к передаваемой информации. Отрицательными факторами являются ограниченность радиуса передачи данных и финансовая затратность, а также ограничено количество связей: одна машина – один пользователь.

Беспроводная передача данных (беспроводная связь) – это перенос информации от одного устройства к другому, которые находятся на определенном расстоянии, без участия проводного подключения. При использовании беспроводной связи возможно расширить радиус действия коммуникации и увеличить число пользователей для одной единицы аппаратной составляющей системы. Ограничение доступа к передаваемой информации реализуется путем настройки шифрования данных.

В сегодняшнем этапе развития беспроводной связи существует множество путей передачи данных: радиоканалы сотовой связи, спутниковые каналы, инфракрасные каналы, лазерные каналы. Каждый из них используется в определенной области и обладает индивидуальными свойствами. Беспроводные сети передачи данных отличаются своими характеристиками, поэтому минимальное и максимальное расстояние между устройствами, в зависимости от вида технологии передачи информации, будет различно. Для синхронизации устройств по радиоканалу устанавливаются специальные адаптеры, которые способны отправлять и получать информацию. Передача осуществляется посредством каналов разных частот и диапазонов, а также за счет протоколов беспроводной передачи данных.

Протоколом передачи данных является определенный набор правил или соглашений интерфейса логического уровня, который определяет обмен данными между различными программами. Установленные правила задают единообразный способ передачи сообщений и обработки ошибок. На основе определенных протоколов созданы такие технологии как Wi-Fi,

Bluetooth, ZigBee, WiMax, LTE и другие. В настоящее время наибольшей популярностью из всех беспроводных технологий передачи данных пользуются Wi-Fi и Bluetooth.

Bluetooth использует максимально низкое потребление энергии при рабочей частоте 2,4 ГГц, обладает слабой мощностью передатчика, небольшой дальностью действия до 100 метров и низкой скоростью передачи данных. Низкая себестоимость чипов передачи данных не слишком отражается на повышении цены на систему. Данная технология предоставила большое разнообразие беспроводных аксессуаров: наушники, колонки, компьютерные мыши, клавиатуры и многое другое. Система беспроводной передачи данных на базе технологии Bluetooth удобна для связи устройств с маленьким расстоянием друг от друга.

Wi-Fi технология обеспечивает высокую скорость передачи данных при рабочей частоте 2,4 ГГц, отличается большим расстоянием действия сетей (до 5 км на открытом пространстве). Принцип действия технологии Wi-Fi следующий: в аппарат встраивается чип, который может дать надежную беспроводную синхронизацию с другим таким же чипом. Если устройств больше, чем два, тогда необходимо использовать роутер или точку доступа Wi-Fi – беспроводной аналог стационарного роутера. При этом появляется возможность подключить сразу несколько устройств. При использовании большого количества девайсов скорость передачи данных будет значительно снижена. Для защиты данных сети Wi-Fi роутер и точку доступа защищают шифрованием.

После выбора способа передачи данных необходимо установить за счет каких средств будет производиться управление системой. Данными способами управления служат мобильное приложение, web-приложение или desktop-приложение.

Мобильное приложение – программный пакет для планшетов и смартфонов, который имеет определенный функционал и дизайн «заточенные» под возможности мобильных платформ. Основными плюсами являются лучшее взаимодействие с клиентом через сообщения, пуш-уведомления, напоминания; выполнение функций в фоновом режиме; поддержка ряда функций при отключенном Интернете. Недостатком являются зависимость от операционной системы, установка приложения на каждое устройство, где будет использоваться функционал, требование обновлять приложение при изменениях кода.

Desktop-приложение – полнофункциональные программы, которые работают вне зависимости от других приложений и требуют наличие оператора. Для работы необходимы достаточные аппаратные ресурсы компьютера, на котором осуществится размещение программы и работа в ней. Такие приложения могут обходиться без подключения к Интернету, взаимодействуют посредством стандартного интерфейса, имеют более высокое быстродействие, зависят от используемой операционной системы и требуют установку на каждый компьютер пользователя, желающего работать с данным приложением. Так как имеется доступ к системным файлам программы, desktop-приложения более уязвимы и полностью зависят от действий пользователя.

Web-приложение – клиент-серверное приложение, в котором клиент взаимодействует с web-сервисом при помощи браузера. Логика и функционал web-приложения построен через личный кабинет, ограничение доступа, расчеты, формирование отчетов и др. Данные

программы работают на нескольких платформах, независимо от ОС или устройства; не устанавливаются на жесткий диск, что устраняет ограничения по месту; сокращают расходы как для бизнеса, так и для конечного пользователя, поскольку требуется меньше поддержки и обслуживания, а также снижает требования к компьютеру или смартфону пользователя. Все клиенты получают доступ к одной и той же версии проекта, что устраняет проблемы совместимости. Серверная часть приложения находится на стороне разработчика, поэтому пользователи не могут взломать лицензионное ограничение, что предотвращает компьютерное пиратство.

За выбором управления следует выбор конкретных протоколов передачи данных в рамках выбранного средства коммуникации. В связи с предыдущими решениями определяются дополнительные необходимые средства хранения и обработки информации и создается архитектура приложения.

В рамках проекта, целью которого является обеспечение безопасности человека путем предотвращения возникновения пожара от электрооборудования в бытовых помещениях, разработана система управления внешними устройствами с применением коммуникации через канал связи Wi-Fi web-приложения и системы удаленного контроля напряжения на линиях электроэнергии в бытовом помещении.

Аппаратная часть системы основана на аналоге платы Arduino, ESP8266-D1 Arduino Compatible Development Board. Данная плата имеет микроконтроллер с интерфейсом Wi-Fi – ESP8266 и не требует дополнительного подключения Wi-Fi модуля. Подключение других функциональных модулей схоже с подключением их к Arduino Uno. Преимуществом данной марки аппаратных составляющих является расширяемость системы при добавлении дополнительных универсальных модулей управления и контроля, а также при написании нового, кардинально отличающегося по функционалу программного обеспечения на одни и те же аппаратные составляющие. При невозможности подключить к плате устройства с большой мощностью принято решение использовать одноканальный модуль реле закрытого типа SRD-05VDC-SL-C на каждой линии электроэнергии.

Функциональный код для платы прописывается в интегрированной среде разработки Arduino IDE от создателей платформы. Язык программирования устройств Arduino основывается на упрощенном языке C/C++. Для приема и передачи информации через Wi-Fi-соединение подключается библиотека ESP8266WiFi.h, включающая в себя также функции доступа к аппаратным ресурсам и базовые функции контроля и управления платой и подключенными модулями, некоторые из которых описаны ниже.

Функция `setup()` используется для настройки платы. Микроконтроллер прорабатывает весь код, который содержится между фигурными скобками после `setup()` только один раз.

