

ISSN 2222-8772

**М**АТЕМАТИЧЕСКИЕ  
**С**ТРУКТУРЫ  
И  
**М**ОДЕЛИРОВАНИЕ

№ 3(47)  
2018



**МИНИСТЕРСТВО НАУКИ  
И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«ОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИМ. Ф.М. ДОСТОЕВСКОГО»**

**МАТЕМАТИЧЕСКИЕ  
СТРУКТУРЫ  
И  
МОДЕЛИРОВАНИЕ**

**№ 3(47)**

Омск  
2018

**Математические структуры и моделирование.** — Омск : Омский государственный университет, 2018. — № 3(47). — 135 с.

ISSN 2222-8772 (print)

ISSN 2222-8799 (online)

#### **Редакционная коллегия**

- А. К. Гуц** — главный редактор, председатель редакционной коллегии, доктор физ.-мат. наук, профессор, зав. кафедрой кибернетики, Омский государственный университет им. Ф.М. Достоевского.
- Д. Н. Лавров** — ответственный за выпуск редактор, зам. глав. редактора, канд. техн. наук, доцент, зав. каф. компьютерных технологий и сетей, Омский государственный университет им. Ф.М. Достоевского.
- Н. Ф. Богаченко** — технический редактор, зам. глав. редактора, канд. физ.-мат. наук, доцент, Омский государственный университет им. Ф.М. Достоевского.
- С. В. Белим** — доктор физ.-мат. наук, профессор, зав. кафедрой информационной безопасности, Омский государственный университет им. Ф.М. Достоевского.
- В. П. Голубятников** — доктор физ.-мат. наук, профессор Новосибирского государственного университета, главный научный сотрудник Института математики СО РАН, г. Новосибирск.
- С. И. Горлов** — доктор физ.-мат. наук, профессор, ректор Нижневарттовского государственного университета.
- А. Г. Гринь** — доктор физ.-мат. наук, профессор, кафедра кибернетики, Омский государственный университет им. Ф.М. Достоевского.
- В. А. Ерошенко** — доктор физ.-мат. наук, профессор, зав. кафедрой общей математики и информатики Белорусского государственного университета, г. Минск, Республика Беларусь.
- V. Zilber** — Dr.Sc. (Phys.-Math.), Professor of Mathematical Logic, Mathematical Institute, University of Oxford, UK.
- А. Н. Кабанов** — канд. физ.-мат. наук, кафедра кибернетики, Омский государственный университет им. Ф.М. Достоевского.
- А. В. Копыльцов** — доктор техн. наук, профессор, кафедра информационных систем, Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина).
- А. Г. Коробейников** — доктор техн. наук, профессор, зам. директора по науке Санкт-Петербургского филиала Института земного магнетизма, ионосферы и распространения радиоволн им. Н.В. Пушкова РАН.
- П. А. Корчагин** — доктор техн. наук, профессор, проректор по научной работе, Сибирская государственная автомобильно-дорожная академия (СибАДИ).
- V. Kreinovich** — Ph.D. (Phys.-Math.), Professor, Computer Science Department, University of Texas at El Paso, Texas, USA.
- В. А. Плетюхов** — доктор физ.-мат. наук, профессор кафедры общей и теоретической физики Брестского государственного университета им. А.С. Пушкина, Республика Беларусь.
- Л. Б. Соколинский** — доктор физ.-мат. наук, профессор, проректор по информатизации, зав. кафедрой системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет), г. Челябинск.
- A. A. Fedorenko** — Ph.D. (Phys.-Math.), Researcher (CR1) at the French National Centre of Scientific Research (CNRS) Laboratoire de Physique de l'ENS-Lyon, France.
- A. Jadczyk** — Ph.D., Professor, Researcher, Laboratoire de Physique, Universite de Toulouse III et CNRS, France.

#### **Учредитель**

Федеральное государственное бюджетное образовательное учреждение высшего образования «Омский государственный университет им. Ф. М. Достоевского».  
Свидетельство о регистрации средства массовой информации ПИ № ФС77-72200 от 15 января 2018 г. выдано Роскомнадзором.

#### **Адрес редакции, издателя и типографии**

644077, Омская обл., г. Омск,  
пр-т Мира, д. 55а.

**Дата выхода в свет:** 28.09.2018.

**Тираж** 100 экз.

Свободная цена.

## МАТЕМАТИЧЕСКИЕ СТРУКТУРЫ И МОДЕЛИРОВАНИЕ

Журнал основан в 1998 году. В журнале публикуются статьи, в которых излагаются результаты исследований по фундаментальной и прикладной математике, теоретической физике, компьютерным наукам, философии и истории математики и информатики, а также размышления, касающиеся окружающей нас природы и общества. Объекты исследования должны быть представлены в форме некоторых математических структур и моделей.

Все статьи журнала проходят обязательное рецензирование. Рефераты статей журнала опубликованы в «Реферативном журнале» и «Mathematical Reviews» (США). Журнал индексируется в РИНЦ (elibrary.ru) и «Zentralblatt für Mathematik» (Германия). Журнал входит в Перечень рецензируемых научных изданий ВАК РФ, в которых должны быть опубликованы основные результаты диссертаций на соискание учёных степеней (Приказ Минобрнауки России от 25 июля 2014 г. № 793).

Все статьи в журнале публикуются под лицензией Attribution 4.0 International (CC-BY).

Электронная версия журнала представлена в сети:

<http://msm.univer.omsk.su>  
<http://msm.omsk.su>

Подписной индекс по каталогу «Пресса России»: 94082

Электронная почта главного редактора:

[guts@omsk.su](mailto:guts@omsk.su)

Электронная почта выпускающего редактора:

[lavrov@omsk.su](mailto:lavrov@omsk.su)

# СОДЕРЖАНИЕ

## Фундаментальная математика и физика

- А.К. Гуц. *Квантовая машина времени и петлевая квантовая гравитация*..... 4  
А.К. Гуц. *Альтернативная новая космогония* .. 15  
А.Н. Кабанов. *Центральный ряд группы унитарных автоморфизмов нильпотентной алгебры Лейбница индекса 2* ..... 27

## Прикладная математика и моделирование

- П.И. Вайнмастер. *Решение актуальных задач обработки данных в системах микросейсмического мониторинга гидроразрыва пластов*... 31  
О.В. Опрышко. *Расчёт параметров специальных трёхмерных стационарных потоков* .... 45  
В.А. Шовин. *Автокорреляционная нейронная сеть факторного анализа* ..... 61  
Т.Ю. Назарова, Д.Н. Лавров. *Компьютерное моделирование идентификации личности по радужной оболочке глаза на основе OpenCV*... 68  
E. Rodriguez Velasquez et al. *Fuzzy Ideas Explain a Complex Heuristic Algorithm for Gauging Pavement Conditions*..... 82  
S. Ayala et al.. *Soft Computing Ideas Can Help Earthquake Geophysics*..... 91  
F. Zapata et al. *How to Select the Best Paper: Towards Justification (and Possible Enhancement) of Current Semi-Heuristic Procedures*..... 101

## Компьютерные науки

- Л.А. Володченкова, С.И. Королёв. *Программное приложение для обучения детей младших классов приёмам скорочтения*..... 107  
А.Д. Городецкий, Д.Н. Лавров. *Полуавтоматическое управление памятью приложения*... 121

## Информационная безопасность

- С.В. Белим, С.Ю. Белим. *Проблемы построения политики безопасности при объединении информационных систем* ..... 126

## КВАНТОВАЯ МАШИНА ВРЕМЕНИ И ПЕТЛЕВАЯ КВАНТОВАЯ ГРАВИТАЦИЯ

А.К. Гуц

профессор, д.ф.-м.н., e-mail: aguts@mail.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

**Аннотация.** Даётся описание квантовой машины времени, осуществляющей межвременные переходы, в рамках петлевой квантовой гравитации. В отличие от квантовой геометродинамики в данном случае становится понятным как производить воздействия, изменяющие современную геометрию на прошлую.

**Ключевые слова:** квантовая машина времени, петлевая квантовая гравитация, спиновая сеть, спиновая пена.

### Введение

Идея квантовой машины времени основывается на разрушении квантовой суперпозиции решений уравнения Уилера–ДеВитта [1, 2].

Разрушение суперпозиции происходит при особой процедуре измерения геометрии 3-пространства. Однако в рамках квантовой геометродинамики Уилера, с помощью которой обосновывалась работа квантовой машины времени, совершенно неясно, на что направлено это измерение, что конкретно измеряется.

Напротив, есть надежда, что использование теории петлевой квантовой гравитации [3–6] позволит установить искомую процедуру измерения, а точнее, определить, на что следует направить энергетическое воздействие, предполагаемое при процедуре измерения. Важно, что в этом случае нет необходимости существенно переформулировать механизм действия квантовой машины времени, поскольку петлевая квантовая гравитация, по сути дела, является переводом квантовой геометродинамики на язык другой пары канонических переменных: вместо пары  $(h_{\alpha\beta}, K_{\alpha\beta})$  рассматривается пара Аштекара  $(A_{\beta}^{(\alpha)}, \tilde{E}_{\beta}^{(\alpha)})$ .

### 1. Петлевая квантовая гравитация

Петлевая гравитация — это переформулировка гамильтоновой формулировки гравитации, основанной на АДМ-формализме, в новых канонических переменных Аштекаря. Петлевая гравитация появляется вместо геометродинамики Уилера.

Петлевая квантовая гравитация — каноническое квантование по схеме Дирака в новых операторах Аштекаря. Квантовая петлевая гравитация появляется

вместо квантовой геометродинамики Уилера; на место *геометродинамики* ставится *связнодинамика*.

Петлевая квантовая гравитация — альтернатива квантовой теории суперструн, но без новых понятий (подобно суперсимметрии) и без новых размерностей (остаётся традиционное 4-мерие).

### 1.1. Переменные Аштекара

Латинские индексы  $i, j, k, \dots = 0, 1, 2, 3$ , греческие  $\alpha, \beta, \gamma, \dots = 1, 2, 3$ .

Метрику пространства-времени  $M^4 = M^3 \times \mathbb{R}$  подвергаем (3+1)-разбиению:

$$g_{ik} = \begin{pmatrix} N^2 & N_\alpha \\ N_\alpha & h_{\alpha\beta} \end{pmatrix}.$$

Вводим триаду — три вектора  $e_\beta^{(\alpha)}$ , где  $(\alpha)$  — номер вектора, такие, что

$$h_{\alpha\beta} = \delta_{(\gamma\delta)} e_\alpha^{(\gamma)} e_\beta^{(\delta)}, \quad h^{\alpha\beta} = \delta^{(\gamma\delta)} e_{(\gamma)}^\alpha e_{(\delta)}^\beta.$$

Состояния в петлевой квантовой гравитации — это решения уравнения Уилера–ДеВитта, но записанные в новых переменных: поля  $\tilde{E}_\beta^{(\alpha)}$ ,  $A_\beta^{(\alpha)}$ .

Опишем эти поля. Поле

$$\tilde{E}_\beta^{(\alpha)} = \sqrt{h} e_\beta^{(\alpha)},$$

$$h = \det(h_{\alpha\beta}) = [\det(e_\beta^{(\alpha)})]^2 = e^2, \quad \det(\tilde{E}) = \sqrt{h} \cdot \det(e) = h,$$

$$\delta^{(\mu\nu)} \tilde{E}_{(\mu)}^\alpha \tilde{E}_{(\nu)}^\beta = h h^{\alpha\beta}.$$

Второе поле задаётся как

$$A_\beta^{(\alpha)} = \Gamma_\beta^{(\alpha)} + \gamma K_\beta^{(\alpha)},$$

где  $\gamma \neq 0 \in \mathbb{C}$  — параметр Иммирци и  $\Gamma_\beta^{(\alpha)}$  — так называемая спиновая связность, которая является решением структурного уравнения Картана

$$\partial_{[\beta} e_{\delta]}^{(\alpha)} + \varepsilon_{(\mu\nu)}^{(\alpha)} \Gamma_{[\beta}^{(\mu)} e_{\delta]}^{(\nu)} = 0$$

и имеет вид

$$\Gamma_\beta^{(\alpha)} = -\frac{1}{2} \varepsilon^{(\alpha\gamma)}_{(\kappa)} e_{(\gamma)}^\delta [\partial_{[\beta} e_{\delta]}^{(\kappa)} + \delta^{(\kappa\lambda)} \delta_{(\mu\sigma)} e_{(\lambda)}^\nu e_{\beta}^{(\mu)} \partial_\delta e_\nu^{(\sigma)}],$$

$$K_\beta^{(\alpha)} = \frac{1}{\sqrt{h}} K_{\beta\gamma} \tilde{E}_{(\mu)}^\gamma \delta^{(\alpha\mu)} = K_{\beta\gamma} e_{(\mu)}^\gamma \delta^{(\alpha\mu)},$$

где  $K_{\beta\gamma}$  — внешняя кривизна 3-пространства  $M^3$ .

Имеем гамильтонову связь [7]

$$H = \frac{G\gamma^2}{2\sqrt{h}} \varepsilon^{(\alpha\beta\gamma)} \tilde{E}_{(\alpha)}^\mu \tilde{E}_{(\beta)}^\nu \left[ F_{\mu\nu(\gamma)} - \frac{1 + \gamma^2}{\gamma^2} \varepsilon_{(\gamma\alpha\beta)} (A_\mu^{(\alpha)} - \Gamma_\mu^{(\alpha)}) (A_\nu^{(\beta)} - \Gamma_\nu^{(\beta)}) \right] = 0,$$

$$F_{\mu\nu}^{(\alpha)} = \partial_\mu A_\nu^{(\alpha)} - \partial_\nu A_\mu^{(\alpha)} + \varepsilon^{(\alpha\beta\gamma)} A_{\mu(\beta)} A_{\nu(\gamma)}.$$

## 1.2. Квантование по схеме Дирака и уравнение Уилера–ДеВитта

Квантование состоит в замене переменных  $(A_\beta^{(\alpha)}, \tilde{E}_\beta^{(\alpha)})$  на эрмитовы операторы

$$\begin{aligned} A_\mu^{(\alpha)} &\rightarrow \hat{A}_\mu^{(\alpha)} = A_\mu^{(\alpha)}, \\ \tilde{E}_{(\alpha)}^\mu &\rightarrow \hat{E}_{(\alpha)}^\mu = -8\pi\gamma G \frac{\delta}{\delta A_\mu^{(\alpha)}}, \\ \{\tilde{E}_{(\alpha)}^\mu(x), A_\nu^{(\beta)}(y)\} &= \gamma G \delta_\nu^\mu \delta_{(\alpha)}^{(\beta)} \delta(x, y), \end{aligned}$$

действующие на функционалы  $\Psi[A]$ .

При  $\gamma = i$

$$H \rightarrow \hat{H} = \varepsilon^{(\alpha\beta\gamma)} F_{\mu\nu(\gamma)} \frac{\delta}{\delta A_\mu^{(\alpha)}} \frac{\delta}{\delta A_\nu^{(\beta)}}.$$

Следовательно, имеем уравнение Уилера–ДеВитта

$$\varepsilon^{(\alpha\beta\gamma)} F_{\mu\nu(\gamma)} \frac{\delta}{\delta A_\mu^{(\alpha)}} \frac{\delta}{\delta A_\nu^{(\beta)}} \Psi[A] = 0. \quad (1)$$

Решением этого уравнения является волновая функция, максимум амплитуды которой даёт наиболее вероятностную связность, а точнее, наиболее вероятную 3-геометрию со связностью, и следовательно, показывает, как меняется вектор в 3-пространстве  $M^3$ , обносимый по петле.

При  $\gamma \neq i$  уравнение Уилера–ДеВитта имеет добавочный член. Отметим, что в настоящее время в петлевой теории гравитации предпочтительным является выбор вещественного параметра  $\gamma$  [7, p. 55].

## 1.3. Квантованность пространства: дискретность площади и объёма

Как известно, площадь 2-мерной поверхности  $S \subset M^3$  вычисляется по формуле:

$$\begin{aligned} A(S) &= \iint_S \sqrt{q} du^1 du^2 = \\ &= \iint_S \sqrt{\hbar} |n| du^1 du^2 = \iint_S \sqrt{n_\alpha h h^{\alpha\beta} n_\beta} du^1 du^2 = \iint_S \sqrt{n_\alpha \tilde{E}_{(\nu)}^\alpha n_\beta \tilde{E}^{(\nu)\beta}} du^1 du^2, \quad (2) \end{aligned}$$

где

$$q = \det||q_{ab}||, \quad q_{ab} = h_{\alpha\beta} \frac{\partial x^\alpha}{\partial u^a} \frac{\partial x^\beta}{\partial u^b}, \quad a, b = 1, 2$$

– индуцированная метрика поверхности  $S : x^\alpha = x^\alpha(u^1, u^2)$ ,  $u^1, u^2$  – координаты на  $S$ ,

$n = (n^\alpha)$  – нормаль к поверхности  $S$ ,  $|n|^2 = h^{\alpha\beta} n_\alpha n_\beta$ .

При квантовании мы в этой формуле меняем  $\tilde{E}_{(\nu)}^\alpha$  на оператор  $\hat{E}_{(\alpha)}^\mu$ . Следовательно, оператором становится площадь, т. е.

$$A(S) \rightarrow \hat{A}(S),$$

спектр которого, как показали исследования [10, 11], является дискретным, а собственные значения имеют вид

$$a(S) = 8\pi G\gamma\hbar c^{-3} \sum_i \sqrt{j_i(j_i + 1)}. \quad (3)$$

Аналогичный результат получается и для объёма  $V(B)$  области  $B \subset M^3$  [13, р. 33]:

$$V(B) = \iiint_B \sqrt{\hbar} d^3x = \iiint_B \sqrt{\frac{1}{6} |\varepsilon_{\alpha\beta\gamma} \tilde{E}_{(\mu)}^\alpha \tilde{E}_{(\nu)}^\beta \tilde{E}_{(\sigma)}^\gamma \varepsilon^{(\mu\nu\sigma)}|} d^3x.$$

а также для длины [14, р. 55].

#### 1.4. Голономия

Определяем в пространстве 1-форму связности  $A$

$$A(x) = A_{(\alpha)}^\beta(x) \tau_\beta dx^{(\alpha)}$$

со значениями в алгебре Ли  $SU(2)$ ,  $A_\beta^{(\alpha)}(x) \in C^\infty(M)$  и  $\tau_\alpha = (i/2)\sigma_\alpha$  — генераторы в  $SU(2)$  ( $\sigma_i$  — матрицы Паули).

Пусть  $\mathcal{A} = \{A\}$  — пространство гладких связностей в 3-пространстве  $M^3$ , и будем обозначать как  $\Psi(A)$  непрерывные функционалы на  $\mathcal{A}$ .

