ISSN 1992-2752

Вестник Сыктывкарского университета

Серия 1:
Математика
Механика
Информатика

1(46) ВЫПУСК **23**

Вестник Сыктывкарского университета	Серия 1: *Математика* *Механика* *Информатика*	12+ ISSN 1992-2752
Основан в 1995 году Выходит 4 раза в год		Выпуск 1 (46) 2023

Учредитель и издатель: Федеральное государственное бюджетное образовательное учреждение высшего образования «Сыктывкарский государственный университет имени Питирима Сорокина» (167001, Республика Коми, г. Сыктывкар, Октябрьский просп., д. 55)

Зарегистрирован Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзор). Свидетельство ПИ № ФС77-37565 от 17 сентября 2009 года

Вестник Сыктывкарского университета. Серия 1: Математика. Механика. Информатика : сборник. Сыктывкар: Изд-во СГУ им. Питирима Сорокина, 2023. — 1 (46). 2023. — 91 с.

Рецензируемый научный журнал. Основан в 1995 г.

Журнал «Вестник Сыктывкарского университета. Серия 1: Математика. Механика. Информатика» включён в перечень отечественных изданий, которые входят в международные реферативные базы данных и системы цитирования и в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук, утвержденный приказом Министерства науки и высшего образования Российской Федерации. Журнал публикует научные статьи по следующим научным специальностям: «1.1.5. Математическая логика, алгебра, теория чисел и дискретная математика (физико-математические науки)», «1.1.8. Механика деформируемого твердого тела (физико-математические науки)», «1.2.2. Математическое моделирование, численные методы и комплексы программ (технические науки)», «5.8.2. Теория и методика обучения и воспитания (математика, уровни общего и профессионального образования) (педагогические науки)».

The peer-reviewed journal was founded in 1995

«Bulletin of Syktyvkar University. Series 1: Mathematics. Mechanics. Informatics» is included in the list of editions included in international abstract databases and citation systems, and in which the main scientific results of dissertations for the degree of Candidate or Doctor of Sciences should be published, approved by the Ministry of Science and Higher Education of the Russian Federation. The journal publishes scientific articles on the following scientific specialties: «1.1.5. Mathematical logic, algebra, number theory, and discrete mathematics (physical and mathematical sciences)», «1.1.8. Mechanics of deformable solids (physical and mathematical sciences)», «1.2.2. Mathematical modeling, numerical methods and software packages (technical sciences)», «5.8.2. Theory and methods of teaching and education (mathematics, general and vocational education levels) (pedagogical sciences)».

Подписной индекс журнала в интернет-каталоге «Пресса России» — 43653.

Адрес редакции
Вестника Сыктывкарского университета
167001, Республика Коми, г. Сыктывкар, Октябрьский просп., д. 55
Тел. (8212)390-308.
Электронный адрес: http://vestnik-mmi.syktsu.ru/

Свободная цена © ФГБОУ ВО «СГУ им. Питирима Сорокина», 2023.

Главный редактор:
д.пед.н., доцент, ректор ФГБОУ ВО «СГУ им. Питирима Сорокина»
Сотникова О. А.

Ответственный редактор:
Ермоленко А. В., к.ф.-м.н., доцент, заведующий кафедрой
(СГУ им. Питирима Сорокина)

Редакционная коллегия:
Аббасов М. Э., д.ф.-м.н., доцент, профессор (СПбГУ)
Беляева Н. А., д.ф.-м.н., доцент, профессор (СГУ им. Питирима Сорокина)
Вечтомов Е. М., д.ф.-м.н., профессор, заведующий кафедрой (ВятГУ)
Головач П. А., к.ф.-м.н., доцент, исследователь (Университет Бергена, Норвегия)
Григорьев С. Г., член-корреспондент РАО, д.т.н., профессор, профессор (МГПУ)
Дорофеев С. Н., д.пед.н., к.ф.-м.н., профессор, профессор (ТолГУ)
Калинин С. И., д.пед.н., к.ф.-м.н., профессор, профессор (ВятГУ)
Колесников Г. Н., д.т.н., профессор, профессор (ПетрГУ)
Колпак Е. П., д.ф.-м.н., профессор (СПбГУ)
Крылатов А. Ю., д.ф.-м.н., доцент, профессор (СПбГУ)
Махнев А. А., член-корреспондент РАН, д.ф.-м.н., профессор, главный научный сотрудник (ИММ УрО РАН)
Одинец В. П., д.ф.-м.н., профессор
Орлов В. В., д.пед.н., профессор, профессор (Российский государственный педагогический университет им. А.И. Герцена)
Парилина Е. М., д.ф.-м.н., доцент, профессор (СПбГУ)
Певный А. Б., д.ф.-м.н., профессор
Петров Н. Н., д.ф.-м.н., профессор, профессор (УдмГУ)
Петраков А. П., д.ф.-м.н., доцент, заведующий кафедрой
(СГУ им. Питирима Сорокина)
Питухин Е. А., д.т.н., профессор, профессор (ПетрГУ)
Попов Н. И., д.пед.н., , к.ф.-м.н., доцент, заведующий кафедрой
(СГУ им. Питирима Сорокина)
Рудикова-Фронхёфер Л. В., к.ф.-м.н., доцент, заведующий кафедрой
(ГрГУ им. Янки Купалы, Респ. Беларусь)
Тихомиров А. Н., д.ф.-м.н., профессор, главный научный сотрудник
(Коми НЦ УрО РАН)
Чермных В. В., д.ф.-м.н., доцент, главный научный сотрудник
(СГУ им. Питирима Сорокина)

Техническая редакция:
Руденко Л. Н., руководитель издательского центра (СГУ им. Питирима Сорокина),
Котелина Н. О., к.ф-м.н., доцент (СГУ им. Питирима Сорокина),
Мазур В. В., к.г.н., преподаватель (СГУ им. Питирима Сорокина)

Содержание

Математика

Созонтова Е. А. *К новым случаям разрешимости задачи Гурса в квадратурах для одной системы гиперболического типа* 4

Информатика

Бабикова Н. Н. *Применение библиотеки NumPy для векторизации кода Python* 14

Гольчевский Ю. В., Ушаков Д. А. *Ускорение криптографических вычислений путем низкоуровневой оптимизации базовых блоков* 30

Методические материалы

Дейнега С. А. *Компоненты геометро-графической компетенции, формируемые при изучении начертательной геометрии в техническом вузе* 50

Дорофеев С. Н., Наземнова Н. В. *Числовые последовательности как основополагающий фактор формирования творческой активности у будущих бакалавров* 64

Одинец В. П. *О работах трёх довоенных математиков из Алма-Аты, Москвы и Ленинграда* 78

.

Вестник Сыктывкарского университета.
Серия 1: Математика. Механика. Информатика. 2023.
Выпуск 1 (46)
Bulletin of Syktyvkar University.
Series 1: Mathematics. Mechanics. Informatics. 2023; 1 (46)

МАТЕМАТИКА

MATHEMATICS

Научная статья

УДК 517.956
https://doi.org/10.34130/1992-2752_2023_1_4

К НОВЫМ СЛУЧАЯМ РАЗРЕШИМОСТИ ЗАДАЧИ ГУРСА В КВАДРАТУРАХ ДЛЯ ОДНОЙ СИСТЕМЫ ГИПЕРБОЛИЧЕСКОГО ТИПА

Елена Александровна Созонтова
Елабужский институт ФГАОУ ВО КФУ, sozontova-elena@rambler.ru

Аннотация. В работе исследована задача Гурса для системы гиперболического типа с двумя независимыми переменными. С помощью факторизации уравнений рассматриваемой системы получены новые случаи разрешимости в квадратурах поставленной задачи.

Ключевые слова: гиперболическая система, задача Гурса, разрешимость в квадратурах

Для цитирования: Созонтова Е. А. К новым случаям разрешимости задачи Гурса в квадратурах для одной системы гиперболического типа // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика.* 2023. Вып. 1 (46). С. 4–13. https://doi.org/10.34130/1992-2752_2023_1_4

© Созонтова Е. А., 2023.

Article

On new cases of solvability of the Goursat problem in quadratures for one hyperbolic type system

Elena A. Sozontova
Elabuga Institute KFU, sozontova-elena@rambler.ru

Abstract. The paper investigates the Goursat problem for a hyperbolic type system with two independent variables. With the help of factorization of the equations of the system under consideration, new cases of solvability in the quadratures of the problem are obtained.

Keywords: hyperbolic system, the Goursat problem, solvability in quadratures

For citation: Sozontova E. A. On new cases of solvability of the Goursat problem in quadratures for one hyperbolic type system. *Vestnik Syktyvkarskogo universiteta. Seriya 1: Matematika. Mekhanika. Informatika* [Bulletin of Syktyvkar University, Series 1: Mathematics. Mechanics. Informatics], 2023, no 1 (46), pp. 4–13. https://doi.org/10.34130/1992-2752_2023_1_4

В области $D = \{x_0 < x < x_1,\ y_0 < y < y_1\}$ рассматривается система

$$\begin{cases} u_{xy} + a_1 u_x + b_1 u_y + c_1 v_x + d_1 v_y + e_1 u + f_1 v = g_1, \\ v_{xy} + a_2 u_x + b_2 u_y + c_2 v_x + d_2 v_y + e_2 u + f_2 v = g_2, \end{cases} \tag{1}$$

гладкость коэффициентов которой определяется включениями

$$a_1,\ a_2,\ c_1,\ c_2 \in C^{(1,0)},\ b_1,\ c_1,\ c_2,\ d_2 \in C^{(0,1)},$$
$$e_1,\ e_2,\ f_1,\ f_2 \in C^{(0,0)}.$$

Задача 1. В области D найти регулярное решение системы (1), удовлетворяющее условиям

$$u(x_0, y) = \varphi_1(y), \quad u(x, y_0) = \psi_1(x),$$
$$v(x_0, y) = \varphi_2(y), \quad v(x, y_0) = \psi_2(x).$$

При этом предполагается, что $\varphi_1,\ \varphi_2 \in C^1(\overline{X}),\ \psi_1,\ \psi_2 \in C^1(\overline{Y})$ и выполняются условия согласования

$$\varphi_1(y_0) = \psi_1(x_0), \quad \varphi_2(y_0) = \psi_2(x_0).$$

Как известно [1], решение задачи 1 существует и единственно. В работах [2], [3] изложены варианты разрешимости этой задачи в квадратурах. Целью данной работы является получение новых случаев разрешимости в квадратурах задачи 1.

Рассмотрим сначала вспомогательную задачу.

Задача 2 (вспомогательная). В области D найти регулярное решение уравнения

$$u_{xy} + au_x + bu_y + cu = f, \tag{2}$$

удовлетворяющее непрерывно дифференцируемым граничным значениям

$$\begin{gathered} u(x_0, y) = \varphi(y),\ u(x, y_0) = \psi(x),\ \varphi(y_0) = \psi(x_0), \\ x \in [x_0, x_1],\ y \in [x_0, x_1]. \end{gathered} \tag{3}$$

Известно [4, с. 172; 5, с. 14], что решение задачи 2 записывается через соответствующие функции Римана, для которых известны [5, с. 15–16; 6–8] случаи их построения в явном виде. Для дальнейшего использования запишем все эти случаи в объединенном виде и сформулируем теорему. В обозначенных выше источниках условия, при которых функции Римана определяются в явном виде, представлены в терминах следующих соотношений:

$$1)\ h \equiv 0;$$

$$2)\ k \equiv 0;$$

$$3)\ 2h - (\ln h)_{xy} - k \equiv 0;$$

$$4)\ 2k - (\ln k)_{xy} - h \equiv 0;$$

$$5)\ a_x \equiv b_y,\ h \equiv \xi_0(x)\eta_0(y) \neq 0;$$

$$6)\ b_y - a_x \equiv h \equiv \xi_1(x)\eta_1(y) \neq 0;$$

$$7)\ a_x - b_y \equiv k \equiv \xi_2(x)\eta_2(y) \neq 0;$$

$$8)\ h \equiv 2\mu_0(x)\tau_0(y) \neq 0,\ k \equiv 3\mu_0(x)\tau_0(y) \neq 0;$$

$$9)\ h \equiv 3\mu_1(x)\tau_1(y) \neq 0,\ k \equiv 2\mu_1(x)\tau_1(y) \neq 0;$$

$$10)\ (\ln h)_{xy} \equiv h - k,\ h \equiv 2b_y \equiv \omega_1;$$

$$11)\ (\ln k)_{xy} \equiv k - h,\ k \equiv 2a_x \equiv \omega_2;$$

$$12)\ m_0 a_x - b_y \equiv m_0 b_y - a_x \equiv (m_0 - 1)(ab - c);$$

$$13)\ h \equiv \omega_0;$$

$$14)\ k \equiv \omega_0, \tag{4}$$

где

$$h = a_x + ab - c,\ k = b_y + ab - c,$$

$$\omega_r = \frac{2s_r^{'}(x)t_r^{'}(y)}{(2 - m_r)[s_r(x) + t_r(y)]^2},\ [s_r(x) + t_r(y)]s_r^{'}(x)t_r^{'}(y) \neq 0. \tag{5}$$

Здесь $\xi_k,\ \eta_k,\ \mu_0,\ \mu_1,\ \tau_0,\ \tau_1\ \in C^1,\ s_k,\ t_k,\ m_k \in C^2$, m_k зависит только от одной из переменных (x, y) и $m_k \neq 2$ $(k = 0,\ 1,\ 2)$. В остальном обозначенные выше функции произвольны, а именно в соответствующем классе должны найтись функции, при которых выполняются условия (4). Коэффициенты $a,\ b,\ c$ имеют гладкость, которая обеспечивает возможность выполнения формул (4). Классы гладкости задаются на замкнутых множествах определения соответствующих функций.

Справедлива теорема

Теорема 1. *Пусть $h,\ k,\ \omega_r$ определяются формулами (5). Тогда построение решения задачи (2) – (3) в квадратурах обеспечивается любым из тождеств 1) – 4), а также существованием функций $\xi_r,\ \eta_r,\ \mu_r,\ \tau_r,\ m_r,\ s_r,\ t_r$, для которых имеет место любая из групп соотношений 5) – 11) или когда вместе с 12) любая из определяемых в (5) комбинаций $h,\ k$ имеет вид, указанный в 13) – 14). При этом зависящая лишь от одной из переменных (x, y) функция ω_0 удовлетворяет условию $\omega_0 \neq 0$, а $\omega_1,\ \omega_2$ – $(\omega_k + 1)(\omega_k - 2) \neq 0$.*

Перейдем теперь к решению задачи 1. Попробуем отыскать такие функции $\alpha_1,\ \beta_1,\ \gamma_1$, чтобы первое уравнение (1) можно было записать в виде

$$\left(\frac{\partial}{\partial y} + \alpha_1\right)(u_x + \beta_1 u + \gamma_1 v) = g_1. \tag{6}$$

Нетрудно убедиться, что первое уравнение (1) совпадает с (6), если имеют место тождества

$$c_1 \equiv 0,\ b_{1y} + a_1 b_1 - e_1 \equiv 0,$$

$$d_{1y} + a_1 d_1 - f_1 \equiv 0, \tag{7}$$

где

$$\alpha_1 = a_1,\ \beta_1 = b_1,\ \gamma_1 = d_1. \tag{8}$$

Рассуждая аналогичным образом, получаем, что при выполнении тождеств

$$b_2 \equiv 0,\ a_{2x} + a_2 d_2 - e_2 \equiv 0,$$
$$c_{2x} + c_2 d_2 - f_2 \equiv 0 \tag{9}$$

второе уравнение (1) можно записать в виде

$$\left(\frac{\partial}{\partial x} + \alpha_2\right)(v_y + \beta_2 u + \gamma_2 v) = g_2,$$

где

$$\alpha_2 = d_2,\ \beta_2 = a_2,\ \gamma_2 = c_2. \tag{10}$$

Итак, задача 1 разбивается на три задачи

$$w_{1y} + \alpha_1 w_1 = g_1, \quad w_1(x, y_0) = \psi_{1x} + \beta_1 \psi_1 + \gamma_1 \psi_2, \tag{11}$$

$$w_{2x} + \alpha_2 w_2 = g_2, \quad w_2(x_0, y) = \varphi_{2y} + \beta_2 \varphi_1 + \gamma_2 \varphi_2, \tag{12}$$

$$\begin{cases} u_x + \beta_1 u + \gamma_1 v = w_1, \\ v_y + \beta_2 u + \gamma_2 v = w_2, \end{cases} \tag{13}$$

$$u(x_0, y) = \varphi_1(y), \quad v(x, y_0) = \psi_2(x). \tag{14}$$

Задачи (11), (12), (13) – (14) необходимо решать последовательно, начиная с первой из них. Функции $w_1,\ w_2$ находятся с помощью интегрирования уравнений из (11), (12), при этом в случае задачи (11) в качестве параметра выступает x, а в случае (12) – y.

Известно [9], что задача (13) – (14) является однозначно разрешимой. Для нахождения условий разрешимости этой задачи в квадратурах воспользуемся возможностью редукции системы (13) к двум уравнениям вида

$$\Theta_{xy} + a\Theta_x + b\Theta_y + c\Theta = f. \tag{15}$$

А именно при выполнении условия

$$\gamma_1 \neq 0 \tag{16}$$

получаем уравнение вида (15) для $\Theta = u$, коэффициенты которого определяются формулами

$$a = \gamma_2 - (\ln \gamma_1)_y,\ b = \beta_1,\ c = \beta_{1y} - \beta_1(\ln \gamma_1)_y - \gamma_1\beta_2 + \gamma_2\beta_1. \tag{17}$$

При

$$\beta_2 \neq 0 \tag{18}$$

получаем уравнение вида (15) для $\Theta = v$ с коэффициентами

$$a = \gamma_2,\ b = \beta_1 - (\ln \beta_2)_x,\ c = \gamma_{2x} - \gamma_2(\ln \beta_2)_x - \beta_2\gamma_1 + \beta_1\gamma_2. \tag{19}$$

Решив первое уравнение, полученное при выполнении условия (16), можно найти функцию $v(x, y)$ из первого уравнения системы (13). И, наоборот, при выполнении неравенства (18), зная $v(x, y)$, функцию $u(x, y)$ можно найти из второго уравнения системы (13). Но для нахождения функций $\Theta = u$ и $\Theta = v$ из (15) условий (14) недостаточно: нужно добавить еще (см. задачу 1)

$$u(x, y_0) = \psi_1(x), \quad v(x_0, y) = \varphi_2(y) \tag{20}$$

и условия согласования

$$\varphi_1(y_0) = \psi_1(x_0), \quad \varphi_2(y_0) = \psi_2(x_0).$$

Очевидно, что первые (вторые) соотношения в (14) и (20) являются граничными условиями первой (второй) задачи Гурса для уравнения вида (15). Таким образом, задачу 1 мы редуцировали к двум задачам Гурса для уравнения (15), которые аналогичны рассмотренной выше вспомогательной задаче 2.

В терминах коэффициентов системы (1) сотношения (16) – (19) запишутся в виде

$$d_1 \neq 0, \tag{21}$$

$$a = c_2 - (\ln d_1)_y,\ b = b_1,\ c = b_{1y} - b_1(\ln d_1)_y - d_1a_2 + c_2b_1, \tag{22}$$

$$a_2 \neq 0, \tag{23}$$

$$a = c_2,\ b = b_1 - (\ln a_2)_x,\ c = c_{2x} - c_2(\ln a_2)_x - a_2d_1 + b_1c_2. \tag{24}$$

Тогда справедливо утверждение

Теорема 2. *Для разрешимости задачи 1 в квадратурах достаточно, чтобы выполнялись тождества (7), (9) и один из двух наборов* a, b, c, *определяемых формулами (21) – (22), (23) – (24), удовлетворял условиям теоремы 1.*

Применим теперь к системе (1) другую факторизацию. А именно уравнения из (1) можно записать в виде

$$\left(\frac{\partial}{\partial x}+\alpha_1\right)(u_y+\beta_1 u+\gamma_1 v)=g_1,\ \left(\frac{\partial}{\partial y}+\alpha_2\right)(v_x+\beta_2 u+\gamma_2 v)=g_2, \tag{25}$$

если имеют место тождества

$$\begin{gathered} d_1\equiv 0,\ a_{1x}+b_1a_1-e_1\equiv 0,\ c_{1x}+b_1c_1-f_1\equiv 0,\\ a_2\equiv 0,\ b_{2y}+c_2b_2-e_2\equiv 0,\ d_{2y}+c_2d_2-f_2\equiv 0, \end{gathered} \tag{26}$$

позволяющие найти

$$\alpha_1=b_1,\ \beta_1=a_1,\ \gamma_1=c_1,\ \alpha_2=c_2,\ \beta_2=b_2,\ \gamma_2=d_2.$$

Тогда задача 1 сводится к трем задачам

$$w_{1x}+\alpha_1 w_1=g_1,\quad w_1(x_0,y)=\varphi_{1y}+\beta_1\varphi_1+\gamma_1\varphi_2, \tag{27}$$

$$w_{2y}+\alpha_2 w_2=g_2,\quad w_2(x,y_0)=\psi_{2x}+\beta_2\psi_1+\gamma_2\psi_2, \tag{28}$$

$$\begin{cases} u_y+\beta_1 u+\gamma_1 v=w_1,\\ v_x+\beta_2 u+\gamma_2 v=w_2, \end{cases} \tag{29}$$

$$u(x,y_0)=\psi_1(x),\quad v(x_0,y)=\varphi_2(y). \tag{30}$$

Функции w_1, w_2 из (27), (28) определяются непосредственным интегрированием, при этом в случае задачи (27) в качестве параметра выступает y, а в случае (28) – x. Задача (29) – (30) аналогична задаче (13) – (14), однако здесь u и v меняются ролями. Учитывая это, формулы (21) – (24) примут соответственно вид

$$b_2\neq 0, \tag{31}$$

$$a = a_1 - (\ln b_2)_y,\ b = d_2,\ c = d_{2y} - d_2(\ln b_2)_y - b_2c_1 + a_1d_2, \tag{32}$$

$$c_1 \neq 0. \tag{33}$$

$$a = a_1,\ b = d_2 - (\ln c_1)_x,\ c = a_{1x} - a_1(\ln c_1)_x - c_1b_2 + d_2a_1. \tag{34}$$

Аналогом теоремы 2 является

Теорема 3. *Для разрешимости задачи 1 в квадратурах достаточно, чтобы выполнялись тождества (26) и один из двух наборов a, b, c, определяемых формулами (31) – (32), (33) – (34), удовлетворял условиям теоремы 1.*

Список источников

1. **Бицадзе А. В.** Некоторые классы уравнений в частных производных. М.: Наука, 1981. 448 с.