Функция `loop()`, используется для постоянного прорабатывания каких-либо процессов в плате. Как следует из названия `loop()`, весь код между фигурными скобками в данной функции повторяется снова и снова, а слово `loop` переводится как «цикл». Другими словами, функция `loop()` – это место, где находится тело программы.



Для написания web-программы удаленного управления системой существует два способа: написание кода web-программы непосредственно в среде программирования Arduino IDE или разработка полноценного проекта с backend и frontend разработкой.

Для двух вариантов после инициации последовательного соединения, определения скорости передачи данных и начальных значений датчиков системы программируется Wi-Fi модуль в пользовательской функции загрузки. Для обозначения к какой сети необходимо подключиться Wi-Fi модулю в режиме станции используется функция `WiFi.begin(ssid, password)`, где `ssid` – название WiFi сети, `password` – пароль от WiFi сети. Функция `WiFi.status()` возвращает информацию о статусе соединения станции с WiFi сетью. Затем необходимо вызвать функцию `server.begin()`, осуществляющую запуск и работу веб-сервера – специальной программы, обеспечивающей работу сайтов и прочих ресурсов, доступных по протоколу `www`. В основной части программы вызывается функция `server.available()` – вылавливает клиента, подключенного к серверу и имеющего данные, доступные для считывания. Соединение сохраняется, даже если клиент исчез из поля зрения. Однако соединение всегда можно закрыть вызовом функции `client.stop()` – отключает клиента от сервера. Функция `client.connect()` – осуществляет подключение к указанному IPадресу и порту. Возвращает значения о том, успешным ли было подключение или нет.

Для разработки web-приложения в среде разработки Arduino IDE определены такие возможности, как авторизация, подача и выключение электроэнергии на подключенных к системе линиях электросети. Авторизоваться может только пользователь, данные о котором внесены в память программы на аппаратной части системы. В общей или в отдельной пользовательской функции необходимо отметить разметку `DOCTYPE HTML` перед использованием логических функций языка программирования HTML, а затем в этой же функции продолжить писать код. Разметку и логические функции языка программирования HTML необходимо передавать в среду разработки Arduino IDE через функцию `client.println(val)` – передает информацию `val` от платы к клиенту на сервере. Сервером в данном случае является сама плата. После завершения описания функционала web-программы необходимо отключить клиента от сервера через вышеуказанную функцию.

При разработке полного проекта web-приложения нет необходимости описывать структуру web-приложения через основную функцию при программировании платы в Arduino IDE. Данные параметры прописываются непосредственно в функциях и на языках программирования backend и frontend разработок.

Backend разработка – это набор аппаратно-программных средств, при помощи которых реализована логика работы web-приложения. Одной из задач разработки backend является разработка API (Application Programming Interface – интерфейс программных приложений) – набор готовых классов, функций, процедур, структур и констант. Вся информация предоставляется самим приложением (или операционной системой). В общем случае данный механизм применяется с целью объединения работы различных приложений в единую систему. Для данного проекта API необходимо для связи программы в аппаратной части системы и программы web-приложения. Языком программирования выбран JavaScript с



использованием расширения Node.js. Такое решение принято по ряду причин: возможность применять JavaScript в разработке клиентской части web-приложения и в backend части; возможность создания приложения с неблокирующим вводом/выводом, который способен обрабатывать несколько запросов одновременно; гибкость языка программирования; стремительное развитие технологии; возможность взаимодействовать с разнообразными базами данных.

Базы данных предоставляют инструменты для организации, добавления, поиска, обновления, удаления и выполнения вычислений над данными. При рассмотрении существующих баз данных, подходящих для разработки API, выбор сделан в пользу MongoDB – документно-ориентированная система управления базами данных с открытым исходным кодом. Преимущества перед другими базами данных: динамическая схема, позволяющая гибко работать со схемой данных без необходимости изменять сами данные; масштабируемость – MongoDB горизонтально масштабируема, что позволяет легко уменьшить нагрузку на сервера при больших объемах данных; удобство в управлении; скорость – высокая производительность при выполнении простых запросов; поддержка всех языков программирования. В рассматриваемом проекте база данных используется для хранения данных с датчиков системы.

Для разработки интерфейса программного приложения между датчиками и клиентской частью веб-приложения, использовались следующие библиотеки и веб-фреймворки:

- Express – это минималистичный и гибкий веб-фреймворк для приложений Node.js, предоставляющий обширный набор функций для мобильных и веб-приложений;
- Mqtt.js – библиотека с открытым исходным кодом, разработанная для работы с протоколом MQTT;
- Mongodb – официальный драйвер MongoDB для Node.js, предоставляет высокоуровневый API поверх mongodb-core.

Описанные библиотеки, предоставляют необходимые для разработки API классы, методы, функции, константы, некоторые из которых описаны ниже.

Функция `callback()`, используется для повторного подключения платы ESP8266 к брокеру MQTT. Функция `reconnect()`, повторно подключает ESP8266 к брокеру MQTT, если этого не произошло. Функция `millis()`, возвращает время в миллисекундах, прошедшее с начала выполнения программы на плате Arduino. Функция `analogRead()`, считывает значение с указанного аналогового входа. Функция `publish()`, обеспечивает передачу данных с модулей и датчиков, брокеру MQTT.

Для формирования клиентской части веб-приложения использовался фреймворк Svelte. Этот фреймворк, являющийся компилятором, который превращает компоненты в автономные JavaScript модули.

Фреймворк Svelte выбран благодаря его маршрутизации. В отличие от React Router, маршрутизация в проекте, созданном с использованием фреймворка Svelte осуществляется за счет файлов разрабатываемого проекта в маршрутном каталоге проекта, каждый из которых является отдельным компонентом, то есть, самостоятельно обрабатываемым модулем,

который можно включать в другие модули проекта. Это обеспечивает удобство навигации в крупных проектах при их разработке. Ключевые особенности фреймворка Svelte: каждая web-страница приложения является компонентом фреймворка, новые web-страницы создаются путем добавления компонентов в директорию src и routes проекта. Web-страницы приложения считываются, обрабатываются и отрисовываются на сервере, передавая клиенту пакет web-страниц, уже готовый для отображения, следовательно, время загрузки web-приложения для пользователя значительно быстрее, чем при обработке страниц на стороне клиента. Также преимуществом является объем поставляемого кода. Svelte имеет способность сжимать написанный код в наиболее оптимальный JavaScript код, который легко подвергается обработке сервером.

Полный функционал приложения включает в себя возможности регистрации, авторизации, выхода из учетной записи, подача и выключение электроэнергии на подключенных к системе линиях электросети.