Голономия — это преобразование в касательном пространстве, которое показывает, как изменяется исходный вектор при параллельном переносе его вдоль кривой. Понятие параллельного переноса появляется благодаря рассмотрению некоторой связности  $A$ , вводимой для того, чтобы обеспечить ковариантность частных производных векторного поля

$$\partial_\mu \rightarrow \nabla_\mu = \partial_\mu + A_\mu.$$

Рассмотрим связность  $A_i^{(a)}$  и кривую  $\alpha : [0, 1] \rightarrow M^3$ . Параллельный перенос вектора  $V_\nu(0)$  вдоль  $\alpha$  из точки  $\alpha(0)$  в точку  $\alpha(s)$  — это преобразование вектора  $V_\nu(0)$

$$V_\nu(0) \rightarrow V_\nu(s) = U[A, \alpha](s) V_\nu(0), \quad U[A, \alpha](s) \in SO(3) \simeq SU(2).$$

Уравнение параллельного переноса вдоль кривой  $\alpha(s)$

$$\frac{d\alpha^\mu}{ds} D_\mu = \frac{d}{ds} + \frac{d\alpha^\mu}{ds} A_\mu(\alpha(s)),$$



$$\frac{d\alpha^\mu}{ds} D_\mu V_\nu(s) = \frac{dV_\nu}{ds} + \underbrace{\frac{d\alpha^\mu}{ds} A_\mu(\alpha(s))}_{A(\alpha(s))} V_\nu = 0.$$

Откуда

$$\frac{d}{ds} U[A, \alpha](s) V_\nu(0) + \underbrace{\frac{d\alpha^\mu}{ds} A_\mu(\alpha(s))}_{A(\alpha(s))} U[A, \alpha](s) V_\nu(0) = 0.$$

Имеем задачу Коши

$$\begin{cases} \frac{d}{ds} U[A, \alpha](s) + \underbrace{\frac{d\alpha^\mu}{ds} A_\mu(\alpha(s))}_{A(\alpha(s))} U[A, \alpha](s) = 0, \\ U[A, \alpha](0) = 1. \end{cases}$$

Интегрируя, получаем

$$U[A, \alpha](s) = 1 - \int_0^s ds_1 A(\alpha(s_1)) U[A, \alpha](s_1).$$

Решение этого уравнения ищем в форме итерации

$$U_{n+1}[A, \alpha](s) = 1 - \int_0^s ds_1 A(\alpha(s_1)) U_n[A, \alpha](s_1).$$

Имеем

$$\begin{aligned} U_0[A, \alpha](s) &= 1, \\ U_1[A, \alpha](s) &= 1 - \int_0^s ds_1 A(\alpha(s_1)), \\ U_2[A, \alpha](s) &= 1 - \int_0^s ds_1 A(\alpha(s_1)) \left(1 - \int_0^{s_1} ds_2 A(\alpha(s_2))\right) = \\ &= 1 - \int_0^s ds_1 A(\alpha(s_1)) + \int_0^s ds_1 A(\alpha(s_1)) \int_0^{s_1} ds_2 A(\alpha(s_2)). \\ &\dots\dots\dots \end{aligned}$$

В результате получаем решение для оператора  $U[A, \alpha](s)$  в виде ряда

$$U[A, \alpha](s) = \sum_{n=0}^{\infty} \int_0^s ds_1 \int_0^{s_1} ds_2 \dots \int_0^{s_{n-1}} ds_n A(\alpha(s_1)) A(\alpha(s_2)) \dots A(\alpha(s_n)),$$

который символически запишем как

$$U[A, \alpha](s) = \mathcal{P} \left( e^{-\int_0^s ds A(\alpha(s))} \right).$$

### 1.5. Петли Вильсона как решения уравнения Уилера–ДеВитта

Рассмотрим функционалы вида

$$\Psi_\alpha[A] = Tr U[A, \alpha](1) = Tr \mathcal{P} \left( e^{-\oint_\alpha ds A(\alpha(s))} \right),$$

которые в случае *петли*  $\alpha$ , т. е. при  $\alpha(0) = \alpha(1)$ , называются *петлями Вильсона* для связности Аштекара.

Если петля Вильсона является гладкой кривой без самопересечений, то она будет решением уравнения Уилера–ДеВитта (1) [5, 8]:

$$\hat{H}\Psi_\alpha[A] = 16\pi^2 G^2 \varepsilon^{(\alpha\beta\gamma)} F_{\mu\nu(\gamma)} \frac{\delta}{\delta A_\mu^{(\alpha)}} \frac{\delta}{\delta A_\nu^{(\alpha)}} \Psi_\alpha[A] = 0. \quad (4)$$

Действительно, когда производная берётся, то она даёт касательный вектор  $\dot{\alpha}^\mu = d\alpha^\mu/dt$  петли  $\alpha$ . Итак, у нас появится что-то вроде выражения

$$F_{\mu\nu(\gamma)} \dot{\alpha}^\mu \dot{\alpha}^\nu.$$

Однако, поскольку  $F_{\mu\nu(\gamma)}$  антисимметричен по индексам  $\mu, \nu$ , а  $\dot{\alpha}^\mu \dot{\alpha}^\nu$  – симметричен, то выражение (4) зануляется.

### 1.6. Кинетическое гильбертово пространство $\mathcal{H}_{kin}$ и физическое гильбертово пространство $\mathcal{H}_{phys}$

Вводим кинетическое гильбертово пространство  $\mathcal{H}_{kin}$  состояний петлевой квантовой гравитации, на котором действуют введённые выше операторы, состоящее из (цилиндрических) функций вида

$$\Psi[A] = f(U(A, s_1), \dots, U(A, s_p)),$$

где  $s_1, \dots, s_p$  — ребра графа  $\Gamma \subset M^3$ ,  $U(A, s_j)$  – параллельный перенос вдоль  $s_j$  относительно связности  $A$ :

$$U(A, s_j) = \mathcal{P} \left( e^{-\oint_{s_j} ds A(\alpha_j(s))} \right).$$

Скалярное произведение в  $\mathcal{H}_{kin}$  определяется как

$$\langle \Psi[A], \Phi[a] \rangle = \int_A \overline{\Psi[A]} \Phi[A] d\mu_{AL}[A],$$

где  $d\mu_{AL}[A]$  — мера Аштекара–Левандовского [9].

Физическое гильбертово пространство  $\mathcal{H}_{phys}$  состояний состоит из тех векторов пространства  $\mathcal{H}_{kin}$ , которые удовлетворяют уравнению Уилера–ДеВитта.

### 1.7. Спиновые сети и спиновая пена

Квантовые (дискретные) состояния объёма и площади изображают в виде графа следующим образом.

Область пространства — объём — изображают как точку (узел). Объём примыкающий к данному — также есть точка, а общую между ними часть поверхности изображают как отрезок. Например, куб изображается как точка, из которой исходят шесть отрезков, каждый из них изображает одну из граней куба (рис. 1). Рядом с точкой указывают величину объёма, а рядом с отрезками — величины площади соответствующих граней. Если объёмы — это любые комбинации различных многогранников, то объёмные полиэдры становятся точками или узлами, а плоские грани — отрезками, линиями, соединяющими узлы.

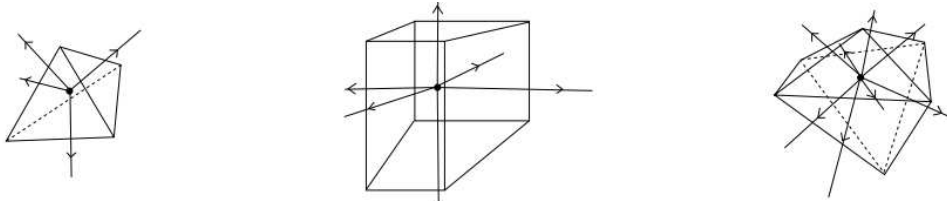


Рис. 1. Объёмы как узлы и отрезки как грани

Как видим, при таком подходе получается то, что математики называют *графом*. Этот граф представляет квантованные объёмы с квантованными площадями, общими для объёмов. Этот граф называют *спиновой сетью*.

Отрезки и точки, или иначе — линии и узлы, образующие граф, — это и есть 3-мерное пространство, геометрия которого определяется тем, как они соединяются.

Спектр оператора площади (ассоциированного с каждым ребром  $I$  графа) оказывается дискретным (associated to the link  $I$  is discrete):

$$A_I = 8\pi\gamma Gc^{-3}\hbar j_I(j_I + 1).$$

Равно дискретным является спектр оператора объёма, ассоциированного с каждым узлом графа.

При квантовом взгляде на геометрию пространства мы должны говорить о её квантовых состояниях, которыми являются спиновые сети. Каждой спиновой сети  $S_i$  приписывается амплитуда вероятности  $c_i$ . Поэтому квантовое состояние геометрии пространства есть суперпозиция спиновых сетей  $\sum_i c_i S_i$ .

Если взять спиновую сеть и рассмотреть её с течением времени, то линии спиновой сети расширяются и становятся двумерными поверхностями, а узлы растягиваются в линии. Это *спиновая пена* (рис. 2). Пространство-время — это спиновая пена. Поперечный срез спиновой пены представляет собой спиновую сеть.

Переходы (шаги), при которых происходит изменение спиновой сети, представляются узлами, в которых сходятся/выходят линии пены. На каждом шаге происходит изменение связности графа, представляющего спиновую сеть.

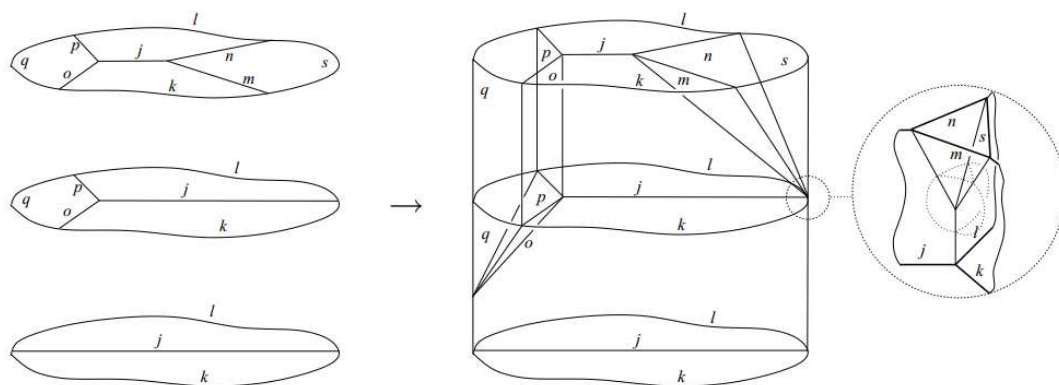


Рис. 2. Спиновая пена: переходы от одной спиновой сети к другой. Узлы превращаются в линии, а линии в грани [15]

Точные выражения для вычисления квантовой вероятности шагов спиновой сети вывел Томас Тиманн (Thomas Thiemann).

### 1.8. Пространство-время в LQG

«Пространство-время в петлевой квантовой гравитации есть квантовая суперпозиция спиновых пен. Спиновая пена является 2-мерный комплекс, срезы которого представляют собой спиновые сети (рис. 3).

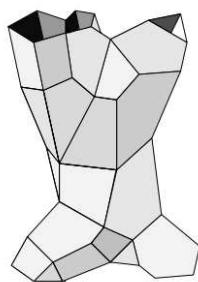


Рис. 3. Спиновая пена как 2-мерный комплекс [12]

Модель спиновой пены определяет, как рассчитать амплитуду вероятности для любой такой спиновой пены — обычно как произведение амплитуд вершин, амплитуды рёбер, амплитуд граней и т. д. Вопрос: можем ли мы найти модель спиновой пены, поведение которой в течение длительного масштаба времени, большего по сравнению с масштабами Планка, сводится к общей теории относительности?» [15]. Если это удастся, то с помощью теории петлевой квантовой гравитации можно будет описывать макроявления.

## 2. Квантовая машина времени

Волновой пакет 3-геометрий

$$\int_{\Omega} c_{\alpha} \Psi_{\alpha}[A] d\alpha, \quad c \in \mathbb{C}, \quad (5)$$

где  $\Omega = \{\alpha\}$  – некоторое множество петель, в результате его интерференции даёт пространство-время, в котором течёт время  $t$ .

В общей теории относительности геометрия пространства меняется со временем. Геометрия характеризуется кривизной в геометродинамике Уилера и связностью  $A$  в петлевой гравитации. Каждому моменту времени отвечает конкретная связность  $A_0$ , конкретная петля  $\alpha_0 \in \mathcal{A}_0$  со значением  $|\Psi_{\alpha_0}[A_0]|^2 = \max |\Psi_{\alpha_0}[A]|^2$ , т.е.

$$A_0 = \arg \max_A |\Psi_{\alpha_0}[A]|^2.$$

С учётом цилиндрических волновых функций логично пространство-время рассматривать как результат интерференции более общего волнового пакета вида

$$\int_K c_k \Psi_k[A] dk, \quad c_k \in \mathbb{C}. \quad (6)$$

Смена волновой функции  $\Psi_k[A]$  — это смена 3-геометрии. Каждая такая 3-геометрия характеризует ту или иную историческую эпоху [1, 2]. Скачкообразный переход из одной исторической эпохи в другую — это межвременной переход их одной эпохи в другую: из настоящего в прошлое, из прошлого в будущее и т. д.

Таким образом, чтобы волновой пакет (6) совершал коллапс в нужную историческую эпоху, в нужную геометрию  $A_0$

$$\int_K c_k \Psi_k[A] dk \rightarrow \Psi_{k_0}[A],$$

следует произвести соответствующие её измерения (так говорят об этом на языке квантовой механики), т. е. запустить машину времени, настроенную на геометрию  $A_0 = \arg \max_A |\Psi_{k_0}[A]|^2$ .

Данная машина времени — это физическое устройство, организующее требуемые энергетические воздействия на совокупность дискретных ячеек 3-пространства, т. е на совокупность узлов графа  $\Gamma$ , представляющую спиновую сеть нашей исторической эпохи, меняющую его связность  $A_{\text{наст.}}$  на связность прошлой эпохи  $A_{\text{прошл.}}$ .

## 3. Заключение

В описании сущности работы квантовой машины времени в случае петлевой квантовой гравитации нет ничего принципиально нового по сравнению с подобным описанием в рамках квантовой геометродинамики: всего лишь вместо

слова «3-метрика» мы говорим «связность». Зачем нам понадобилась петлевая квантовая гравитация (LQG)?

А дело в том, что в петлевой квантовой гравитации описывается структура 3-пространства — она на планковских расстояниях является пушистой! Пространство описывается как спиновая сеть, которая допускает перестройку в другую спиновую сеть при квантовых скачках времени.

Поэтому есть шанс найти способ локального энергетического воздействия, направленного в «точку» (конкретный объём пространства), итогом которого будет перестройка спиновой сети, а значит, смена связности  $A$ , скачок кривизны.

Следует отметить, что для описания квантовой машины времени мы вынуждены использовать либо квантовую геометродинамику, либо петлевую квантовую гравитацию, больше приспособленные для описания микроявлений, поскольку сегодня мы не имеем квантовой теории, которая равным образом успешно бы описывала как микро- так и макроявления. Но лучше иметь приблизительное математическое представление о работе интересующего нас аппарата, чем не иметь ничего.

## ЛИТЕРАТУРА

1. Гуц А.К. Не-гёделевская машина времени // Математические структуры и моделирование. 2016. № 3(31). С. 48–56.
2. Гуц А.К. Временные эффекты коллапса волнового пакета в суперпространстве Уилера // Международный научный семинар «Нелинейные модели в механике, статистике, теории поля и космологии» GRACOS-16. Лекции школы и материалы семинара (5 – 7 ноября 2016 г., Казань). Казань: Казанский (Приволжский) федеральный университет, 2016. С. 273–280.
3. Gambini R., Pullin J. A First Course in Loop Quantum Gravity. Oxford University Press Inc.: New York, 2011. 183 p.
4. Rovelli C., Gaul M. Loop Quantum Gravity and the Meaning of Diffeomorphism Invariance // Lecture Notes in Physics. 2000. No. 541. P. 277–324.
5. Rovelli C. The strange equation of quantum gravity. 2015. arXiv:1506.00927v1 [gr-qc].
6. Rovelli C. Loop Quantum Gravity. 1997. URL: <https://arxiv.org/abs/gr-qc/9710008v1> (дата обращения: 01.09.2018).
7. Casares P.A.M. An review on Loop Quantum Gravity. 2018. rXiv:1808.01252v1 [gr-qc]
8. Jacobson T., Smolin L. Nonperturbative Quantum Geometries // Nucl. Phys. 1988. V. B299. P. 295–345.
9. Ashtekar A., Lewandowski J. Projective techniques and functional integration // J. Math. Phys. 1995. V. 36. P. 2170.
10. Rovelli C., Smolin L. Discreteness of area and volume in quantum gravity // Nucl. Phys. 1995. B442. P. 593–622.
11. Rovelli C. et al. Discreteness of area and volume in quantum gravity. 1994. URL: <http://xxx.lanl.gov/abs/gr-qc/9411005> (дата обращения: 01.09.2018).
12. Perez A. Spin Foam Models for Quantum Gravity. 2003. URL: <https://arxiv.org/pdf/gr-qc/0301113v2.pdf> (дата обращения: 01.09.2018).

13. Perez A. Introduction to loop quantum gravity and spin foams. 2005. URL: <https://arxiv.org/pdf/gr-qc/0409061v3.pdf> (дата обращения: 01.09.2018).
14. Loop quantum gravity: the first 30 years / Eds: A. Ashtekar, J. Pullin. World Scientific Publishing Co. Pte. Ltd., 2017.
15. Baez J.C. Loop Quantum Gravity, Quantum Geometry and Spin Foams. Dublin, 2004.

## QUANTUM TIME MACHINE AND LOOP QUANTUM GRAVITY

**А.К. Guts**

Dr.Sc. (Phys.-Math.), Professor, e-mail: aguts@mail.ru

Dostoevsky Omsk State University, Omsk, Russia

**Abstract.** The description of a quantum time machine, implementing intertemporal transitions, within the framework of loop quantum gravity is given. In contrast to quantum geometrodynamics, in this case it becomes clear how to produce influences that change the modern geometry to the past one.

**Keywords:** quantum time machine, loop quantum gravity, spin net, spin foam.

*Дата поступления в редакцию: 09.09.2018*

## АЛЬТЕРНАТИВНАЯ НОВАЯ КОСМОГОНИЯ

А.К. Гуц

профессор, д.ф.-м.н., e-mail: aguts@mail.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

**Аннотация.** В статье указывается математический аппарат, посредством которого можно реализовать идеи Гегеля о бытии, об отчуждении идей в Природе, а также о теории рождения Вселенной посредством деятельного участия сознания, которая альтернативна теории Большого взрыва.

**Ключевые слова:** Космогония, рождение Вселенной, бытие, ничто, фрактальное сжатие, материализация (локализация) идей.

### Введение

Современная релятивистская космология пришла к тому, что заявляет о рождении Вселенной из *ничего* в результате Большого взрыва. Идея рождения Вселенной из ничего путём квантового туннелирования была впервые предложена Аткатцем и Пагелсом [1, 2] и развита Виленкиным [3].

Однако это рождение Вселенной из ничего происходит в соответствии со взглядами современной научной космологии без участия человека. Человек появляется во Вселенной, согласно этой космологии, лишь через миллиарды лет после того, как образуются атомы, звезды, галактики, планетные системы, биосфера, зародится живое вещество, которое проэволюционирует случайным образом до человека.