2. **Созонтова Е. А.** Об условиях разрешимости граничных задач в квадратурах для гиперболических систем второго порядка // *Уфимск. матем. журн.* 2016. Т. 8. № 3. С. 135–140.

3. **Созонтова Е. А.** К новым случаям разрешимости задачи Гурса в квадратурах для системы второго порядка // *Труды Математического центра имени Н.И. Лобачевского : материалы XVI Молодежной научной школы-конференции «Лобачевские чтения – 2017».* Казань, 2017. С. 140–141.

4. **Бицадзе А. В.** Уравнения математической физики. М.: Наука, 1982. 336 с.

5. **Жегалов В. И., Миронов А. Н.** Дифференциальные уравнения со старшими частными производными. Казань: Казанское матем. об-во, 2001. 226 с.

6. **Жегалов В. И.** К случаям разрешимости гиперболических уравнений в терминах специальных функций // *Неклассические уравнения математической физики.* Новосибирск: ИМ СО РАН, 2002. С. 73–79.

7. **Жегалов В. И., Сарварова И. М.** К условиям разрешимости задачи Гурса в квадратурах // *Изв. вузов. Математика.* 2013. № 3. С. 68–73.

8. **Жегалов В. И., Созонтова Е. А.** Дополнение к случаям разрешимости задачи Гурса в квадратурах // *Дифференц. уравнения.* 2017. Т. 53. № 2. С. 270–272.

9. **Чекмарев Т. В.** Решение гиперболической системы двух дифференциальных уравнений в частных производных с двумя неизвестными функциями // *Изв. вузов. Математика.* 1959. № 6. С. 220–228.

References

1. **Bicadze A. V.** *Nekotorye klassy uravnenij v chastnyh proizvodnyh* [Some classes of partial differential equations]. Moscow: Nauka, 1981, 448 p. (In Russ.)

2. **Sozontova E. A.** On solvability by quadratures conditions of boundary value problems for second order hyperbolic systems. *Ufimskij matematicheskij zhurnal* [Ufa mathematical journal], 2016, vol. 8, no 3, pp. 135–140. (In Russ.)

3. **Sozontova E. A.** On new cases of solvability of the Goursat problem in quadratures for a second-order system. *Trudy Matematicheskogo centra imeni N. I. Lobachevskogo: materialy XVI molodezhnoj nauchnoj shkoly-konferencii "Lobachevskie chteniya – 2017"* [Proceedings of the N. I. Lobachevsky: materials of the XVI Youth Scientific School-Conference "Lobachevsky Readings – 2017"], 2017, pp. 140–141. (In Russ.)

4. **Bicadze A. V.** *Uravneniya matematicheskoj fiziki* [Equations of mathematical physics]. Moscow: Nauka, 1982, 336 p. (In Russ.)

5. **Zhegalov V. I., Mironov A. N.** *Differencial'nye uravneniya so starshimi chastnymi proizvodnymi* [Differential equations with higher partial derivatives]. Kazan: Kazanskoe matematicheskoe obshchestvo, 2001, 226 p. (In Russ.)

6. **Zhegalov V. I.** On solvability cases for hyperbolic equations in terms of special functions. *Neklassicheskie uravneniya matematicheskoi fiziki* [Nonclassical Equations of Mathematical Physics]. Novosibirsk: Mathematical Institute, Russian Academy of Science, Siberian Branch, 2002, pp. 73–79. (In Russ.)

7. **Zhegalov V. I., Sarvarova I. M.** Solvability of the Goursat problem in quadratures. *Izvestiya vuzov. Matematika* [Russian Mathematics], 2013, no 3, pp. 68–73. (In Russ.)

8. **Zhegalov V. I., Sozontova E. A.** An addition to the cases of solvability of the goursat problem in quadratures. *Differencial'nye uravneniya* [Differential equations], 2017, vol. 53, no 2, pp. 270–272. (In Russ.)

9. **Chekmaryov T. V.** Solution of a hyperbolic system of two partial differential equations with two unknown functions. *Izvestiya vuzov. Matematika* [Russian Mathematics], 1959, no 6, pp. 220–228. (In Russ.)

Сведения об авторе / Information about author
Созонтова Елена Александровна / Elena A. Sozontova
к.ф.-м.н., доцент кафедры математики и прикладной информатики Елабужского института ФГАОУ ВО КФУ / Ph.D. in Physics and Mathematics, Associate Professor of Mathematics and Applied Informatics Department of the Elabuga Institute KFU
Елабужский институт ФГАОУ ВО КФУ / Elabuga Institute KFU
423603, г. Елабуга, ул. Казанская, д.89. / 423603, Russia, Elabuga, Kazanskaja st., 89

Статья поступила в редакцию / The article was submitted 21.02.2023
Одобрено после рецензирования / Approved after reviewing 27.02.2023
Принято к публикации / Accepted for publication 01.03.2023

Вестник Сыктывкарского университета.
Серия 1: Математика. Механика. Информатика. 2023.
Выпуск 1 (46)
Bulletin of Syktyvkar University.
Series 1: Mathematics. Mechanics. Informatics. 2023; 1 (46)

ИНФОРМАТИКА

INFORMATICS

Научная статья

УДК 378.14, 004.432
https://doi.org/10.34130/1992-2752_2023_1_14

ПРИМЕНЕНИЕ БИБЛИОТЕКИ NUMPY ДЛЯ ВЕКТОРИЗАЦИИ КОДА PYTHON

Надежда Николаевна Бабикова
Сыктывкарский государственный университет
им. Питирима Сорокина, valmasha@mail.ru

Аннотация. Векторизация кода — процесс перехода от операций над отдельными элементами массивов к операциям, происходящим над целыми массивами или их частями. В статье рассматриваются инструменты библиотеки NumPy, позволяющие векторизовать код на языке Python: векторные функции, укладывание, маскирование, прихотливая индексация. Эффективность применения этих инструментов продемонстрирована на примере двух задач машинного обучения.

Ключевые слова: NumPy, Python, векторизация, многомерные массивы, циклы

Для цитирования: Бабикова Н. Н. Применение библиотеки NumPy для векторизации кода Python // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика.* 2023. Вып. 1 (46). С. 14–29. https://doi.org/10.34130/1992-2752_2023_1_14

© Бабикова Н. Н., 2023.

Article

Using NumPy to vectorization of Python code

Nadezhda N. Babikova
Pitirim Sorokin Syktyvkar State University, valmasha@mail.ru

Abstrakt. Code vectorization is the process of moving from operations on individual elements of arrays to operations that occur on entire arrays or their parts. The NumPy library tools that allow to vectorize Python code are discussed in the article: vector functions, broadcasting, masking, fancy indexing. The effectiveness of these tools is demonstrated on the example of two machine learning problems.

Keywords: NumPy, Python, vectorization, multidimensional arrays, loops

For citation: Babikova N. N. Using NumPy to vectorization of Python code . *Vestnik Syktyvkarskogo universiteta. Seriya 1: Matematika. Mekhanika. Informatika* [Bulletin of Syktyvkar University, Series 1: Mathematics. Mechanics. Informatics], 2023, no 1 (46), pp. 14–29. https://doi.org/10.34130/1992-2752_2023_1_14

Введение

Библиотека Numerical Python, сокращенно NumPy, – это основа научной экосистемы языка Python. NumPy лежит в основе почти каждой библиотеки Python, предназначенной для научных или числовых вычислений, включая SciPy, Matplotlib, Pandas, Scikit-learn и Scikit-image [1, с. 357]. Библиотека NumPy предоставляет удобный и гибкий интерфейс для оптимизированных вычислений над массивами данных. Векторизованные операции в NumPy делегируют выполнение цикла высоко оптимизированным функциям C и Fortran, что делает код Python более чистым и быстрым.

Векторизация кода – процесс перехода от скалярных операций над отдельными элементами массивов к векторным операциям, происходящим над целыми массивами или их частями. С точки зрения сложности реализации, а также изучения или обучения способы векторизации кода можно разделить на непосредственные и опосредованные. *Непосредственная векторизация* заключается в прямой замене поэлементной обработки массивов в циклах на векторные операции и использовании *универсальных функций* (universal function) NumPy. *Опо-*

средованная векторизация подразумевает дополнительно к этому использование специальных возможностей библиотеки NumPy, таких как *укладывание* (broadcasting), *маскирование* (masking) и *прихотливое индексирование* (fancy indexing). Цель статьи – показать на примере задач машинного обучения эффективность применения инструментов векторизации NumPy с точки зрения повышения производительности кода Python.

Непосредственная векторизация

Рассмотрим непосредственную векторизацию кода. Основой библиотеки NumPy является объект *ndarray* для представления однородного n-мерного массива. У любого массива есть атрибут *shape* (форма) – кортеж, описывающий размер по каждому измерению, атрибут *dtype* – объект, определяющий тип данных в массиве, атрибут *strides* – кортеж, который задает шаги перемещения к следующему элементу данных по каждому измерению. Синтаксис индексации и доступа к подмассивам с помощью *срезов* (slicing) библиотеки NumPy аналогичен синтаксису для стандартных списков языка Python.

Срезы массивов возвращают *представления* (views), а не *копии* (copies) данных массива. Представления NumPy – это метод доступа к данным массива, при котором изменяются только метаданные dtype и/или strides, но не меняется буфер данных. Один и тот же массив в NumPy может иметь разные представления и математически и программно обрабатываться по-разному. При изменении значений в представлении изменяются данные в исходном массиве. Использование представлений полезно при векторизации кода – создание копии занимает больше времени и памяти, чем создание представления [2].

Векторизованные вычисления в библиотеке NumPy реализованы посредством универсальных функций. Универсальной (*u*-функцией) называется функция, которая выполняет поэлементные операции над данными, хранящимися в объектах ndarray. Можно считать, что это векторные обертки вокруг простых функций. Универсальные функции библиотеки NumPy очень просты в использовании, поскольку применяют нативные арифметические операторы языка Python. Все арифметические операции – удобные адаптеры для встроенных функций библиотеки NumPy. Например, оператор + является адаптером для функции add().

В табл. 1 приводится сравнение средней скорости выполнения поэлементной обработки массивов, списков Python и векторной опера-

ции NumPy. Проверка проводилась в среде IDLE (Python 3.10 64 bit), NumPy 1.22.3, процессор Intel Core i5-5200U (ноутбук), 4GB DDR3 L Memory.

Таблица 1

Среднее время выполнения циклов и векторных операций

Двумерные массивы		

`# Векторная операция`
`l = np.ones((n,n), dtype=np.int32)`
`l += 10`
	n= 100
	n=1000
	n=5000
		t= 0.0000
		t= 0.0009
		t= 0.0211

`# Циклы по списку`
`l = [[1.0 for i in range(n)]`
`for j in range(n)]`
`for i in range(n):`
`for j in range(n):`
`l[i][j] += 10.0`
	n= 100
	n=1000
	n=5000
		t= 0.0019
		t= 0.2545
		t= 6.0717

`# Циклы по массиву`
`l = np.ones((n,n), dtype=np.int32)`
`for i in range(n):`
`for j in range(n):`
`l[i, j] += 10`
	n= 100
	n=1000
	n=5000
		t= 0.0081
		t= 0.7472
		t=20.6330

Одномерные массивы		
`l = np.array(range(n), dtype=np.int32)` `l += 10`	n= 100000 n=10000000	t= 0.0001 t= 0.0098

`l = list(range(n))`
`for i in range(n):`
`l[i] += 10`
	n= 100000
	n=10000000
		t= 0.0217
		t= 2.4403

`l = np.array(range(n), dtype=np.int32)`
`for i in range(n):`
`l[i] += 10`
	n= 100000
	n=10000000
		t= 0.0675
		t= 6.5120

Таблица 2

Среднее время выполнения циклов с проверкой условия и векторной функции

# Векторная операция 	
a = np.random.randint(1, 10000,
size = (n,n), dtype=np.int32)
z = np.where(a > 5, a, a+11)
	n= 100
	n=1000
	n=5000
		t= 0.00015
		t= 0.0059
		t= 0.1468

# Циклы по списку 	
aa = list(np.random.randint(1, 10000,
size = (n,n), dtype=np.int32)
z = list(np.zeros((n,n),dtype=np.int32))
for i in range(n):
for j in range(n):
if aa[i][j]>5:
z[i][j]=aa[i][j]
else:
z[i][j]=aa[i][j]+11
	n= 100
	n=1000
	n=5000
		t= 0.0092
		t= 0.9637
		t=24.7588

# Циклы по массиву 	
a = np.random.randint(1, 10000,
size = (n,n), dtype=np.int32)
z = np.zeros((n,n), dtype=np.int32)
for i in range(n):
for j in range(n):
if a[i, j]>5:
z[i, j]=a[i, j]
else:
z[i ,j]=a[i, j]+11
	n= 100
	n=1000
	n=5000
		t= 0.0107
		t= 1.1528
		t=29.6370

В табл. 2 приводится сравнение средней скорости выполнения поэлементной обработки массивов, списков Python при наличии условного оператора и функции NumPy. Функция numpy.where – это векторный вариант тернарного выражения «x if condition else y».

Преимущество векторных операций очевидно, но удивительно то, что в больших циклах использование массивов ndarray менее эффективно, чем использование списков чистого Python.

Опосредованная векторизация

Рассмотрим опосредованную векторизацию кода на примере применения метода кластеризации k-means для уменьшения количества цветов в изображении. В листинге 1 приведен код считывания трехканального изображения и подготовки данных для кластеризации. Далее в цикле по количеству итераций выполняется вычисление квадрата расстояния от каждого пикселя изображения до центра каждого кластера, для каждого пикселя определяется номер центра с минимальным расстоянием (листинг 2). Затем рассчитываются средние арифметические RGB для каждого кластера, если эти значения отличаются от центров кластеров менее чем на заданную точность, то цикл завершается. Иначе вычисленные значения становятся новыми центрами. После завершения цикла перекрашиваем все пиксели каждого кластера в цвет центра.

Листинг 1

```
from time import time
import matplotlib.pyplot as plt
path_to_png_file = r"C:\0110.png"
import matplotlib.image as mpimg
import numpy as np
# функция для вычисления квадрата расстояния
def dist(s1,s2):
    d = 0
    for si,sj in zip(s1,s2):
        d += (si-sj)**2
    return d
k = 5 # количество кластеров
max_iter = 10
# оригинальный рисунок
img = mpimg.imread(path_to_png_file)
```

```
# определяем высоту и ширину изображения
img_x = img.shape[0]
img_y = img.shape[1]
# массив размера img_x*img_y - номера кластеров для пикселей
z = np.zeros((img_x, img_y), dtype=np.int32)
# выбор центров кластеров, массив k*3
cl_center = np.zeros((k,3))
for i in range(3):
    cl_center [:, i] = np.linspace(0.25, 0.95, k)
```

Листинг 2

```
iter = 0
while iter != max_iter:
# вычисление квадратов расстояний до центров кластеров и
#                                 номера ближайшего центра
    for x in range(img_x):
        for y in range(img_y):
            z[x, y] = np.argmin([dist(img[x, y], cl_center[j])
                                 for j in range(k)])
# вычисление средних значений RGB в каждом кластере
    sum = np.zeros((k, 3))    # средние значения компонент RGB
                              #              в каждом кластере
    sum_n = np.zeros(k, dtype=np.int32)  # количество пикселей
                                         #   в каждом кластере
    for x in range(img_x):
        for y in range(img_y):
            sum_n[z[x, y]] +=1
            sum[z[x, y], :] += img[x, y, :]
    for i in range(k):
        if sum_n[i] != 0:
            sum[i, :] = sum[i, :]/sum_n[i]
# проверка достижения точности
    if np.allclose(sum, cl_center, 0.02):
        print('ups')
        break
# назначаем новые центры
    cl_center = np.copy(sum)
    iter += 1
```

```
# перекрашиваем все пиксели каждого кластера в цвет центра
for x in range(img_x):
    for y in range(img_y):
        img[x, y, :] = cl_center[z[x, y], :]
```

В программе много циклов, которые невозможно реализовать непосредственно как векторные операции. Среднее время выполнения одной итерации цикла для изображения размером (582, 800) и 5 кластеров составляет около 62 секунд (рис. 1, 2).

Очевидно, самый тяжеловесный цикл в программе – это вычисление расстояний от каждого пикселя до каждого центра кластера и определения номера ближайшего кластера. Для того чтобы избавиться от него, потребуется укладывание (транслирование) – способ выполнения арифметических операций над массивами разной формы. Это очень мощный механизм, но даже опытные пользователи иногда испытывают затруднения с его пониманием. Автор книги «Python и анализ данных» Уэс Маккинли пишет: «Даже я, опытный пользователь Numpy, иногда вынужден рисовать картинки, чтобы понять, как будет применяться правило укладывания» [3]. В последних версиях библиотеки Numpy появилась функция np.broadcast_shapes(), которая позволяет проверить правильность предполагаемой процедуры укладывания. Эта функция принимает на вход кортежи с размерностями исходных массивов и возвращает размерность результирующего массива после укладывания.

Укладывание в библиотеке NumPy следует строгому набору правил, определяющему взаимодействие двух массивов [4]:

1. Если размерность двух массивов отличается, форма массива с меньшей размерностью дополняется единицами с ведущей (левой) стороны.

2. Если форма двух массивов не совпадает в каком-то измерении, массив с формой, равной 1 в данном измерении, растягивается вплоть до соответствия форме другого массива.

3. Если в каком-либо измерении размеры массивов различаются и ни один не равен 1, генерируется ошибка.

Рассматриваемый цикл можно заменить одной строкой кода (листинг 3). Массив img имеет размерность (img_x, img_y, 3), массив

Рис. 1. Исходное изображение

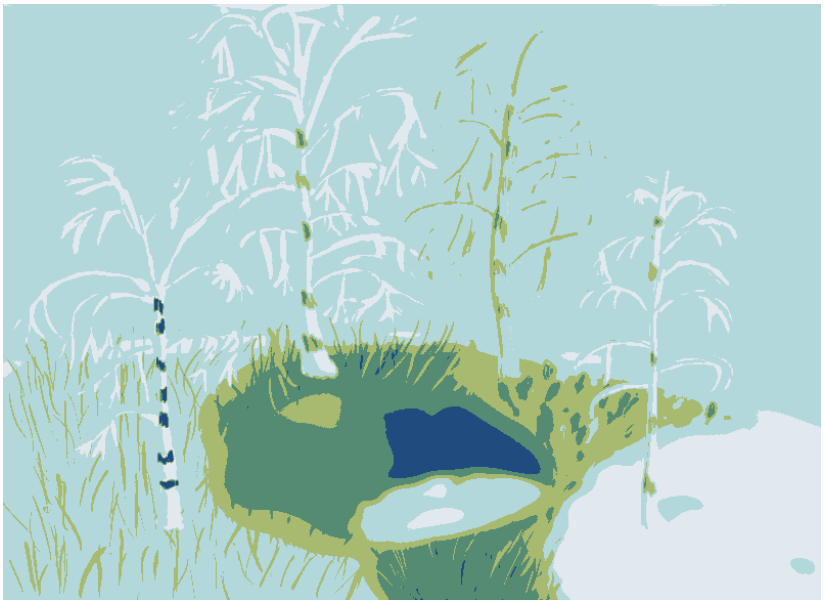

Рис. 2. Изображение после выполнения одной итерации

cl_center – (k, 3), запись cl_center[:, None, None, :] подразумевает размерность (k, 1, 1, 3). Размерность разности массивов (img – cl_center) определяется по правилам укладывания – (k, img_x, img_y, 3) (табл. 3). Ключевое слово None (синоним np.newaxis) вводит дополнительное измерение. Ключевое слово axis задает измерение массива, которое будет «схлопнуто».