Регистрация подразумевает под собой, что пользователь, который еще не был зарегистрирован на сайте, внесет свои логин и пароль в общую базу, получив при этом возможности зарегистрированного пользователя. Регистрация уже зарегистрированного пользователя невозможна, а ее попытка приведет к возникновению ошибки.

Авторизоваться может только зарегистрированный пользователь. Авторизация возможно двумя способами: после регистрации происходит автоматическая авторизация; с помощью формы авторизации. Авторизация при помощи формы авторизации возможна, как только пользователь введет логин и пароль, указанные в базе данных. Если пользователь укажет логин и пароль, не совпадающие в комбинации с логином и паролем ни у одного из зарегистрированных пользователей, на web-странице формы авторизации высветится ошибка. После авторизации гость попадает в учетную запись пользователя.

В учетной записи находится возможность выхода из учетной записи. Это делается при помощи кнопки, расположенной в верхнем правом углу экрана. Кнопка появляется только у авторизованных пользователей, заменяя собой кнопку авторизации. При нажатии на кнопку выхода сервис перестает хранить сессию текущего пользователя и возвращает ему статус гостя.

Таким образом, коммуникация внешних устройств разнообразна по технологии передачи данных, протоколу связи, средствам программирования управления, которые включают в себя функционал управления, язык программирования, среду разработки.

© Пикуль М.А., Бощенко Е.С., Матолыгин А.А., 2021

УДК 004

**Гардт В.Е.**

*Научный руководитель:* **Казиахмедов Т.Б.**, канд. пед. наук  
Низhevартoвский государственный университет,  
г. Низhevартoвск, Россия

## АНАЛИЗ API И АВТОМАТИЗАЦИЯ ИХ ТЕСТИРОВАНИЯ

API (Application Programming Interface пер. с англ. интерфейс прикладного программирования) - это набор процедур, протоколов и инструментов взаимодействия, для создания приложений. То есть API - это набор программных функций, которые могут быть выполнены другой программой.

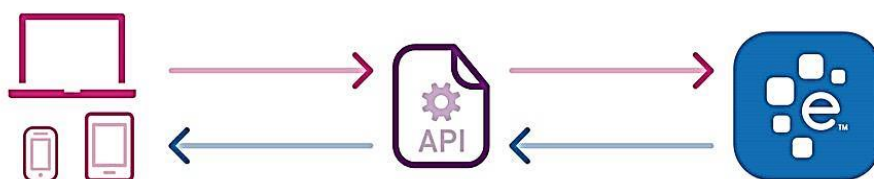


Рис. 1. Простой API

API определяет, как одна программа должна взаимодействовать с другой, действует как интерфейс между ними, позволяет им общаться. А тестирование этих функций как раз называется тестированием API.

В общем выделяют четыре общих архитектурных стиля веб-API:

HTTP для связи клиент-сервер;

XML / JSON для форматирования;

Простой URI как адрес для сервисов;

Коммуникация без указания статуса[1].

Во время тестирования подразделений и API обеих целевых исходных кодов, если метод API использует код, основанный на .NET, инструмент, который его поддерживает, должен иметь .NET

Можно использовать такие средства автоматизации для тестирования интерфейсов программных приложений:

NUnit для .NET

JUnit для Java

HP UFT

Soap UI

Так, например, самой популярной платформой по продаже авто является «AUTO.RIA». Этот ресурс является лидером автомобильной интернет-торговли уже несколько лет подряд. В частности, «AUTO.RIA» единственный ресурс, который предоставляет свой прикладной программный интерфейс в открытом доступе с возможностью отсылать тысячу запросов в час

бесплатно. Однако, API предоставляет доступ к общей базе платформы, то есть пользователь может выполнять операции только над данными, которые отображаются на сайте [2].

Для того, чтобы пользоваться прикладным программным интерфейсом AUTO.RIA, необходимо перейти на веб-страницу <https://developers.ria.com/> и зарегистрироваться. Также здесь можно найти подробную документацию о том, как пользоваться данным прикладным программным интерфейсом.

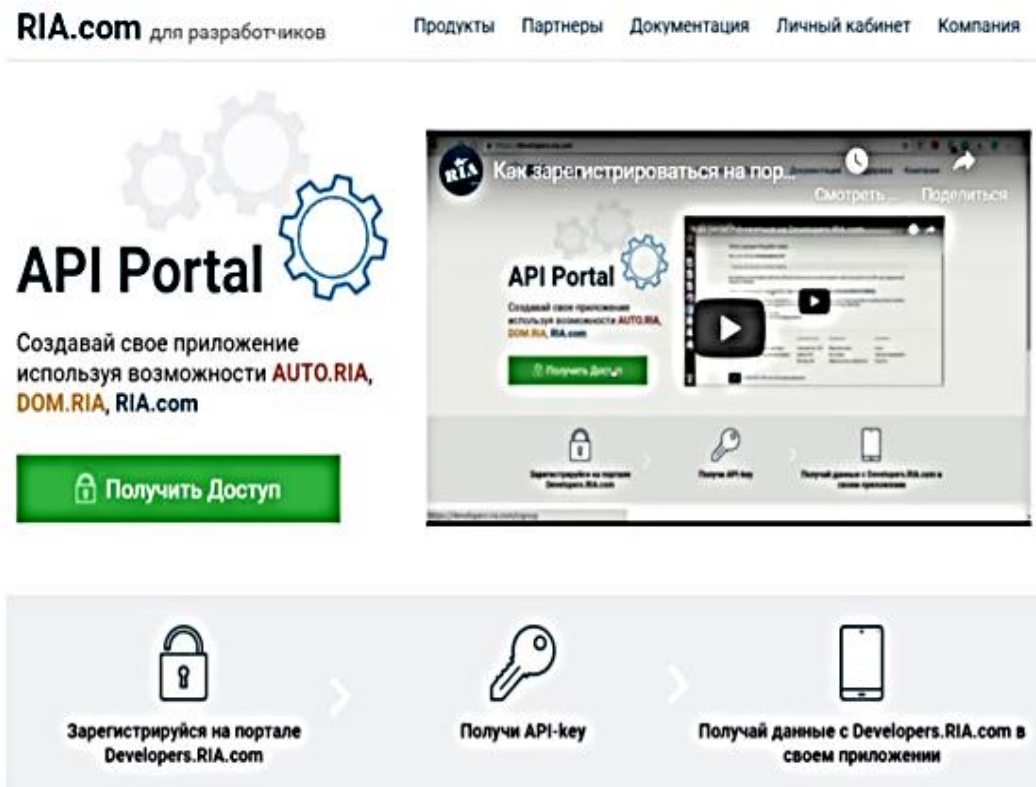


Рис. 2. Главное окно портала для разработчиков AUTO.RIA

После регистрации, в личном кабинете пользователя будет доступен API-ключ, который необходимо использовать при работе с прикладным программным интерфейсом AUTO.RIA, чтобы пройти аутентификацию.