Такова парадигма современной науки в части космологии и космогонии. Человеку не дано её проверить экспериментально даже с помощью машины времени, поскольку на ранних этапах эволюции Вселенной либо конструкция машины времени вряд ли сможет выдержать без разрушения экстремальные внешние физические условия, либо наблюдения сведутся к фиксации получаемых изображений (что-то вроде телескопов или телесеансов), т. е. к разглядыванию картинок того, что нельзя потрогать.

В любом случае то, что мы будем считать научными фактами, подтверждающими теорию Большого взрыва, оказывается способом самообмана, в котором не захочется признаться. Ведь в любую историческую эпоху люди, которых в современной терминологии следовало бы отнести к научной элите эпохи, были убеждены в справедливости современной им космогонии и космологии. Для древних индусов — это учение о плоской Земле на спинах слонов, для более поздних эпох — учение о Солнце, вращающемся вокруг Земли и т. д. Наши современные учения ничуть не лучше в глазах учёных мужей иных эпох, которые мы относим к будущим эпохам.



## 1. Рождению Вселенной из *ничего*

Тем не менее следует серьёзно отнестись к идее рождения Вселенной из «ничего» и посмотреть на этот тезис с иной точки зрения, в которой рождение Вселенной есть акт, неотделимый от человека.

Что такое *ничто*<sup>1</sup>? Космологи и физики не дают ответа на этот вопрос. Чтобы сказать, что есть *ничего*, физик должен сделать *ничто чем-то*. Делает он это посредством сопоставления неопisanного *ничему* с описанным *нечто*. Это *нечто*, как правило, описано, представлено в форме математических формул, и этому математическому описанию предписывается материальное *существование*, т. е. это математическое описание должно *быть*. Иначе говоря, должно *пребывать*, находиться в состоянии *бытия*.

То, что физик предписывает математической абстракции, находится в материальном бытии, означает, что это бытие достаточно конкретно, т. е. обладает вполне определённым набором конкретных свойств. Бытие математической абстракции *налично*, поскольку налична, конкретна сама эта математическая абстракция (она, её конструкции и структуры наличествуют, т. е. являются набором конкретных математических свойств).

Философски образованный читатель уже отметил, что автор явно использует понятия и рассуждения Гегеля из «Науки логика». Да, это так. Почему Гегель?

Гегель оставил нам произведение, в котором он фактически пытался ответить на проклятый вопрос: «Что есть Вселенная, в которой мы живём, откуда она и откуда мы... как мы появляемся и как появляется Вселенная?» Однако в его эпоху не было необходимого математического инструментария, и это заставило его строить свою космогонию и космологию вербально. Сложность решаемой им задачи при отсутствии нужных средств привела к порождению словесных логических конструкций, которые были названы диалектической логикой Гегеля. Наиболее успешно ею пользовались Маркс и Ленин.

Поскольку в наши дни Маркс — признанный, как минимум в социологии, авторитет, успешность теории которого во много обязана логике Гегеля, то естественно повторить эксперимент Марса с гегелевскими рассуждениями применительно к космологии.

Итак, продолжаем...

Наличное бытие появилось благодаря конкретному математическому описанию, чтобы заменить абсолютно неналичное *ничего*. Иначе говоря, оно появилось вместо *чистого ничего*, или вместо *чистого ничто*.

*Ничего*, из которого родилась Вселенная, — это *чистое ничего*, *чистое ничто*. А *чистое ничто*, как пояснил Гегель, есть *чистое бытие* [4, с. 139-140]. «Категории чистое бытие и ничто виртуальны. Причём виртуальны не

---

<sup>1</sup>*Ничто* — это именительный падеж слова *ничего*, а *ничего* — это родительный падеж слова *ничто*.

только в том смысле, что воображаемы, но и в том смысле, что они постоянно переходят друг в друга» [5, с. 31].

«Внутреннее беспокойство мысли, перебегающей от чистого бытия к ничто и обратно, есть *становление* <...> В ходе становления что-то *возникает* и тут же исчезает, *преходит*» [5, с. 31],[4, с. 166].

Спонтанное сосредоточение мысли, концентрация внимания на нечто неуловимом по ходу становления есть *непустое мышление*. Оно определяет устойчивый образ *наличных вещей*. Этой ступени осознания соответствует обнаружение *наличного бытия* (которое представляет собой единство возникновения и прехождения). «Наличное бытие есть *определённое* бытие; его определённость есть *сущая* определённость, *качество*» [4, с. 169]. Качество *изменчиво*, оно означает явление того, что именуется *временем*.

Используемые слова «родилась», «возникает», «становится» не означают, что речь идёт о процессе, переходах во времени. Нет, они понимаются как логические следствия. Переходы осуществляются логически, в логике, а не во времени.

Логика — это для человека, для сознания, которое ему присуще, а не для окружающих человека вещей, которые возникли как наличное бытие из ничего посредством той же логики сознания. Иначе говоря, всё вокруг — результат логических переходов в сознании людей. Или, если хотите, вещи могут пребывать в состоянии ничего, т. е. в состоянии чистого бытия, либо, что логически допустимо, в состоянии наличного бытия. В современной терминологии смена состояния происходит не во времени, а является мгновенным квантовым переходом, подобно тому, как электрон меняет орбиту в атоме. А поскольку речь идёт о сознании, то это «квантовый переход» из ничего к осознанию своего существования по аналогии с тем, как ребёнок вдруг себя стал помнить! Каждый человек знает, что в его жизни было два состояния: себя не помнил, и это есть ничего, или чистое бытие, и с этого момента себя уже помню, т. е. оказался в состоянии наличного бытия.

## 2. Сознание из ничего

Как возникло само сознание? Человек мыслит, мысль — продукт мышления. Мысль, что я существую — это осознание собственного наличного бытия. Мышление связано с фактом наличного бытия. Мышление без конкретной мысли есть *пустое мышление*, а пустое мышление есть чистое ничто, т. е. есть чистое бытие. Значит сознание было всегда.

Если я был в состоянии чистого бытия, т. е. был ничем, то и всё было ничем; мир был в состоянии ничего.

Если мы в состоянии чистого бытия, т. е. мы — ничто, то и всё ничто; мир был в состоянии ничего, и не было бы, в частности, этой статьи, и никто не задавался бы вопросом — откуда мы, и как появилась Вселенная.

Но если я в состоянии наличного бытия, то мир вокруг не может быть ничем, в нем я различаю нечто, т. е. наличное бытие чего-то ещё. Чего? А того, что смогу мысленно представить. *Индивидуальное наличное бытие начинает*

*конструировать* из окружающего *ничего* наличное бытие этого окружающего меня чистого бытия. Скореллированные мысли-представления совокупности индивидуальных наличных бытийствующих сознаний дают реальную коллективную материальную среду обитания для носителей этих сознаний [7].

Элементарная частица рождается из вакуума *готовой*, т. е., например протон, со всей своей внутренней структурой: массой, зарядом, спином, изотопическим спином и т. д. Физиков почему-то не смущает то обстоятельство, что Вселенную они собирают из хорошо организованных готовых кирпичей — элементарных частиц.

Частица с момента рождения способна выполнять свои функции: перемещаться, вступать во взаимодействия, распадаться или превращаться в другие частицы при определённых условиях.

Аналогично мозг, а точнее организм его содержащий, возникает из *ничего готовым*, т. е. со всей своей структурой. Мозг с момента *появления на свет* способен выполнять свои функции.

Мозг (организм) рождает новый мозг (организм) подобно тому, как частица распадается на другие частицы. Ребёнок (новый организм) формируется в организме матери, но точно также нейтрон формируется внутри протона

$$p \rightarrow n + e^+ + \nu. \quad (1)$$

Правда, в отличие от биологии, физика мало интересуется механизмом зарождения нейтрона внутри протона<sup>2</sup>.

Процесс однотипного рождения частиц — это сброс энергии, например в форме распада нуклонного резонанса  $\Delta$

$$\Delta \rightarrow (p \text{ или } n) + \pi.$$

Данный пример всего лишь приведён в качестве образца того, как в науке физике вполне допускается рождение готовых структур, а не для того, чтобы во всех подробностях отыскать в физике элементарных частиц аналог рождения матерью ребёнка.

Таким образом, Вселенная была всегда, то как чистое бытие, то как наличное бытие. Равно, люди, — носители сознания, были всегда. И их наличное бытие осуществляется в форме исторических эпох, разные совокупности которых дают различные, развивающиеся во времени, истории [7].

Люди то пребывают в небытии, и их дух (сознание), как материя, находится в состоянии «ничего», либо в бытии. Будучи в наличном бытии, они порождают и время, и пространство. Историки это чувствуют:

«Берлинский историк географии Г.Д. Шульц высказал легко запоминающуюся формулу «пространства не существуют сами по себе, пространства создаются». Они являются продуктом человеческого сознания, результатом интеллектуальных построений. Это заявление

<sup>2</sup>Однако члены ряда возмущений посредством фейнмановских диаграмм демонстрируют, отчасти, что происходит «внутри».

является принципиальным изменением парадигмы в исследовании пространства» [9, 10].

В.О. Ключевский отмечал, что «история складывается из двух великих параллельных движений — из определения отношений между людьми и развития власти мысли над внешним фактом, т. е. над природой. <...> Мир факта есть мир, совершенно чуждый духу <...> » [6, с. 242].

Вселенная без людей, без сознания не существует. Посредством мышления порождается не только Вселенная [7], но и физические законы. Механизм изменения физических законов предсказан Станиславом Лемом, и может быть описан с помощью структур Кулакова–Владимирова [8]. А теория Большого взрыва — это всего лишь паттерн (образец) наличного бытия Вселенной, — иначе говоря, картина, выработанная в нашей исторической эпохе. И как картина, характерная только своей эпохе, со временем окажется в руках историков, и только им будет интересна.

### 3. Конечность конкретного сознания

Речь фактически идёт об ответе на вопрос: почему человек не живёт вечно?

Конструктор реализует конкретное сознание, т. е. человека, обладающего телом, как компактную область  $V$  в 4-мерном пространстве-времени  $\mathbb{R}^3 \times \mathbb{R}$ . Область  $V$  с точностью до гомеоморфизма — это цилиндр  $B^3 \times [0, T]$ , где  $B^3 \subset \mathbb{R}^3$  — 3-мерный замкнутый шар.

Если бы человек жил вечно, то имели бы  $V = B^3 \times [0, +\infty)$ . В каждый момент времени для поддержания сознания тело человека должно иметь ресурс  $r(t) < +\infty$ , который не может быть ниже величины  $r_0$ , т. е.

$$r_0 \leq r(t) < +\infty.$$

Интегральный жизненный ресурс также должен быть конечным

$$\int_0^{+\infty} r(t) dt < +\infty. \quad (1)$$

Сходимость несобственного интеграла (1) означает, что имеет место оценка

$$r(t) \leq \frac{const}{t^\alpha}, \quad t \in [0, +\infty), \quad \alpha > 1.$$

Следовательно, существует момент времени  $t_0 \in [0, +\infty)$ , для которого  $r(t_0) < r_0$ . Это означает разрушение тела человека, поскольку не хватает ресурса для поддержания жизнедеятельности.

#### 4. Материализация мыслей посредством субстанции

Наличные вещи, рождённые непустым мышлением, представляющие собой в общем случае сложные системы, могут физически воплощаться во Внешнем мире. Воплощёнными оказываются и металлические детали станков, и живые люди. Поэтому введём «понятие «субстанция», подразумевая под этим термином всё то конкретное физическое, во что воплощены элементы сложного объекта. Следовательно, субстанцией может быть и строительный материал, и живой организм, и цепочка букв на бумаге, и любые другие формы внешнего проявления материальности элементов системы» [11, с. 6].

Общий квантовый механизм материализации мыслей был нами описан в публикациях [7, 8, 14, 16, 17].

Опишем ниже само появление вещей из ничего на классическом уровне, т. е. вне аппарата квантовой механики.

#### 5. Порождение Мира из ничего или из любого нечего

Вселенная, Мир — это изображение, которое появляется в нашей голове.

Новая космогония должна пояснить, как это изображение возникает из ничего, из тьмы. Точнее, она должна пояснить, как наши мысли порождают нечего во Внешнем мире буквально из ничего.

Библия даёт следующий ответ: и сотворил Бог человека по образу Своему, по образу Божьему сотворил его. И коль человек (его сознание) подобен Богу, то и творить человек может то, что может Бог. Поэтому «в начале сотворил человек небо и землю. Земля же была безвидна и пуста, и тьма над бездною и сказал человек: «Да будет свет». И стал свет». Иначе говоря, человек способен из тьмы родить свет, породить Мир, Вселенную из тьмы, из ничего. И всё это по воле Бога. Почему в Библии говорится не о человеке, а о Боге? Просто потому, что один человек, даже созданный подобным Богу, не может в одиночестве сотворить Вселенную. Но «много» могут сделать множества людей. Соответствующую математическую формализацию сказанного мы описали в [7, 14]. Как результат, в тексте Библии человек появляется лишь на пятый день. Это и понятно: сознание с первого дня, а осознание на пятый.

Однако если забыть о Боге и придерживаться теории Большого взрыва, то её мы и увидим в Библии: «... Да будет свет. И стал свет». И после этого наступил день первый — потекло время, и на пятый день появится человек.

Текст Библии свят для верующего, в нём в поэтической форме отражены идеи космогонии лучших умов прошлых эпох. Мы позволили себе переработать его так, чтобы более явно проявилась изложенная *новая космогония*, которая альтернативна современной научной космогонии и теории Большого Взрыва, в частности.

Итак, возвращаясь к сказанному в начале параграфа, ищем ответ на вопрос: «Как можно породить из ничего изображение, т. е. нечего?»

Нам известен способ хранения изображений методом фрактального сжатия: хранят не изображение  $Im$ , а алгоритм  $A_{Im}$  его воссоздания. Последователь-

ные вычисления (итерации), совершаемые мозгом с помощью алгоритма  $A_{Im}$ , поэтапно собирают, проявляют, усиливая резкость, изображение  $Im$ .

Алгоритм  $A_{Im}$  — это итерации  $f^n = \underbrace{f \circ \dots \circ f}_n$  сжимающего отображения  $f : \Omega \rightarrow \Omega$  области  $\Omega$  размещения изображения  $Im$ , для которого исходное изображение (или некоторое близкое к нему) является неподвижной точкой, т. е.  $f(\Omega) = \Omega$ . Для восстановления изображения  $Im$  достаточно многократно применить это отображение к любому стартовому изображению. По теореме Банаха такие итерации всегда приводят к неподвижной точке, то есть к исходному изображению [15].

Где хранится алгоритм  $A_{Im}$ ? Он, естественно, в битовом коде хранится в микротрубочках нейронов головного мозга. Там же и создаётся изображение в процессе мышления, порождающем мысли (идеи-фантазии) о структуре окружающего мира. Мысли воспринимаются, осознаются человеком как изображения  $Im$  (кубитовые состояния димеров в микротрубочках нейронов [12, 13]), которые кодируются в форме алгоритма  $A_{Im}$ .

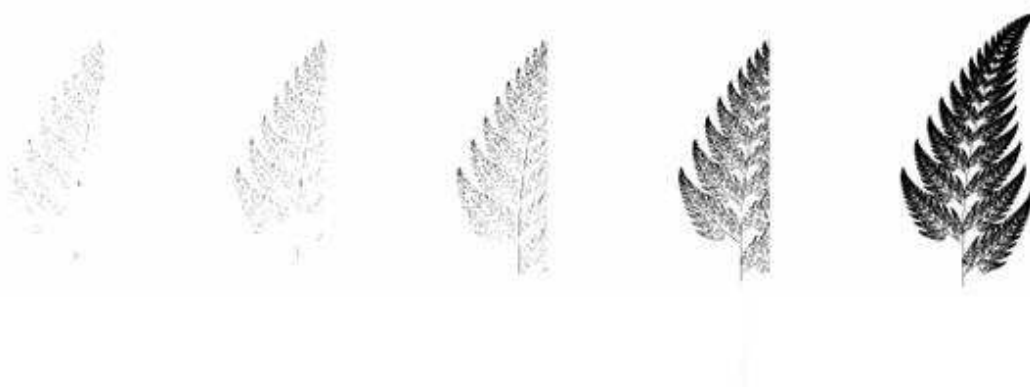


Рис. 1. Так появляется лист перед человеком: настраивается зрительное восприятие деревьев = рождение деревьев. Показаны этапы воссоздания изображения через 2000, 4000, 10000, 50000 и 200000 итераций

Реализация алгоритма  $A_{Im}$  во внешнем материальном мире — это процесс корреляции идей-фантазий в голове с соответствующим состоянием внешнего мира [7, 14]. Во времени полная сборка идеи-фантазии  $Im$  во внешнем мире требует миллионов итераций (рис. 1), т. е. длительного времени работы алгоритма  $A_{Im}$ , которое подчас больше времени существования мозга. Поэтому работа по сборке передаётся из поколения в поколение, поскольку отдельный человек имеет конечный отрезок хранения сознания (§ 3). Передача реализуемой идеи-фантазии при этом происходит посредством системы образования подрастающего поколения.

Почему человеческий мозг не вечен? Потому что работа алгоритма  $A_{Im}$  — это неизбежные необратимые тепловые процессы стирания битов кода  $kodA_{Im}$ , т. е. это износ материала мозга. Впрочем, об этом мы уже говорили в § 3.

Очень важный момент: не играет роли, чем заполнена область  $\Omega$ , к которой применяются итерации отображения  $f$ . На рис. 2 — это чёрный прямоугольник,



Рис. 2. Поэтапное появление мамы в глазах новорождённого ребёнка или «рождение Вселенной из тьмы» (из ничего) [15]



Рис. 3. «Рождение Вселенной» из чего угодно [15]

а на рис. 3 — это изображение кота. Результат один — появляется изображение известной шведской модели Lena Söderberg.

Однако восстановление плоского изображения — это всего лишь упрощённый вариант того, что происходит в пространстве. В реальности восстанавливаются по фрактальному алгоритму объёмные, 3-мерные тела в 3-пространстве. Точнее, создается Вселенная с её материей, звёздами, галактиками, планетами, биосферой планет. Конечно, отдельно приходится рассматривать процесс созидаение структуры, которую мы называем пространством. В нашем примере восстановления фото Лены мы имели готовый прямоугольник экрана, т. е. готовый набор пикселей. В реальности создается и сам экран по мере того, как возникает необходимость в размещении нового пикселя. *Пространство даёт место вещам.* В своё время о появлении нового кубического метра, пристраиваемого к «закончившемуся» пространству прозорливо писал Энгельс.

## 6. Иллюзия плотности тела

Визуально материализованная из ничего (чёрный квадрат) или из нечто (кот) Lena Söderberg порождает иллюзию, что наша дама занимает часть пространства, что она плотна, непроницаема, а вне её нечто воздушное, разряженное и уж точно отличное от тела Лены.

В действительности это работа мозга, сознания, порождающего конкретное бытие из ничего. Теоретически, *конструктивно*, разницы между телом Лены и воздухом вне её нет.