Листинг 3

```
iter = 0
while iter != max_iter:
# вычисление расстояний до центров кластеров и номера
#                                    ближайшего центра
   z = np.argmin(np.sum((img-cl_center[:, None, None, :])**2,
                        axis=3), axis=0)
# вычисление средних значений RGB в каждом кластере
   sum = np.zeros((k,3)) # средние значения компонент RGB
                         # в каждом кластере
   for i in range(k):
     for j in range(3):
       sum[i, j] = np.sum(img[:, :, j][z==i])/(np.sum(z==i)+1)
# проверка достижения точности
   if np.allclose(sum, cl_center, 0.02):
     print('ups')
     break
# назначаем новые центры
   cl_center = np.copy(sum)
   iter += 1
# перекрашиваем все пиксели каждого кластера в цвет центра
   img = cl_center[z]
```

Таблица 3

Применение правила укладывания

img	1	Img_x	Img_y	3
cl_center	k	1	1	3
img – cl_center	k	Img_x	Img_y	3

После замены тройного цикла для расстояний на векторную операцию среднее время выполнения одной итерации цикла составляет около 4.5 секунд, т. е. сокращается приблизительно в 15 раз.

Вычисление средних значений компонент цвета для каждого кластера в первоначальной версии программы содержит два цикла. Использование маскирования (наложения маски) позволит заменить их одним (листинг 3). Маскирование заключается в индексировании по булеву массиву.

В результате применения к массиву z операции сравнения получаем булев массив (z==i) той же размерности, что и массив z. Вычисляем np.sum(z==i) – сумма истинных значений в булевом массиве, т. е. количество пикселей в кластере с номером i. Вырезаем j компонент RGB из матрицы изображения и накладываем маску, тем самым получаем все элементы, которые относятся к кластеру i: img[:, :, j][z==i].

Конечно, можно было бы еще больше упростить код, используя функцию np.mean() для вычисления средних значений. Но возможна ситуация, когда в некоторый кластер не попадет ни один пиксель, а функция np.mean() в этом случае выдает сообщение об ошибке.

После замены циклов для вычисления средних значений время выполнения одной итерации цикла составляет около 1.5 секунд, т. е. сокращается еще приблизительно в 3 раза.

Последний цикл в программе перекрашивает каждую точку изображения в цвет соответствующего центра кластера. Для векторизации цикла можно использовать прихотливую (причудливую) индексацию (листинг 3). Суть прихотливой индексации заключается в передаче целочисленного массива индексов с целью одновременного доступа к нескольким элементам массива. В случае прихотливой индексации форма результата отражает форму массива индексов, а не форму индексируемого массива.

После замены цикла перекрашивания пикселей среднее время выполнения одной итерации цикла составляет около 0.25 секунды, т. е. сокращается еще приблизительно в 5 раз.

Векторизация задачи

В некоторых случаях векторизация кода невозможна без предварительной векторизации задачи [5; 6], т. е. переосмысления алгоритма ее решения. Рассмотрим векторизацию задачи на примере фильтрации изображения. К изображению в оттенках серого применяется верти-

кальный фильтр Собеля (листинг 4), размер ядра 3 на 3, страйд равен 1. Исходное изображение имеет размер 400 на 400 пикселей, в результате фильтрации получаем изображение размером 398 на 398. Ядро скользит по матрице изображения, на каждом шаге вычисляется сумма попарных произведений соответствующих элементов ядра и текущего окна изображения, равного по форме ядру. Программа выполняется приблизительно 2.3 секунды. Стандартная функция cv2.filter2D() библиотеки OpenCV выполняется за наносекунды, результаты фильтрации идентичны (рис. 3, 4).

Листинг 4

```
import cv2
import numpy as np
from time import time
img = cv2.imread('C:\\804.png', cv2.IMREAD_GRAYSCALE)
img = img/255
=========================================================
yadro = np.array([[-1,0,1],[-2,0,2],[-1,0,1]])
res = np.zeros((398,398))
# фильтрация 1
for i in range(398):
   for j in range(398):
      res[i, j] = np.sum(yadro*img[i: i+3, j: j+3])
=========================================================
#использование библиотечной функции фильтра
im = cv2.filter2D(img, -1, yadro)
=========================================================
cv2.imshow('NO FILTER ', img)
cv2.waitKey(0)
cv2.imshow('LIBRARY FILTER', im)
cv2.waitKey(0)
cv2.imshow('MY FILTER', res)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Рис. 3. Исходное изображение

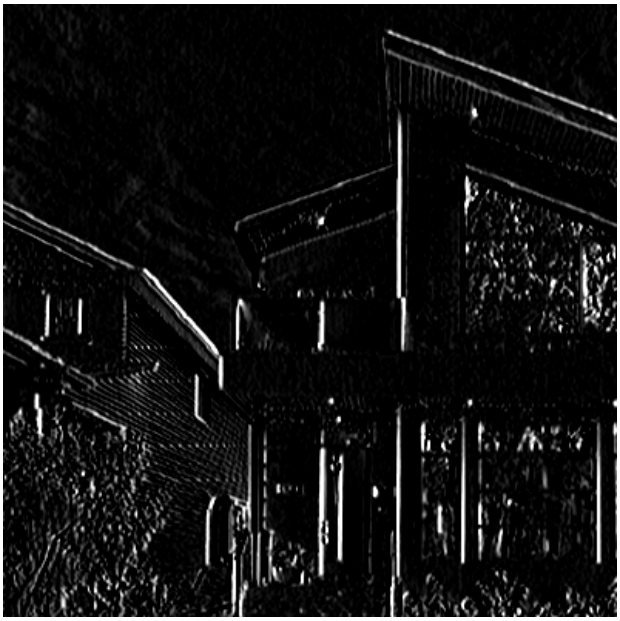

Рис. 4. Изображение после фильтрации

Индексы среза меняются на каждом шаге цикла, поэтому от цикла нельзя избавиться. Но слова «сумма попарных произведений» любого математика наводят на мысль о произведении матриц: если бы элементы ядра были строкой матрицы, а элементы окон – столбцами матрицы (или наоборот), то можно было бы за одну операцию выполнить все вычисления. Для этого нужно предварительно сформировать массив из всех окон. Имеет место проблема пространственно-временного компромисса: размер массива всех окон будет в 9 раз больше (для ядра 3 на 3) размера исходного изображения. Цикл формирования массива окон занимает меньше времени, чем цикл вычисления результата для каждого окна (листинг 5). В этом случае алгоритм фильтрации выполняется приблизительно за 0.5 секунды.

Идея скользящего (сканирующего) окна используется не только в алгоритме фильтрации, но и во многих других алгоритмах обработки изображений. Начиная с версии 1.20.00, в Numpy появилась функция np.lib.stride_tricks.sliding_window_view() [2], которая строит массив из всех возможных окон изображения (в общем случае произвольного массива) (листинг 5). Применение этой функции сокращает время выполнения алгоритма до приблизительно 0.016 секунды.

Листинг 5