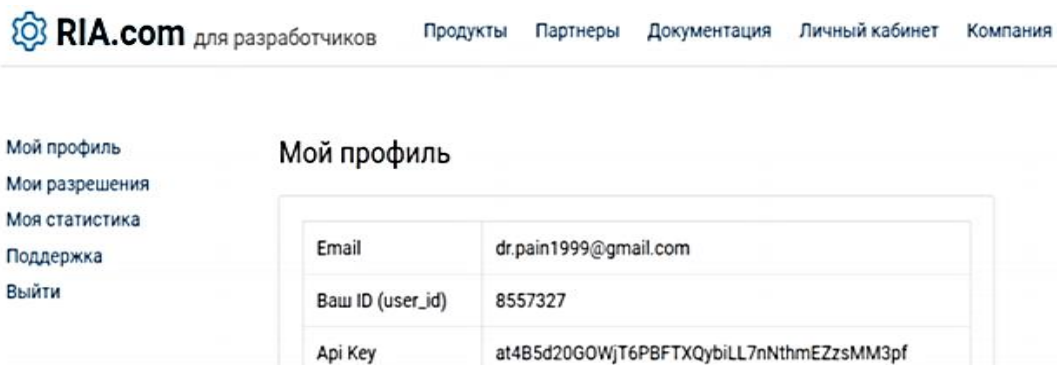


Рис. 3. Личный кабинет разработчика

Теперь API ключ можно использовать, чтобы получить данные, используя прикладной программный интерфейс AUTO.RIA. Почасовые счетчики API-ключа аннулируются на постоянной основе. Например, если пользователь выполняет 500 запросов в 10:15 и 500 запросов

в 10:25, то ключ будет временно заблокирован и любые действия с API будут недоступны. Блок будет снят через час, в 11:15, после чего пользователь сможет отправить еще 500 запросов. Если ключ превышает разрешенный лимит запросов, то пользователь получит в ответ HTTP-статус 429 (слишком много запросов). Если пользователь хочет увеличить количество разрешенных запросов к API для своей определенных потребностей, ему рекомендуют связаться с представителями AUTO.RIA и обсудить коммерческое сотрудничество.

Прикладной программный интерфейс AUTO.RIA можно использовать как разработчик, используя API ключ и работать с ним непосредственно, или же использовать сайт AUTO.RIA как обычный пользователь.

Обращаться до прикладного программного интерфейса веб-приложения можно несколькими способами, в частности:

- Поле для URL веб-браузера;
- Утилита командной строки «cUrl»;
- Приложения для отправки HTTP-запросов «Postman».

Используем Postman, чтобы отправить запрос в API AUTO.RIA, поскольку это приложение имеет понятный пользователю интерфейс и широкий функционал для настройки и отправки HTTP-запросов. Также Postman может конвертировать ответ сервера в разные форматы, что помогает отразить данные в удобном для чтения виде.

Выполним запрос, который позволяет искать объявления по заданным параметрам, то есть поиск. Для того, чтобы обратиться к поиску в API AUTO.RIA, необходимо выполнить HTTP-запрос с методом GET такого вида:

[https://developers.ria.com/auto/search?api\\_key=YOUR\\_API\\_KEY&PARAMETERS](https://developers.ria.com/auto/search?api_key=YOUR_API_KEY&PARAMETERS)

или:

```
curl -i -g -
```

```
X GET https://developers.ria.com/auto/search?api\_key=YOUR\_API\_KEY&PARAMETERS
```

при использовании утилиты cURL.

Отметим, что максимальное количество отраженных id объявлений за один запрос равно 100. В случае, если результат превышает это число, можно добавить значение параметра page (порядковый номер страницы), с помощью которого можно разбить результат на части и посмотреть.

Пример ответа на успешный запрос, который был описан выше:

```
[
  {
    "additional_params": {
      "lang_id": 2,
      "page": 0,
      "view_type_id": 0,
      "target": "search",
    },
    ...
  },
  "result": {
    "search_result": {
      "ids": [
        "26956399",
        "25855206",
        ...
        "26874853"
      ],
      "count": 1,
      "last_id": 0,
      "qs": {
```

API Testing обычно требует меньше времени, чем функциональное тестирование графического интерфейса (GUI testing). Веб-элементы в процессе тестирования графического интерфейса обрабатываются медленнее. В частности, автоматизация тестирования API требует меньше кода, чтобы обеспечить лучше и быстрее тестовое покрытие по сравнению с автоматизацией GUI тестов графического интерфейса пользователя. А это экономия средств для проекта тестирования.

### Литература

1. Application Programming Interface <https://clck.ru/UJPAh>
2. Купер А., Рейман Р., Кронин Д. Алан Купер об интерфейсе. Основы проектирования взаимодействия. СПб., 2019. 688 с.
3. Чарнецки К., Айзенекер У. Порождающее программирование. Методы, инструменты, применение. СПб., 2015. 736 с.

© Гардт В.Е., Казиахмедов Т.Б., 2021



УДК 004.94:37

**Забудская М.П.**

*Научный руководитель:* **Никонова Е.З.**, канд. пед. наук  
Нижевартовский государственный университет,  
г. Нижевартовск, Россия

## ИНСТРУМЕНТЫ АВТОМАТИЗАЦИИ РАБОТЫ ЛОГОПЕДА-ДЕФЕКТОЛОГА ДОШКОЛЬНОГО УЧРЕЖДЕНИЯ

В настоящее время 1,7 млн. детей, проживающих в Российской Федерации, т. е. 4,5% всей детской популяции, относятся к категории детей с ограниченными возможностями здоровья (ОВЗ) и нуждаются в специальном образовании. В это число входит более 353 тыс. детей дошкольного возраста; 63,6% таких детей находятся в дошкольных образовательных учреждениях вместе с обычными детьми. Также можно заметить, что рост детей с нарушениями речи растет с течением времени (рис. 1). Занимаются с детьми, имеющими отклонения в развитии, коррекционные педагоги – дефектологи. Работников этого профиля можно встретить не только в детских садах и школах. Их можно найти в реабилитационных и социальных центрах, поликлиниках.



Рис. 1. Рост нарушений речи

Проблема автоматизации рабочего места логопеда-дефектолога дошкольных образовательных учреждениях приобретает сегодня все большее значение. Образовательная практика учителя-дефектолога убедительно доказывает, что все основное время уходит на обучение детей и практически не остается времени на заполнение отчетной и методической документации. Это сказывается на том, что многим работникам приходится выполнять эту работу сверхурочно, в свое нерабочее время. Неоценимую помощь в этом оказывает активное применение информационно-коммуникационных технологий (ИКТ).