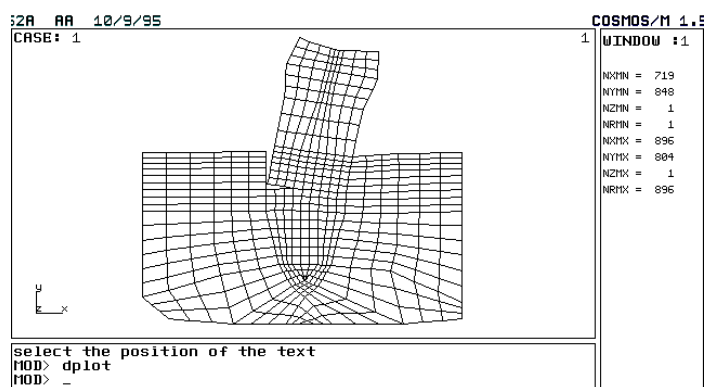


Рис. 4. Метод конечных элементов представляет очень плотный стальной штифт, корень зуба, десну и пустое пространство вне их

Чтобы это понять, достаточно представить квадрат с Леной и воздухом в рамках метода конечных элементов. На рис. 1 изображены тела разной плотности. Помимо картинки имеются ещё данные с численными наборами для коэффициентов, характеризующих свойства тел.



## 7. Старение теорий

Любые устоявшиеся научные теории со временем устаревают: признаются неточными, а то и ошибочными. Человечество находится в гонке за правильной научной теорией. Почему это происходит?

С точки зрения современной научной парадигмы мы находимся в состоянии раскрытия тайн Природы. Другими словами, когда-то Вселенная появилась, развилась до нынешнего состояния, и наука «докапывается» до сути винтиков и шурупчиков, из которых состоит механизм Вселенной.

Это довольно примитивный тезис. Во-первых, он порождает иллюзию, что когда-нибудь все тайны работы механизма Вселенной будут раскрыты, а, во-вторых, отстраняет «человека сознающего» от акта творения механизма Вселенной.

Однако стоит подвергнуть сомнению иллюзию окончательного познания механизма Вселенной, как мы придём к мысли, что механизм Вселенной эволюционирует и его эволюция опережает человеческую сообразительность. Поэтому все тайны не будут раскрыты.

Но, что более важно, — само предполагаемое опережение эволюции Вселенной эволюции человеческой мысли говорит о том, что Вселенная обязана своим развитием развитию Человечества. Ведь она должна опережать то, что никак не участвовало при рождении Вселенной. Микроб, случайно возникший и сидящий на стрелке механизма Вселенной, вдруг определяет скорость вращения стрелки!

Вывод прост: именно люди, их мысли, представления об окружающей их Вселенной, бытие в которой они осознают, и *порождают* механизм самой Вселенной изначально, при одновременном возникновении конкретной мысли из чистой мысли, и конкретного бытия Вселенной их чистого её бытия, т. е. из ничего.

Следовательно, чем больше людей, чем больше мыслей о Вселенной, тем больше коррекций механизма Вселенной. И эти коррекции идут постоянно и непрерывно с рождением каждого нового поколения, каждого нового гения. Иначе говоря, Вселенную люди всё время перестраивают, и в силу этого научные теории вынуждены всё время улавливать эти перестройки и соответствующим образом изменяться. Таким образом, старение теорий, их постоянное совершенствование не есть следствие недомыслия людей конкретной эпохи из-за чрезвычайной сложности механизма Вселенной, а нарастающее разнообразие людских мыслей, связанное с ростом численности людей<sup>3</sup>.

А как же законы, физические законы, управляющие Вселенной? Ведь если каждая мышь, глядя на Вселенную, её меняет, то развитие Вселенной непредсказуемо. Это не нравилось Эйнштейну в квантовой механике, не нравился её

---

<sup>3</sup>При этом в рамках конкретной исторической эпохи наука вполне довольна господствующими теориями, а создавшие их учёные купаются в славе до конца жизни (Бор, Гейзенберг, Планк и др.). Но взаимодействие исторических эпох, их интерференция приводит к тому, что усиливается натиск фактов (материализованные новые мысли новых людей), говорящих о недостатках этих теорий. Как результат — они «стареют».

вероятностный характер научных предсказаний. Ответ прост: чем больше людей, чем больше желаний изменить Вселенную, т. е. чем больше навязываемых Вселенной физических законов, тем с большей силой эти навязываемые физические законы подпадают под действие принципов статистической математики.

Так хаос движения молекул, как выяснил Максвелл, управляется нормальным законом распределения. Следует принять, что и хаос человеческих мыслей, хаос придумываемых ими физических законов управляется неким законом распределения. В работе [8] мы показывали, как люди способны менять физические законы.

#### ЛИТЕРАТУРА

1. Atkatz D., Pagels H. Origin of the universe as a quantum tunneling event // *Phys. Rev.* 1982. V. 25. P. 2065–2073.
2. Atkatz D. Quantum cosmology for pedestrians // *Am. J. Phys.* 1994. V. 62. P. 619–627.
3. Vilenkin A. Creation universes from nothing // *Phys. Let. B.* 1982. V. 117. P. 25–28.
4. Гегель Г.В. Наука логики в трёх томах. Т. 1. М. : Мысль, 1970. 472 с.
5. Труфанов С.Н. «Наука логики» Гегеля в доступном изложении. Самара : Издательство «Парус», 1999. 192 с.
6. Ключевский В.О. Письма. Дневники. Афоризмы и мысли об истории. М., 1968.
7. Гуц А.К. Физика реальности. Омск : Изд-во КАН, 2012. 424 с.
8. Гуц А.К. Формализация новой космогонии Лема // *Математические структуры и моделирование.* 2014. № 3(31). С. 48–56.
9. Власюк О.С. Представления о пространстве русских историков второй половины XIX в.: дис. ... канд. истор. наук: Ом. гос. пед. ун-т. Омск, 2009. 220 с.
10. Шульц Г.Д. Пространства существуют не сами по себе, а создаются: к вопросу о происхождении понятия «Центральная Европа» в немецкой историографии. [Электронный ресурс]. URL: <http://www.ifl-leipzig.com/fileadmm/daten/downloads/НОМЕ/Publikatto11ea1opa1/1997> (дата обращения: 01.06.2018).
11. Мельников Г.П. Азбука математической логики. М. : Изд-во «Знание», 1967. 104 с.
12. Пенроуз Р. Тени разума. В поисках науки о сознании. II. М.–Ижевск : Ин-т комп. исследований, 2005.
13. Hameroff S., Penrose R. Orchestrated reduction of quantum coherence in brain microtubules: a model for consciousness? // *Mathematics and Computers in Simulation.* 1996. V. 40. P. 453–480.
14. Гуц А.К. Квантовое рождение физической реальности и математическое описание осознания // *Математические структуры и моделирование.* 2007. Вып. 17. С. 47–52.
15. Анисимова Э.С. Фрактальное кодирование изображений // *Scientific research and their practical application. modern state and ways of development 2013.* Физика и математика — Информатика и кибернетика. URL: <http://www.sworld.com.ua/index.php/ru/conference/the-content-of-conferences/archives-of-individual-conferences/oct-2013> (дата обращения: 01.06.2018).
16. Гуц А.К. Созидание мира с машиной времени // *Математические структуры и моделирование.* 2013. Вып. 28. С. 5–14.
17. Гуц А.К. Основы квантовой кибернетики. Изд. 2, испр. и доп. М. : УРСС, 2016. 216 с.

**ALTERNATIVE NEW COSMOGONY****A.K. Guts**

Dr.Sc. (Phys.-Math.), Professor, e-mail: aguts@mail.ru

Dostoevsky Omsk State University, Omsk, Russia

**Abstract.** In the article the mathematical apparatus is indicated, by means of which we can realize the ideas of Hegel about being, about the alienation of ideas in Nature, as well as the theory of the creation of the Universe by active consciousness, which is alternative to the theory of the Big Bang.

**Keywords:** Cosmogony, the creation of the universe, being, nothing, fractal compression, materialization (localization) of ideas.

*Дата поступления в редакцию: 30.06.2018*

## ЦЕНТРАЛЬНЫЙ РЯД ГРУППЫ УНИТРЕУГОЛЬНЫХ АВТОМОРФИЗМОВ НИЛЬПОТЕНТНОЙ АЛГЕБРЫ ЛЕЙБНИЦА ИНДЕКСА 2

**А.Н. Кабанов**

к.ф.-м.н., e-mail: kabanovan@omsu.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

**Аннотация.** Получено описание центрального ряда группы унитарных автоморфизмов свободной нильпотентной алгебры Лейбница индекса 2 над произвольным полем.

**Ключевые слова:** алгебра Лейбница, нильпотентная алгебра, унитарный автоморфизм, центральный ряд.

В статьях автора [1–3] было представлено описание гиперцентральной серии группы унитарных автоморфизмов свободной алгебры Лейбница.

В данной работе представлено описание центральной серии указанной группы для свободной нильпотентной алгебры Лейбница с индексом нильпотентности 2.

Напомним, что неассоциативная алгебра  $L$  над полем  $F$  с билинейным произведением  $[\ ]$  называется (правой) алгеброй Лейбница, если для любых элементов  $x, y, z \in L$  выполняется (правое) тождество Лейбница:

$$[[x, y], z] = [x, [y, z]] + [[x, z], y].$$

Или, что то же самое,

$$[x, [y, z]] = [[x, y], z] - [[x, z], y].$$

Отсюда видно, что  $[x, [y, y]] = 0$ .

Из тождества Лейбница также следует, что любой элемент алгебры  $L$  можно представить как линейную комбинацию элементов вида  $[[[a, b], c], d], \dots$ , поэтому для краткой записи будем опускать скобки, положив

$$[[a, b], c] = abc.$$

Более того, примем записи

$$[[a, b], b] = ab^2, [[[a, b], b], b] = ab^3 \text{ и т. п.}$$

Обозначим  $L^1 = L$ ,  $L^k = [L^{k-1}, L]$ . Если  $L^p = 0$  для некоторого натурального  $p$ , то алгебра  $L$  называется нильпотентной.

Минимальное  $p$ , для которого выполняется равенство  $L^p = 0$ , называется индексом нильпотентности алгебры  $L$ .

Пусть  $L_n$  — свободная алгебра Лейбница над полем  $F$  с множеством свободных порождающих  $X_n = x_1, \dots, x_n$ .

Выделим в группе  $\text{Aut}L_n$  всех автоморфизмов алгебры  $L_n$  подгруппу  $U_n$ , порождённую автоморфизмами вида:

$$\tau_i(y_i) : \begin{cases} x_i \rightarrow x_i + y_i, \\ x_j \rightarrow x_j, \quad j \neq i, \end{cases}$$

где  $y_i$  принадлежит подалгебре, порождённой  $x_{i+1}, \dots, x_n$ . Такая подгруппа называется группой унитарных автоморфизмов алгебры  $L_n$ .

Для краткости будем записывать произвольный автоморфизм  $\varphi$  свободной алгебры  $L_n$  с множеством свободных порождающих  $X_n$  как  $\varphi = (F_1, F_2, \dots, F_n)$ , где  $\varphi(x_i) = F_i$ ,  $i = 1, \dots, n$ .

Тогда произвольное отображение вида

$$\varphi = (x_1 + f_1(x_2, \dots, x_n), \dots, x_i + f_i(x_{i+1}, \dots, x_n), \dots, x_n), \quad (1)$$

где для любого  $i$  многочлен  $f_i(x_{i+1}, \dots, x_n) \in L_n$ , определяет автоморфизм из  $U_n$ , и группа  $U_n$  состоит из всех таких автоморфизмов.

Пусть алгебра Лейбница  $L_n$  является нильпотентной с индексом нильпотентности  $p$ .

В этой статье рассмотрим случай  $p = 2$ , т. е.  $L_n$  — алгебра с нулевым умножением.

Выделим в группе  $U_n$  подгруппы  $Z_k$ , состоящие из автоморфизмов вида

$$(x_1 + g_1(x_{n-k+1}, \dots, x_n), x_2 + g_2(x_{n-k+2}, \dots, x_n), \dots, x_n). \quad (2)$$

Очевидно, что  $Z_k \subseteq Z_{k+1}$ .

**Теорема 1.** Подгруппы  $Z_k$  составляют центральный ряд группы  $U_n$ .

*Доказательство.* Возьмём произвольные автоморфизмы  $\varphi \in Z_1$  и  $\psi \in U_n$ . Согласно описанию группы  $Z_1$ , автоморфизм  $\varphi$  имеет вид

$$\varphi = (x_1 + \lambda x_n, x_2, \dots, x_n),$$

где  $\lambda \in F$

Пусть  $\psi$  имеет вид (1). Вычислим  $\varphi\psi$  и  $\psi\varphi$ . Имеем

$$\varphi\psi = (x_1 + f_1(x_2, \dots, x_n) + \lambda x_n, x_2 + f_2(x_3, \dots, x_n), \dots, x_n).$$

Далее

$$\psi\varphi = (x_1 + \lambda x_n + f_1(x_2, \dots, x_n), x_2 + f_2(x_3, \dots, x_n), \dots, x_n).$$

Убеждаемся, что эти композиции равны. Это по определению означает, что  $\varphi \in Z(U_n)$ .

Теперь предположим, что  $\varphi = (x_1 + f_1, x_2, \dots, x_{i-1}, x_i + f_i, x_{i+1}, \dots, x_n) \in Z(U_n)$ . Возьмём автоморфизм  $\psi = (x_1 + x_i, x_2, \dots, x_n)$ . Тогда

$$\varphi\psi = (x_1 + x_i + f_1, x_2, \dots, x_{i-1}, x_i + f_i, x_{i+1}, \dots, x_n),$$

$$\psi\varphi = (x_1 + f_1 + x_i + f_i, x_2, \dots, x_{i-1}, x_i + f_i, x_{i+1}, \dots, x_n).$$

Таким образом, поскольку  $\varphi\psi = \psi\varphi$ , следовательно,  $f_i = 0$ .

Пусть теперь  $\varphi = (x_1 + f_1(x_i, x_n), x_2, \dots, x_n) \in Z(U_n)$ . Возьмём автоморфизм  $\psi = (x_1, \dots, x_i + x_n, \dots, x_n)$ . Тогда

$$\varphi\psi = (x_1 + f_1(x_i + x_n, x_n), x_2, \dots, x_i + x_n, \dots, x_n),$$

$$\psi\varphi = (x_1 + f_1(x_i, x_n), x_2, \dots, x_i + x_n, \dots, x_n).$$

Таким образом,  $f_1(x_i + x_n, x_n) = f_1(x_i, x_n)$ . Но т. к.  $f_1(x_i, x_n) = \lambda_i x_i + \lambda_n x_n$ , где  $\lambda_i, \lambda_n \in F$ , следовательно,  $f_1(x_i, x_n) = f_1(x_n)$ .

Таким образом, мы доказали, что  $Z_1$  — центр группы  $U_n$ .

Напомним, что в центральном ряде  $Z_k(U_n)/Z_{k-1}(U_n) = Z(U_n/Z_{k-1})$ .

Допустим, что для некоторого  $k \geq 1$  множества  $Z_1 \subseteq \dots \subseteq Z_{k-1}$  являются частью центрального ряда. Пусть  $\varphi \in Z_k$  имеет вид (2), а  $\psi$  — произвольный унитарный автоморфизм вида (1).

Составляя их композиции, видим, что

$$\varphi\psi = (x_1 + f_1 + g_1(x_{n-k+1} + f_{n-k+1}, \dots, x_n), x_2 + f_2 + g_2(x_{n-k+2} + f_{n-k+2}, \dots, x_n), \dots, x_n),$$

$$\psi\varphi = (x_1 + g_1(x_{n-k+1}, \dots, x_n) + f_1(x_{n-k+1} + g_{n-k+1}, \dots, x_n), x_2 + g_2(x_{n-k+2}, \dots, x_n) + f_2(x_{n-k+2} + g_{n-k+2}, \dots, x_n), \dots, x_n).$$

Так как  $L_n$  — алгебра с нулевым умножением, то многочлены  $g_i$  и  $f_i$  линейные. Из вида  $Z_{k-1}$  следует, что в  $U_n/Z_{k-1}$  композиции  $\varphi\psi$  и  $\psi\varphi$  совпадают.

Теперь докажем, что других автоморфизмов в  $Z_\alpha$  нет.

Используя автоморфизм  $\psi = (x_1 + x_i, x_2, \dots, x_n)$  и рассуждения, аналогичные случаю  $Z_1$ , получим, что в автоморфизме  $\varphi$  многочлены  $g_i = 0$  для любого  $i \geq k + 1$ .

Используя автоморфизм  $\psi = (x_1, \dots, x_i + x_n, \dots, x_n)$ , получим, что в автоморфизме  $\varphi$  многочлен  $g_1$  не содержит переменных  $x_i$  для  $i \leq n - k$ .

Таким образом, множество  $Z_k$  также входит в центральный ряд при условии, что туда входит  $Z_{k-1} \subseteq Z_k$ .

Так как для  $Z_1$  утверждение уже доказано, по индукции получаем то, что и требовалось доказать. ■

Из вида (2) следует, что подгруппа  $Z_k$  совпадает с группой  $U_n$  при  $n - k + 1 = 2$ . Отсюда видим следующее свойство этой группы.

**Следствие 1.** *Группа унитарных автоморфизмов свободной алгебры Лейбница с нулевым умножением гиперцентральна длины  $n - 1$ .*

## ЛИТЕРАТУРА

1. Кабанов А.Н. Центр группы унитарных автоморфизмов свободной алгебры Лейбница // Математическое и компьютерное моделирование: сборник материалов IV Междунар. науч. конф. (Омск, 11 ноября 2016 г.). 2016. С. 78–80.
2. Кабанов А.Н. Центральный ряд группы унитарных автоморфизмов свободной алгебры Лейбница // Математические структуры и моделирование. 2017. № 3(43). С. 12–15.
3. Кабанов А.Н. Верхний гиперцентральный ряд группы унитарных автоморфизмов свободной алгебры Лейбница // Математические структуры и моделирование. 2017. № 4(44). С. 49–52.

**CENTRAL SERIES OF THE GROUP OF UNITRIANGULAR AUTOMORPHISMS  
OF A FREE NILPOTENT LEIBNIZ ALGEBRA WITH INDEX 2****A.N. Kabanov**

Ph.D. (Phys.-Math.), Associate Professor, e-mail: kabanovan@omsu.ru

Dostoevsky Omsk State University, Omsk, Russia

**Abstract.** The central series of the group of unitriangular automorphisms of a free nilpotent Leibniz algebra with index 2 over an arbitrary field is described.

**Keywords:** Leibniz algebra, nilpotent algebra, unitriangular automorphism, central series.

*Дата поступления в редакцию: 10.09.2018*

## **РЕШЕНИЕ АКТУАЛЬНЫХ ЗАДАЧ ОБРАБОТКИ ДАННЫХ В СИСТЕМАХ МИКРОСЕЙСМИЧЕСКОГО МОНИТОРИНГА ГИДРОРАЗРЫВА ПЛАСТОВ**

**П.И. Вайнмастер**

аспирант, e-mail: wmaster91@mail.ru

Новосибирский государственный технический университет, Новосибирск, Россия

**Аннотация.** Выполнение операций по гидроразрыву пластов (ГРП) требует контроля за ростом формирующейся трещины. Распространённой технологией, обеспечивающей такой контроль, является микросейсмический мониторинг. Он основан на решении задач локации источников сейсмической эмиссии и их визуализации на трёхмерной координатной сетке. Существующие методы мониторинга ГРП имеют ряд недостатков, среди которых — низкая точность измерения геометрических параметров трещин. В настоящей статье рассматривается оригинальный подход к обработке данных микросейсмического мониторинга, позволяющий представить зону трещиноватости в виде трёхмерных непрерывных объектов — одного или нескольких слоев трещины, заполненных проппантом. Предложенные алгоритмы базируются на применении целого ряда процедур цифровой обработки сигналов: обнаружении импульса на фоне шума и локации его источника, пространственной фильтрации источников, построении траекторий раскрытия трещины, интерполяции набора источников и визуализации трёхмерной графической модели.