```
yadro1 = np.array([-1,0,1,-2,0,2,-1,0,1])
=======================================================
# фильтрация 2
k=0
res2 = np.zeros((9,158404))
for j in range(398):
   for i in range(398):
      res2[0:9, k] = img[i: i+3, j: j+3].flatten()
      k+=1
res2 = np.dot(yadro1, res2)
res2 = np.transpose(res2.reshape((398,398)))
=======================================================
# фильтрация 3
v = np.lib.stride\_tricks.sliding\_window\_view(img, (3, 3))
v = v.reshape(398,398,9)
res1 = np.dot(v, yadro1.T)
res1 = res1.reshape((398,398))
```

Заключение

Библиотека NumPy хорошо документирована и имеет довольно простой и продуманный синтаксис, что важно в процессе обучения. Изучение библиотеки студентами направления подготовки «Прикладная информатика» происходит в рамках дисциплин «Интеллектуальный анализ данных» и «Математические методы в экономике». Опыт показал, что студенты легко справляются с освоением универсальных функций и маскирования. Затруднения возникают при необходимости использования механизмов укладывания и прихотливой индексации, что не удивительно – использование этих механизмов требует элементов творчества и, конечно, опыта.

Список источников

1. **Harris C. R., Millman K. J., van der Walt S. J. et al.** Array programming with NumPy // *Nature.* 2020. No. 585. Pp. 357–362. https://doi.org/10.1038/s41586-020-2649-2.

2. NumPy documentation. Version: 1.25.dev0. URL: https://numpy.org/devdocs/user/basics.copies.html (дата обращения: 07.02.2023).

3. **Уэс Маккинли.** Python и анализ данных / пер. с англ. А. А. Слинкин М.: ДМК Пресс, 2015. 482 с.

4. **Плас Дж. Вандер.** Python для сложных задач: наука о данных и машинное обучение. СПб.: Питер, 2018. 576 с.

5. **Nicolas P. Rougier.** From-python-to-numpy. URL: https://www.labri.fr/perso/nrougier/from-python-to-numpy/#code-vectorization (дата обращения: 07.02.2023).

6. **Shenoy A.** How Are Convolutions Actually Performed Under the Hood. URL: https://towardsdatascience.com/how-are-convolutions-actually-performed-under-the-hood-226523ce7fbf (дата обращения: 07.02.2023).

References

1. **Harris C. R., Millman K. J., van der Walt S. J. et al.** Array programming with NumPy. *Nature*, 2020, no. 585. pp. 357–362. https://doi.org/10.1038/s41586-020-2649-2.

2. *NumPy documentation. Version: 1.25.dev0.* Available at: https://numpy.org/devdocs/user/basics.copies.html (accessed: 07.02.2023).

3. **Ues Makkinli.** *Python i analiz dannyh / per. s angl. A. A. Slinkin* [Python for data analysis] M.: DMK Press, 2015. 482 p.

4. **Plas Dzh. Vander.** *Python dlya slozhnyh zadach: nauka o dannyh i mashinnoe obuchenie* [Python for complex tasks. Data Science and Machine Learning]. SPb.: Piter, 2018. 576 p. (In Russ.)

5. **Nicolas P. Rougier.** *From-python-to-numpy.* Available at: https://www.labri.fr/perso/nrougier/from-python-to-numpy/#code-vectorization (accessed: 07.02.2023).

6. **Shenoy A.** *How Are Convolutions Actually Performed Under the Hood.* Available at: https://towardsdatascience.com/how-are-convolutions-actually-performed-under-the-hood-226523ce7fbf (accessed: 07.02.2023).

Сведения об авторе / Information about author
Бабикова Надежда Николаевна / Nadezhda N. Babikova
к.пед.н., доцент, доцент кафедры прикладной информатики / PhD (Pedagogy), Associate Professor, Associate Professor of Applied Informatics Department
Сыктывкарский государственный университет им. Питирима Сорокина / Pitirim Sorokin Syktyvkar State University
167001, Россия, г. Сыктывкар, Октябрьский пр., 55 / 167001, Russia, Syktyvkar, Oktyabrsky Ave., 55

Статья поступила в редакцию / The article was submitted 02.03.2023
Одобрено после рецензирования / Approved after reviewing 06.03.2023
Принято к публикации / Accepted for publication 10.03.2023

Вестник Сыктывкарского университета.
Серия 1: Математика. Механика. Информатика. 2023.
Выпуск 1 (46)
Bulletin of Syktyvkar University.
Series 1: Mathematics. Mechanics. Informatics. 2023; 1 (46)

Научная статья

УДК 517.9, 539.3
https://doi.org/10.34130/1992-2752_2023_1_30

УСКОРЕНИЕ КРИПТОГРАФИЧЕСКИХ ВЫЧИСЛЕНИЙ ПУТЕМ НИЗКОУРОВНЕВОЙ ОПТИМИЗАЦИИ БАЗОВЫХ БЛОКОВ

Юрий Валентинович Гольчевский[1], **Дмитрий Александрович Ушаков**[2]
[1,2]Сыктывкарский государственный университет
им. Питирима Сорокина, yurygol@mail.ru

Аннотация. В данной работе представлено исследование проблемы оптимизации программного кода при реализации алгоритмов шифрования. Выделены базовые блоки криптографического алгоритма на примере алгоритма «Кузнечик». Реализованные варианты алгоритма с использованием различных версий векторных инструкций и их комбинаций протестированы на процессорах различных микроархитектур. Некоторые разработанные варианты реализации алгоритма показывают большую скорость шифрования, чем существующие программные продукты.

Ключевые слова: криптографические вычисления, низкоуровневая оптимизация, базовые блоки, алгоритм «Кузнечик»

Для цитирования: Гольчевский Ю. В., Ушаков Д. А. Ускорение криптографических вычислений путем низкоуровневой оптимизации базовых блоков // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика.* 2023. Вып. 1 (46). С. 30–49. https://doi.org/10.34130/1992-2752_2023_1_30

© Гольчевский Ю. В., Ушаков Д. А., 2023.

Article

Cryptographic Calculations Acceleration by Low-Level Optimization of Basic Blocks

Yuriy V. Golchevskiy[1], **Dmitriy A. Ushakov**[2]
[1,2]Pitirim Sorokin Syktyvkar State University, yurygol@mail.ru

Abstract. The paper presents a study of optimizing the program code problem when implementing encryption algorithms. The basic blocks of the cryptographic algorithm are highlighted on the example of the Kuznechik algorithm. Implemented variants of the algorithm using different versions of vector instructions and their combinations have been tested on processors of various microarchitectures. Some developed algorithm implementation variants show a higher encryption speed than existing software products.

Keywords: cryptographic computing, low-level optimization, basic blocks, algorithm Kuznechik

For citation: Golchevskiy Yu. V., Ushakov D. A. Cryptographic Calculations Acceleration by Low-Level Optimization of Basic Blocks. *Vestnik Syktyvkarskogo universiteta. Seriya 1: Matematika. Mekhanika. Informatika* [Bulletin of Syktyvkar University, Series 1: Mathematics. Mechanics. Informatics], 2023, no 1 (46), pp. 30−49. https://doi.org/10.34130/1992-2752_2023_1_30

Введение

В настоящее время оптимизация программного обеспечения (ПО) важна как для программ общего назначения (веб-браузеры, офисные пакеты), так и для специализированных программ (монтаж видео, моделирование различных процессов). К последним можно также отнести системы криптографической защиты информации (СКЗИ). Они в силу своей специфики (например, многократное применение блоков преобразований над открытым текстом) нуждаются в оптимизации участков кода, называемых базовыми блоками.

В качестве примера работ, представляющих результаты изучения проблемы оптимизации криптографических вычислений с различных точек зрения, можно привести работы [1–7]. В [1; 2] рассматриваются комплексный подход к ускорению криптографических вычислений

и метод оптимизации целочисленного деления посредством ассемблерных вставок. В [3–5] описаны способы оптимизации блочных алгоритмов шифрования, включая ГОСТ 28147-89 («Магма») с использованием технологии SSE. В работе [6] представлен подход к оптимизации программных решений. Ускорению ГОСТ 34.10-2015 («Кузнечик») посвящена работа [7], где предлагается усовершенствованный алгоритм умножения вектора на столбец матрицы без применения низкоуровневых инструкций.

Сказанное выше обусловливает некоторые задачи данного исследования:

- изучить основные направления оптимизации программного кода при реализации алгоритмов шифрования и выделить базовые блоки криптографического алгоритма на примере исследуемого ГОСТ Р 34.12-2015;
- исследовать код, содержащий intrinsic функции и сгенерированный компиляторами с разными уровнями оптимизации;
- реализовать оптимизированные версии алгоритма с использованием различных версий векторных инструкций и их комбинаций и протестировать их на процессорах различных микроархитектур, сравнить полученные результаты и сделать выводы о возможности и эффективности низкоуровневой оптимизации на основе численной оценки.

Способы оптимизации программ

Существует несколько направлений оптимизации программного кода: алгоритмическая, оптимизация потребления аппаратных ресурсов и низкоуровневая. Стоит подчеркнуть, что оптимизация не всегда возможна. Например, для программ, реализующих графический пользовательский интерфейс, скорость выполнения может оказаться низкой из-за недостаточной производительности графической подсистемы компьютера.

Использование алгоритмической оптимизации в большинстве случаев повышает производительность прикладных программ. Но использование этого метода не всегда оправдано для системного и специализированного ПО. Для него могут потребоваться низкоуровневые оптимизации путем ассемблерных вставок или полностью ассемблерной реализации «горячих точек» программы.

Повышение эффективности использования ресурсов процессора и памяти дает возможность возложить часть работы по увеличению быстродействия программы на компилятор с незначительными изменениями исходного кода приложения.

Основными препятствиями на пути повышения быстродействия программ являются вызовы функций, циклы, условные переходы и зависимости по данным [8–10]. Снижение производительности при использовании циклов обусловлено накладными расходами на инициализацию цикла, проверку условия выполнения, совершение условных переходов. Одним из приемов оптимизации циклов является их раскрутка – за одну итерацию совершается больше полезной работы и уменьшается доля накладных расходов.

Проблема условных переходов или ветвлений может решаться с помощью процесса предвыборки. В современных процессорах существует блок предсказания переходов, который может повысить точность предсказания переходов при достаточном количестве совершенных переходов.

Проблему зависимости данных (если для начала работы над одними данными нужны предварительные вычисления над другими данными, и процессор должен дождаться завершения работы над ними) можно решать путем использования нескольких временных переменных, операции над которыми могут быть исполнены параллельно.

Также существует проблема оптимизации программ по использованию памяти, тесно связанная с эффективным использованием ресурсов процессора. Это обусловлено тем, что обращение к памяти разных типов (регистры, кэш, оперативная память) занимает определенное количество тактов процессора [11]. Для оптимизации программы целесообразно отказаться от частого использования указателей, а значит, и обращений к медленной оперативной памяти, в пользу таких концепций, как локальность и «дружелюбный к кэшу» (cache-friendly) код [8].

Низкоуровневую оптимизацию следует производить после изучения кода, сгенерированного компилятором. Тогда появляется возможность использовать регистры процессора более эффективно, так как компиляторы не всегда используют все доступные регистры. Примером этого метода может служить использование векторных регистров XMM, YMM, ZMM для выполнения операций одновременно над несколькими наборами чисел, хранящихся в данных регистрах [12; 13].

Анализ кода, сгенерированного компиляторами

В ГОСТ Р 34.12-2015 определены три этапа преобразований входного вектора [14]:

1. Побитовое сложение 128-битного входного вектора с итерационным ключом с помощью операции исключающее «ИЛИ».

2. Нелинейное преобразование, заключающееся в применении к каждому 8-битному подвектору входного вектора фиксированной подстановки.

3. Линейное преобразование, сводящееся к операциям умножения вектора-строки промежуточного текста на коэффициенты в поле Галуа.

Кроме алгоритмических оптимизаций для ускорения производительности можно учитывать особенности ассемблера и планировщика, который может поставить инструкции на параллельное исполнение на разных исполняющих устройствах [15; 16].

Предлагается реализация алгоритмической оптимизации алгоритма, основанная на использовании набора инструкций SSE2 и XMM регистров. Данный выбор объясняется тем, что регистры XMM могут вместить блок данных объемом 128 бит целиком, что равно принятой длине входного блока ГОСТ Р 34.12-2015. Кроме того, набор инструкций SSE2 доступен на всех современных процессорах [17; 18].

Для того чтобы сделать выводы о целесообразности приведенного метода оптимизации, необходимо определить реализации шифра «Кузнечик», производительность которых будет оцениваться:

- версия на языке C без использования intrinsic функций (далее src-версия);

- версия на языке C с использованием intrinsic функций, приведенная в статье [19] (https://github.com/aprelev/libgost15) (далее itr-версия).

Тестируемые версии компилировались при помощи GNU Compiler Collection 4.9.2 (GCC) и Intel C++ Compiler из набора Intel Parallel Studio XE 2016 (ICL). При компиляции обеих версий высокоуровневой реализации использовались следующие опции компиляции GCC [20]:

- -shared -c -m32 (-m64) -mmmx -msse -msse2 -Wall -std=c99 для всех версий;
- -Os для оптимизации программы по размеру;
- -O3 для оптимизации программы по скорости выполнения.

По результатам проведенного анализа были сделаны следующие выводы:

- версии, оптимизированные по размеру кода (ключ -Os), показывают неудовлетворительную производительность, поэтому были исключены из дальнейшего тестирования;
- src-версии используют только регистры общего назначения (РОН), т. е. компилятор не смог распознать возможность оптимизации и использовать более эффективные технологии (MMX, SSE, AVX);
- в 32-битной itr-версии компилятор произвел оптимизацию, что позволило сэкономить 1 регистр;
- в 64-битной itr-версии компилятор использовал все доступные РОН для сохранения указателей на данные и раундовые ключи шифрования.

Для дальнейших исследований было решено оставить 64-битную itr-версию, так как ее код был сгенерирован наиболее оптимальным образом.

При компиляции обеих версий высокоуровневой реализации использовались следующие опции компиляции ICL в различных комбинациях [21]:

- /QaxSSE2,SSE3,SSSE3,SSE4.1,SSE4.2,AVX,CORE-AVX-I,CORE-AVX2;
- /Qipo для включения межпроцедурной оптимизации;
- /O3 для оптимизации программы по скорости выполнения;
- /Od для отключения оптимизации.

По итогам проведенного анализа сгенерированного кода было решено оставить 32-битную и 64-битную src-версии. Это объясняется тем, что в 32-битной версии компилятор распознал возможность оптимизации и использовал младшие 64 бита XMM регистров для вычислений. В 64-битной версии компилятор не использовал XMM регистры, а производил вычисления на 64-битных РОН.

Ручная оптимизация (ассемблерная реализация)

Согласно [22], низкоуровневую оптимизацию необходимо начинать после анализа листингов, сгенерированных компиляторами, поэтому в тестирование были включены высокоуровневые версии на языке C. Современные компиляторы могут генерировать достаточно оптимальный код с точки зрения производительности, используя возможности векторных расширений, даже если это явно не указано в коде. Однако возможности для оптимизации остаются. С другой стороны, прирост быстродействия от техник «ручной» низкоуровневой оптимизации должен составлять не менее 20 %. Если прирост производительности меньше этого порогового значения, то усилия, потраченные на оптимизацию, не имеют смысла согласно [23]. Это утверждение позволяет сделать вывод об успешности примененных методов.

Для ассемблерных реализаций выбор компилятора не имеет значения, так как весь код оформлен в виде объектного файла, созданного FASM, содержимое которого переносится компилятором в исполняемый файл без изменений. Компиляция таких версий проводилась без использования оптимизации [24].

Приведем несколько пояснений для дальнейшего изложения проблемы.

Ядром будет называться минимальная совокупность команд и используемых ими регистров, достаточная для осуществления табличной подстановки.

Используемые регистры можно разделить:

1) на регистры-накопители (далее – РН) – содержат результат одного раунда, который затем складывается с ключом;

2) регистры-маски (далее – РМ) – хранят маску (могут отсутствовать, если маска загружается из памяти);

3) регистры-экстракторы (далее – РЭ) – из них извлекаются байты для подстановки по таблице LSP-преобразований.

Всего было подготовлено несколько десятков ассемблерных реализаций алгоритма как для 32-битного, так и для 64-битного режима работы процессора. Наименования всех исполняемых файлов версий алгоритма, приведенных на графиках, будем строить по определенной схеме:

ex-архитектура-[компилятор][используемые наборы векторных расширений]-[ядро][модификаторы].exe,

которая для экономии места будет преобразована в следующую схему:

ex-архитектура-[g|i][r,m,s,s3,s4,a,a2]-[Ln|Wn|Bn|Un|Mn][Pn,A,E,R,C,M,X,G,K[n],f,n,l,p,t].exe.

Компилятор:
— g – GCC, i – ICL.

Используемые наборы векторных расширений:
— r – регистры общего назначения;
— m – MMX;
— s – SSE или SSE2, s3 – SSSE3, s4 – SSSE4.x;
— a – AVX, a2 – AVX2.

Ядро (n – число задействованных регистров для одного раунда):
— L – из РЭ байты извлекаются командой pextrw;
— W – из РЭ байты извлекаются командой pmovzxbw;
— B – из РЭ байты извлекаются командой pextrb;
— U – из РЭ байты извлекаются командами punpcklbw/punpckhbw;
— M – РЭ отсутствуют, байты извлекаются прямо из специально выделенной области памяти;
— R – в качестве РЭ используются регистры общего назначения;
— F – из РЭ байты извлекаются командой pshufb.

Модификаторы:
— K – раундовый ключ «кеширован» в XMM регистрах;
— E – раундовый цикл полностью развернут;
— Pn – за одну итерацию раундового цикла обрабатываются n блоков;
— A – вместо умножения/сдвига на 16 используется сложение с самим собой и множитель 8 при обращении к памяти;
— C – константы-смещения для таблицы подстановки помещаются в РЭ вместо указания их при обращении к памяти;
— X – группировка блоков вычисления РЭ;

— G – группировка блоков РН;
— n – ближний порядок, f – дальний порядок, l – реассоциация регистров;
— t – использование свободных регистров для «кеширования» значений памяти.

Тестирование производительности

Итоговое тестирование проводилось путем измерения скорости шифрования сгенерированного случайным образом массива данных (усредненной по многим испытаниям).

Вычисление скорости шифрования производилось путем деления объема шифруемых данных на среднее время выполнения цикла шифрования.

Данные для оценки скорости шифрования генерируются случайным образом на основе алгоритма «Вихрь Мерсенна» (Mersenne Twister) и его свободно распространяемой реализации TinyMT (https://github.com/MersenneTwister-Lab/TinyMT).

На первом этапе проводилось шифрование полного объема тестовых данных размером 512 МБ, затем его половины и т. д. — до размера блока. Также следует учитывать, что с уменьшением размера шифруемых данных время шифрования уменьшается, поэтому количество испытаний должно увеличиваться пропорционально.

Схема проведения исследования представлена в работе [25].

Тестирование производительности реализованных версий производилось на процессорах различных микроархитектур (Haswell, Vishera, Sandy Bridge, Arrandale, Wolfdale).

Результаты тестирования

На рис. 1, 2 представлены значения скорости шифрования тестируемых версий для 32- и 64-битного режима работы процессора соответственно.

Близость кривых позволяет говорить о том, что особенности организации конвейера, присущие конкретной микроархитектуре, являются наиболее существенным фактором, определяющим эффективность тестируемых способов низкоуровневой оптимизации. Напротив, такие факторы, как различия в типе процессоров, рабочих частотах, размерах кэшей, не оказывают существенного влияния.

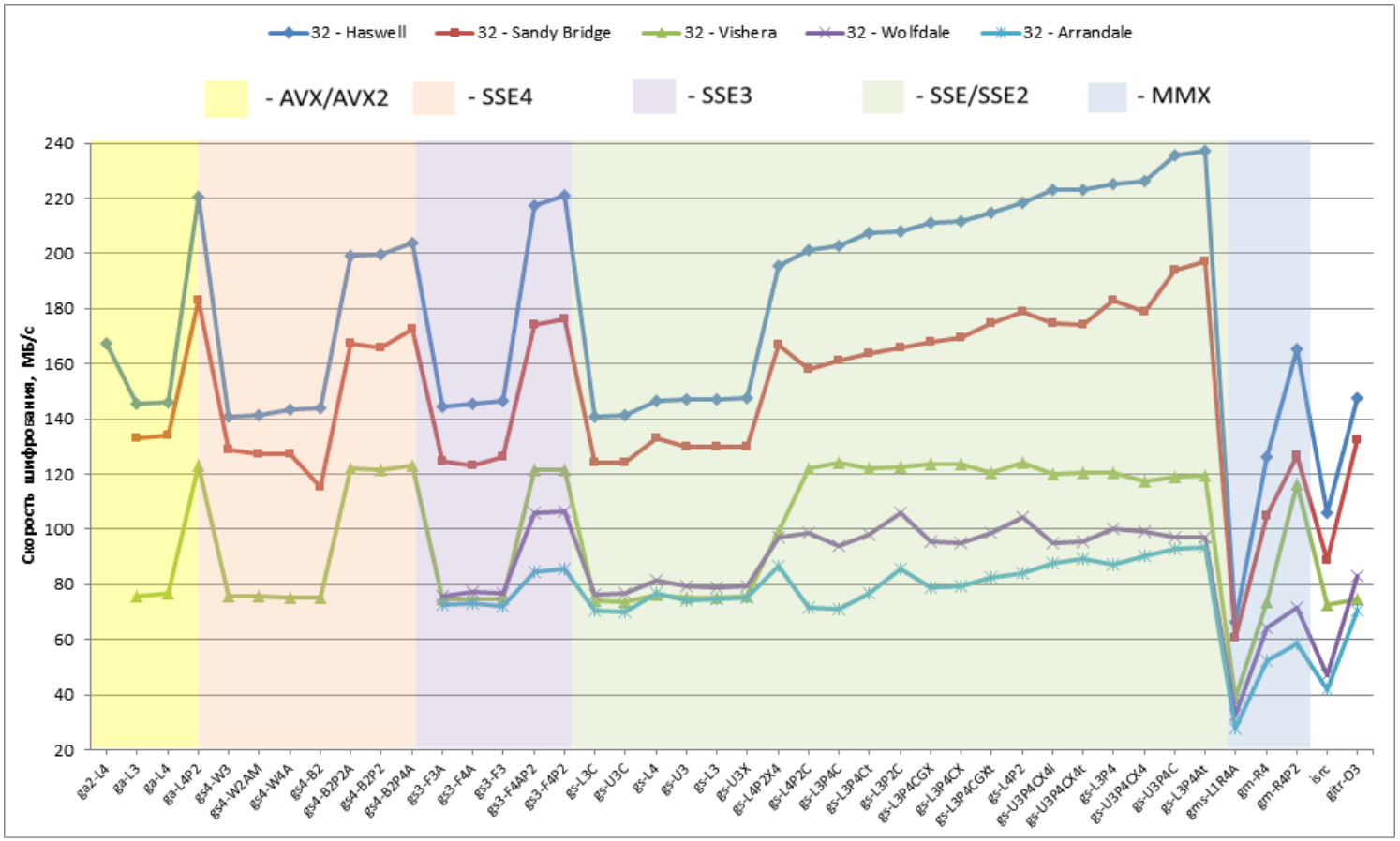

Рис. 1. Результаты тестирования для 32-битного режима работы процессора

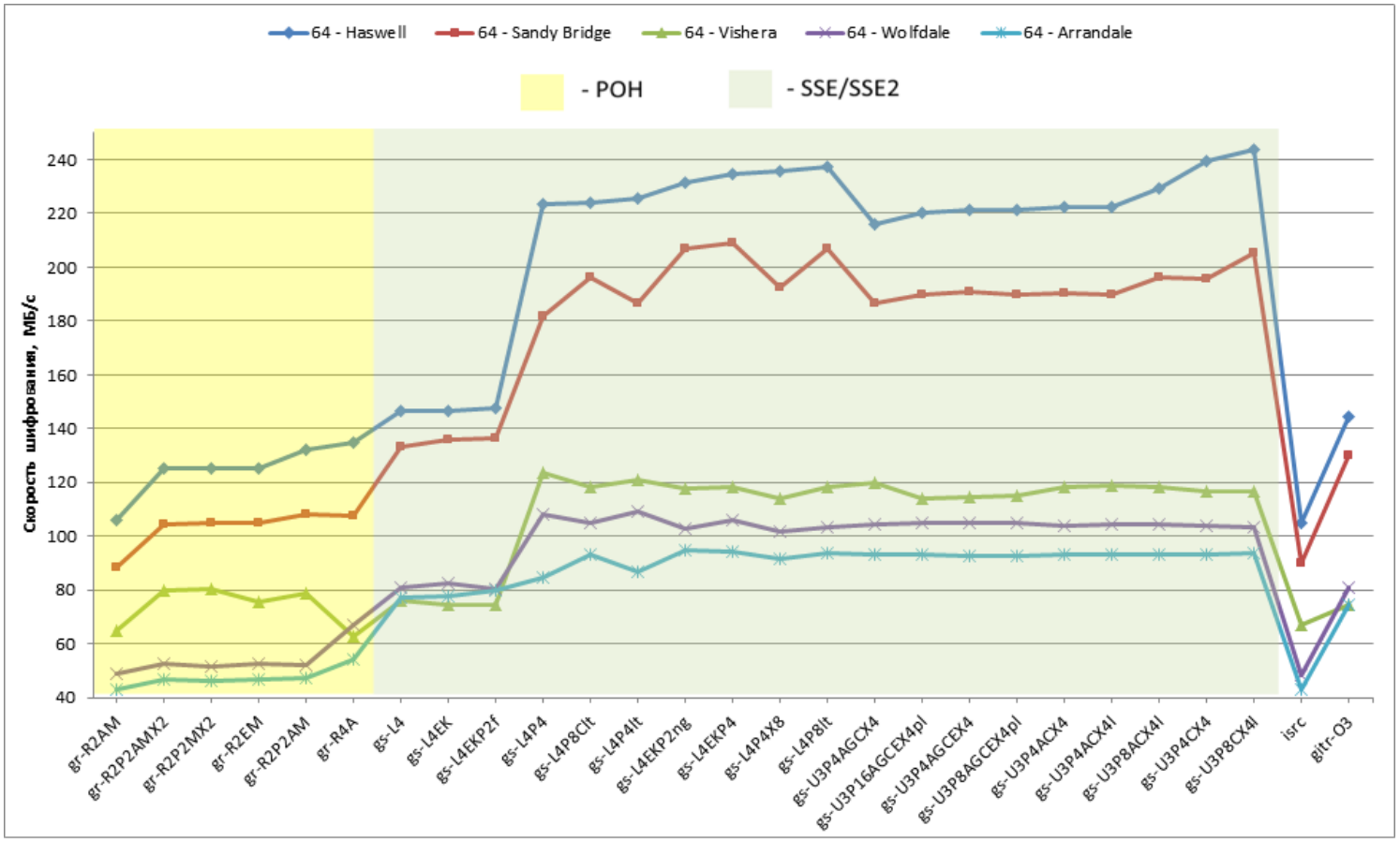

Рис. 2. Результаты тестирования для 64-битного режима работы процессора

Рассмотрим скорости шифрования для 32-битных версий, реализованных с использованием разных наборов векторных инструкций (рис. 1). Тестирование показывает, что производительность алгоритма шифрования «не пропорциональна» версии используемых векторных расширений, что является следствием ее принадлежности к классу «преимущественно конвейеризируемых» программ [26]. Решения SSE/SSE2 в целом наиболее оптимальны с точки зрения производительности. Использование MMX инструкций в ряде случаев приводит к снижению производительности в несколько раз по сравнению с наиболее оптимизированными реализациями.

Далее необходимо привести сравнение версий, использующих инструкцию pextrw для извлечения данных из РЭ и четыре регистра для обработки блока открытого текста, одна из которых сгенерирована компилятором, а другая реализована на ассемблере. Версия girt-O3 (сгенерированная компилятором) дает прирост скорости шифрования для микроархитектур Haswell и Wolfdale и ее падение до 6 МБ/с для остальных микроархитектур по сравнению с версией gs-L4 (ассемблерная реализация).

Для 64-битного режима работы процессора (рис. 2) результаты аналогичны представленным ранее, за исключением версий gr-R4A, gs-L4, gs-L4EK и gs-L4EKP2f для микроархитектуры Vishera. Такое поведение может быть объяснено особенностями микроархитектуры Vishera (более низкая ассоциативность кеша данных L1 и его меньший объем по сравнению с процессорами Intel).

Для процессоров Intel прирост производительности составляет в среднем 2,25 раза. Для процессоров микроархитектуры Vishera прирост производительности ниже как для 32-битного, так и для 64-битного режима работы процессора.

Анализ зависимости скорости шифрования от частоты процессора, не углубляясь в особенности микроархитектуры, которые обусловливают различия производительности тестируемых версий, показал, что в данном эксперименте процессор AMD уступал по характеристикам быстродействия сопоставимым процессорам Intel, несмотря на более высокую тактовую частоту. Возможно, одна из основных причин заключается в более низкой ассоциативности кеша данных L1 процессора и его меньшем объеме, что приводит к снижению эффективности кеширования, поскольку одни и те же кеш-линейки вынуждены попеременно замещать свое содержимое данными из разных участков памяти.

Выводы