В этой работе мы будем рассматривать упрощенную версию той части структуры, в которой находится одно из звеньев данной структуры, деятельность которого понадобится в ходе работы – логопед-дефектолог.

Часть структуры предметной области, содержащая звено, для которого будет производиться автоматизация, является линейной (рис. 2). Линейная структура управления имеет ряд следующих достоинств:

- четкие системы взаимосвязи между руководителем и подчиненными;
- быстрота реакции в ответ на прямые указания;
- личная ответственность руководителя за конечные результаты предприятия.

Данная структура управления представляет организацию как набор взаимосвязанных элементов. Каждый отдельный элемент имеет свои цели и задачи. Заведующий дошкольным учреждением управляет заместителем заведующего по воспитательно-методической работе, имея при этом представление о деятельности психолога, логопеда-дефектолога и воспитателей, стоящих в конце иерархии.



Рис. 2. Организационная структура дошкольного учреждения

Логопед-дефектолог участвует в построение системы коррекционно-развивающей работы в логопедических группах для детей с общим недоразвитием речи в возрасте с 5 до 7 лет, предусматривающей полное взаимодействие и преемственность действий всех специалистов дошкольного образовательного учреждения и родителей дошкольников.

Направления деятельности:

- изучение уровня речевого, познавательного, социально-личностного, физического развития и индивидуальных особенностей детей, нуждающихся в логопедической поддержке, определение основных направлений и содержание работы с каждым ребенком;
- коррекционно-развивающее – создание условий, направленных на коррекцию речевого развития детей и обеспечивающих достижение ребенком, имеющим нарушения речи, уровня речевого развития, соответствующего возрастной норме;
- профилактическое – создание условий для повышения компетентности педагогов и родителей в вопросах развития речи детей, социально-эмоциональных и познавательных потребностей и возможностей развития;
- информационно-методическое – создание условий для освоения и внедрения инновационных технологий в области коррекции речевых нарушений, позволяющих повысить результативность коррекционного процесса в целом.

В настоящий момент в работе учителя-дефектолога в основном используются следующие программные продукты:

- Microsoft PowerPoint. Используется для подготовки материала для работы с детьми;
- Microsoft Office Excel. Используется для работы с документами учета посещаемости детей и данных о детях и их родителях, для формирования отчета и наблюдения за эффективностью и динамики развития воспитанников;
- Microsoft Office Word. Так же используется для формирования типовых документов, таких как табеля, бланки, отчеты.

Информационные технологии, компьютерная и ксероксно-копировальная техника используются в управленческой деятельности для:

- получения информации, нормативно-правовых документов по электронной почте;
- распечатка домашних заданий для детей;
- распечатка табелей посещаемости и других бланков.

Также необходимо проанализировать информационные потоки присутствующие в работе логопеда-дефектолога, т. е. рассмотреть все процессы возникновения, движения и обработки информации, а также направленность и интенсивность документооборота.

Каждый информационный поток, представляющий собой единичное перемещение информации, имеет следующие параметры:

- документ (физический носитель информации);
- проблематика (к какой сфере деятельности относится информация);
- исполнитель (человек, который эту информацию передает);
- периодичность (частота передачи: ежемесячно, ежеквартально, ежедневно).

В нашем случае можно выделить следующие потоки: «Дети», «ТПМПК», «Занятия».

В таблице 1 представлены документы каждого информационного потока и соответствующие ему реквизиты.

Таблица 1

Информационные потоки и соответствующие им документы

Документ	Реквизиты	Исполнитель	Периодичность
<b>ДЕТИ</b>			
Личное дело	ФИО Дата рождения Дата прохождения ТПМПК ФИО родителей \ опекунов Телефон родителей \ опекунов Адрес	Логопед-дефектолог	При добавлении и изменении информации
Динамика развития	ФИО ребенка Дата занятия Образовательная область Изменения	Логопед-дефектолог	Ежедневно
Рекомендации для родителей \ законных представителей и воспитателей	ФИО ребенка ФИО родителей \ опекунов ФИО воспитателя Образовательная область Задания	Логопед-дефектолог	Еженедельно

Документ	Реквизиты	Исполнитель	Периодичность
Приказ об открытии коррекционных групп	Список детей Дата формирования Дата расформирования Вид образовательной программы Ответственные за работу Количество занятий	Заместитель заведующего по ВМР	Ежегодно
<b>ТПМПК</b>			
Заключение	Номер заключения ФИО ребенка Дата рождения Дата проведения Результат Рекомендованный вариант программы Статус ОВЗ	Логопед-дефектолог	При формировании и изменении
График проведения	Дата проведения Список детей Характеристики от специалистов	Логопед-дефектолог	Ежедневно
Отчет	Дата проведения Наименование учреждения Количество коррекционных групп Количество детей Диагнозы Результат Специалисты		
<b>ЗАНЯТИЯ</b>			
Расписание занятий	День недели Время Тип занятия Продолжительность	Логопед-дефектолог	При формировании и изменении договора
Журнал проведения занятий	Дата проведения Тема Количество детей	Логопед-дефектолог	Ежедневно
Табель посещаемости занятий	ФИО ребенка Дата Присутствие	Логопед-дефектолог	Ежедневно
Рабочая программа	Задачи работы Особенности детей Нагрузка в неделю Нагрузка в год Тематическое планирование	Логопед-дефектолог	Ежегодно

Для автоматизации деятельности организаций, предприятий и частных лиц в настоящее время наиболее часто используется система 1С:Предприятие 8.3, благодаря ее технологической платформе, в возможности которой входят создание и модификация самых разнообразных бизнес-приложений.

Помимо технологичности платформы, система обладает такими возможностями как: решение задач планирования и оперативного управления, ведения бухгалтерского и управленческого учета, расчета заработной платы, а также комплексной автоматизации

организационной и хозяйственной деятельности производственных предприятий, торговых и финансовых организаций, бюджетных учреждений и предприятий сферы услуг.

Именно поэтому мной было принято решение осуществлять проектирование информационной системы с помощью технологической платформы «1С: Предприятие 8.3».

Конфигурация учитывает деятельность логопеда-дефектолога и состоит из трех подсистем.

Первой в конфигурации является подсистема «Дети», она включает в себя справочники «Сведения о детях», «Сведения о родителях или законных представителях» и позволяет оперативно вносить необходимые изменения.

К подсистеме «ТПМПК», которая расшифровывается как территориальная психолого-медико-педагогическая комиссия, относятся справочники «Результаты комиссии», «Сведения о детях», «График проведения» позволяющие получать актуальные данные о проведениях комиссий и их заключениях.

Подсистема «Занятия» содержит справочники «Методические материалы», «Посещаемость», «Динамика развития воспитанников», «Тема занятий».