**Ключевые слова:** гидравлический разрыв пласта, трещина, сейсмический импульс, обнаружение, фильтрация, построение траекторий, триангуляция Делоне.

### **Введение**

В последние годы наблюдается истощение запасов нефтяных месторождений с высокопроницаемыми коллекторами. Нефтедобывающие компании вынуждены осваивать коллекторы с низкой проницаемостью, вследствие чего получил распространение гидравлический разрыв пласта (ГРП) — один из эффективных методов интенсификации добычи нефти. Суть метода заключается в том, что на забое скважины создаётся высокое давление, разрывающее пласт по направлениям минимальных напряжений. Для создания такого давления в пласт закачивается специальная жидкость, содержащая расклинивающий материал — проппант. В пласте образуется зона трещиноватости, формируя высокопроницающий канал для проникновения нефти к забою скважины.



Отсутствие контроля за развитием трещиноватой зоны может привести к обводнению скважины, что потребует её ремонта или полной остановки добычи. Одним из методов такого контроля, наиболее распространённым в наши дни, является микросейсмический мониторинг ГРП.

Классический мониторинг включает в себя несколько этапов, основные из которых — сбор данных о трещине, их обработка и интерпретация. Сбор данных заключается в регистрации сейсмических трасс в процессе гидроразрыва. Обработка записанных сейсмотрасс позволяет выполнить локацию источников микросейсмических событий в пласте. Завершающим этапом является интерпретация полученных данных для установления пространственных характеристик трещины. Полученная информация используется для управления основными параметрами гидроразрыва — объёмом и темпами закачки жидкости в пласт.

Наиболее сложными задачами обработки данных в микросейсмическом мониторинге считаются задача *локации* микросейсмических событий и задача их *визуализации* в виде графической модели. В данной работе рассматриваются оригинальные подходы к их решению.

## 1. Постановка задачи

Сейсмическим событием считается колебание пород пласта, вызванное увеличением трещиноватой зоны в размере 10-25 м в результате гидродинамического воздействия на пласт. Сейсмические импульсы, возбуждаемые при таких событиях, регистрируются датчиками вместе с естественными сейсмическими шумами и искусственными помехами. Задача *локации* сейсмических источников заключается в вычислении их координат, что даёт косвенную оценку траектории распространения трещины.

Известные методы локации точечных источников основаны на различных решениях обратной кинематической задачи сейсморазведки (ОКЗС). Считается, что скоростная модель пласта состоит из нескольких горизонтальных слоёв, имеющих глубинные границы и постоянную (различную для каждого слоя) скорость распространения сейсмических волн. Делается предположение о том, что по сейсмотрассам, записанным на земной поверхности, можно с достаточной точностью определить времена прихода волн к датчикам антенны и, на основе заданной скоростной модели, восстановить координаты их источника.

Решение ОКЗС опирается на подбор модели границ среды и вычислении пространственных координат сейсмического источника  $(x_s, y_s, z_s)$  и скорости распространения сейсмических волн  $V(z)$  от источника до точек наблюдений антенны. Требуется так подобрать параметры модели, чтобы результат решения прямой задачи для источника  $(x_s, y_s, z_s)$  совпадал (в пределах заданной погрешности) с наблюдаемыми временами прихода отражённой волны от глубинных границ.

Такая обратная задача является некорректной ввиду отсутствия единственного и устойчивого решения, а подбор параметров для её решения сопряжён с устранением целого ряда существенных неопределённостей разного рода [1].

Во-первых, решение задачи зависит от выбора модели: чрезвычайно сложно корректно определить количество пластов в среде и измерить времена прихода всех отражённых волн от всех границ. Во-вторых, существенное влияние на результат оказывает выбор метода решения задачи. Например, решение ОКЗС может заключаться в минимизации отклонений рассчитанных времён отражений от фактически наблюдаемых. Но даже эту оптимизационную задачу можно решать в рамках различных критериев: среднеквадратическое отклонение, абсолютное отклонение и т. п.

Низкая эффективность существующих методов, сложность получения единственного и устойчивого решения, а также необходимость проведения дополнительных исследований геофизической модели пласта обуславливают актуальность задачи локации сейсмических источников.

В работе [2] предложен альтернативный подход к решению задачи локации. В зоне приёма вводится трёхмерная координатная сетка с заданным шагом между узлами. Каждый узел сетки является кандидатом на проверку наличия или отсутствия сейсмического события. Единственным параметром, характеризующим пластовую среду, является средняя скорость распространения волн  $V$ . Для каждого узла сетки, с учётом его известных координат и скорости  $V$ , решается задача обнаружения сейсмического импульса на сейсмотрассах, зарегистрированных антенной. Если импульс обнаружен, узел считается точечным источником сейсмической эмиссии. Преимуществом данного подхода является направленное обнаружение микросейсмических источников: решение существует в рамках определённой погрешности единственно и устойчиво, даже в смысле структурной устойчивости.

В исходном алгоритме, описанном в [2], для обнаружения импульса предлагается выполнить статистическое накопление сигнала — суммирование сейсмотрасс с учётом рассчитанных времён задержек. Для принятия решения о наличии импульса используется простой критерий: максимальная амплитуда на суммарной сейсмотрассе должна превышать заданный порог. Но этот подход неустойчив к изменению амплитуды искомого импульса и к наличию в среде высокоамплитудных сигналов-помех, вызванных работой аппаратуры ГРП. Поэтому актуальной задачей является исследование и применение алгоритмов обнаружения сейсмического импульса, обладающих устойчивостью к изменению мешающих параметров модели.

Задача *визуализации* зоны трещиноватости заключается в представлении результатов обработки сейсмических данных в виде графической модели, позволяющей оценить её пространственные характеристики. От вида модели зависит принятие решений о дальнейшей разработке пласта и следующем после ГРП дизайне скважины. Поэтому модель должна отражать физические особенности трещины в удобном для восприятия виде.

Производители современного программного обеспечения для мониторинга ГРП используют трёхмерную модель зоны трещиноватости в виде множества источников микроземлетрясений на координатной сетке. Модели такого вида позволяют обнаружить сгустки сейсмической эмиссии, но накладывают определённые ограничения на восприятие результата обработки. Во-первых, реальные

данные ГРП характеризуются большой интенсивностью сейсмической активности (количество источников в модели может достигать нескольких десятков тысяч). Во-вторых, сгустки микросейсмических источников не имеют чётких контуров и не позволяют с достаточной точностью и достоверностью определить форму и толщину раскрывающихся трещин.

Альтернативным способом представления сейсмических данных является графическая модель в виде трёхмерных объектов — одного или нескольких слоев трещины, заполненных пропантом. Однако визуализация такой модели сопряжена с рядом трудностей. Избыточность сейсмических данных ограничивает корректное построение графической модели с помощью известных интерполяционных методов, т. к. количество сейсмических источников может составлять десятки тысяч.

В связи с этим, актуальной является задача пространственной фильтрации данных. Одним из её решений является переход от множества зарегистрированных источников сейсмической эмиссии к заведомо меньшему набору их гипоцентров, представляющих энергетические очаги локальных сейсмических событий.

Информация о гипоцентрах микросейсмической эмиссии даёт возможность восстановить пространственные и временные параметры трещиноватой зоны, построить траектории разлома и представить трещину в графическом виде, удобном для оператора ГРП и последующей обработки.

## 2. Теория

Модель сейсмического квазидетерминированного импульса, излучаемого точечным источником внутри пласта, в общем виде представляется в виде квазигармонического сигнала [3]

$$S(t, \varphi) = U(t) \cos(\omega_0 t + \varphi),$$

где  $U(t)$  — нормированная по амплитуде известная (при фиксированном значении  $\varphi$ ) функция времени, задающая форму импульса;  $\varphi$  — априорно неизвестная фаза импульса, которая равномерно распределена на интервале  $[-\pi, \pi]$ ;  $\omega_0 = 2\pi/T_0$  — опорная частота импульса;  $T_0$  — длительность импульса. Пример модели сейсмического импульса, излучаемого источником, приведён на рис. 1.

Фрагмент сейсмотрассы, зарегистрированный  $i$ -м приёмником наземной антенны на временном интервале  $[0; T]$ , представляет колебание, состоящее из суммы сейсмического импульса  $S(t, \varphi)$  и шума  $n(t)$

$$y_i(t) = \delta S(t, \varphi) + n(t), 0 \leq t \leq T,$$

где  $\delta$  — случайная величина, которая принимает значение 1, если сейсмический импульс был излучён точечным источником и 0 — импульс излучён не был. Фрагмент сейсмотрассы, зарегистрированной на интервале  $[0; 6 \text{ мс}]$ , представлен на рис. 2.

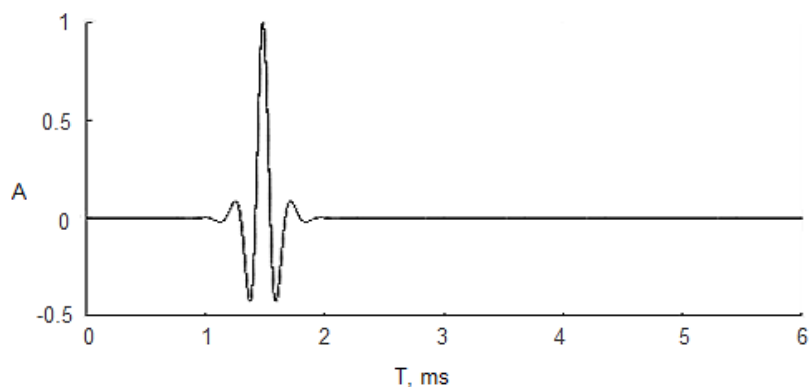


Рис. 1. Модель сейсмического импульса

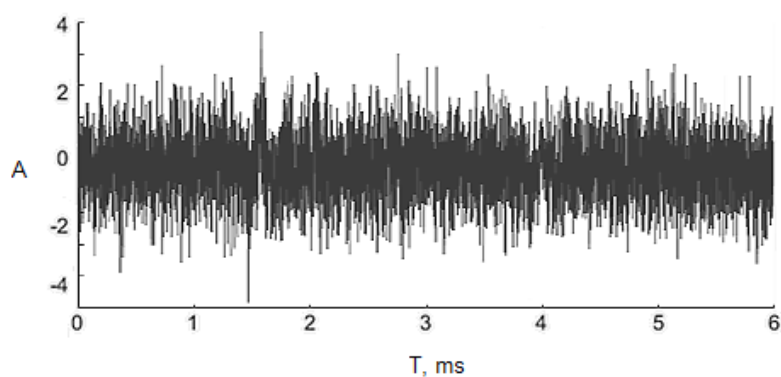


Рис. 2. Вид фрагмента сейсмограммы

Требуется определить правило, которое по результатам обработки набора сейсмограмм  $y_i(t)$  позволит сделать вывод о наличии или отсутствии сейсмического импульса в заданном точечном узле координатной сетки. Алгоритм должен удовлетворять двум основным требованиям: стабильная вероятность ложной тревоги при любых изменениях спектральной плотности мощности шума и инвариантность вероятности правильного обнаружения к начальной фазе сейсмического импульса.

Для решения этой задачи применены и исследованы два известных в радиотехнике алгоритма обнаружения сигнала, адаптированные к обнаружению сейсмических импульсов и удовлетворяющие описанным условиям: квадратурный корреляционный и статистический.

Квадратурный корреляционный алгоритм [4] предполагает вычисление «суммарной» сейсмограммы  $y(t)$  (она образуется путём суммирования сейсмограмм  $y_i(t)$  с учётом времён задержек распространения сигнала) и сравнение комплексных огибающих фрагмента сейсмограммы  $y(t)$  и шаблона сейсмического импульса  $S(t)$ .

Для определения коэффициента корреляции  $\rho_{ys}$  используется формула

$$\rho_{ys} = 2\sqrt{X_c^2 + X_s^2},$$

где  $X_c = \int_0^T y(t)S(t) \cos(\omega t)dt$ ,  $X_s = \int_0^T y(t)S(t) \sin(\omega t)dt$ .

Пороговая величина  $h$ , используемая для принятия решения об обнаружении, основана на выбранной вероятности ложной тревоги  $P_F$ :

$$h = 2\sqrt{N_0 E \ln\left(\frac{1}{P_F}\right)}.$$

Если коэффициент корреляции  $\rho_{ys}$  превышает порог  $h$ , принимается решение о наличии сейсмического импульса.

На рисунке 3 приведены графики функции корреляции комплексных огибающих  $\rho_{ys}(t)$  (верхняя толстая кривая чёрного цвета) и фрагмента суммарной сейсмограммы  $y(t)$  (нижняя тонкая кривая). Для уточнения взаимного расположения функций к ним добавлен график сейсмического импульса  $S(t, \varphi)$  (кривая серого цвета). Заметно, что начальная фаза сейсмического импульса  $S(t, \varphi)$  совпадает по времени с максимальным значением  $\rho_{ys}(t)$ , а максимум импульса  $S(t, \varphi)$  совпадает по времени с локальным максимумом фрагмента суммарной сейсмограммы  $y(t)$ .

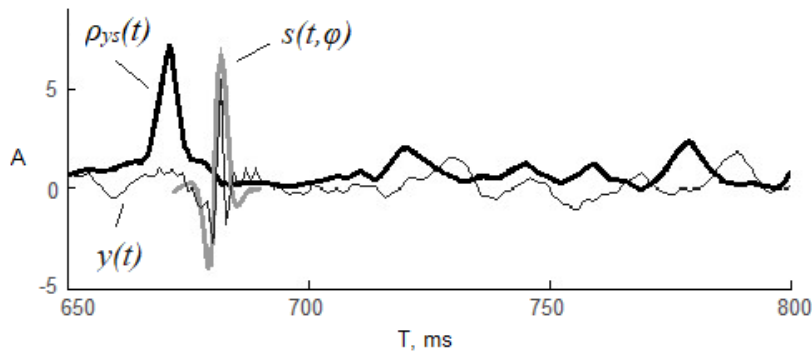


Рис. 3. Функция корреляции огибающих, сейсмограмма и импульс

Статистический алгоритм обнаружения [5, 6], в свою очередь, предполагает представление каждой сеймотрассы  $y_i(t)$  в виде ряда Фурье и вычисление коэффициентов Фурье  $\{u_{li}, v_{li}\}$  в качестве наблюдаемой выборки. Для описания семейства их совместных распределений вводится понятие достаточной статистики  $Z = z_1, \dots, z_I$  и производится расчёт плотности вероятности  $w(Z)$ . Плотность вероятности используется для построения решающей функции. Если решающая функция превышает пороговую величину, принимается решение о наличии импульса.

Решающая функция статистического алгоритма имеет вид

$$\varphi(Z) = \begin{cases} 1, & t(Z) > C(\alpha) \\ 0, & t(Z) < C(\alpha) \end{cases}$$

Здесь  $t(Z) = \sum_{i=1}^I \exp\left\{-\frac{\delta_i^{*2}}{2}\right\} M[n(L_2 - L_1 + 1), 1, \frac{\delta_i^{*2} z_i}{2}]$ ,  $M[*]$  — функция Куммера;  $\delta_i^*$  — пороговое отношение «сигнал/шум» для  $i$ -й гармоники, определяющее границу «зоны безразличия» алгоритма;  $L_1, L_2$  — границы полосы пропускания приёмников сигнала;  $z_i$  —  $Z$ -статистика для  $i$ -й гармоники. Пороговая величина  $C(\alpha)$  вычисляется на основе заданного уровня вероятности ложной тревоги  $\alpha$ .

Узлы виртуальной сетки, в которых было принято решение о наличии сейсмического импульса, принимаются локальными очагами микроземлетрясений. Таким образом с помощью задачи обнаружения, решается задача *локации* сейсмических источников.

Известные системы мониторинга ГРП осуществляют визуализацию графической модели трещины на основе полученного набора источников, при этом трещина представляется либо в виде «облака» этих точек на координатной сетке, либо в виде тепловой карты (двумерное изображение в горизонтальной проекции, где области большего скопления источников имеют более тёплую цветовую гамму). Оба этих подхода дают приблизительное представление о распределении зон трещиноватости в пласте, но они не позволяют оценить такие параметры трещиноватой зоны, как форма и толщина. В данной работе предлагается представить зону трещиноватости в виде слоёв, заполненных пропантом.

Если на основе имеющегося набора источников построить поверхность (с помощью триангуляции Делоне), то она будет иметь вид, показанный на рисунке 4. Из рисунка видно, что набор источников избыточен: во многих близко расположенных узлах трёхмерной виртуальной сетки обнаруживаются очаги микросейсмической эмиссии. Чтобы устранить эту избыточность, очередным этапом обработки является фильтрация узлов-источников.

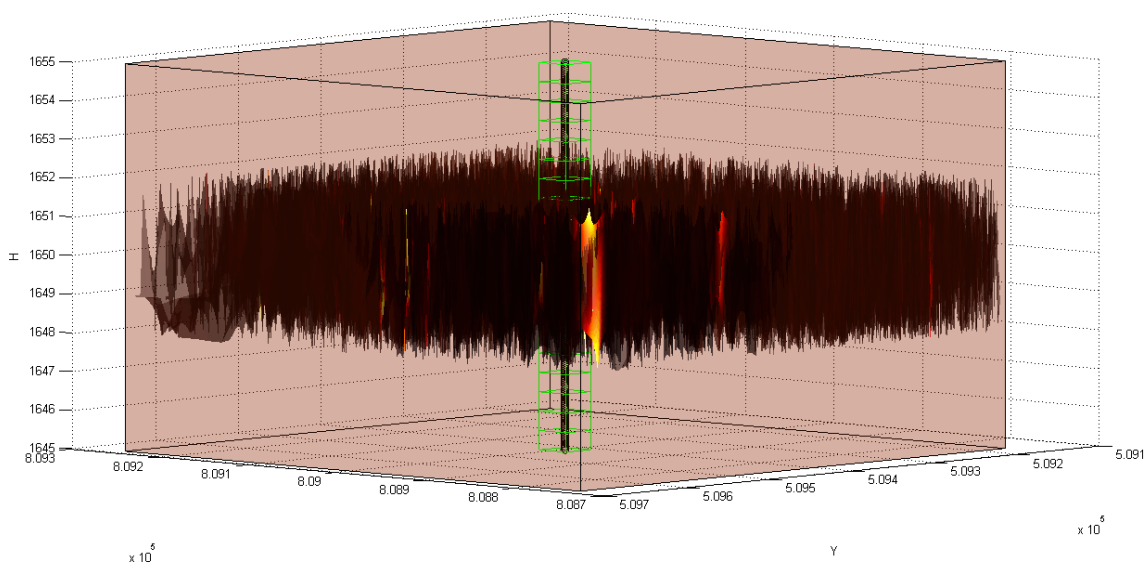


Рис. 4. Исходная трёхмерная модель трещиноватости целевого пласта

Фильтрация заключается в обнаружении кластеров («облаков» микросейсмической эмиссии) и объединении узлов внутри каждого кластера в один пространственный объект, являющийся гипоцентром «облака». Разрыв среды, формирующий острие раскрывающейся трещины, происходит в этой точке модели с наибольшей вероятностью. Для осуществления фильтрации используются методы компонентного и кластерного анализа [7].