Проведенный анализ указанных работ и данных работ [27; 28] позволяет сказать, что некоторые разработанные варианты реализации алгоритма шифрования «Кузнечик» показывают большую скорость шифрования, являясь как минимум не менее эффективными, чем существующие программные продукты. Оптимизации, основанные на использовании различных версий векторных расширений, позволяют добиться значительного (от 1,7 до 3,2 раз) улучшения производительности шифрования ассемблерных реализаций над src-реализациями, это означает, что усилия, потраченные на низкоуровневую оптимизацию, согласно [23], имеют смысл.

Список источников

1. **Северин П. А., Гольчевский Ю. В.** Комплексный подход к ускорению криптографических вычислений // *Информационные технологии в управлении и экономике.* 2012. № 2. С. 36–39.

2. **Гольчевский Ю. В., Северин П. А.** Об оптимизации криптографических алгоритмов посредством ассемблерных вставок при целочисленном делении // *Известия ТулГУ. Технические науки.* 2013. Вып. 3. С. 295–301.

3. Скоростное поточное шифрование // SecurityLab.ru [Электронный ресурс]. URL: http://www.securitylab.ru/analytics/436620.php (дата обращения: 03.12.2022).

4. **Павлов В. Э., Удальцов В. А.** Оптимизация скорости работы блочных алгоритмов шифрования // *Приоритетные направления развития образования и науки : материалы междунар. науч.-практ. конф. (Чебоксары, 9 апр. 2017 г.).* Чебоксары: ЦНС «Интерактив плюс», 2017. Т. 2. С. 76–80. doi: 10.21661/r-129822.

5. Особенности национальной криптографии // SecurityLab.ru [Электронный ресурс]. URL: http://www.securitylab.ru/analytics/480357.php (дата обращения: 03.12.2022).

6. **Гашин Р. А., Гольчевский Ю. В.** Разработка криптобиблиотеки на базе алгоритмов проекта eSTREAM // *Безопасность информационных технологий.* 2015. № 4 (22). С. 52–57.

7. **Ищукова Е. А., Кошуцкий Р. А., Бабенко Л. К.** Разработка и реализация высокоскоростного шифрования данных с использованием алгоритма «Кузнечик» // *Auditorium.* 2015. № 4 (8).

8. Оптимизация кода: процессор // Хабрахабр [Электронный ресурс]. URL: https://habrahabr.ru/post/309796/ (дата обращения: 03.12.2022).

9. **Bryant R., Hallaron D.** Computer Systems A Programmer's Perspective. Printice Hall, 2015. 1120 p.

10. **Гербер Р., Бик А., Смит К., Тиан К.** Оптимизация ПО : сборник рецептов. СПб.: Питер, 2010. 352 с.

11. **Кунин Р.** Оптимизация кода: память [Электронный ресурс]. URL: https://habrahabr.ru/post/312078/ (дата обращения: 03.12.2022).

12. Intel Instruction Set Extensions Technology [Электронный ресурс]. URL: https://www.intel.com/content/www/us/en/support/articles/000005779/processors.html (дата обращения: 07.12.2022).

13. **Киреев С. Е., Калгин К. В.** Эффективное программирование современных микропроцессоров и мультипроцессоров [Электронный ресурс]. URL: https://ssd.sscc.ru/sites/default/files/content/attach/317/lecture2016_01_intro.pdf (дата обращения: 03.12.2022).

14. ГОСТ Р 34.12-2015 Информационная технология. Криптографическая защита информации. Блочные шифры.

15. Intel 64 and IA-32 Architectures Software Developer Manuals [Электронный ресурс]. URL: https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html (дата обращения: 07.12.2022).

16. Intel 64 and IA-32 Architectures Optimization Reference Manual [Электронный ресурс]. URL: https://cdrdv2-public.intel.com/671488/248966-046A-software-optimization-manual.pdf (дата обращения: 07.02.2023).

17. Intel SSE4 Programming Reference [Электронный ресурс]. URL: https://www.intel.com/content/dam/develop/external/us/en/documents/d9156103-705230.pdf (дата обращения: 07.02.2023).

18. Intel Architecture Instruction Set Extensions Programming Reference [Электронный ресурс]. URL: https://cdrdv2-public.intel.com/671368/architecture-instruction-set-extensions-programming-reference.pdf (дата обращения: 07.02.2023).

19. ГОСТ Р 34.12 '15 на SSE2, или Не так уж и плох Кузнечик // Хабрахабр [Электронный ресурс]. URL: https://habrahabr.ru/post/312224/ (дата обращения: 03.12.2022).

20. Using the GNU Compiler Collection // GCC Online Documentation [Электронный ресурс]. URL: https://gcc.gnu.org/onlinedocs/gcc.pdf (дата обращения: 03.12.2022).

21. Intel C++ Compiler 19.1 Developer Guide and Reference // Intel Developer Zone [Электронный ресурс]. URL: https://www.intel.com/content/www/us/en/develop/documentation/cpp-compiler-developer-guide-and-reference/top.html (дата обращения: 07.12.2022).

22. **Fog A.** Optimizing Subroutines In Assembly Language: An Optimization Guide For x86 Platforms [Электронный ресурс]. URL: http://www.agner.org/optimize/optimizing_assembly.pdf (дата обращения: 03.12.2022).

23. **Касперски К.** Техника оптимизации программ. Эффективное использование памяти. СПб.: БХВ-Петербург, 2003. 464 с.

24. flat assembler 1.73 Programmer's Manual | flat assembler [Электронный ресурс]. URL: http://flatassembler.net/docs.php?article=manual (дата обращения: 03.12.2022).

25. **Гольчевский Ю. В.** Автоматизация исследования оптимизации скорости шифрования информации // *Информационные технологии в моделировании и управлении: подходы, методы, решения : сборник научных статей V Всероссийской научной конференции с международным участием.* Тольятти: Изд-во ТГУ, 2022. С. 208–215.

26. **Северин П. А., Гольчевский Ю. В.** Низкоуровневая оптимизация производительности на примере функции хеширования по ГОСТ Р 34.11-2012 // *Системный администратор.* 2017. № 1–2. С. 170–171.

27. **Ahmetzyanova L. R., Alekseev E. K., Oshkin I. B. и др.** On the properties of the CTR encryption mode of the Magma and Kuznyechik block ciphers with re-keying method based on CryptoPro Key Meshing // *IACR Cryptol.* ePrint Arch., 2016, 628 p.

28. **Гафуров И. Р.** Высокоскоростная программная реализация алгоритмов шифрования из ГОСТ Р 34.12-2015 // *Ученые записки УлГУ. Серия: Математика и информационные технологии.* 2022. Вып. 2. С. 38–48.

References

1. **Severin P. A., Golchevskiy Yu. V.** Comprehensive Approach for Cryptographic Computation Acceleration. *Informatsionnyye tekhnologii v upravlenii i ekonomike* [Information technologies in management and economics]. 2012, no 2, pp. 36–39. (In Russ.)

2. **Golchevskiy Yu. V., Severin P. A.** Cryptographic Algorithms Optimization by Means of Assembly Inserts in Integer Division. *Izvestiya TulGU. Tekhnicheskiye nauki* [News of TulGU. Technical sciences]. 2013, vol. 3, pp. 295–301. (In Russ.)

3. Fast Stream Encryption. *SecurityLab.ru.* Available at: http://www.securitylab.ru/analytics/436620.php (accessed: 03.12.2022).

4. **Pavlov V. E., Udaltsov V. A.** Optimizing the performance of block encryption algorithms. *Prioritetnyye napravleniya razvitiya obrazovaniya i nauki* [Priority directions of science and education development]. Cheboksary: SCC "Interaktiv plus LLC, 2017, 2(2), pp. 76–80. DOI: 10.21661/r-129822. (In Russ.)

5. Features of national cryptography. *SecurityLab.ru.* Available at: http://www.securitylab.ru/analytics/480357.php (accessed: 03.12.2022). (In Russ.)

6. **Gashin R. A., Golchevskiy Yu. V.** Development of Cryptographic Library Based on Algorithms from eSTREAM Project. *Bezopasnost' informatsionnykh tekhnologiy* [Information Technology Security], 2015, no 4 (22), pp. 52–57. (In Russ.)

7. **Ischukova E. A., Koshutsky R. A., Babenko L. K.** Development and implementation of high-speed data encryption using the Kuznechik algorithm. *Auditorium*, 2015, no 4 (8). (In Russ.)

8. Code Optimization: CPU. *Habrahabr.* Available at: https://habrahabr.ru/post/309796/ (accessed: 03.12.2022). (In Russ.)

9. **Bryant R., Hallaron D.** *Computer Systems A Programmer's Perspective.* Printice Hall, 2015. 1120 p.

10. **Gerber R., Bik A. J. C., Smith K., Tian X.** The Software Optimization Cookbook. St. Petersburg: Piter, 2010. 352 p. (In Russ.)

11. **Kunin R.** *Optimizatsiya Koda: pamyat'* [Code Optimization: Memory]. Available at: https://habrahabr.ru/post/312078/ (accessed: 03.12.2022). (In Russ.)

12. *Intel Instruction Set Extensions Technology.* Available at: https://www.intel.com/content/www/us/en/support/articles/000005779/processors.html (accessed: 07.12.2022).

13. **Kireev S. E. Kalgin K. V.** *Effektivnoye programmirovaniye sovremennykh mikroprotsessorov i mul'tiprotsessorov* [Efficient programming of modern microprocessors and multiprocessors]. Available at: https://ssd.sscc.ru/sites/default/files/content/attach/317/lecture2016_01_intro.pdf (accessed: 03.12.2022). (In Russ.)

14. *GOST R 34.12-2015 Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Blochnyye shifry* [GOST R 34.12-2015. Information technology. Cryptographic data security. Block ciphers]. (In Russ.)

15. *Intel 64 and IA-32 Architectures Software Developer Manuals.* Available at: https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html (accessed: 07.12.2022).

16. *Intel 64 and IA-32 Architectures Optimization Reference Manual.* Available at: https://cdrdv2-public.intel.com/671488/248966-046A-software-optimization-manual.pdf (accessed: 07.02.2023).

17. *Intel SSE4 Programming Reference.* Available at: https://www.intel.com/content/dam/develop/external/us/en/documents/d9156103-705230.pdf (accessed: 07.02.2023).

18. *Intel Architecture Instruction Set Extensions Programming Reference.* Available at: https://cdrdv2-public.intel.com/671368/architecture-instruction-set-extensions-programming-reference.pdf (accessed: 07.02.2023).

19. *GOST R 34.12 '15 na SSE2, ili Ne tak uzh i plokh Kuznechik* [GOST R 34.12 '15 on SSE2, or Not So Bad Kuznechik]. *Habrahabr.* Available at: https://habrahabr.ru/post/312224/ (accessed: 03.12.2022). (In Russ.)

20. Using the GNU Compiler Collection. *GCC Online Documentation.* Available at: https://gcc.gnu.org/onlinedocs/gcc.pdf (accessed: 03.12.2022).

21. Intel C++ Compiler 19.1 Developer Guide and Reference. *Intel Developer Zone.* Available at: https://www.intel.com/content/www/us/en/develop/documentation/cpp-compiler-developer-guide-and-reference/top.html (accessed: 07.12.2022).

22. **Fog A.** *Optimizing Subroutines In Assembly Language: An Optimization Guide For x86 Platforms.* Available at: http://www.agner.org/optimize/optimizing_assembly.pdf (accessed: 03.12.2022).

23. **Kaspersky K.** *Tekhnika optimizatsii programm. Effektivnoye ispol'zovaniye pamyati* [Program optimization technique. Efficient Memory Usage]. St. Petersburg, BHV-Petersburg, 2003. 464 p. (In Russ.)

24. *flat assembler 1.73 Programmer's Manual | flat assembler.* Available at: http://flatassembler.net/docs.php?article=manual (accessed: 03.12.2022).

25. **Golchevskiy Yu. V.** Automation of Encryption Speed Optimization Research. *Informatsionnyye tekhnologii v modelirovanii i upravlenii: podkhody, metody, resheniya : sbornik nauchnykh statey V Vserossiyskoy nauchnoy konferentsii s mezhdunarodnym uchastiyem* [Information technologies in modeling and management: approaches, methods, solutions: Collection of scientific articles: V All-Russian scientific conference with international participation]. Tolyatti: Publishing House of TSU, 2022, pp. 208–215. (In Russ.)

26. **Severin P. A., Golchevskiy Yu. V.** Low-Level Performance Optimization on the Example of the Hash Function GOST R 34.11-2012. *Sistemnyy administrator* [System Administrator]. 2017, no 1–2, pp. 170–171. (In Russ.)

27. **Ahmetzyanova L. R., Alekseev E. K., Oshkin I. B. et al.** On the properties of the CTR encryption mode of the Magma and Kuznyechik block ciphers with re-keying method based on CryptoPro Key Meshing. *IACR Cryptol.* ePrint Arch., 2016, 628 p.

28. **Gafurov I. R.** High-speed software implementation of encryption algorithms from GOST R 34.12-2015. *Uchenyye zapiski UlGU. Seriya: Matematika i informatsionnyye tekhnologii* [Scientific notes of UlGU. Series: Mathematics and Information Technology], 2022, no 2, pp. 38–48. (In Russ.)

Сведения об авторах / Information about authors
Гольчевский Юрий Валентинович / Yuriy V. Golchevskiy
к.ф.-м.н, доцент, заведующий кафедрой прикладной информатики / Ph.D. in Physics and Mathematics, Associate Professor, Head of Applied Informatics Department
Сыктывкарский государственный университет им. Питирима Сорокина / Pitirim Sorokin Syktyvkar State University
167001, Россия, г. Сыктывкар, Октябрьский пр., 55 / 167001, Russia, Syktyvkar, Oktyabrsky Ave., 55

Ушаков Дмитрий Александрович / Dmitriy A. Ushakov
технический специалист по информационным системам / Information Systems Technician
Сыктывкарский государственный университет им. Питирима Сорокина / Pitirim Sorokin Syktyvkar State University

167001, Россия, г. Сыктывкар, Октябрьский пр., 55 / 167001, Russia, Syktyvkar, Oktyabrsky Ave., 55

Статья поступила в редакцию / The article was submitted 06.02.2023
Одобрено после рецензирования / Approved after reviewing 14.02.2023
Принято к публикации / Accepted for publication 20.02.2023

Вестник Сыктывкарского университета.
Серия 1: Математика. Механика. Информатика. 2023.
Выпуск 1 (46)
Bulletin of Syktyvkar University.
Series 1: Mathematics. Mechanics. Informatics. 2023; 1 (46)

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

METHODICAL MATERIALS

Научная статья

УДК 378.1
https://doi.org/10.34130/1992-2752_2023_1_50

КОМПОНЕНТЫ ГЕОМЕТРО-ГРАФИЧЕСКОЙ КОМПЕТЕНЦИИ, ФОРМИРУЕМЫЕ ПРИ ИЗУЧЕНИИ НАЧЕРТАТЕЛЬНОЙ ГЕОМЕТРИИ В ТЕХНИЧЕСКОМ ВУЗЕ

Светлана Александровна Дейнега
Ухтинский государственный технический университет, deynega07@mail.ru

Аннотация. В статье рассматриваются обобщенные компоненты профессиональных компетенций студентов технического вуза. Раскрывается значимость формирования познавательно-созидательной компоненты на начальном этапе профессиональной подготовки. Показаны возможности формирования познавательно-созидательной компоненты геометро-графической компетенции при изучении начертательной геометрии.

Ключевые слова: начертательная геометрия, геометро-графическая компетенция, техническое образование

Для цитирования: Дейнега С. А. Компоненты геометро-графической компетенции, формируемые при изучении начертательной геометрии в техническом вузе // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика.* 2023. Вып. 1 (46). С. 50–63. https://doi.org/10.34130/1992-2752_2023_1_50

© Дейнега С. А., 2023.

Article

Components of Geometric-graphic Competence, Formed in the Study of Descriptive Geometry at a Technical University

Svetlana A. Deynega
Uchta State Technical University, deynega07@mail.ru

Abstract. The article considers the generalized components of the professional competencies of students of a technical university. The significance of the formation of a cognitive and creative component at the initial stage of professional training is revealed. The possibilities of forming a cognitive and creative component of geometry and graphic competence in the study of descriptive geometry are shown.

Keywords: projective geometry, geometric-graphic competence, technical education

For citation: Deynega S. A. Components of Geometric-graphic Competence, Formed in the Study of Descriptive Geometry at a Technical University. *Vestnik Syktyvkarskogo universiteta. Seriya 1: Matematika. Mekhanika. Informatika* [Bulletin of Syktyvkar University, Series 1: Mathematics. Mechanics. Informatics], 2023, no 1 (46), pp. 50–63. https://doi.org/10.34130/1992-2752_2023_1_50

В современных условиях развития техники и технологий значение графической компетенции как составляющей профессиональной компетентности выпускников технических вузов является общепризнанным. Высокий уровень графической компетенции необходим для успешной профессиональной деятельности выпускника. Формирование данной компетенции начинается при изучении базовых графических дисциплин в техническом вузе, к которым относят начертательную геометрию, инженерную и компьютерную графику.

Теоретическим ядром данных дисциплин является начертательная геометрия, так как содержит основные теоретические положения геометрического моделирования, что позволяет обеспечить дальнейшее изучение не только инженерной и компьютерной графики, но и многих других технических дисциплин высшего образования.

Графическая подготовка в системе высшего образования направлена на изучение свойств пространственных форм, методов их изображения

на плоскости, способов решения задач на плоскости с пространственными формами, создание технической документации (текстовой и графической) [1]. Законы представления пространственных форм в плоские изображения, плоского чертежа к пространственным моделям изучает учебная дисциплина «Начертательная геометрия» [2].

На современном этапе графическую подготовку рассматривают как систему, состоящую из двух частей: предметно-содержательной (начертательная геометрия) и профессионально-деятельностной (инженерная и компьютерная графика) [4]. Предметно-содержательный компонент обеспечивает фундаментальную графическую подготовку выпускника с направленностью их деятельности:

– на повышение эффективности техники и технологий;

– освоение новых знаний, способов и опыта профессиональной деятельности;

– разработку и использование графических изображений в практической деятельности.

Профессионально-деятельностный компонент графической подготовки инженера ориентирован на готовность к комплексной проектно-конструкторской, исследовательской и научной деятельности в профессиональной области.

Обзор научной литературы показал, что именно фундаментальная подготовка выпускника является основой его будущей профессиональной гибкости, трансформации на протяжении всей профессиональной жизни, предоставляет ему возможность понять и освоить новое оборудование и технологии, новые принципы организации производства [3]. Кроме того, изучение фундаментальных дисциплин раскрывает общие закономерности развития материального мира, формируя научное мировоззрение будущих выпускников технических вузов.

Таким образом, фундаментальность предметно-содержательной компоненты направлена на формирование универсальных, долгосрочных и системообразующих знаний и навыков в общепрофессиональных и профессиональных компетенциях. Одну из фундаментальных позиций в подготовке инженера в техническом вузе занимает начертательная геометрия. Решение многих технических задач можно производить аналитически и графически, выбирая наиболее целесообразный метод

решения или комбинируя их вместе. Графическое решение задач в определенных случаях быстрее и проще, чем аналитическое, но менее точное. Методы графического решения задач рассматривают в курсе начертательной геометрии и успешно применяют для решения прикладных инженерных задач.

Начертательная геометрия изучает и обосновывает способы изображений пространственных форм (линий, поверхностей, тел) на плоскости и способы решений задач геометрического характера по заданным изображениям указанных форм. Для успешного освоения начертательной геометрии студенты должны иметь достаточные знания в области стереометрии и планиметрии.

Исследователи и преподаватели технических вузов подчеркивают, что именно изучение этой науки о геометрическом моделировании реальных объектов на плоскости является основополагающим компонентом в формировании профессиональных инженерных компетенций студента, а потому требует высокого качества обучения по сформированности геометро-графических компетенций.

С этой позиции важным является рассмотрение компонентов геометро-графической компетенции, формирование которой осуществляется в процессе изучения начертательной геометрии.

Существуют различные подходы исследователей к определению профессиональной компетентности (В. И. Байденко, Е. В. Бондаревская, И. А. Зимняя, Н. В. Кузьмина, А. К. Маркова, Н. В. Матяш, Н. С. Розов, В. В. Сериков и др.), во многих из которых исследователи выделяют личностную составляющую и владение профессиональными навыками и знаниями. В Федеральном государственном образовательном стандарте высшего образования модель выпускника определена общепрофессиональными и профессиональными компетенциями, на основе которых Е. В. Савченко [5] выделяет общие компетенции для всех инженерных специальностей и разделяет их по направлениям:

– компетенции, характеризующиеся способностями к обработке и анализу информации;

– компетенции, характеризующиеся способностями принимать обоснованное решение, находить решение задачи;

– компетенции, включающие способности к графической, математической и информационной обработке процесса решения задачи.

Выпускники высших учебных заведений инженерных специальностей начинают формировать данные компетенции при изучении дисциплин фундаментальной подготовки.

В данной статье мы рассматриваем компоненты геометрографической компетенции, которую относим к базовой общепрофессиональной компетенции инженера.

Анализ научных работ исследователей (И. А. Зимняя, Н. В. Кузьмина, В. В. Краевский, В. А. Сластенин, Ю. Г. Татур, А. В. Хуторский и др.) позволил определить, что структура компетенций в общем случае состоит из следующих компонентов:

– когнитивного – характеризуется совокупностью знаний, умений и навыков в предметной области;

– личностного – заключается в направленности на профессиональную деятельность и непрерывное самообучение, сформированности профессионально важных качеств, ценностного отношения к содержанию и результату, положительной мотивации к выполнению учебной деятельности;

– операционального – отвечает за способность приобретать и использовать систему знаний, умений и навыков, необходимых для решения учебных задач;

– рефлексивного – связан с оценкой собственной деятельности, формирует способность к самоанализу, самосовершенствованию, самодиагностике.

Анализ работ исследователей по графической подготовке студентов (А. Д. Ботвинников, К. А. Вольхин, В. В. Вязанкова, В. Н. Гузненков, И. Д. Столбова, В. И. Якунин и др.) показал, что формирование компетенции рассматривается в процессе обучения в комплексе графических дисциплин: «Начертательная геометрия», «Инженерная графика», «Компьютерная графика». При этом структура графической компетенции состоит из компонентов, которые в основном отражают совокупность знаний, умений и навыков, ориентированных на выполнение и понимание чертежей (инженерная графика), решения задач разного уровня сложности (начертательная геометрия), составление конструкторской и проектной документации (инженерная и компьютерная графика), в т. ч. с помощью компьютера.