Справочник «Дети» используется для хранения информации о детях, которая может применяться в работе учителя-дефектолога.

Таблица 2

Реквизиты справочника «Дети»

Идентификатор	Синоним	Тип
Код	Код	Число
Наименование	ФИО	Строка
ДатаРождения	ДатаРождения	Дата
Адрес	Адрес	Строка
Группа	Группа	Число
ОВЗ	ОВЗ	Булево

В справочнике «Родители» планируется хранить информацию о родителях или законных представителях воспитанников, посещающих логопеда-дефектолога. Он является подчиненным для справочника «Дети».

Таблица 3

Реквизиты справочника «Родители»

Идентификатор	Синоним	Тип
Код	Код	Число
Владелец	Ребенок	Строка
Мать	Мать	Строка
ДатаРожденияМатери	Дата рождения матери	Дата
ТелефонМатери	Телефон матери	Число
ДолжностьМатери	Должность матери	Строка
Отец	Отец	Строка
ДатаРожденияОтца	Дата рождения отца	Строка
ТелефонОтца	Телефон отца	Дата
ДолжностьОтца	Должность отца	Строка
Адрес	Адрес	Строка

«Динамика развития» это справочник, в котором отслеживается динамика развития детей на занятиях, их успеваемость.

Таблица 4

Реквизиты справочника «Динамика развития»

Идентификатор	Синоним	Тип
Код	Номер	Число
ФИО	ФИО	Справочник Ссылка. Дети
Дата Занятия	Дата занятия	Дата
Прогресс	Прогресс	Булево
Образовательная Область	Образовательная область	Перечисление Ссылка. Образовательная Область
Изменение	Изменение	Строка

В справочнике «Результаты комиссии» хранятся данные связанные с заключениями территориальной психолого-медико-педагогической комиссии.

Таблица 4

Реквизиты справочника «Результаты комиссии»

Идентификатор	Синоним	Тип
Код	Код	Число
Номер Заключения	Номер заключения	Число
ФИО	ФИО	Справочник Ссылка. Дети
Дата Рождения	Дата рождения	Дата
Код Комиссии	Код комиссии	Справочник Ссылка. График Проведения
СтатусОВЗ	Статус ОВЗ	Булево
Инвалидность	Инвалидность	Булево
Рекомендованный Вариант Программы	Рекомендованный вариант программы	Перечисление Ссылка. Программа

В «График проведения» будут располагаться сведения о графиках и расписаниях проведения ТПМПК.

Таблица 6

Реквизиты справочника «График проведения»

Идентификатор	Синоним	Тип
Код	Код	Число
Дата	Дата	Дата
Место	Место	Строка

«Методические материалы» позволяют хранить данные об используемых в работе с детьми методических материалах, учебниках и т. д.

Таблица 7

Реквизиты справочника «Методические материалы»

Идентификатор	Синоним	Тип
Код	Код	Число
Наименование	Название	Строка
Образовательная Область	Образовательная область	Перечисление Ссылка. Образовательная Область
Тема	Тема	Справочник Ссылка. Тема Занятий
Автор	Автор	Строка



В справочнике «Посещаемость» ведется учет посещаемости детей на групповых и индивидуальных занятиях.

Таблица 8

Реквизиты справочника «Посещаемость»

Идентификатор	Синоним	Тип
Код	Код	Число
ФИО	ФИО	Справочник Ссылка. Дети
Дата	Дата	Дата
Тема	Тема	Справочник Ссылка. Тема Занятий
Присутствие	Присутствие	Булево

В справочнике «Тема занятий» ведется план занятий.

Таблица 9

Реквизиты справочника «Тема занятий»

Идентификатор	Синоним	Тип
Код	Код	Число
Наименование	Тема	Строка
Образовательная Область	Образовательная область	Перечисление Ссылка. Образовательная Область
Цель	Цель	Строка

Ниже представлены связи между справочниками (рис. 3).



Рис. 3. Схема связей справочников

### Литература

1. Андерсен Б.Б. Мультимедиа в образовании: информационные технологии в образовании: специализированный учебный курс / Б.Б.Андерсен, К. Бринк. – М., Дрофа, 2017. – 223 с.
2. Гладких, Т. В. Разработка прикладных решений для информационной системы 1С: Предприятие 8.2 [Текст] / Т. В. Гладких, Е. В. Воронова. – Воронеж : ВГУИТ, 2016. – 56 с.

3. Еропкина, А. С. Современные информационные технологии для автоматизации бизнес-процессов / А. С. Еропкина, Ю. А. Зобнин. – Тюмень: ТИУ, 2018. – 156 с.
4. Затеса, А. В. Инструментарии выбора информационной системы. Выбор информационной системы на предприятии: проблемы и способы их преодоления [Текст] / А. В. Затеса // Креативная экономика. – 2010. – No 11. – С. 64–71.
5. Информатика: Энциклопедический словарь для начинающих / Сост. Д.А. Поспелов. – М.: Педагогика-Пресс, 1994. – 352 с.: ил.
6. Тубеева Ф.К. Формирование информационной компетентности будущих бакалавров дефектологического образования в педагогическом вузе: монография / Ф.К. Тубеева; Сев.-Осет. гос. пед. ин-т. – Владикавказ: Издательство СОГПИ, 2017. – 148 с.
7. Шаховой, А. В. Использование информационных технологий в развитии бизнеса : дисс. ... канд. эконом. наук [Текст] / А. В. Шаховой. – Москва, 2015. – 215 с.