Каждый узел-источник представлен пятью параметрами: это координаты  $X$ ,  $Y$  в горизонтальной плоскости, глубина расположения  $H$ , амплитуда сейсмического события  $A$  и время его возникновения  $T$ . Чтобы сократить время обработки узлов, а также устранить влияние схожих и близких параметров, предварительно выполняется переход к меньшему количеству компонент, число которых равно трём —  $F_1$ ,  $F_2$  и  $F_3$ . Для отсева «лишних» компонент используется критерий Кайзера.

Для всех узлов-источников, с учётом новой системы признаков, выполняется кластеризация с помощью агломеративного алгоритма. Отметим, что исходная область целевого пласта разбивается на области одинакового размера (параллелепипеды), в рамках каждого из которых происходит независимая обработка узлов-источников. Такой подход позволяет значительно улучшить качество кластеризации, т. к. обнаруженные кластеры являются компактными и рассредоточены по всей области мониторинга. Центры масс обнаруженных кластеров принимаются в качестве гипоцентров соответствующих «облаков» сейсмичности.

Если исходное количество источников, расположенных в области локации размером  $512 \times 512$  метров, может составлять тысячи, то количество гипоцентров после пространственной фильтрации не превышает нескольких сотен. Это позволяет выполнить интерполяцию и представить трёхмерную модель трещины в виде одного или нескольких слоёв.

Задачу построения графической модели трещины можно условно разделить на две подзадачи: восстановление траекторий разрыва зоны трещиноватости в пласте коллектора и визуализация трёхмерной поверхности на их основе.

Раскрытие и рост трещин при гидроразрыве, как правило, происходят в нескольких направлениях, соответствующих направлениям перфорации (прострела) стенок скважины. Области перфорации характеризуются излучением сейсмических импульсов высокой амплитуды вблизи точек разрыва. Поэтому для локации основания трещины следует проанализировать амплитудные характеристики гипоцентров вблизи забоя: максимальные значения амплитуд соответствуют местам первичного вскрытия породы. По мере дальнейшего роста трещины её траектория сначала является гладкой, затем она становится шероховатой и, наконец, приобретает ветвящийся (фрактальный) характер.

Для восстановления траекторий трещиноватости используется оригинальный алгоритм [8]. Исследуемая область пласта, с учётом наложенной трёхмерной сетки локации, представляется в виде двух цилиндров: внутреннего и внешнего. Внутренний цилиндр ограничивает область вблизи ствола скважины. Эта область содержит очаги микроземлетрясений, образующихся при разрыве основания трещины. Внешний цилиндр охватывает всю исследуемую

область целевого пласта, и, в том числе содержит внутренний цилиндр.

На основе ряда закономерностей, которые вычисляются с помощью исходных данных, внешний цилиндр разбивается на  $N$  областей (секторов) одинакового размера. Чтобы задать взаимное расположение этих секторов, во внутреннем цилиндре отыскивается гипоцентр  $r'_i$ , расположенный вблизи ствола скважины и имеющий наибольшую амплитуду  $A_i$  (высокая амплитуда колебаний фиксируется в момент прорыва породы разрывающей жидкостью, т. е. найденный узел с высокой вероятностью лежит в основании трещины). Первый сектор располагается так, что отрезок прямой, проведённый в горизонтальной проекции из центра координатной сетки к найденному узлу и продлённый до границы внешнего цилиндра, делит проекцию первого сектора на две равные части. Остальные секторы строятся по часовой стрелке от первого.

При последовательном просмотре каждого сектора происходит восстановление траекторий трещиноватости с помощью рекурсивного алгоритма. Построение заключается в последовательном отборе гипоцентров  $r'_1, \dots, r'_n$ , лежащих внутри первого сектора, начиная с найденного ранее узла  $r'_i$  (назовём его «базовым»).

Обозначим последний известный узел траектории как  $r'_k$ , а выбираемый узел-претендент —  $r'_j$ . Для того чтобы узел  $r'_j$  был включён в траекторию, должны одновременно выполняться следующие условия:

- узел  $r'_j$  должен лежать не ближе к стволу скважины, чем  $r'_k$ , и быть ближайшим к  $r'_k$  среди всех остальных узлов данного сектора;
- время возникновения сейсмического события в  $r'_j$  должно превышать время возникновения эмиссии в узле  $r'_k$ ;
- амплитуда импульса в узле  $r'_j$  должна превышать заданный порог. Порог амплитуды  $A_{min}$  выбирается по гистограмме распределения амплитуд гипоцентров сектора на уровне, соответствующем одному среднеквадратическому отклонению.

Гипоцентры, для которых параметр «плотности»  $D$  принимает значения меньше пороговой величины  $D_{min}$ , также отсеиваются и не участвуют в дальнейших расчётах. Под «плотностью» здесь понимается количество узлов-источников, ранее объединённых в кластер, центром которого является рассматриваемый гипоцентр. Порог  $D_{min}$  вычисляется по формуле

$$D_{min} = M(D) \times C_D,$$

где  $M(D)$  — математическое ожидание выборки  $D$  наблюдаемой плотности гипоцентров сектора;  $C_D$  — пороговый коэффициент.

Отбор гипоцентров завершается, если ни один из оставшихся гипоцентров  $r'_j$  не удовлетворяет описанным критериям и не может быть включён в траекторию трещиноватости. Выбранные гипоцентры соединяются отрезками, образуя ломаную линию, задающую траекторию трещины в рамках сектора.

Для восстановления «ветвистой» структуры траектории выполняется рекурсивное погружение: каждый из выбранных на первом этапе гипоцентров  $r'_k$  представляется новой «базовой» точкой, и для него рекурсивно повторяется



процесс поиска гипоцентров, образующих ответвления от первоначально рассчитанной траектории. Затем процесс поиска повторяется для гипоцентров, найденных на этом этапе, и т. д. После завершения обработки в первом секторе процедура отбора гипоцентров и построения ветвящейся траектории выполняется поочерёдно для всех оставшихся секторов.

На рисунке 5 представлен вид пласта в горизонтальной проекции, на котором выделены внутренний и внешний цилиндры, секторы, а также построенные траектории трещиноватости.

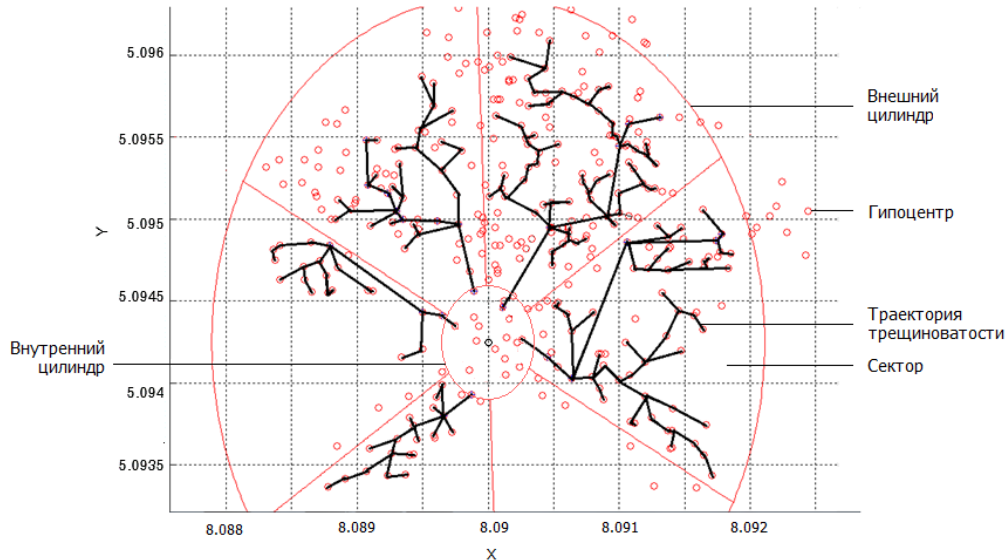


Рис. 5. Траектории раскрытия трещины ГРП (горизонтальная проекция)

Завершающим этапом обработки данных является визуализация трёхмерной графической модели. Для отображения динамики (роста трещины с течением времени), вводится временное окно, которое делит набор гипоцентров на несколько групп по времени возникновения. Построение траекторий и визуализация модели осуществляются поэтапно для каждой из таких групп.

Для визуализации модели трещиноватой зоны используются триангуляция Делоне. Интерполяция имеющегося набора данных основана на методе естественного соседа, а для учёта фрактального характера роста зоны трещиноватости используется метод срединного смещения.

### 3. Результаты экспериментов

Предлагаемые алгоритмы позволяют с достаточной точностью решать задачи обработки данных в системах микросейсмического мониторинга и восстанавливать пространственные и временные параметры фрактальной зоны трещиноватости, образованной при гидравлическом разрыве пласта.

Применение алгоритмов обнаружения сигнала (квадратурного корреляционного и статистического) позволяет обнаружить наличие сейсмического импульса на сейсмотрассах и осуществить локацию источников микросейсмических

событий. В результате экспериментов установлено, что статистический алгоритм устойчиво обнаруживает импульс при отношении сигнал / шум  $-20$  Дб и ниже. Квадратурный корреляционный алгоритм имеет порог обнаружения несколько ниже. Кроме того, статистический алгоритм позволяет обнаруживать импульсы, близко расположенные на сейсмограмме или даже перекрывающиеся. Корреляционный алгоритм не обнаруживает такие импульсы (точнее, не разрешает их), что ограничивает его применение в задачах сейсморазведки [9].

Применение алгоритма пространственной фильтрации позволяет избавиться от избыточности числа узлов локации, возникающей из-за непрерывного пространственного характера колебания частиц разрываемого пласта. Согласно проведённым экспериментам, наилучшее устранение избыточности достигается при использовании в качестве меры расстояния между объектами меры Евклида, а в качестве стратегии кластеризации — метода Average-link. При количестве исходных узлов в несколько тысяч количество гипоцентров после фильтрации не превышает нескольких сотен, а в среднем избыточность данных сокращается почти в 50 раз [7].

Для ускорения процесса кластеризации и устранения влияния схожих характеристик объектов дополнительно применяется компонентный анализ признаков и их сокращение с пяти до трёх. Помимо устранения помеховых составляющих, искажающих результат кластеризации, это позволяет достичь значительного ускорения процесса выделения кластеров (на 25–30 %).

Использование рекурсивного алгоритма построения траекторий позволяет восстановить фрактальную геометрию и динамику распространения трещиноватой зоны. Особенностью алгоритма является выделение трёхмерных секторов и построение фрагментов трещины в каждом из них. Поскольку расположение секторов соответствует прострелам перфорации пласта, каждая траектория трещиноватости определяет одно из «крыльев» раскрывшейся трещины.

Модель трещиноватой зоны, построенная согласно предложенному методу мониторинга ГРП, представлена на рисунках 6–8.

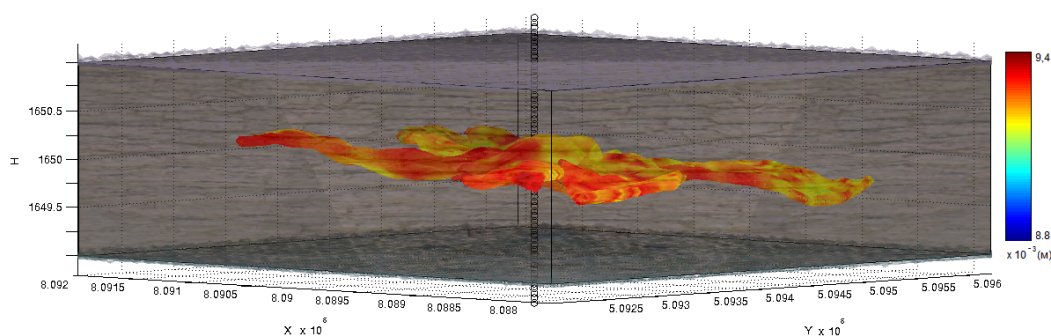


Рис. 6. Модель зоны трещиноватости пласта (3D)

На рисунке 6 изображена трёхмерная модель. Ось в центральной части рисунков представляет ствол скважины. На рисунке 7 изображён вид зоны трещиноватости сверху. Сравнивая его с рисунком 5, можно заметить, что

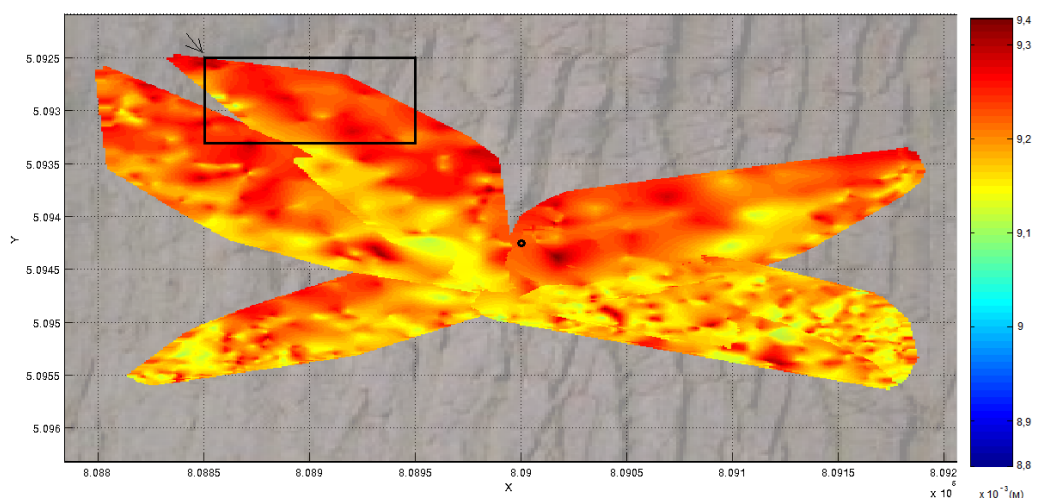


Рис. 7. Модель зоны трещиноватости пласта (вид сверху)

«крылья» трещины соответствуют выделенным траекториям трещиноватости. Рисунок 8 иллюстрирует фрагмент трещины, заполненный проппантом (его расположение указано чёрным прямоугольником на рисунке 7).

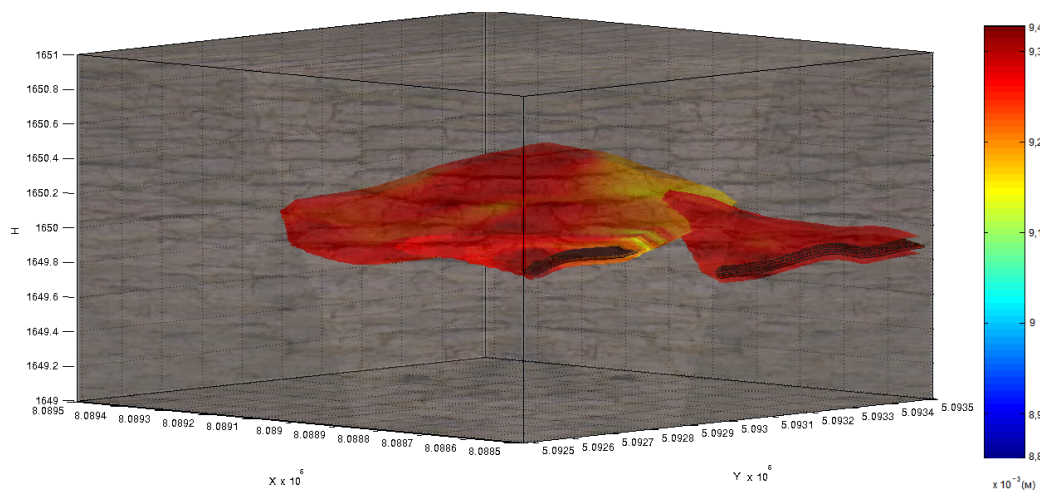


Рис. 8. Фрагмент модели зоны трещиноватости пласта (3D)

Цвет элементов поверхности пропорционален амплитуде колебаний и косвенно отражает толщину трещины: более тёплые оттенки соответствуют большей толщине.

#### 4. Заключение

Предложены и реализованы оригинальные алгоритмы обработки данных микросейсмического мониторинга ГРП. Они базируются на применении целого

ряда информационных процедур цифровой обработки сигналов: обнаружении сейсмического импульса и локации источников микросейсмической эмиссии; фильтрации узлов-источников для устранения их избыточности; восстановлении траекторий раскрытия трещины; визуализации трёхмерной графической модели.

Преимуществами данного метода по сравнению с существующими аналогами являются:

- точность и достоверность получаемых пространственных моделей трещиноватой зоны;
- учёт резонансных свойств среды и нелинейности распространения сейсмического сигнала;
- представление модели трещиноватости пласта в виде поверхностей — слоёв, заполненных проппантом;
- визуализация динамики распространения зоны растрескивания.

Применение представленных алгоритмов позволяет решать актуальные задачи обработки данных в системах микросейсмического мониторинга и получать адекватные изображения динамических пространственных объектов трещиноватости пласта.

#### ЛИТЕРАТУРА

1. Глоговский В.М., Лангман С.Л. Свойства решения обратной кинематической задачи сейсморазведки // Технологии сейсморазведки. 2009. № 1. С. 10–17.
2. Рабинович Е.В., Туркин А.С. Наземная локация микросейсмических сигналов для мониторинга гидравлического разрыва пласта // Доклады ТУСУР. 2012. № 1(25), Ч. 1. С. 104–112.
3. Модель сейсмического импульса, возникающего при гидравлическом разрыве пласта / Е.В. Рабинович [и др.] // Математические структуры и моделирование. 2014. № 4(32). С. 105–111.
4. Rabinovich E.V., Shefel G.S., Vaynmaster P.I. Correlation detection of seismic pulses for hydraulic fracturing monitoring // 16 International conference of young specialists on micro/nanotechnologies and electron devices (EDM). IEEE, 2015. P. 162–165.
5. Богданович В.А., Вострецов А.Г. Теория устойчивого обнаружения, различения и оценивания сигналов. М. : Физматлит, 2004. 320 с.
6. Rabinovich E.V., Vaynmaster P.I. Statistical detection of seismic pulses for hydraulic fracturing monitoring // 11 International forum on strategic technology (IFOST 2016). Novosibirsk, 2016. Pt. 1. P. 375–378.
7. Рабинович Е.В., Вайнмастер П.И., Шефель Г.С. Устранение избыточности данных сейсмического мониторинга гидроразрыва пласта // Математические структуры и моделирование. 2018. № 1(45). С. 81–94.
8. Рабинович Е.В., Вайнмастер П.И. Определение траекторий разрыва и визуализация модели трещины, возникающей при ГРП // Вестник СибГУТИ. 2016. № 4. С. 77–85.
9. Rabinovich E. V., Vaynmaster P.I. Comparison of two algorithms for detection of seismic pulses // The 17 international conference of young specialists on micro/nanotechnologies and electron devices, EDM 2016: proc. Novosibirsk. 2016. P. 71–74.