Для успешной профессиональной деятельности будущему инженеру необходимо постоянно обучаться и самосовершенствоваться, поэтому важно уметь оценивать собственные знания по предметному материалу и выделять недостаток теоретических знаний и практических умений. Познавательная готовность способствует формированию новой системы знаний, дополнительных умений и навыков с опорой на имеющиеся знания.

Не менее важной является способность оценивания процесса нахождения оптимального пути решения поставленной задачи и его обоснования. Выбор технологического решения и его конструирование определяется постановкой решаемой задачи и связан с созидательными качествами личности.

Следовательно, умения анализировать собственные знания, оценивать полученные результаты, критически относиться к собственной деятельности необходимы при оценивании проблемной ситуации, для мотивации к решению инженерной задачи в самостоятельном обучении и в практической деятельности. В структуре инженерных компетенций данные умения относят к личностной и рефлексивной компонентам, которые направлены на решение проблемных ситуаций, возникающих в инженерной деятельности, на поиск новых инженерных решений, постановку инженерных задач и конструирования решений. Таким образом, можно утверждать, что данные умения являются познавательно-созидательной составляющей в структуре инженерных компетенций, так как направляют деятельность инженера на познание с целью создания субъективно нового «продукта».

Формирование познавательно-созидательной компоненты компетенций инженера необходимо начинать на ранних этапах вузовской подготовки в процессе изучения дисциплин образовательной программы в рамках организации учебно-познавательной деятельности. В техническом образовании с первого курса студенты изучают курс начертательной геометрии, нацеленный на формирование профессиональных компетенций. Содержание дисциплины является фундаментальным и необходимым заделом для изучения дисциплин профессионального блока в подготовке инженера.

Именно в курсе начертательной геометрии формируется геометро-графическая компетенция, которая характеризуется способностью оперирования абстрактными понятиями реальных предметов и отношений – абстракция отождествления, абстракция потенциальной беско-

нечности, выходящими за геометрию, связью количественных отношений и пространственных форм, использованием формализованной символьной записью, связью геометрических и математических моделей.

Геометро-графическая компетенция даст возможность сформировать у студентов технического вуза познавательно-созидательную составляющую компоненты структуры профессиональной компетентности, что позволит выпускнику в профессиональной деятельности выполнять постановку инженерных задач, оптимально решать различные проблемные ситуации, осуществлять поиск новых путей их решения.

Особенность изучения содержания начертательной геометрии базируется на решении тематических задач по алгоритмическим предписаниям, которые определяют последовательность решения. Логическая структура изучения начертательной геометрии определяет переход от простых алгоритмических предписаний к более сложным при решении тематических задач. В зависимости от сложности решаемых задач, изменяется последовательность алгоритмических предписаний. Эта последовательность определяется совокупностью тех типовых тематических задач, которые определяют весь процесс решения. Поэтому в содержании начертательной геометрии можно выделить процессуальный аспект, который базируется на выполнении сходных между собой действий при решении типовых тематических задач. Эти действия наращиваются при переходе от одной темы к следующей, образуя систему «вложенных» задач. Условием этого перехода при изучении начертательной геометрии является критерий сформированности действий по реализации алгоритмов, в т. ч. действий по их распознаванию и реализации, по применению совокупности алгоритмов при решении основных тематических задач в обучении. В этом случае важным для студента является действие по выявлению типовых задач в содержании учебного материала, т. е. действие по постановке задач.

Необходимо также отметить, что процесс решения задач начертательной геометрии связан с их графической, математической и информационной обработкой и конструированием геометрических моделей. В данном случае существенные свойства геометрических моделей отражают взаимосвязи между элементами геометрических объектов и системой образов, содержащей информацию о правилах выполнения действий с ними. Создание системы образов является одним из важных действий геометро-графического моделирования.

Система образов может быть представлена:

– информационными моделями (электронными геометрическими моделями на плоскости и в пространстве — представление с использованием компьютерного моделирования);

– графическими моделями (образно-знаковыми в виде изображения геометрических элементов в форме чертежей и схем в ортогональных проекциях, в перспективе, в числовых отметках);

– математическими моделями (символьной записью геометрических элементов, записью определенной логической последовательности решения задач).

В содержании начертательной геометрии существенные свойства понятий отражены во взаимосвязи между элементами геометрических объектов и системой их образов. Действия по установлению соотношений между системой понятий и системой образов абстрактной модели могут рассматриваться в различных цепочках связей: «понятие – образ» и «образ – понятие»; «образ – знак» и «знак – образ»; «понятие – образ – знак» и «знак – образ – понятие». Поэтому познавательно-созидательная компонента геометро-графической компетенции связана с установлением связей в указанных цепочках, выполнением действий по геометро-графическому моделированию.

При решении тематических задач начертательной геометрии у студентов формируются отдельные действия по решению инженерной задачи, в основном это действия по установлению связей «понятие – образ» и «образ – понятие» с геометро-графическим моделированием. В качестве примера на рис. 1 показано условие тематической задачи и ее решение.

В процессе решения задачи, показанной на рис. 1, при построении проекций по заданным координатам пространственных точек на плоскостях проекций студентом выполняются следующие действия:

– установление связей понятийного аппарата с их образом, образа с понятием (комплексного чертежа точек A, D, E; отрезков прямых AD и E; их позиционного расположения относительно плоскостей проекций – $AE \parallel \Pi_1,\ AD \nparallel \Pi_1, \Pi_2, \Pi_3$);

– установление связей между графическим образом комплексного чертежа точек A, D, E, отрезков прямых AD и AE с ее математической моделью через пространственные координаты – $A(100, 0, 25)$, $D(70, 25, 50)$, $E(45, 20, 25)$;

– моделирование геометрического образа.

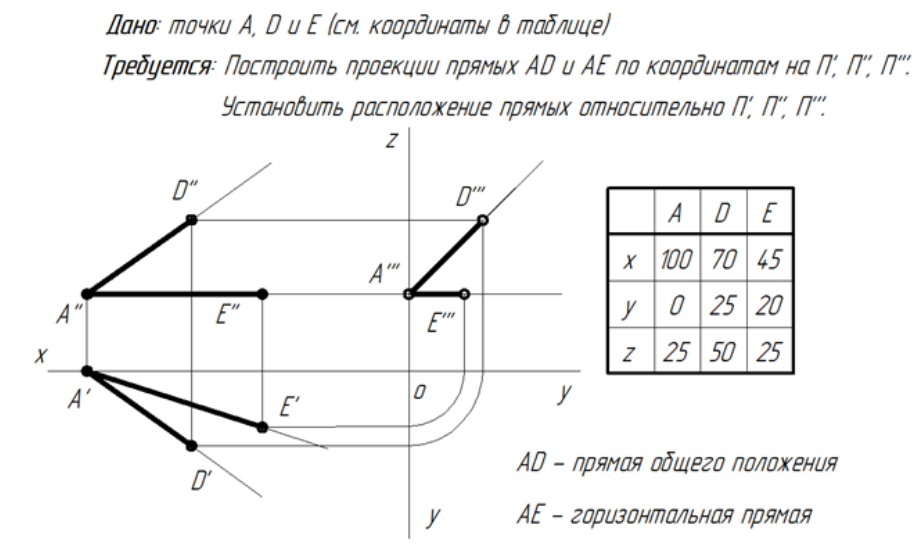

Рис. 1. Пример решения тематической задачи по построению комплексного чертежа проекций прямых

Если рассматривать систему «вложенных» задач, то в данном случае выполняются действия по установлению связей предыдущего уровня («понятие – образ» и «образ – понятие») и действия по установлению связей «образ – знак» и «знак – образ», так как в этом случае рассматривается запись алгоритмического предписания по решению. В действиях данных уровней результатом является моделирование геометрических образов и начинает формироваться умение применять геометро-графические методы, что важно для созидательной компоненты геометро-графической компетенции.

Система «вложенных» задач представляет собой комбинирование тематических задач, выполняемых в соответствии с логическим следованием решения задачи. Такие задачи называют комбинированными. При их решении важен путь следования или возможные пути при использовании разных методов решения. При решении комбинированных

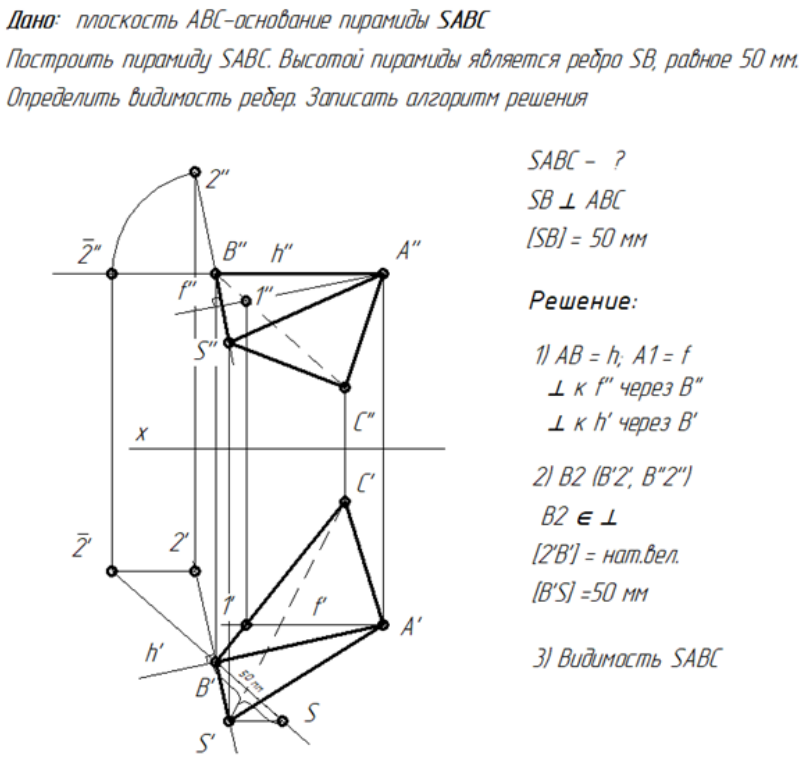

Рис. 2. Пример решения комбинированной задачи

задач (вложенные задачи) формируются обобщенные способы действия. В данном случае необходим дополнительный практический материал по решению задач подобного типа. Условие задачи и пример решения комбинированной задачи показан на рис. 2. Кроме описанных действий, выполняемых студентом при решении тематических задач (по установлению понятийного аппарата, соотношений между понятийной моделью и образом геометрической модели, между образом геометрической модели и понятийной модели, по моделированию геометрического образа), добавляются действия по постановке учебной задачи (конкретизация цели — найти высоту пирамиды SB, определить натуральную величину SB, определить видимость ребер), установлению последовательности выполнения алгоритмических операций (построение горизонтали h и фронтали f, построение к h и f перпендикуляра из точки B, метод поворота для определения натуральной величины SB), связей между геометрической и знаковой моделью (символьная запись алгоритма ре-

шения), между знаковой моделью и ее образом (графическое воспроизведение записи алгоритма решения задачи).

В полной мере действия студентов по установления соотношений между элементами геометрических объектов и системой их образов в направлениях «понятие – образ – знак» и «знак – образ – понятие» будут раскрыты в работе над учебным проектом, ориентированным на выполнение этих действий. В основе учебного проекта, который студент выполняет самостоятельно в рамках учебной деятельности, представлена учебная проблемная задача, условие которой поставлено в формате инженерного технического задания и ранее им не рассматривалось. Отличительной особенностью данных учебных инженерно-ориентированных задач от обычных условий тематических и комбинированных задач является их нацеленность на последовательное выполнение действий в технологии инженерной задачи при решении:

– постановка технического задания;

– описание средств и методов его выполнения;

– проработка возможных путей решения;

– выбор и принятие решений;

– проектирование, применение и корректировка.

В задании на учебный проект, помимо условия задачи, дано поэтапное описание решения в виде понятийно-знаковой модели обобщенного алгоритма решения. При работе над учебным проектом студентом выполняется постановка учебной задачи (конкретизация цели, последовательность решения, прогнозирование результата), определяются методы и средства решения, осуществляется выбор методов решения, выполняется решение учебной задачи и его графическое моделирование. При этом выполняются следующие действия:

– установление связей между системой понятий (поэтапное описание решения) и образом (комплексный чертеж точек, прямых) в обоих направлениях (геометрическое моделирование решения);

– установление двунаправленных связей между системой образов и знаков (символьная запись характеристики объектов и описания решения, символьное обозначение графических моделей);

– установление связей в виде «понятие – образ – знак» (символьная запись решения в виде алгоритма) и «знак – образ – понятие» (оформление учебного проекта, решения, подготовка учебного проекта к защите).

Таким образом, формирование познавательно-созидательной компоненты геометро-графической компетенции при изучении начертательной геометрии студентами технических вузов будет осуществляться при выполнении действий по постановке учебных задач, использованию методов геометро-графического моделирования при решении учебных задач. Создание ситуации, в которой студент сможет выполнить данные действия комплексно, возможно в процессе организации его учебной деятельности при решении учебных инженерно-ориентированных задач в рамках выполнения учебного проекта.

Список источников

1. **Якунин В. И., Гузненков В. Н.** Геометро-графические дисциплины в техническом университете // *Теория и практика общественного развития.* 2014. № 17. С. 191–195.

2. **Гузненков В. Н., Журбенко П. А.** Модель как ключевое понятие геометро-графической подготовки // *Alma mater (Вестник высшей школы).* 2013. № 4. С. 82–87.

3. **Вязанкова В. В.** Формирование графической компетентности бакалавров технических направлений подготовки в условиях информационно-образовательной среды // *Современные проблемы науки и образования.* 2021. № 2. URL: https://science-education.ru/ru/article/view?id=30663 (дата обращения: 03.03.2023).

4. **Гузненков В. Н., Якунин В. И., Серегин В. И. и др.** Компьютерная графика – основа геометро-графической подготовки // *Международный научно-исследовательский журнал.* 2016. № 4 (46). URL: https://research-journal.org/archive/4-46-2016-april/kompyuternaya-grafika-osnova-geometro-graficheskoj-podgotovki (дата обращения: 07.03.2023). doi: 10.18454/IRJ.2016.46.298

5. **Савченко Е. В.** Компоненты информационной компетенции будущего инженера, формируемые при изучении фундаментальных

дисциплин // *Современное образование.* 2020. № 4. С. 37–48. URL: https://nbpublish.com/library_read_article.php?id=31606 (дата обращения: 07.03.2023). doi: 10.25136/2409-8736.2020.4.31606

6. **Коваленко А. В.** Графическая компетенция как одна из составляющих профессиональной компетентности бакалавра профессионального обучения по направлению «051000.62. Профессиональное обучение (по отраслям)» // *Вестник ЮУрГГПУ.* 2011. № 10. С. 83–95. URL: https://cyberleninka.ru/article/n/graficheskaya-kompetentsiya-kak-odna-iz-sostavlyayuschih-professionalnoy-kompetentnosti-bakalavra-professionalnogo-obucheniya-po/viewer (дата обращения: 09.03.2023).

7. **Вольхин К. А., Лейбов А. М.** Проблемы формирования графической компетентности в системе высшего профессионального образования // *Философия образования.* 2012. № 4 (43). С. 16–22.

References

1. **Yakunin V. I., Guznenkov V. N.** Geometric and graphic disciplines at the technical University. *Teoriya i praktika obshchestvennogo razvitiya* [Theory and practice of social development]. 2014, no 17, pp. 191–195. (In Russ.)

2. **Guznenkov V. N., Zhurbenko P. A.** Model as a key concept of geometric-graphic training. *Alma mater (Vestnik vysshei shkoly)* [Alma mater (Bulletin of the Higher School)]. 2013, no 4, pp. 82–87. (In Russ.)

3. **Vyazankova V. V.** Formation of graphic competence of bachelors of technical areas of training in the conditions of information and educational environment. *Sovremennyye problemy nauki i obrazovaniya* [Modern problems of science and education]. 2021, no 2. Available at: https://science-education.ru/ru/article/view?id=30663 (accessed: 03.03.2023). (In Russ.)

4. **Guznenkov V. N., Yakunin V.I., Seregin V.I. et al.** Computer graphics – the basis of geometric-graphic training. *Mezhdunarodnyy nauchno-issledovatel'skiy zhurnal* [International Research Journal].

2016, no 4 (46). Available at: https://research-journal.org/archive/4-46-2016-april/kompyuternaya-grafika-osnova-geometro-graficheskoj-podgotovki (accessed: 07.03.2023). doi: 10.18454/IRJ.2016.46.298 (In Russ.)

5. **Savchenko E. V.** Components of the information competence of the future engineer, formed in the study of fundamental disciplines. *Sovremennoye obrazovaniye* [Modern Education]. 2020, no 4, pp. 37–48. Available at: https://nbpublish.com/library_read_article.php?id=31606 (accessed: 07.03.2023). doi: 10.25136/2409-8736.2020.4.31606 (In Russ.)

6. **Kovalenko A. V.** Graphic competence as one of the components of the professional competence of a bachelor of vocational training in the direction "051000.62 Vocational training (by industry)". *Vestnik YuUrGGPU* [Bulletin of the SUSUGPU]. 2011, no 10, pp. 83–95. Available at: https://cyberleninka.ru/article/n/graficheskaya-kompetentsiya-kak-odna-iz-sostavlyayuschih-professionalnoy-kompetentnosti-bakalavra-professionalnogo-obucheniya-po/viewer (accessed: 09.03.2023). (In Russ.)

7. **Volkhin K. A., Leibov A. M.** Problems of formation of graphic competence in the system of higher professional education. *Filosofiya obrazovaniya* [Philosophy of education]. 2012, no 4 (43), pp. 16–22.

Сведения об авторе / Information about author
Дейнега Светлана Александровна / Svetlana A. Deynega
старший преподаватель кафедры механики / senior lecturer of the Department of Mechanics
Ухтинский государственный технический университет / Ukhta State Technical University
69300 Россия, г. Ухта, ул. Первомайская, д. 13 / 69300 Russia, Ukhta, Pervomayskaya str., 13

Статья поступила в редакцию / The article was submitted 10.03.2023
Одобрено после рецензирования / Approved after reviewing 14.03.2023
Принято к публикации / Accepted for publication 17.03.2023

.

Вестник Сыктывкарского университета.
Серия 1: Математика. Механика. Информатика. 2023.
Выпуск 1 (46)
Bulletin of Syktyvkar University.
Series 1: Mathematics. Mechanics. Informatics. 2023; 1 (46)

Научная статья

УДК 372.851
https://doi.org/10.34130/1992-2752_2023_1_64

ЧИСЛОВЫЕ ПОСЛЕДОВАТЕЛЬНОСТИ КАК ОСНОВОПОЛАГАЮЩИЙ ФАКТОР ФОРМИРОВАНИЯ ТВОРЧЕСКОЙ АКТИВНОСТИ У БУДУЩИХ БАКАЛАВРОВ

Сергей Николаевич Дорофеев[1], **Наталия Владимировна Наземнова**[2]
[1] Тольяттинский государственный университет,
komrad.dorofeev2010@yandex.ru
[2] Пензенский государственный университет

Аннотация. В данной статье изучаются проблемы подготовки инженерных кадров к творческой деятельности в процессе изучения основ высшей математики. Введение адаптивного курса математики и разбиение самого курса высшей математики на три важных раздела: основы высшей алгебры и аналитической геометрии, основы интегрального и дифференциального исчисления, основы теории дифференциальных уравнений и теории вероятностей, создает благоприятный фундамент, на котором будущие инженеры могут не только успешно трудиться в сфере своей инженерной деятельности, но и успешно осваивать новые для них профессии, такие как учитель математики, конечно, после соответствующей переподготовки. Как свидетельствуют наши наблюдения, нынешнее образование можно охарактеризовать притоком в основные и общеобразовательные учреждения специалистов, закончивших технические вузы. Согласно современным требованиям, предъявляемым к профессиональной подготовке учителя, они должны пройти обучение в магистратуре по направлению

© Дорофеев С. Н., Наземнова Н. В. , 2023.

«44.04.01. Педагогическое образование», профиль «математическое образование», и получить диплом магистра по соответствующему профилю или закончить курсы перепрофилизации в соответствующем объеме.

В нашей работе мы проанализировали некоторые методические подходы и особенности к определению таких главных понятий курса математики, как последовательности и интегралы. Эти понятия играют исключительно важную роль не только в сфере деятельности будущего инженера, но и в сфере деятельности будущего учителя математики. Посредством этих понятий в сознании учителя формируется его математическое мировоззрение. Обосновали недостаточность одних и преимущество других в смысловом толковании этих понятий. Предлагаемая нами методика введения этих понятий базируется на числовых последовательностях, составленных из нижних или верхних сумм Дарбу, и обусловливает формирование у обучающихся приемов и умений творческой активности.

Ключевые слова: математическое образование, преемственность, фундаментальность, качество математической подготовки, числовые последовательности, интегралы

Для цитирования: Дорофеев С. Н., Наземнова Н. В. Числовые последовательности как основополагающий фактор формирования творческой активности у будущих бакалавров // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика.* 2023. Вып. 1 (46). С. 64–77. https://doi.org/10.34130/1992-2752_2023_1_64

Article

Numerical sequences as a fundamental factor in the formation of creative activity in future bachelors

Sergej N. Dorofeev[1], Natalija V. Nazemnova[2]
[1] Togliatti State University, komrad.dorofeev2010@yandex.ru
[2] Penza State University

Abstract. This article examines the problems of training engineering personnel for creative activity in the process of studying the basics of higher mathematics. The introduction of an adaptive mathematics course and the division of the higher mathematics

course into three important sections: the basics of higher algebra and analytical geometry, the basics of integral and differential calculus, the basics of the theory of differential equations and probability theory creates a favorable foundation on which future engineers can not only successfully work in the field of their engineering activities, but also successfully master new professions such as a math teacher, of course, after appropriate retraining. As our observations show, the current education can be characterized by the influx of specialists who have graduated from technical universities to basic and general education institutions. According to the modern requirements for the professional training of teachers, they must complete a master's degree in the direction "44.04.01. Pedagogical education" profile "mathematical education" and receive a master's degree in the appropriate profile or complete courses of retraining at an appropriate level.

Keywords: mathematical education, continuity, fundamentality, quality of mathematical training, numerical sequences, integrals

For citation: Dorofeev S. N., Nazemnova N. V. Numerical sequences as a fundamental factor in the formation of creative activity in future bachelors. *Vestnik Syktyvkarskogo universiteta. Seriya 1: Matematika. Mekhanika. Informatika* [Bulletin of Syktyvkar University, Series 1: Mathematics. Mechanics. Informatics], 2023, no 1 (46), pp. 64–77. https://doi.org/10.34130/1992-2752_2023_1_64

Введение

Подготовка инженерных кадров представляет одно из важных стратегических направлений развития экономики и обороноспособности нашей страны. Этот процесс должен быть неразрывным и отвечать самым высоким требованиям нашего общества. Современная сфера производства, как никогда, испытывает острую потребность в специалистах инженерного профиля высокого уровня, способных решать не только шаблонные задачи, но, прежде всего, задачи нестандартного характера, требующие проявления способности к поиску неалгоритмических путей их решения, проявления творческой активности, связанной с разработкой новых способов и методов. Решением таких задач могут заниматься специалисты, владеющие не только специальными знаниями, но и приемами составления математических моделей реальных ситуаций более высокого порядка, способные применять свои математические знания и умения на самом высоком уровне [1–3]. Нет сомнения в том, что это

должны быть специалисты, способные проявлять творческую инициативу и смекалку.

Обзор литературы