© Забудская М.П., Никонова Е.З., 2021

## АВТОРСКИЙ УКАЗАТЕЛЬ

- Абрамов В.В. **98, 108**  
Аджиева А.А. **196**  
Алиева А.Э. **149**  
Алмакаев Д.Р. **33**  
Андреева А.Ю. **185**  
Антипова Е.С. **135**  
Банников К.Д. **65**  
Бобко В.А. **192**  
Бобокало Д.С. **9**  
Болхудере Е.И. **91**  
Бощенко Е.С. **207**  
Брагин К.А. **87**  
Бражникова А.М. **14**  
Бренер Л.И. **122**  
Гардт В.Е. **213**  
Горшков П.В. **98, 103**  
Григорьева Д.Д. **76**  
Гробер Т.А. **158**  
Добровольская Н.Ю. **18, 22**  
Долгов А.Е. **172**  
Домнина М.В. **143**  
Ехлаков Ю.П. **61**  
Забудская М.П. **217**  
Зайнуллина Л.М. **76**  
Закутняя Л.А. **135**  
Захаров Д.М. **118**  
Земенков А.В. **109**  
Игнатов С.В. **38**  
Корженко М.А. **129**  
Кривошеков Д.С. **202**  
Крошилов А.Ф. **172**  
Кузьменко Т.А. **129**  
Кузьмичева Е.А. **72**  
Кульмяков А.А. **126**  
Кушев В.О. **86**  
Лазоренко Е.В. **81**  
Левкин Л.В. **153**  
Лосева А.В. **113, 181**  
Магомедшарипов А.М. **158**  
Мамедли Р.Э. **38**  
Марунов А.А. **18**  
Насыпова Э.А. **41**  
Осмаков Н.А. **46**  
Пикуль М.А. **207**  
Покорная И.Ю. **113, 181**  
Попова Д.Е. **71**  
Прохоренков А.О. **177**  
Развеева И.Ф. **41**  
Свинцова М.А. **56**  
Севальников А.С. **4**  
Семенова А.А. **113, 181**  
Силантьева И.А. **166**  
Сорокин А.О. **162**  
Фурса И.И. **40**  
Харитонов Л.С. **51**  
Хорина И.В. **14**  
Цыпурдиенко А.С. **158**  
Чеверева С.А. **122**  
Чекалина Л.А. **196**  
Чернявская А.Э. **61**  
Шипилова А.А. **27**

## СОДЕРЖАНИЕ

### ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ. МАТЕМАТИКА

---

---

Севальников А.С. <b>ПРОГНОЗИРОВАНИЕ ПО ДВУХПАРАМЕТРИЧЕСКОЙ ТРЕНДОВОЙ МОДЕЛИ.....</b>	<b>4</b>
Бобокало Д.С. <b>ФАЛЬШИВЫЕ НОВОСТИ. FAKE NEWS .....</b>	<b>9</b>
Бражникова А.М., Хорина И.В. <b>УМНЫЕ ПАЙЕТКИ - ЭЛЕКТРОМЕХАНИЧЕСКИЕ ЦВЕТОВОСПРОИЗВОДЯЩИЕ УСТРОЙСТВА ДЛЯ ДИЗАЙНЕРСКОЙ ОДЕЖДЫ .....</b>	<b>14</b>
Марунов А.А., Добровольская Н.Ю. <b>ИСПОЛЬЗОВАНИЕ LSTM СЕТЕЙ ПРИ КЛАССИФИКАЦИИ ТЕКСТОВЫХ СООБЩЕНИЙ.....</b>	<b>18</b>
Фурса И.И., Добровольская Н.Ю. <b>РЕШЕНИЕ ЗАДАЧИ ПРОГНОЗИРОВАНИЯ ВОЛАТИЛЬНОСТИ С ПОМОЩЬЮ НЕЙРОННОЙ СЕТИ С ДОЛГОЙ КРАТКОСРОЧНОЙ ПАМЯТЬЮ .....</b>	<b>22</b>
Шипилова А.А. <b>АНАЛИЗ ДИНАМИКИ КУРСА ВАЛЮТЫ СРЕДСТВАМИ ПЛАТФОРМЫ METATRADER 5 НА ПРИМЕРЕ ВАЛЮТНОЙ ПАРЫ USD/EUR .....</b>	<b>27</b>
Алмакаев Д.Р. <b>АВТОМАТИЗАЦИЯ ПРОЦЕССОВ ВЗАИМОДЕЙСТВИЯ ПОЛЬЗОВАТЕЛЯ ПУТЕМ СОЗДАНИЯ СПЕЦИАЛИЗИРОВАННЫХ ИНФОРМАЦИОННО-СПРАВОЧНЫХ СИСТЕМ..</b>	<b>33</b>
Игнатов С.В., Мамедли Р.Э. <b>ФУНКЦИИ ОБРАТНОГО ВЫЗОВА В AJAX .....</b>	<b>38</b>
Насыпова Э.А., Развеева И.Ф. <b>ПРИМЕНЕНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ФОРМИРОВАНИЯ ПОРТРЕТА КЛИЕНТА ПСИХОЛОГИЧЕСКОГО ЦЕНТРА .....</b>	<b>41</b>
Осмаков Н.А. <b>АВТОМАТИЗИРОВАННЫЙ КОВШ СНЕГОУБОРОЧНОЙ МАШИНЫ ДЛЯ УДАЛЕНИЯ НАЛЕДИ.....</b>	<b>46</b>
Харитонов Л.С. <b>КОМПЬЮТЕРНАЯ ВИЗУАЛИЗАЦИЯ АНАТОМИИ И ФИЗИОЛОГИИ ГОЛОВНОГО МОЗГА ЧЕЛОВЕКА С ПОМОЩЬЮ ПРОГРАММНОГО ПАКЕТА WOLFRAM MATHEMATICA.....</b>	<b>51</b>
Свинцова М.А. <b>МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ В ЭКОНОМИКЕ .....</b>	<b>56</b>
Чернявская А.Э., Ехлаков Ю.П. <b>КРИТЕРИИ ПРИОРИТЕЗАЦИИ ТРЕБОВАНИЙ ПРИ РАЗРАБОТКЕ ПРОГРАММНОГО ПРОДУКТА .....</b>	<b>61</b>