## **SOLUTION FOR DATA PROCESSING IN SYSTEMS OF MICROSEISMIC MONITORING OF HYDRAULIC FRACTURING**

**P.I. Vaynmaster**

Postgraduate Student, e-mail: wmaster91@mail.ru

Novosibirsk State Technical University, Novosibirsk, Russia

**Abstract.** Performing operations on Hydraulic fracturing (HF) requires control of the fracture growth. A common technology providing such control is microseismic monitoring. It's based on solving problems of location of sources of seismic emission and their visualization on a three-dimensional coordinate grid. The existing methods of HF monitoring have a number of drawbacks, among which is the low accuracy of measuring the geometrical parameters of fractures and the dynamics of their growth. In this article, the original approach to the processing of microseismic monitoring data is described, which allows us to represent the fracturing zone in the form of three-dimensional continuous objects — one or several fracture layers filled with proppant. The proposed algorithms are based on the application of a number of digital signal processing procedures: detection of a pulse on the background of noise and location of its source, spatial filtering of sources, construction of fracture trajectories, interpolation of a set of sources and visualization of a three-dimensional graphic model.

**Keywords:** hydraulic fracturing, fracture, seismic pulse, detection, filtering, trajectory construction, Delaunay triangulation.

*Дата поступления в редакцию: 09.08.2018*

## РАСЧЁТ ПАРАМЕТРОВ СПЕЦИАЛЬНЫХ ТРЁХМЕРНЫХ СТАЦИОНАРНЫХ ПОТОКОВ

**О.В. Опрышко**

аспирант, e-mail: OpryshkoOV@gmail.com

Снежинский физико-технический институт НИЯУ МИФИ, Снежинск, Россия

**Аннотация.** Торнадо и тропические циклоны представляют собой природное явление, которое характеризуется тем, что возникает из обычного, поднимающегося вверх течения воздуха, прогретого солнцем. При определённых условиях такой поток приобретает высокие скорости, вследствие которых происходят множественные разрушения. Чтобы математически смоделировать течение восходящего закрученного потока воздуха в рамках предложенной С.П. Баутиным схемы, используются натурные наблюдения, собранные в шкалу Фудзиты. В шкале имеются данные о ширине полосы разрушения торнадо и скорости ветра, характерные для определённого класса торнадо. В настоящей работе для системы уравнений газовой динамики ставится одна задача Коши с начальными условиями, заданными на непроницаемой горизонтальной плоскости  $z = 0$ , при численно-аналитическом способе решения этой задачи определяются значения газодинамических параметров, в том числе для нахождения кинетической энергии восходящих закрученных потоков. Расчёты проведены для различных классов торнадо и тропического циклона из расширенной таблицы Фудзиты при разных значениях высот рассматриваемых потоков.

**Ключевые слова:** система уравнений газовой динамики, газодинамические параметры, кинетическая энергия потока, шкала Фудзиты.

### Введение

Восходящие закрученные потоки (ВЗП), а именно вихри, смерчи, торнадо возникают при определённых условиях. Эти потоки представляют собой семейство интересных природных явлений, которые имеют характерные особенности: закрутка воздуха и высокие скорости потока. Для существования такой системы необходим постоянный приток энергии. Судя по времени существования вихрей, этот приток энергии обеспечивает сила, которая постоянно присутствует и обладает вращательным моментом — сила Кориолиса, вызванная вращением Земли вокруг своей оси, что обосновано в работах С.П. Баутина [1–4].

Целью данной работы является представление коэффициентов газодинамических параметров для расчёта кинетической энергии ВЗП, действующих в

условиях сил тяжести и Кориолиса, а также расчётов самой кинетической энергии для торнадо различных классов и тропического циклона численно-аналитическим способом при различных значениях высот, рассматриваемых потоков.

## 1. Постановка задачи Коши с начальными данными на непроницаемой горизонтальной плоскости $z = 0$

Баутиным С.П. была предложена и обоснована схема возникновения природных восходящих закрученных потоков [1–4]. В рамках этой схемы для системы уравнений газовой динамики (1) рассматривается одна задача Коши с заданными начальными данными на непроницаемой горизонтальной плоскости  $z = 0$ . Вертикальная составляющая вектора скорости газа полагается равной нулю и поэтому в цилиндрической системе координат система уравнений газовой динамики при учёте действия сил тяжести и Кориолиса имеет следующий вид [1–4]:

$$\left\{ \begin{array}{l} c_t + uc_r + \frac{v}{r}c_\varphi + wc_z + \frac{(\gamma-1)}{2}c \left( u_r + \frac{u}{r} + \frac{v_\varphi}{r} + w_z \right) = 0, \\ u_t + uu_r + \frac{v}{r}u_\varphi - \frac{v^2}{r} + wu_z + \frac{2}{(\gamma-1)}cc_r = av - bw \cos \varphi, \\ v_t + uv_r + \frac{uv}{r} + \frac{v}{r}v_\varphi + wv_z + \frac{2}{(\gamma-1)}\frac{c}{r}c_\varphi = -au + bw \sin \varphi, \\ w_t + ww_r + \frac{v}{r}w_\varphi + ww_z + \frac{2}{(\gamma-1)}cc_z = bu \cos \varphi - bv \sin \varphi - g, \end{array} \right. \quad (1)$$

где  $r$  — полярный радиус,  $\varphi$  — полярный угол в плоскости  $xOy$ ;  $c = \rho^{(\gamma-1)/2}$  — скорость звука газа;  $u, v, w$  — радиальная, окружная и вертикальная составляющие вектора скорости газа соответственно;  $a = 2\Omega \sin \psi$ ;  $b = 2\Omega \cos \psi$ ,  $g = \text{const} > 0$  — ускорение свободного падения;  $\Omega$  — модуль вектора угловой скорости вращения Земли,  $\psi$  — широта точки, в которой находится начало цилиндрической системы координат, вращающейся вместе с Землёй. Для системы (1) при  $z = 0$  ставятся следующие начальные данные:

$$\left\{ \begin{array}{l} c(t, r, \varphi, z)|_{z=0} = c_0(t, r, \varphi), \\ u(t, r, \varphi, z)|_{z=0} = u_0(t, r, \varphi), \\ v(t, r, \varphi, z)|_{z=0} = v_0(t, r, \varphi), \\ w(t, r, \varphi, z)|_{z=0} = 0. \end{array} \right. \quad (2)$$

Последнее из начальных условий (2) обеспечивает условие непротекания газа через плоскость  $z = 0$ .

## 2. Формулы для расчёта газодинамических параметров

Ранее [1–4] установлена разрешимость поставленной задачи в виде бесконечного сходящегося ряда по степеням  $z$ :

$$\mathbf{U}(r, \varphi, z) = \sum_{k=0}^{\infty} \mathbf{U}_k(r, \varphi) \frac{z^k}{k!}; \quad \mathbf{U}_k(r, \varphi) = \left. \frac{\partial^k \mathbf{U}(r, \varphi, z)}{\partial z^k} \right|_{z=0}, \quad (3)$$

где вектор  $\mathbf{U}$  имеет в качестве координат искомые функции  $c, u, v, w$ . В настоящей работе для нахождения значений компонент вектора  $\mathbf{U}$  рассматривается стационарное течение, когда искомые функции зависят от пространственных координат  $r, \varphi, z$ . Из бесконечного ряда учитывается конечное число первых слагаемых:

$$\left\{ \begin{array}{l} c = c_0 + c_1 z + c_2 \frac{z^2}{2} + c_3 \frac{z^3}{6} + c_4 \frac{z^4}{24}; \\ u = u_0 + u_1 z + u_2 \frac{z^2}{2} + u_3 \frac{z^3}{6}; \\ v = v_0 + v_1 z + v_2 \frac{z^2}{2} + v_3 \frac{z^3}{6}; \\ w = w_0 + w_1 z + w_2 \frac{z^2}{2} + w_3 \frac{z^3}{6} + w_4 \frac{z^4}{24}, \end{array} \right. \quad (4)$$

при этом

$$w_0 = w_1 = 0.$$

Для нахождения коэффициентов  $c_0, u_0, v_0$  в системе (1) полагается  $z = 0$ , а также учитываются начальные условия (2). Таким образом, для коэффициентов  $c_0(r), u_0(r)$  получается система обыкновенных дифференциальных уравнений следующего вида:

$$\left\{ \begin{array}{l} c'_0 = -\frac{(\gamma-1)}{2} c_0 \frac{\left[ u_0^2 + \frac{a^2(r_{in}^4 - r^4)}{4r^2} \right]}{r(u_0^2 - c_0^2)}, \\ u'_0 = u_0 \frac{\left[ c_0^2 + \frac{a^2(r_{in}^4 - r^4)}{4r^2} \right]}{r(u_0^2 - c_0^2)}, \\ c_0(r) |_{r=r_{in}} = 1, \\ u_0(r) |_{r=r_{in}} = u_{in}, \end{array} \right. \quad (5)$$

а коэффициент  $v_0(r)$  определяется в явном виде

$$v_0 = -\frac{a(r_{in}^2 - r^2)}{2r}, \quad v_0(r_{in}) = 0.$$



Следующие коэффициенты  $c_1, u_1, v_1$  ряда (3) находятся с помощью дифференцирования по переменной  $z$  системы (1), подстановки в результат дифференцирования  $z = 0$  и уже найденных коэффициентов. С учётом возможности частичного разделения переменных получается новая СОДУ для  $c_1, u_1, v_1$ .

Подробности получения разложения коэффициентов (4) представлены в работе [5]. С использованием формул из работ [4, 5] далее представлены расчёты для нахождения коэффициентов с индексом один:

$$\begin{aligned} c(r, \varphi, z) &= c_0(r) + c_1(r, \varphi)z, \\ c_1(r, \varphi, z) &= c_{10}(r) + c_{11}(r)\cos(\varphi) + c_{12}(r)\sin(\varphi); \\ c_{10}(r) &= -g\frac{(\gamma-1)}{2}\frac{1}{c_0(r)}; \quad c_{11}(r) = b\frac{(\gamma-1)u_0(r)}{2c_0(r)}; \quad c_{12}(r) = -b\frac{(\gamma-1)v_0(r)}{2c_0(r)}; \\ u(r, \varphi, z) &= u_0(r) + u_1(r, \varphi)z, \\ v(r, \varphi, z) &= v_0(r) + v_1(r, \varphi)z. \end{aligned}$$

С помощью разделения переменных коэффициенты  $u_1, v_1$  можно представить в следующем виде:

$$\begin{aligned} u_1(r, \varphi, z) &= u_{10}(r) + u_{11}(r)\cos(\varphi) + u_{12}(r)\sin(\varphi), \\ v_1(r, \varphi, z) &= v_{10}(r) + v_{11}(r)\cos(\varphi) + v_{12}(r)\sin(\varphi), \\ u_{10} &= v_{10} = 0. \end{aligned}$$

Коэффициенты  $u_{11}(r), u_{12}(r), v_{11}(r), v_{12}(r)$  будем находить по формулам из [4, 5]:

$$u_0 u_{1ir} = F_{u_i}; \quad u_0 v_{1ir} = F_{v_i}; \quad i = 1, 2,$$

где

$$\begin{aligned} F_{u_{11}} &= -\left[u_{0r}u_{11} + \frac{v_0}{r}u_{12} - \frac{2v_0}{r}v_{11} + bu_{0r} - av_{11}\right]; \\ F_{u_{12}} &= -\left[u_{0r}u_{12} - \frac{v_0}{r}u_{11} - \frac{2v_0}{r}v_{12} - bu_{0r} - av_{12}\right]; \\ F_{v_{11}} &= -\left[v_{0r}u_{11} + \frac{v_0}{r}u_{11} + \frac{u_0}{r}v_{11} + \frac{v_0}{r}v_{12} - b\frac{v_0}{r} + au_{11}\right]; \\ F_{v_{12}} &= -\left[v_{0r}u_{12} + \frac{v_0}{r}u_{12} + \frac{u_0}{r}v_{12} - \frac{v_0}{r}v_{11} - b\frac{u_0}{r} + au_{12}\right]. \end{aligned}$$

### 3. Представление основных газодинамических параметров с учётом нулевых коэффициентов для торнадо класса F40

В данном пункте представлены результаты расчёта основных газодинамических параметров:  $c_0, u_0, v_0$  для торнадо класса F40. В дальнейшем будем называть расчёт с нулевыми коэффициентами базовым расчётом. Основные характеристики этого класса взяты из шкалы Фудзиты [6]. В табл. 1 представлены

значения радиуса стока торнадо и вектора скорости ветра, который состоит из двух компонент: окружной и радиальной  $|\vec{V}| = \sqrt{u^2 + v^2}$ .

Таблица 1. Данные из шкалы Фудзиты

Класс торнадо	$r_0$ , радиус стока, м	$ \vec{V} $ , скорость ветра, м/с
F40	273.5	93

Расчёт газодинамических параметров реализован с помощью численного метода Рунге-Кутты четвёртого порядка точности. Для расчётов использовалась пространственная сетка в области  $r \in [r_0, r_{in}]$  с шагом  $h = \frac{(r_{in}-r_0)}{n}$ , где  $n = 10^6$  — число интервалов. Выбранный шаг обеспечивает требуемую точность полученных результатов. При решении СОДУ (5) получаются следующие решения для

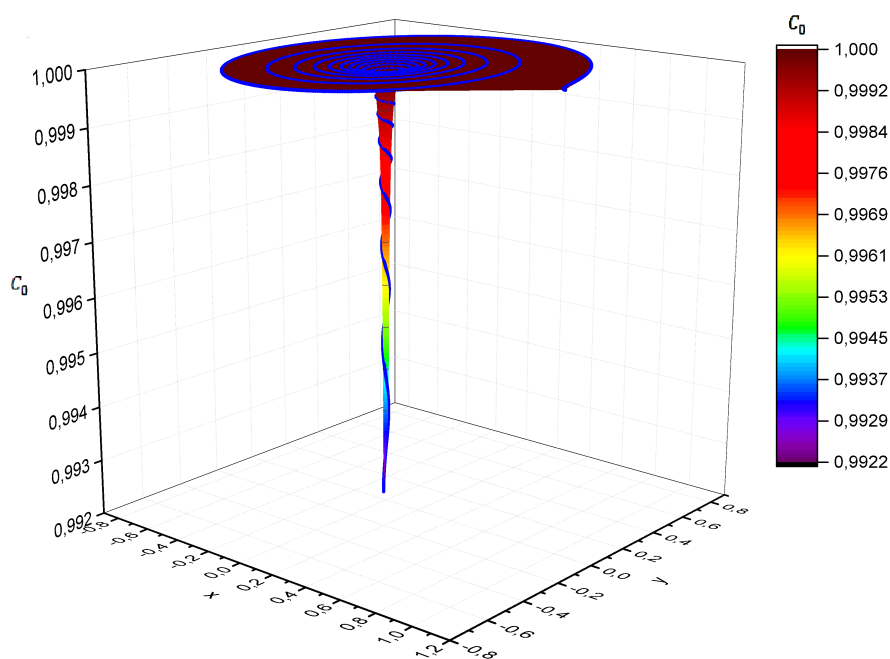


Рис. 1. Скорость звука газа  $c_0$  и траектория движения частицы для торнадо класса F40

$c_0, u_0, v_0$ : на рис. 1–3 представлены поверхности газодинамических распределений:  $c_0$  — скорости звука газа,  $u_0$  — радиальной и  $v_0$  — окружной компонент вектора скорости в декартовых координатах  $xOy$  с выделенными траекториями движения частиц газа.

Значение скорости звука  $c_0$  убывает при приближении к радиусу стока  $r \rightarrow r_0$ , а значит, убывают значения плотности и давления газа. Скорость звука меняется от единичного безразмерного значения (333 м/с — размерное значение) до 0.992 (330.391 м/с — размерное значение), что соответствует

изменению на 0.78 %. Плотность — от единичного значения до 0.961, что соответствует изменению на 4.86 %.

Таким образом, подтверждается то, что в центре ВЗП находится область пониженного давления газа.

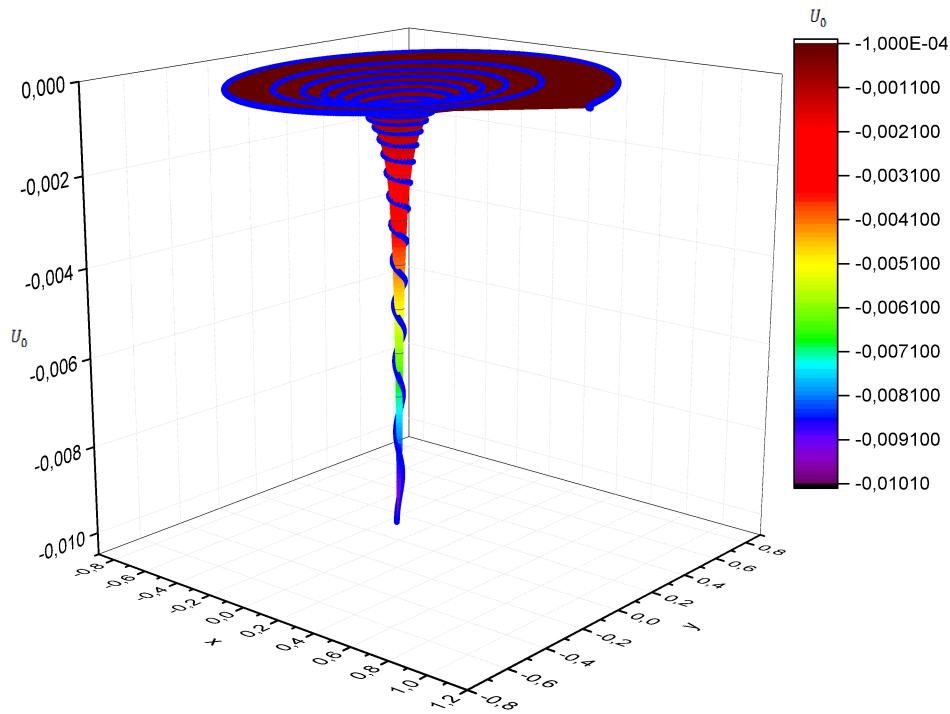


Рис. 2. Радиальная скорость ветра  $u_0$  и траектория движения частицы для торнадо класса F40

Радиальная составляющая вектора скорости газа (рис. 2) имеет отрицательный знак. Основные изменения для радиальной компоненты скорости происходят на отрезке  $[r_0, 0.1]$  от значения  $u \approx -0.0101$  до  $u \approx -0.001$ , что соответствует размерным значениям  $-3.34961$  м/с и  $-0.3332$  м/с.

Таким образом, наличие области пониженного давления ускоряет стремление газа в центр ВЗП.