Перед современным инженером в сфере его профессиональной деятельности могут возникать задачи, в ходе решения которых ему необходимо будет использовать свои математические знания, умения и навыки, способности к переформулированию производственной задачи на язык математики. Творческая активность будущего специалиста инженерного профиля связана как с открытием нового в самой проблеме, так и в деятельности, направленной на поиск решения этой проблемы. Открытие «нового» невозможно как без отрицания «старого», так и без сохранения всего положительного, что было накоплено в прежнем, — преемственности. По своему внутреннему содержанию преемственность, как мы уже отмечали в работах [4; 5], можно трактовать как категорию, отражающую объективный и всеобщий характер, проявляемый в природе, обществе и познании. Следует отметить, что основными принципами подготовки будущих бакалавров инженерного профиля к творческой деятельности, связанной с созиданием новых идей и открытий, мы относим не только преемственность, но и доступность, научность, системность, наглядность, принцип постепенного восхождения от простого к сложному. Некоторые пути формирования творческой активности изучались нами в работах [6–9].

Обсуждения

В курсе высшей математики для бакалавров инженерного профиля неотъемлемой частью служит раздел «Основы интегрального исчисления». Многие студенты с небольшим восторгом вспоминают те времена, когда им пришлось изучать этот раздел. Сложности восприятия основных понятий этого раздела связаны с тем, что студенты, к сожалению, не смогли понять сущность определений самых простых понятий, таких как определенный, криволинейный, поверхностный и кратные интегралы [10; 11]. Подробнее нам бы хотелось остановиться именно на сущностной основе этих понятий. Усвоение сущностной основы определенного интеграла, как показывает наш опыт, служит фундаментом, на основании которого будущие инженеры с незначительной помощью преподавателя приходят к самостоятельному открытию таких понятий, как криволинейный, двойной, тройной и поверхностный интеграл. Сущностная основа – это то, что является самым важным и самым главным

в определении этих понятий [11; 12]. К нашему сожалению, многие преподаватели при введении понятия, например, определенного интеграла стараются избежать таких определений, как нижняя или верхняя сумма Дарбу, а следовательно, и возникающих при этом числовых последовательностей. Чаще говорят о числовых последовательностях, составленных из интегральных сумм, но при этом не обращают внимание обучающихся на тот факт, что эта числовая последовательность почему-то сходится. В результате в сознании студентов складывается мнение, что это утверждение не является составляющим фактором определения определенного, криволинейного, поверхностного или кратного интегралов. На наш взгляд, каждому ведущему преподавателю необходимо четко представлять всю линию развития теории кратных и поверхностных интегралов. Следует отметить, что во всех этих понятиях важную значимость приобретают числовые последовательности, составленные из нижних или верхних сумм Дарбу. Нам следует подробнее остановиться на этих фактах, ибо возникающие перед современным инженером некоторые профессиональные задачи не содержат в себе конкретных определенных или криволинейных интегралов, их надо еще построить и лишь потом, используя математические знания, изучать их поведение. Как правило, решение проводят через нижние или верхние суммы Дарбу или через интегральные суммы, затем через возникающие посредством их числовые последовательности и, наконец, через предельные значения числовых последовательностей. Кто-то может возразить по поводу того, зачем так делать, надо использовать компьютерные технологии. Сразу предостережем таких поклонников компьютерных технологий о том, что перед компьютером тоже надо поставить задачу. Для этого надо описать все начальные условия на определенном языке программирования, в нашем случае это есть конкретные нижние или верхние суммы Дарбу или интегральные суммы, порождаемые ими числовые последовательности. При этих условиях компьютер, возможно, укажет нам соответствующие им предельные значения. По нашему мнению, с целью реализации творческого подхода к обучению студентов математическим методам необходимо при изучении каждой темы, прежде всего, обращать внимание на ее системообразующие понятия, развивать у студентов способность отслеживать пути развития этого понятия, генезис его происхождения и предвидеть следующие направления его реализации [13]. Акцентирование внимания на главных понятиях означает, что необходимо выделять все его существенные признаки, последова-

тельность их участия в определении, на возможности и невозможности перестановки этих признаков при формулировке определения [2; 12; 14]. Итак, как же нам следует организовать процесс обучения наших студентов – будущих специалистов инженерного профиля математическим основам интегрального исчисления, чтобы они могли успешно использовать их при изучении проблемных ситуаций профессионального значения, обусловливающих открытие новых знаний и методов? Во-первых, следует отметить, что этот раздел состоит из следующих частей:

1. Числовые последовательности. Предел числовой последовательности. Теоремы о существовании предела возрастающей (убывающей) и ограниченной сверху(снизу) числовой последовательности.
2. Числовые функции. Предел функции. Производная функции.
3. Неопределенный интеграл. Свойства. Методы вычисления неопределенных интегралов.
4. Определенный интеграл. Свойства. Методы вычисления. Формула Ньютона – Лейбница.
5. Кратные интегралы.
6. Криволинейные интегралы.
7. Поверхностные интегралы.

С целью формирования понятия «предел числовой последовательности» и развития способности к осмысленному восприятию новой информации мы считаем целесообразным использовать визуализацию или графическую иллюстрацию как приемы отражения существенных признаков этого понятия. На наш взгляд наиболее эффективным следует признать принцип постепенного восхождения к сущностным основам нового понятия. Для этого мы рекомендуем в качестве первоначального примера использовать числовую последовательность вида $x_n = \frac{1}{n}$. Как известно, предел этой последовательности при $n \to \infty$ равен 0. Используя этот частный факт, мы начнем формирование у обучающихся представления о пределе числовой последовательности, используя принцип визуализации. Для этого на числовой прямой отметим точку 0 и несколько членов нашей последовательности, например

$$x_1 = 1, x_2 = \frac{1}{2}, x_5 = \frac{1}{5}, x_{10} = \frac{1}{10}, x_{20} = \frac{1}{20}, x_{25} = \frac{1}{25}.$$

Теперь рассмотрим какую-нибудь ε-окрестность точки 0, например, при $\varepsilon = 0.1$. С помощью обучающихся определим из обозначенных нами членов нашей последовательности те, которые лежат внутри ε-окрестности точки 0 при $\varepsilon = 0.1$, и те, которые не лежат внутри этой окрестности. Важно обратить их внимание на тот факт, что внутрь этой окрестности попадают те члены нашей последовательности, номера которых строго больше 10. Теперь возьмем $\varepsilon = 0.2$ и с помощью обучающихся назовем те члены нашей последовательности, которые попадают внутрь ε-окрестности точки 0 при $\varepsilon = 0.2$, и те, которые не лежат внутри этой окрестности. В этом случае при наводящем вопросе, с какого номера все члены нашей последовательности попадают в ε-окрестность точки 0 при $\varepsilon = 0.2$, большая часть студентов смело отвечают, что с номера больше чем 5. Таким образом, на данном примере мы доказали важный факт: факт существования предела числовой последовательности. В математике такой результат относят к важным теоремам: теоремам существования. Далее рассмотрим произвольную числовую последовательность x_n с заранее известным пределом b. Используя прием визуализации, на числовой прямой отметим точку b и несколько членов нашей произвольной последовательности. Затем возьмем достаточно малое положительное число ε_1 и на числовой прямой отметим ε_1-окрестность точки b так, чтобы какие-то члены последовательности попали внутрь этой окрестности, а какие-то остались вне ее. Важно отметить, что внутрь этой окрестности попадет достаточно большое бесчисленное множество членов последовательности, а вот вне ее останется только конечное множество. Теперь возьмем достаточно малое положительное число ε_2 и на числовой прямой отметим ε_2-окрестность точки b. Отметим так, чтобы какие-то члены последовательности не попали внутрь этой окрестности, а все остальные попали бы внутрь этой окрестности. Как показывает наш опыт, на основе предыдущего примера и на основе этого факта у обучающихся формируется устойчивое представление о пределе числовой последовательности. Они уже самостоятельно могут выделить и озвучить все существенные признаки этого понятия. Теперь необходимо систематизировать их представления и сформулировать определение, используя все существенные признаки понятия: число b называется пределом числовой последовательности x_n, если для любого достаточно малого положительного числа ε найдется такое натуральное число N, что для всех $n > N$ выполняется неравенство $|x_n - b| < \varepsilon$ [15; 16].

Следующий важный этап – это доказательство единственности существования предела числовой последовательности. Необходимость реализации этого этапа развития теории пределов продиктована принципами развития математической теории. В самом начале мы доказали, что теория пределов изучает не пустые объекты, а существующие математические понятия, теперь следует доказать, что всякая сходящаяся числовая последовательность имеет только один предел. Среди многих свойств предела числовой последовательности мы считаем необходимым заострить особое внимание на теоремах:

1. Если три числовые последовательности x_n, y_n, z_n удовлетворяют условиям: $x_n \leq y_n \leq z_n$ и $\lim\limits_{n\to\infty} x_n = \lim\limits_{n\to\infty} z_n = b$, то $\lim\limits_{n\to\infty} y_n = b$ (теорема о трех солдатах).

2. Если числовая последовательность x_n возрастает и ограничена сверху, то она сходится.

3. Если числовая последовательность x_n убывает и ограничена снизу, то она сходится.

Эти теоремы служат фундаментальной основой определения определенного, двойного, тройного, криволинейных и поверхностных интегралов. Вторая и третья теоремы обеспечивают сходимость числовых последовательностей, составленных из верхних и нижних сумм Дарбу, а первая обеспечивает сходимость числовой последовательности, составленной из интегральных сумм. В свою очередь, сходимость этих числовых последовательностей посредством теоремы о трех солдатах обеспечивает сходимость числовой последовательности, составленной из интегральных сумм [15]. К сожалению, во многих учебных пособиях по высшей математике доказательство этого факта умышленно пропускается. С методической точки зрения это может быть и оправдано, но с точки зрения научности методики преподавания это слишком произвольно. По нашему мнению, вводить понятия определенного, криволинейного, поверхностного и кратных интегралов необходимо через числовые последовательности, составленные из нижних и верхних сумм Дарбу. После того как ввели понятие каждого из этих интегралов, уже можно доказать теорему о том, что числовая последовательность, составленная из интегральных сумм, тоже сходится к соответствующему интегралу. Как известно, интегральные суммы удобнее использовать

в приложениях. В каждой конкретной задаче всегда можно выбрать такие точки из области определения функции, для которых значения самой функции станут легко вычислимыми [15].

Творческий подход к изучению некоторых разделов математики, в частности числовых последовательностей и основ дифференциального и интегрального исчисления, на инженерных специальностях университетов и технических вузов, реализованный по предлагаемой нами системе, обеспечивает эффективность усвоения знаний не только по вертикали, но и по горизонтали. Усвоение будущими инженерами на более осмысленном уровне основных математических методов в течение трех семестров обусловливает фундаментальность их математического образования. Следует отметить, что творческий подход к изучению курса математики на инженерных специальностях, как мы уже отмечали в [5], «ориентирован не только на студентов с достаточно высоким уровнем сформированности операционно-содержательного компонента, но и на студентов с низким уровнем сформированности мотивационного и волевого компонентов».

Процесс организации аудиторных и внеаудиторных занятий на очных и дистанционных формах обучения по математике, разделенной на отдельные дисциплины «Высшая алгебра и аналитическая геометрия», «Основы дифференциального и интегрального исчисления», «Основы дифференциальных уравнений» и «Теории вероятностей» «способствует не только повышению у обучающихся математической грамотности, но и формированию у них умения выделять существенные стороны исследуемой проблемной ситуации» [6; 17].

Заключение

Таким образом творческий подход к обучению студентов инженерных специальностей математическим методам познания окружающего мира способствует повышению качества их математического образования, а также создает прочный фундамент для овладения новыми профессиями, в частности профессией учителя математики, и формирует в их сознании потребности и умения разрешать производственные задачи посредством построения математических моделей на более высоком творческом уровне и, используя их, находить оптимальные способы решения.

Список источников

1. **Дорофеев С. Н., Есетов Е. Н., Наземнова Н. В.** Аналогия как основа обучения школьников векторному методу решения геометрических задач // *Вестник Сыктывкарского университета. Серия 1: Математика. Механика. Информатика.* Вып. 4 (41). 2021. С. 69–79.

2. **Менчинская Н. А.** Проблемы обучения и психического развития студентов. М.: Педагогика, 1989. 224 с.

3. **Слепкань З. И.** Психолого-педагогические основы преподавания математики : метод. пособие. Киев, 1983. 192 с.

4. **Дорофеев С. Н.** Теория и практика формирования творческой активности будущих учителей математики в педагогическом вузе : дис. ... д-ра пед. наук. Пенза, 2000. 410 с.

5. **Дорофеев С. Н., Иванова Т. А., Утеева Р. А. и др.** Преемственность в подготовке будущих бакалавров педагогического образования (профиль «Математика») к творческой деятельности // *Гуманитарные науки и образование.* 2018. Т. 9. № 4 (36). С. 25–30.

6. **Дорофеев С. Н.** Компетентностный подход к математическому образованию студентов технических вузов // *Педагогическое образование и наука.* 2009. № 1. С. 88–91.

7. **Дорофеев С. Н.** УДЕ как метод подготовки будущих бакалавров педагогического образования к профессиональной деятельности // *Гуманитарные науки и образование* 2013. № 1. С. 14–17.

8. **Dorofeev S. N., Pavlov I. I., Shichiyakh R. F., Prikhodko A. N.** Differentiated Training as a Form of Organization of Education fnd Cognitive Activity of Future Masters of Pedagogical Education // *Applied Lingvistics Research Jounal.* 2021. 5 (3). Pp. 216–222.

9. **Dorofeev S., Shichiyach R. A., Khasimova L. N.** Devoloping creative activity abilities of students in higer educaitional esteblishments // *Revista on line de politica e gistao educational.* 2021. Т. 25. № S2. Pp. 883–900.

10. **Фридман Л. М.** Психолого-педагогические основы преподавания математики в школе: учитель математики о педагогической психологии. М.: Просвещение, 1983. 160 с.

11. **Талызина Н. Ф.** Формирование математических понятий // *Формирование методов математического мышления* / под ред. Н. Ф. Талызиной. М.: Московский государственный университет им. М. В. Ломоносова; ТОО «Вентана-Граф», 1995. С. 13–28.

12. **Болл Г. А.** Теория образовательных проблем: психолого-педагогический аспект. М.: Педагогика, 1990. 184 с.: ил.

13. **Зак А. З.** Как определить уровень развития мышления студента. М.: Знание, 1982. 96 с.

14. **Выготский Л. С.** Собрание сочинений : в 6 т. Т. 2. Проблемы общей психологии / под ред. В. В. Давыдова. М.: Педагогика, 1982. 504 с.: ил.

15. **Дорофеев С. Н.** Высшая математика. М.: ООО «Издательство "Мир и образование"», 2011. 592 с.: ил.

16. **Кудрявцев Л. Д.** Мысли о современной математике и ее изучении. М.: Наука, 1977. 123 с.

17. **Выготский Л. С.** Собрание сочинений : в 6 т. Т. 3. Проблемы развития психики / под ред. А. М. Матюшкина. М.: Педагогика, 1983. 368 с.: ил.

References

1. **Dorofeev S. N., Esetov E. N., Nazemnova N. V.** Analogy as a basis for teaching schoolchildren the vector method of solving geometric problems. *Vestnik Syktyvkarskogo universiteta. Seriya 1: Matematika. Mekhanika. Informatika* [Bulletin of Syktyvkar University. Series 1: Mathematics. Mechanics. Computer science]. 2021, issue 4 (41), pp. 69–79. (In Russ.)

2. **Menchinskaya N. A.** *Problemy obucheniya i psikhicheskogo razvitiya studentov* [Problems of teaching and mental development of students]. M.: Pedagogy, 1989, 224 p. (In Russ.)

3. **Slepkan Z. I.** *Psikhologo-pedagogicheskiye osnovy prepodavaniya matematiki : metod.posobiye* [Psychological and pedagogical foundations of teaching mathematics: method.stipend]. Kiev, 1983, 192 p. (In Russ.)

4. **Dorofeev S. N.** *Teoriya i praktika formirovaniya tvorcheskoy aktivnosti budushchikh uchiteley matematiki v pedagogicheskom vuze* [Theory and practice of formation of creative activity of future teachers of mathematics at a pedagogical university, dissertation for the degree of Doctor of Pedagogical Sciences]. Penza, 2000. 410 p. (In Russ.)

5. **Dorofeev S. N., Ivanova T. A., Uteeva R. A. et al.** Continuity in the preparation of future bachelors of pedagogical education (profile "Mathematics") for creative activity. *Gumanitarnyye nauki i obrazovaniye* [Humanities and Education]. 2018, vol. 9, no 4 (36), pp. 25–30. (In Russ.)

6. **Dorofeev S. N.** Competence-based approach to mathematical education of students of technical universities. *Pedagogicheskoye obrazovaniye i nauka* [Pedagogical education and science]. 2009, no 1, pp.88–91. (In Russ.)

7. **Dorofeev S. N.** UDE as a method of preparing future bachelors of pedagogical education for professional activity. *Gumanitarnyye nauki i obrazovaniye* [Humanities and Education]. 2013, no 1, pp. 14–17. (In Russ.)

8. **Dorofeev S. N., Pavlov I. I., Shichiyakh R. F., Prikhodko A. N.** Differentiated Training as a Form of Organization of Education fnd Cognitive Activity of Future Masters of Pedagogical Education. *Applied Lingvistics Research Jounal*, 2021, 5 (3), pp. 216–222.

9. **Dorofeev S. N., Shichiyach R. A., Khasimova L. N.** Devoloping creative activity abilities of students in higer educaitional esteblishments. *Revista on line de politica e gistao educational.* 2021, vol. 25, no S2, pp. 883–900.

10. **Friedman L. M.** *Psikhologo-pedagogicheskiye osnovy prepodavaniya matematiki v shkole: uchitel' matematiki o pedagogicheskoy psikhologii* [Psychological and pedagogical foundations of teaching mathematics at school: a teacher of mathematics about pedagogical psychology]. M.: Enlightenment, 1983, 160 p. (In Russ.)

11. **Talyzina N. F.** Formation of mathematical concepts. *Formirovaniye metodov matematicheskogo myshleniya* [Formation of methods of mathematical thinking] / edited by N. F. Talyzina. M.: Lomonosov Moscow State University; Ventana-Graf LLP, 1995, pp. 13–28. (In Russ.)

12. **Ball G. A.** *Teoriya obrazovatel'nykh problem: psikhologo-pedagogicheskiy aspekt* [Theory of educational problems: psychological and pedagogical aspect]. M.: Pedagogy, 1990. 184 p. (In Russ.)

13. **Zak A. Z.** *Kak opredelit' uroven' razvitiya myshleniya studenta* [How to determine the level of development of a student's thinking]. M.: Knowledge, 1982. 96 p. (In Russ.)

14. **Vygotsky L. S.** *Sobraniye sochineniy : v 6 t. T. 2. Problemy obshchey psikhologii / pod red. V. V. Davydova* [Collected works: in 6 vols. 2. Problems of general psychology / edited by V. V. Davydov]. M.: Pedagogika, 1982. 504 p.: ill. (In Russ.)

15. **Dorofeev S. N.** *Vysshaya matematika* [Higher Mathematics]. M.: LLC "Publishing House "Mir i obrazovanie"", 2011. 592 p.: ill. (In Russ.)

16. **Kudryavtsev L. D.** *Mysli o sovremennoy matematike i yeye izuchenii* [Thoughts on modern mathematics and its study]. M.: Science, 1977, 123 p. (In Russ.)

17. **Vygotsky L. S.** *Sobraniye sochineniy : v 6 t. T. 3. Problemy razvitiya psikhiki / pod red. A. M. Matyushkina* [Collected works: in 6 vols. 3. problems of the development of the psyche / edited by A. M. Matyushkin]. M.: Pedagogy, 1983. 368 p.: ill. (In Russ.)

Сведения об авторах / Information about authors
Дорофеев Сергей Николаевич / Sergej N. Dorofeev
д. пед. н., профессор кафедры «Высшая математика и математическое образование»/ Doctor of Pedagogical Sciences, Professor of the Department of Higher Mathematics and Mathematical Education
Тольяттинский государственный университет / Togliatti State University
Россия, 445020, Самарская область, г. Тольятти, ул. Белорусская, д. 14 / 445020, Russia, Samara region, Togliatti, Belorusskaya str., 14

Наземнова Наталия Владимировна / Natalija V. Nazemnova
к. пед. н., старший преподаватель кафедры «Высшая математика»/ Candidate of Pedagogical Sciences, Senior Lecturer of the Department of Higher Mathematics
Пензенский государственный университет / Penza State University
440020, Россия, Пензенская область, г. Пенза, ул. Красная, д. 40 / 440020, Russia, Penza region, Penza, Krasnaya str., 40

Статья поступила в редакцию / The article was submitted 14.03.2022
Одобрено после рецензирования / Approved after reviewing 29.03.2022
Принято к публикации / Accepted for publication 30.03.2022

Вестник Сыктывкарского университета.
Серия 1: Математика. Механика. Информатика. 2023.
Выпуск 1 (46)
Bulletin of Syktyvkar University.
Series 1: Mathematics. Mechanics. Informatics. 2023; 1 (46)

Научная статья

УДК 004.42
https://doi.org/10.34130/1992-2752_2023_1_78

О РАБОТАХ ТРЁХ ДОВОЕННЫХ МАТЕМАТИКОВ ИЗ АЛМА-АТЫ, МОСКВЫ И ЛЕНИНГРАДА

Владимир Петрович Одинец
W.P.Odyniec@mail.ru

Аннотация. В работе рассмотрены труды трёх математиков: И. Акбергенова, специалиста в области интегральных уравнений Фредгольма, ученика профессора Л. В. Канторовича, С. Е. Аршона, специалиста в области комбинаторики и теории функций, и профессора Б. И. Извекова, специалиста в области обучения математике, погибших в 1938–1942 гг., и проживавших, соответственно, в Алма-Ате, Москве и Ленинграде.

Ключевые слова: интегральное уравнение, уравнение Фредгольма 2-го рода, правило Саррюса, комбинаторика, ассиметричная последовательность, векторный анализ