Банников К.Д. <b>ВНЕДРЕНИЕ ПРОГРАММЫ АВТОМАТИЗАЦИИ ПРОЦЕССА СКАНИРОВАНИЯ ДОКУМЕНТАЦИИ</b> .....	65
Попова Д.Е., Кузьмичева Е.А. <b>ПРИМЕНЕНИЕ ИКТ В ОБРАЗОВАТЕЛЬНОМ ПРОЦЕССЕ</b> .....	71
Григорьева Д.Д., Зайнуллина Л.М. <b>ПОДХОДЫ К УТОЧНЕНИЮ ПОНЯТИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ ЕГО ПРИМЕНЕНИЯ В КОНЦЕПЦИИ ЭЛЕКТРОННОГО ПРАВИТЕЛЬСТВА</b> .....	76
Лазоренко Е.В. <b>ЧПУ СТАНОК – ЛАЗЕРНЫЙ ГРАВЕР</b> .....	81
Брагин К.А., Кушев В.О. <b>РАЗРАБОТКА ОПТИМАЛЬНОЙ МЕТОДИКИ ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ</b> .....	86
Болхударе Е.И. <b>СРАВНЕНИЕ СИСТЕМ РАЗРАБОТКИ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ</b> .....	91
Горшков П.В., Абрамов В.В. <b>МАЛОЕ СТАЦИОНАРНОЕ РЕШЕНИЕ НЕЛИНЕЙНОЙ СИСТЕМЫ В КОНЕЧНЫХ РАЗНОСТЯХ</b> .....	98
Горшков П.В. <b>УСТОЙЧИВОСТЬ ЛИНЕЙНОЙ СИСТЕМЫ В КОНЕЧНЫХ РАЗНОСТЯХ</b> .....	103
Земенков А.В., Абрамов В.В. <b>УСЛОВИЯ СУЩЕСТВОВАНИЯ ПЕРИОДИЧЕСКОГО РЕШЕНИЯ СИСТЕМЫ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ</b> .....	108
Лосева А.В., Семенова А.А., Покорная И.Ю. <b>ЗНАЧИМОСТЬ СПЛАЙНА В РАЗЛИЧНЫХ СФЕРАХ ЖИЗНЕДЕЯТЕЛЬНОСТИ И ОСНОВЫ ЕГО ПОСТРОЕНИЯ</b> .....	113
Захаров Д.М. <b>ДВИЖОК ВИЗУАЛИЗАЦИИ ДАННЫХ ROASSAL</b> .....	118
Бренер Л.И., Чеверева С.А. <b>ЭФФЕКТИВНОСТЬ ИСПОЛЬЗОВАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ПРОЕКТАМИ</b> .....	122
Кульмяков А.А. <b>ПОСТРОЕНИЕ ЦИФРОВОГО ОБЩЕСТВА</b> .....	126
Корженко М.А., Кузьменко Т.А. <b>МОДЕЛИРОВАНИЕ БИЗНЕС-ПРОЦЕССОВ ИНСТРУМЕНТАМИ РАЗЛИЧНЫХ КОМПЬЮТЕРНЫХ СРЕДСТВ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ</b> .....	129
Закутняя Л.А., Антипова Е.С. <b>МОДЕЛЬ КОНКУРЕНТНОЙ БОРЬБЫ ДВУХ ИДЕОЛОГИЙ</b> .....	135
Домнина М.В. <b>РЕАЛИЗАЦИЯ МНОГОПОТОЧНОГО СЕРВЕРНОГО ПРИЛОЖЕНИЯ НА ПЛАТФОРМЕ .NET CORE</b> .....	143

Алиева А.Э. <b>РЕАЛИЗАЦИЯ КЛИЕНТСКОГО ПРИЛОЖЕНИЯ НА ОСНОВЕ ТЕХНОЛОГИИ WPF .....</b>	<b>149</b>
Левкин Л.В. <b>РАЗРАБОТКА ЛИЧНОГО КАБИНЕТА СТУДЕНТА НА ПЛАТФОРМЕ ANDROID .....</b>	<b>153</b>
Магомедшарипов А.М., Цыпурдиенко А.С., Гробер Т.А. <b>РЕШЕНИЕ ЗАДАЧИ О КОНКУРЕНТНОМ ПРЕИМУЩЕСТВЕ СРЕДСТВАМИ СОВРЕМЕННЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ .....</b>	<b>158</b>
Сорокин А.О. <b>СОЗДАНИЕ ПРИЛОЖЕНИЯ ДЛЯ ПРОСМОТРА IPTV НА ПЛАТФОРМЕ ANDROID.....</b>	<b>162</b>
Силантьева И.А. <b>МОДЕЛИРОВАНИЕ ДИНАМИКИ БИЗНЕС-ПРОЦЕССОВ С ПРИМЕНЕНИЕМ ЛАГОВЫХ МОДЕЛЕЙ С ИСПОЛЬЗОВАНИЕМ СТАТИСТИКИ ПОИСКОВЫХ ЗАПРОСОВ СЕРВИСА GOOGLTRENDS .....</b>	<b>166</b>
Долгов А.Е., Крошилов А.Ф. <b>МАТЕМАТИЧЕСКОЕ ОБОСНОВАНИЕ ПРИМЕНЕНИЯ БЕРЕЖЛИВЫХ ТЕХНОЛОГИЙ .....</b>	<b>172</b>
Прохоренков А.О. <b>СОЗДАНИЕ КОДОВОГО УСТРОЙСТВА НА ПЛАТФОРМЕ ARDUINO .....</b>	<b>177</b>
Семенова А.А., Лосева А.В., Покорная И.Ю. <b>ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ МАТРИЦ ПОВОРОТА В РОБОТОТЕХНИКЕ.....</b>	<b>181</b>
Андреева А.Ю. <b>ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ: ОСНОВНЫЕ ПОНЯТИЯ, ПРОБЛЕМЫ И ПРОГНОЗЫ РАЗВИТИЯ.....</b>	<b>185</b>
Бобко В.А. <b>МЕТОДЫ ГЕНЕТИЧЕСКОГО ПРОГРАММИРОВАНИЯ .....</b>	<b>192</b>
Аджиева А.А., Чекалина Л.А. <b>РОЛЬ ИНТЕРЕНТ-ВЕЩЕЙ В РАЗВИТИИ ТРАНСПОРТНОЙ ЛОГИСТИКИ .....</b>	<b>196</b>
Кривошеков Д.С. <b>ИСПОЛЬЗОВАНИЕ NIBERNATE ДЛЯ ХРАНЕНИЯ ЗАПИСЕЙ .....</b>	<b>202</b>
Пикуль М.А., Бощенко Е.С. <b>СПОСОБЫ КОММУНИКАЦИИ ПРИ УПРАВЛЕНИИ ВНЕШНИМИ УСТРОЙСТВАМИ.....</b>	<b>207</b>
Гардт В.Е. <b>АНАЛИЗ API И АВТОМАТИЗАЦИЯ ИХ ТЕСТИРОВАНИЯ .....</b>	<b>213</b>
Забудская М.П. <b>ИНСТРУМЕНТЫ АВТОМАТИЗАЦИИ РАБОТЫ ЛОГОПЕДА-ДЕФЕКТОЛОГА ДОШКОЛЬНОГО УЧРЕЖДЕНИЯ .....</b>	<b>217</b>
<b>АВТОРСКИЙ УКАЗАТЕЛЬ .....</b>	<b>225</b>



Научное издание

**XXIII Всероссийская студенческая  
научно-практическая конференция  
Нижевартовского государственного университета**

**Часть 4**

*Информационные технологии. Математика.*

*Нижевартовск, 6-7 апреля 2021 года*

ISBN 978-5-00047-598-0



Под общей редакцией: *Д.А. Погоньшева*

Редакторы: *Е.С. Овечкина, И.С. Анцева, Е.В. Вилявина*

Технический редактор: *Д.В. Вилявин*

Обложка: *Д.В. Вилявин*

Подписано в печать 15.04.2021

Формат 60×84/8

Гарнитура Times New Roman. Усл. печ. листов 9,85

Электронное издание. Объем 14,64 МБ. Заказ 2190

Издательство НВГУ

628615, Тюменская область, г. Нижневартовск, ул. Маршала Жукова, 4

Тел./факс: (3466) 24-50-51, E-mail: red@nvsu.ru, izdatelstvo@nggu.ru