Для цитирования: Одинец В. П. О работах трёх довоенных математиков из Алма-Аты, Москвы и Ленинграда // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика.* 2023. Вып. 1 (46). С. 78–90. https://doi.org/10.34130/1992-2752_2023_1_78

© Одинец В. П., 2023.

Article

On the works of three prewar mathematicians from Alma-Ata, Moscow, and Leningrad

Vladimir P. Odinets
W.P.Odyniec@mail.ru

Abstract. The article considers the works of three mathematicians: I. Akbergenov, specialist in Fredholm integral equations, a student of Professor L. Kantorovich, S. Arshon, specialist in combinatorics and function theory and Professor B. Izvekov, in the field of teaching higher mathematics, who lived accordingly, in Alma-Ata, Moscow and Leningrad and perished in 1938–1942.

Keywords: integral equations, Fredholm equation of second kind, Sarrus rule, combinatorial analysis, asymmetric sequence, vector analysis

For citation: Odinets V. P. On the works of three prewar mathematicians from Alma-Ata, Moscow, and Leningrad. *Vestnik Syktyvkarskogo universiteta. Seriya 1: Matematika. Mekhanika. Informatika* [Bulletin of Syktyvkar University, Series 1: Mathematics. Mechanics. Informatics], 2023, no 1 (46), pp. 78−90. https://doi.org/10.34130/1992-2752_2023_1_78

1. Акбергенов Ибадулла родился в 1907 г. в Сузакском районе Туркестанской области[1]. В 1931 г. окончил физико-математический факультет Ташкентского Средне-Азиатского университета. В 1932 г. он поступает в аспирантуру того же университета [1]. В конце 1933 г. – начале 1934 г. И. Акбергенов приезжает в Ленинград для продолжения учебы. Его научным руководителем становится самый молодой профессор в СССР, Леонид Витальевич Канторович (1912–1987) [2], будущий Нобелевский и Ленинский лауреат.

В июне того же года И. Акбергенов уже участвует в работе 2-го Всесоюзного Математического съезда, где на секции «Приближенные вычисления» делает сообщение [3], посвященное оценке погрешности

[1]В советское время до 1936 г. – Южно-Казахская обл, после – до 2018 г. – Южно-Казахстанская обл.

Акбергенов Ибадулла

приближенного решения уравнения Фредгольма[2] 2-го рода по методу Е. Нистрёма[3]. Как заметил И. Акбергенов, в статье Е. Нистрёма (Acta Matematics, Т. 54 , (1930), p. 189) даётся приближенный способ решения интегрального уравнения Фредгольма второго рода, но без оценки точности решения.

Напомним, что интегральным уравнением Фредгольма 2-го рода называется интегральное уравнение вида

$$\varphi(x) = f(x) + \lambda \int\limits_a^b K(x,y)\varphi(y)dy, \ (a < b).$$

В 1935 г. в журнале «Математический сборник» вышла статья И. Акбергенова [4], послужившая основой защищенной в том же году кандидатской диссертации, на тему «Способы решения интегральных

[2]Э. И. Фредгольм (Erik Fredholm: 1866–1927) – шведский математик, профессор Стокгольмского университета (с 1906 г.); защитил в 1898 г. докторскую диссертацию под руководством М. Миттаг-Лефлера (1846–1927). Э. Фредгольм ввёл и изучал класс интегральных уравнений, названных его именем.

[3]Е. Нистрём (Nystroem Evert Johannes: 1895–1960) – финский математик, окончил Хельсинский университет в 1921 г., защитил докторскую диссертацию в 1926 г. под руководством Э. Линделёфа (1870–1946). С 1844 г. профессор прикладной математики. Основные труды у Е. Нистрёма посвящены численному анализу.

уравнений» [2]. Тем самым Ибадулла Акбергенов стал первым кандидатом физико-математических наук Казахстана.

В статье, кроме предисловия, было четыре параграфа. В предисловии анализировались существующие методы решения, в том числе способ Джона фон Неймана (1903–1957)[4]. Другим методом решения стал метод продолжения, разработанный Л. В. Канторовичем. Еще одним методом решения стал метод Е. Нистрёма, заключающийся в замене исходного уравнения конечной системой линейных алгебраических уравнений. Наконец, одним из наиболее эффективных способов приближенного решения интегральных уравнений стал способ замены ядра $K(x, y)$ на вырожденное ядро, в частности, такой способ предложил Г. Бейтмен[5].

В первых трёх параграфах статьи представлено решение линейных интегральных уравнений Фредгольма путем замены ядра на близкое и исходя из решений приближенного уравнения доказывается единственность и существование решения данного интегрального уравнения. Кроме того, дается оценка погрешности решения. Параграф 4 посвящен определению области расположения собственных значений, т. е. собственных значений данного уравнения, внутри некоторого круга, с определенной точностью.

В 1936 г. с 1-го сентября Ибадуллу Акбергенова пригласили в Казахский государственный университет, открытый в январе 1934 г. в г. Алма-Ата, на должность заведующего кафедрой математического анализа [2]. Вскоре он получает звание доцента, имея стаж научной работы 6 лет, а общий педагогический стаж – 7 лет [2].

Перед переходом в Казахский государственный университет И. Акбергенов сдаёт в печать книгу [6]. Она вышла уже в 1937 г. Книга [6] существенно дополняет результаты статьи [4].

---

[4]Дж. фон Нейман (John von Neuman: 1903–1967) – американо-венгерский математик и физик, один из основателей современных компьютеров (архитектура фон Неймана), один из создателей теории операторов в квантовой механике (алгебра фон Неймана), участник американского атомного проекта. Дж. (по венгерски – Янош) фон Нейман родился в состоятельной еврейской семье в Будапеште. В 1826 г. получил степень доктора философии по математике. С этого же года стал приват-доцентом Берлинского университета. В 1930 г. был приглашен в Принстон и остался в США.

[5]Гарри Бейтмен (Harry Bateman: 1882–1946) – англо-американский математик, специалист по решению дифференциальных уравнений в математической физике, профессор Калифорнийского технологического института.

В Казахском государственном университете И. Абергенов за активную работу был отмечен благодарностями [2].

Тем не менее, 14 марта 1938 г. И. Акбергенов был арестован и приговорён к расстрелу. Приговор приведен в исполнение 11 ноября 1938 г. 29 августа 1957 г. Ибадулла Акбергенов был полностью реабилитирован [2].

2. Аршон Соломон Ефимович, родился в Тобольске в 1892 г. [7]. Вероятно, закончил в Томске физико-математический факультет университета. Пока мы не знаем, чем С. Е. Аршон занимался до 1934 г. Однако с 1934 г. знаем, что он уже работает в Москве.

В июне 1934 г. С. Е. Аршон участвует в работе 2-го Всесоюзного Математического съезда в Ленинграде, будучи зарегистрирован от Москвы. Здесь на секции «Теория чисел и алгебра» он делает сообщение «Об одном методе комбинаторного анализа» [8]. В начале С. Е. Аршон замечает, что комбинаторика до сих пор не имеет оперативного математического аппарата. В своём сообщении он делает попытку введения такого аппарата, введя операции комбинаторного умножения и комбинаторного дифференцирования. В качестве примера применения этого аппарата рассматривается задача Эйлера об определении числа всех членов миноров $(n-1)$-го, $(n-2)$-го и т. д. порядков квадратной матрицы n-го порядка. Другим примером служит классическая задача о числе расстановок на шахматной доске из n^2 клеток n небьющих друг друга слонов.

В 1935 г. в журнале «Математический сборник» появляется статья С. Е. Аршона «Обобщенное правило Саррюса»[6]. Пусть дана квадратная матрица n-го порядка и её детерминант. К каждой строке данной матрицы припишем справа в последовательном порядке $(n-1)$ элементов той же строки, начиная с первого. Путем всех возможных перестановок строк данной матрицы получим $n!$ матриц (семейство матриц), члены которых будут и членами исходной матрицы. Семейство матриц содержит $(2n)n!$ членов. Задача сводится к тому, чтобы

[6]Пьер Фредерик Саррюс (Pier-Frederic Sarrus: 1798–1881) – французский математик; с 1826 г. преподавал в Страсбургском университете, там же профессор с 1829 г., в 1839–1852 гг. — декан. В 1843 г . в работе по вариационному исчислению дал мнемоническое правило вычисления определителя квадратной матрицы третьего порядка.

1) доказать, что среди членов семейства встретим все $n!$ членов исходного детерминанта;

2) выявить, какие именно из матриц семейства необходимы и достаточны для получения всех членов детерминанта;

3) установить критерий для определения знака у членов матрицы.

Кроме того, на основе обобщения правила Саррюса получена возможность механизации подсчета детерминанта n-го порядка.

В 1936 г. в журнале «Математическое просвещение» вышла статья[7] С. Е. Аршона «Некоторые свойства арифметических пропорций» [10], в которой рассмотрены четверки чисел, соответствующие проекции равнобедренной трапеции на числовую ось, где основания трапеции параллельны оси. При этом две крайние вершины считаются выше оси, а две оставшиеся – ниже. Вводятся также понятия обратной трапеции, звена четверок и порядок звеньев. Доказывается теорема о свойствах звеньев. В качестве примера применения находятся неограниченное количество различных целых решений для некоторого класса неопределенных уравнений при условии, что известно одно решение.

В 1937 г. вышла последняя статья С. Е. Аршона «Доказательство существования бесконечной n-значной ассимметрической последовательности» [11]. В этой статье указано, что С. Е. Аршон – сотрудник Математического института АН СССР. Начинается статья с определения: под *n-значной последовательностью* понимается последовательность знаков, среди которых только n различных. Далее, i каких-либо последовательных знаков данной n-значной последовательности образуют её *часть*. Если в последовательности существует часть из px знаков ($x = 1, 2, 3, \dots$), которая может быть разбита на p частей, одинаковых как по знакам, так и порядку их следования, то такую часть (из px знаков) будем называть *p-кратным повторением*, а о последовательности будем говорить, что она содержит повторение. Наконец, n-значную последовательность, не содержащую повторений, будем называть *ассиметричной*. Статья посвящена доказательству существования n-значных бесконечных ассиметричных последовательностей.

[7]Эта статья не вошла в список статей С. Е. Аршона в книге [5].

Проблема существования бесконечной n-значной ассимметричной последовательности была поставлена А. Я. Хинчиным[8] в январе 1933 г.

В конце 1938 г. С. Е. Аршон был арестован и приговорен к расстрелу [7]. Исполнен приговор был в 1939 г. Точная дата неизвестна. Перед арестом С. Е. Аршон был главным редактором издательства технико-теоретической литературы при АН СССР [12].

3. Извеков Борис Иванович родился в 1891 г. в Калуге в семье священника, происходившего из дворян Калужской губернии. В 1909 г. Б. И. Извеков окончил гимназию в Калуге с золотой медалью и в том же году поступил в Петербургский университет на физико-математический факультет.

Извеков Борис Иванович

В 1914 г. окончил учебу в университете с дипломом 1-й степени и был призван на военную службу вольноопределяющимся. Воевал под Ригой и Двинском (ныне Даугавпилс). Февральскую революцию 1917 г. встретил в должности заведующего автомобильным парком 59-го тяжёлого

[8]Алексадр Яковлевич Хинчин (1894–1959), профессор МГУ. Чл.-корреспондент АН СССР (1939), один из крупнейших специалистов в теории чисел и теории вероятностей. А. Я. Хинчин был одним из основателей Академии наук РСФСР.

артиллерийского дивизиона. Пользуясь уважением сослуживцев, был выбран командиром того же подразделения. Демобилизовался в феврале 1918 г., а осенью того же года поступил на службу в Павловскую магнито-метеорологическую обсерваторию. Через год (1919) вступил в должность приватного преподавателя авиашколы летчиков Красного воздушного флота. Чуть позже был призван на службу в Красную армию [13].

В 1920 г. был демобилизован и перешел на службу в Главную геофизическую обсерваторию (ГГО) в Петрограде. С 1922 г. состоял штатным преподавателем по математике Высшего военно-морского училища, совмещая со службой в ГГО. В тот же год ему было присвоено звание доцента, и он становится действительным членом Петроградского (с 1924 г. Ленинградского) физико-математического общества. В 1925 г. на заседании этого общества памяти скончавшегося А. А. Фридмана[9] он делает доклад «Работы А. А. Фридмана по теоретической метеорологии». Жил Б. И. Извеков тогда на 11-й линии Васильевского острова, д. 44, кв.14.

В 1923 г. Б. И. Извеков назначается ученым секретарем ГГО и избирается старшим физиком ГГО.

В 1925 г. Б. И. Извеков в течение трех месяцев находился в командировке по линии ГГО в Норвегии. В тот же год он сдаёт в печать книгу «Состав и строение земной атмосферы по современным понятиям» (Изд-во ГГО, (1926)). В 1928 г. Б. И. Извеков в течение двух месяцев был в командировке в Германии во Франкфурте-на-Майне, где познакомился со своей будущей женой Ольгой Алексеевной Косаревой, находившейся в Германии также в командировке. О. А. Косарева, будучи на 2 года старше Бориса Ивановича, мать уже троих детей, не сразу согласилась связать себя узами нового брака. Но в 1929 г. у неё уже родилась дочь Татьяна [13]. Борису Ивановичу теперь надо было думать, как кормить четырех детей. По совместительству он работает уже 5 лет доцентом в ЛЭТИ, преподавая математику [14]. Позже добавилась Военно-морская академия им. К. Е. Ворошилова. Здесь он даже стал

[9]Александр Александрович Фридман (1988–1925) – советский математик, механик и физик, профессор (1918), окончил Петербургский университет (1910); в 1915–1917 – преподаватель военной школы летчиков, 1918–1920 – профессор механики Пермского университета. Еще гимназистом, а позже студентом 1-го курса университета вместе с Я. Д. Тумаркиным опубликовал статьи по теории чисел. В 1922 и 1924 гг. опубликовал ставшие знаменитыми статьи по космологии. В сентябре 1925 г. умер от тифа, возвращаясь из Крыма.

заведующим кафедрой высшей математики и получил звание профессора.

В 1934 г. у него выходит книга «Основы векторного анализа» [15], а менее чем через год — книга «Сборник задач по прикладной математике для студентов, аспирантов и преподавателей втузов» [16]. Книга состояла из двух частей:

1) Математический анализ. Задачи и упражнения;

2) Прикладная математика. Задачи и упражнения.

Эти книги несомненно повлияли на приглашение Б. И. Извекова в 1939 г. преподавать на математико-механическом (мат.-мех) факультете ЛГУ.

Возможно, другим поводом послужила закрытая тематика мат.-меха: повышение точности артиллерийской стрельбы в зависимости от метеоусловий: скорость и направление ветра, температура и давление воздуха.

После ухода из Военно-морской академии в 1940 г. единственным местом работы профессора Б. И. Извекова оставался мат.-мех ЛГУ.

4 февраля 1942 г. Борис Иванович был арестован по делу № 555 — делу якобы контрреволюционной группы. Жена Бориса Ивановича была арестована и сослана в Саратов еще во времена «ежовщины» [13]. И хотя расстрельный приговор Борису Ивановичу Извекову был в мае 1942 г. заменён на 10-летнюю ссылку в Республику Коми, он умер в тюремной больнице 22 июня 1942 г. от дистрофии. Реабилитация пришла в 1954 г. На доме 44, по 11-й линии Васильевского острова, где жил Б. И. Извеков, в 2015 г. установлен памятный знак «Последний адрес».

Список источников

1. Акбергенов Ибадулла // *Национальная энциклопедия.* Алматы: Казак энциклопедиясы, 2004. Т. 1. С. 13.

2. **Ахметжанова А. Т.** Судьба ученого – последствие имперской политики советского государства // *Вестник КазНУ.* Алматы, 2012. С. 7–21.

3. **Акбергенов И. А.** Об оценке погрешности приближенного решения интегрального уравнения Фредгольма второго рода по способу

E. Nistrom'a // *Труды 2-го Всесоюзного математического съезда.* 1934. Т. 2 (Секционные доклады). 1935. С. 386–387.

4. **Акбергенов И. А.** О приближенном решении интегрального уравнения Фредгольма и об определении его собственных значений // *Математический сборник.* 1935. Т. 42. № 6. С. 679–698.

5. Математика в СССР за сорок лет 1917–1947. Т. 2. Биобиблиография. М.: Физматлит, 1959. 819 с.

6. **Акбергенов И. А.** О приближенном решении интегрального уравнения Фредгольма и об определении его собственных значений // Труды Средне-Азиатского университета. Ташкент: Математика (V), 16, 1937. С. 1–49.

7. Аршон С. Е. Жертвы политического террора в СССР // Архивное дело: П-48248.

8. **Аршон С. Е.** Об одном методе в комбинаторном анализе // *Труды 2-го Всесоюзного математического съезда.* Л., 1934. Т. 2 (Секционные доклады). Л.: Изд-во АН СССР, 1935. С. 24–26.

9. **Аршон С. Е.** Обобщение правила Саррюса // *Матем. сб.* 42, 1935. С. 121–128.

10. **Аршон С. Е.** Некоторые свойства арифметических пропорций // *Матем. просв.* 1936. Вып. 5. С. 24–28.

11. **Аршон С. Е.** Доказательство существования n-значных бесконечных ассимметричных последовательностей // *Матем. сб.* 44, 1937. № 4. С. 769–779.

12. **Кирсанов В. С.** Уничтоженные книги: эхо сталинского террора в советской истории науки // *Семь искусств.* 05.01.(2015). № 12. С. 13–19.

13. **Булах-Извекова Т. Б.** Воспоминания моей жизни. СПб., 2008. (Возвращение). 173 с.; 2009 (Продолжение). 114 с.; 2010 (Эпилог). 120 с.

14. Наука и научные работники в СССР. Ч. V. Научные работники Ленинграда. Л.: Изд-во АН СССР, 1934. 746 с.

15. **Извеков Б. И.** Основы векторного анализа. Л.: Кубуч, 1934. 176 с.

16. **Извеков Б. И.** Сборник задач по прикладной математике для студентов, аспирантов и преподавателей втузов. Л., М.: Гос. технико-теоретическое изд-во, 1935. Ч. 1. 407 с.

References

1. Akbergenov Ibadulla. *The National encyclopaedia.* Almaty. Kazak encyclopedijasy, 2004. Vol. 1, p. 13. (In Russ.)

2. **Ahmetzhanova A. T.** The fate of the academic – consequence of imperial policy of Soviet State. *Vestnik KazNU* [KazNU Bulletin]. Almaty, 2012, pp. 7–21. (In Russ.)

3. **Akbergtnov I. A.** On the estimation of the mistake of the approximate solution of the Fredholm integral equation of a second kind by E. Nistrom method. Leningrad. *Trudy 2-go Vsesouznogo Matematicheskogo s'ezda (1934). T. 2. Sekcionnye doklady* [Proceedings of the 2nd All-Union Mathematical Congress. 1934. Vol. 2 (Sectional reports)], 1935, pp. 386–387. (In Russ.)

4. **Akbergenov I. A.** Upon an approximate solution of the Fredholm integral equations and the determination of its eigenvalues. *Mat. sb.* [Mathematical collection], 1935, vol. 42. no 6, pp. 679–698. (In Russ.)

5. *Matematika v SSSR za sorok let 1917–1947. T. 2. Biobibliograph* [Mathematics in the U.S.S.R. after forty years 1917–1957. Vol. 2. Biobibliography]. M.: Fizmatlit, 1959, 819 p. (In Russ.)

6. **Akbergenov I. A.** Upon an approximate solution of the Fredholm Integral equations and the determination of its eigenvalues. Tashkent: Izd-vo Sredne-Aziatskogo universiteta, T.16, 1937, 48 p. (In Russ.)

7. Arshon S. E. Victims of political terror in U.S.S.R. *Arhivnoe delo: P- 48248* [Archival file: P-48248]. (In Russ.)

8. **Arshon S.E.** Upon a method of combinatorial analysis. *Trudy 2-go Vsesouznogo Mathematicheskogo s'ezda, 1934. T. 2. Sekcionnye doklady* [Proceedings of the 2nd All-Union Mathematical Congress.

1934. Vol. 2 (Sectional reports)]. L.: Izd-vo AN U.S.S.R., 1935, pp. 24–26. (In Russ.)

9. **Arshon S. E.** A generalization of the Sarrus rule. *Mat. sb.* [Mathematical collection], 1935, vol. 42, no 1, pp. 121–128. (In Russ.)

10. **Arshon S. E.** A property of the arithmetic proportion. *Mat. prosv.* [Mathematical education], 1936, no 5, pp. 24–28. (In Russ.)

11. **Arshon S. E.** A proof of the existence of n-valued infinite asymmetrical sequence. *Mat. sbornik* [Mathematical collection], 1937, vol. 44, no 4, pp. 769–779. (In Russ.)

12. **Kirsanov V. S.** The books Destroyed: an echo by the Stalin's terror in Soviet historical science. *Sem' iskustv* [Seven arts], no 12. 05.01.(2015), pp. 21–34. (In Russ.)

13. **Balach-Izvekova T. B.** *Vospominaniya moyey zhizni* [Memories of my life]. SPb.,2008 (Return). 171 p.; 2009 (Continuation). 114 p.; 2010 (Epilogue). 120 p. (In Russ.)

14. *Nauka i nauchnyye rabotniki v SSSR. Ch. V. Nauchnyye rabotniki Leningrada* [Science and the scientific workers in the USSR. Part V. Scientific workers of Leningrad]. L.: Izd-vo AN U.S.S.R., 1934. 746 p. (In Russ.)

15. **Izvekov B. I.** *Osnovy vektornogo analiza* [A basis of vector analysis]. L.: Izd-vo Kubuch, 1934, 176 p. (In Russ.)

16. **Izvekov B. I.**. *Sbornik zadach po prikladnoy matematike dlya studentov, aspirantov i prepodavateley vtuzov* [A collection of problems in applied mathematics for students, past-graduate students and the instructors of higher technical education]. M., L.: Gos.techniko-teoreticheskoe izd-vo, 1935. Part 1. 218 p. (In Russ.)

Сведения об авторе / Information about author
Одинец Владимир Петрович / Vladimir P. Odinets
д.ф.-м.н., профессор / Doctor of in Physics and Mathematics, Professor

Статья поступила в редакцию / The article was submitted 11.01.2022
Одобрено после рецензирования / Approved after reviewing 18.01.2022
Принято к публикации / Accepted for publication 23.01.2022

Contents

Mathematics

Sozontova E. A. *On new cases of solvability of the Goursat problem in quadratures for one hyperbolic type system* 4

Computer sciences

Babikova N. N. *Using NumPy to vectorization of Python code* 14

Golchevskiy Yu. V., Ushakov D. A. *Cryptographic Calculations Acceleration by Low-Level Optimization of Basic Blocks* 30

Methodical materials

Deynega S. A. *Components of Geometric-graphic Competence, Formed in the Study of Descriptive Geometry at a Technical University* . . 50

Dorofeev S. N., Nazemnova N. V. *Numerical sequences as a fundamental factor in the formation of creative activity in future bachelors* 64

Odinets V. P. *On the works of three prewar mathematicians from Alma-Ata, Moscow, and Leningrad* 78

Научное периодическое издание

Вестник Сыктывкарского университета
Серия 1: Математика. Механика. Информатика
Выпуск 1 (46) 2023

Гл. редактор О. А. Сотникова
Отв. редактор А. В. Ермоленко

Редактор Е. М. Насирова
Компьютерный макет Е. Н. Старцева
Корректор Л. Н. Руденко

Подписано в печать 28.04.2023. Дата выхода в свет 15.05.2023.
Формат $70 \times 100\frac{1}{16}$. Бумага офсетная.
Гарнитура Computer Modern.
Печать ризографическая. Усл. печ. л. 10,5.
Тираж 32 экз. Заказ № 27.

Отпечатано в соответствии с предоставленными
материалами в ООО «Коми республиканская типография»
167982, Республика Коми, г. Сыктывкар, ул. Савина, 81
Тел. 8(8212)-28-46-60
E-mail: ceo@komitip.ru
Сайт: komitip.ru