Окружная составляющая вектора скорости газа (рис. 3) возрастает до безразмерного значения 0.28, что соответствует размерному значению 92.972 м/с для торнадо класса F40. Положительное значение окружной компоненты говорит о том, что закрутка газа направлена против часовой стрелки. В следствие большого отношения окружной компоненты к радиальной, которая составляет примерно 30 раз, наблюдается, что частица при движении к центру успевает совершить около 17–18 полных оборотов вокруг стока торнадо.

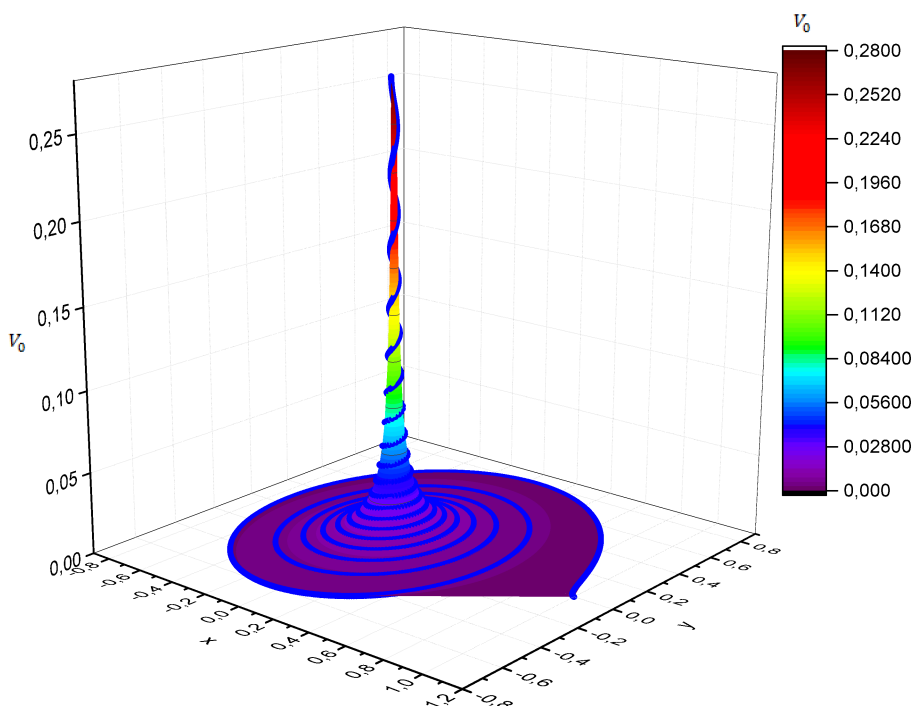


Рис. 3. Окружная скорость ветра  $v_0$  и траектория движения частицы для торнадо класса F40

#### 4. Представление основных газодинамических параметров с учётом нулевых и первых коэффициентов для торнадо класса F40

На рисунках 1–3 изображены поверхности скорости звука газа, окружной и радиальной компонент вектора скорости газа и траектории движения частицы, находящиеся с помощью построения решения задачи Коши [3]:

$$\frac{d\varphi}{dr} = \frac{v_0}{ru_0}; \quad \varphi(r;n) = \varphi_0; \quad \varphi_0 = const.$$

Полученные траектории — линии тока, по которым движутся частицы. На рисунках 4–5 изображены четыре траектории частицы, исходящие из углов  $\frac{\pi}{2}$ ,  $\pi$ ,  $\frac{3\pi}{2}$  и  $2\pi$ . Такие линии тока в многократном количественном увеличении образуют так называемые струи, которые можно увидеть, например, на воде при торнадо над морем, как на фотографии [7] на рисунке 7.

На рисунке 4 показаны линии тока в квадратной области  $[-1; 1] \times [-1; 1]$ . На рисунке 5 представлены трёхмерные линии тока, распределение скорости звука вдоль третьего направления. На рисунке 6 изображён фрагмент линий тока вблизи радиуса стока торнадо в квадрате  $[-0.1; 0.1] \times [-0.1; 0.1]$ .

Для нахождения газодинамических параметров с учётом первого индекса

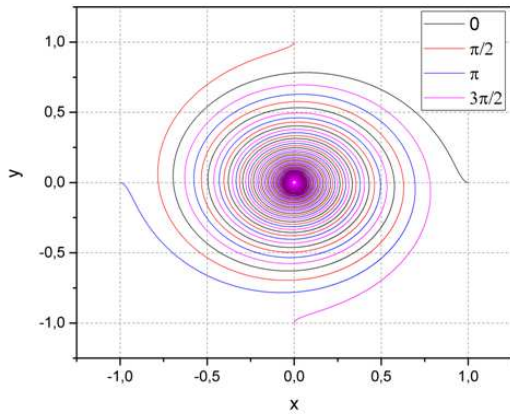


Рис. 4. Линии тока для торнадо класса F40

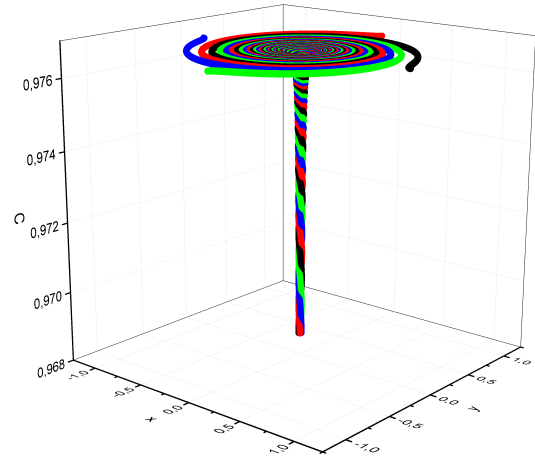


Рис. 5. Линии тока для торнадо класса F40

из бесконечного ряда (3) будем учитывать два слагаемых: с нулевым и первым индексами. Ниже представлены сравнительные расчёты для  $c$ ,  $u$ ,  $v$ , входящие в систему уравнений (4).

Таблица 2. Скорость звука газа

$c_0$ безразм.	$c_0$ разм.	$c_0 + c_1 z$ безразм., 10м	$c_0 + c_1 z$ разм., 10м	$c_0 + c_1 z$ безразм., 50м	$c_0 + c_1 z$ разм., 50м
0.99216	<b>330.38928</b>	0.9919862	<b>330.3314</b>	0.991274	<b>330.09424</b>

В таблице 2 показаны расчёты скорости звука газа, её размерное и безразмерное значения для торнадо класса F40 с безразмерным значением радиуса стока  $r_0 = 0.01034$ . В таблицах 3, 4 — радиальная и окружная компоненты вектора скорости ветра соответственно.

На рисунках 8–9 представлены траектории движения частицы и скорость звука газа для торнадо класса F40. Верхняя линия — график скорости звука газа (базовый расчёт) соответствует значению 330.39 м/с. Средняя линия — график скорости звука газа с учётом коэффициента с индексом один:  $c_0 + c_1 z$ , при значении  $z = 10$  м. Безразмерное значение величины  $z = 0.000378$ . Скорость звука газа достигает 330.33 м/с. Изменение относительно базового расчёта на 0.058 м/с. Нижняя линия — график скорости звука газа с учётом коэффициента с индексом один:  $c_0 + c_1 z$  при  $z = 50$  м соответствует значению 330.094 м/с, что даёт уменьшение относительно базового значения на 0.295 м/с.

С учётом дополнительных слагаемых картина течения потока воздуха существенно не меняется, но с увеличением высоты пропорционально увеличивается вклад первого коэффициента для скорости звука газа.

На рисунках 10, 11 представлены траектории движения частицы и радиальная составляющая вектора скорости газа для торнадо класса F40. В точке  $r_0$  радиальная компонента скорости ветра достигает 3.35 м/с по модулю. При

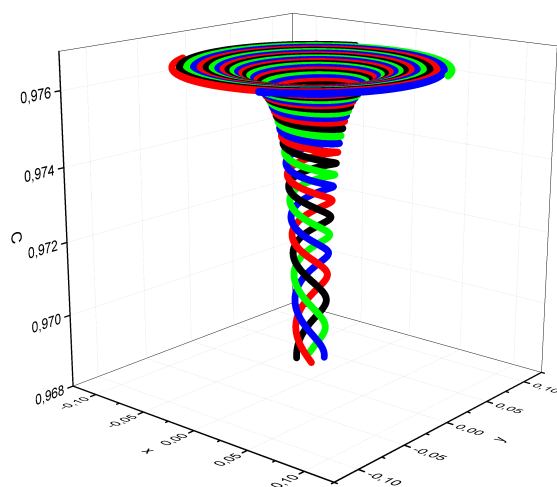


Рис. 6. Линии тока для торнадо класса F40  
вблизи радиуса стока



Рис. 7. Торнадо над морем

Таблица 3. Радиальная скорость ветра

$u_0$ безразм.	$u_0$ разм.	$u_0 + u_1z$ безразм., 10м	$u_0 + u_1z$ разм., 10м	$u_0 + u_1z$ безразм., 50м	$u_0 + u_1z$ разм., 50м
-0.0100589	<b>-3.34961</b>	-0.00993151	<b>-3.30719</b>	-0.009422	<b>-3.13753</b>

$z = 10$  м радиальная компонента скорости ветра с учётом первого коэффициента равна 3.307 м/с. Разница относительно базового расчёта равна 0.042 м/с (0.013 %).

Большее изменение даёт расчёт радиальной компоненты скорости при  $z = 50$  м. Радиальная компонента скорости ветра достигает 3.138 м/с — изменение на 0.212 м/с (0,063%) относительно базового расчёта.

Таким образом, с увеличением высоты пропорционально увеличивается вклад первого коэффициента для радиальной компоненты вектора скорости.

Таблица 4. Окружная скорость ветра

$v_0$ безразм.	$v_0$ разм.	$v_0 + v_1z$ безразм., 10м	$v_0 + v_1z$ разм., 10м	$v_0 + v_1z$ безразм., 50м	$v_0 + v_1z$ разм., 50м
0.2791951	<b>92.97197</b>	0.2791964	<b>92.97240</b>	0.279202	<b>92.9727</b>

На рисунках 12–13 представлены траектории движения частицы и окружная составляющая вектора скорости газа для торнадо класса F40. Окружная компонента скорости ветра достигает 92.97197 м/с. При  $z = 10$  метров окружная компонента соответствует значению 92.9724 м/с. Изменение относительно базового расчёта составляет 0.00043 м/с.

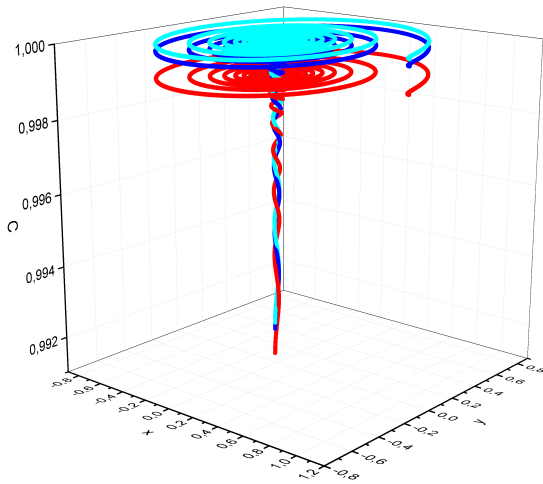


Рис. 8. Торнадо класса F40:  $c_0$ ;  $c_0 + c_1 z$ , высота 10 м;  $c_0 + c_1 z$ , высота 50 м.

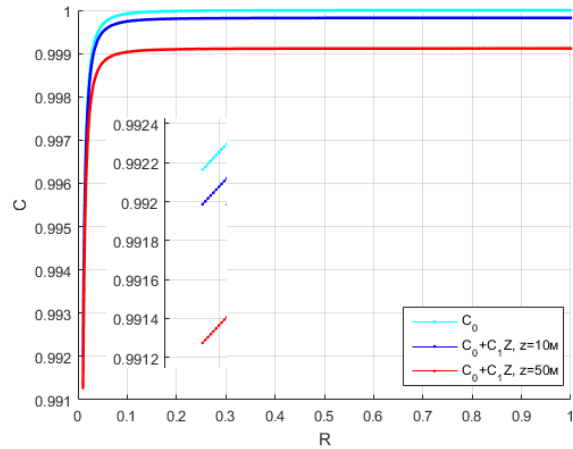


Рис. 9. Торнадо класса F40:  $c_0$ ;  $c_0 + c_1 z$ , высота 10 м;  $c_0 + c_1 z$ , высота 50 м.

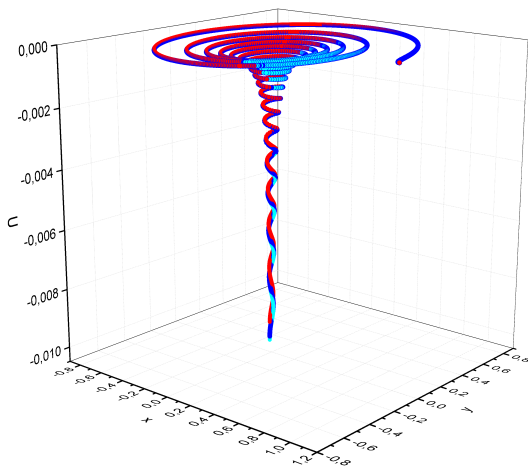


Рис. 10. Торнадо класса F40:  $u_0$ ;  $u_0 + u_1 z$ , высота 10 м;  $u_0 + u_1 z$ , высота 50.

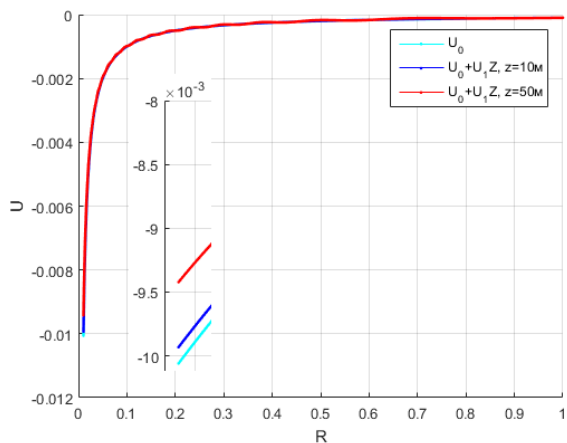


Рис. 11. Торнадо класса F40:  $u_0$ ;  $u_0 + u_1 z$ , высота 10 м;  $u_0 + u_1 z$ , высота 50.

При  $z = 50$  метров окружная скорость ветра достигает 92.9727 м/с. Изменение относительно базового расчёта составляет 0.00073. Учёт коэффициента с первым индексом для окружной компоненты скорости не даёт значимых изменений в приведённых расчётах.

При добавлении коэффициентов с первыми индексами наблюдается значимость полученного базового решения.

## 5. Результаты расчётов кинетической энергии ВЗП

Кинетическая энергия газа, движущегося в области

$$(D): \{r_0 \leq r \leq r_{in}; 0 \leq \varphi \leq 2\pi; 0 \leq z \leq h\},$$

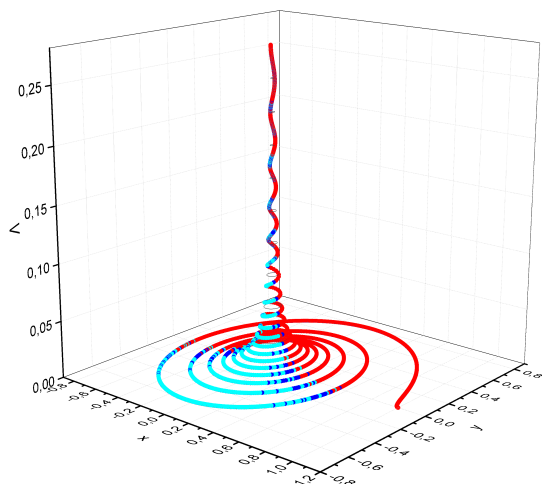


Рис. 12. Торнадо класса F40:  $v_0$ ;  $v_0 + v_1z$ , высота 10 м;  $v_0 + v_1z$ , высота 50

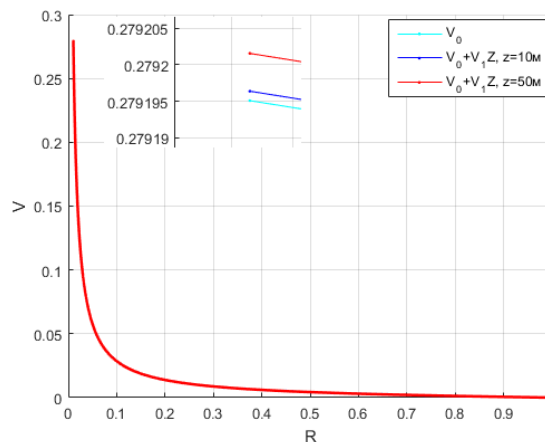


Рис. 13. Торнадо класса F40:  $v_0$ ;  $v_0 + v_1z$ , высота 10 м;  $v_0 + v_1z$ , высота 50

задаётся тройным интегралом

$$W = \frac{1}{2} \iiint_{(D)} \rho(x, y, z) \vec{V}^2 dx dy dz =$$

$$= \frac{1}{2} \int_0^{z_0} \left\{ \int_0^{2\pi} \left[ \int_{r_0}^{r_{in}} c^{\frac{2}{(\gamma-1)}}(r, \varphi, z) (u^2(r, \varphi, z) + v^2(r, \varphi, z) + w^2(r, \varphi, z)) r dr \right] d\varphi \right\} dz,$$

где  $\vec{V}$  – вектор скорости газа.

В работе [8] представлен полный расчёт кинетической энергии. Вычисление кинетической энергии ведётся по первым двум слагаемым для каждого параметра из (4): скорости звука газа, окружной и радиальной компонент вектора скорости газа. Вертикальная составляющая вектора скорости газа для нулевых и первых коэффициентов равна нулю[2].

Далее представлены таблицы расчёта кинетической энергии для базового расчёта для всех классов торнадо и тропического циклона (ТЦ) средней интенсивности из расширенной таблицы Фудзиты, для базового расчёта (табл. 5), расчётов с учётом первых коэффициентов при значении  $z = 10$  м (табл. 6),  $z = 50$  м (табл. 7),  $z = 100$  м (табл. 8).

При сравнении таблиц 5 и 6 отметим, что для классов торнадо от F00 до F30 включительно значения радиальной компоненты для расчёта с учётом коэффициентов с первым индексом при  $z = 10$  м практически совпадают с базовым. Для классов F31–ТЦ увеличение радиальной составляющей кинетической энергии приближается к двукратному, хотя окружная компонента для соответствующих классов остаётся практически неизменной.

Сравнивая таблицы 6 и 7, стоит отметить, что для радиальной и окружной составляющих кинетической энергии расчёты при  $z = 50$  м показывают пяти-



Таблица 5. Кинетическая энергия для базового расчёта для широты  $\pi/6$ 

Класс торнадо	$W_U$ МДж	$W_V$ МДж	$W$ МДж	$W_V/W$	$W$ кТн
F00	0.256	0.254	0.510	0.498	1.219E-07
F01	0.822	2.658	3.480	0.764	8.317E-07
F10	1.787	12.681	14.468	0.877	3.458E-06
F11	4.615	87.459	92.073	0.949	2.201E-05
F20	8.690	315.003	323.693	0.973	7.737E-05
F21	20.399	1809.309	1829.708	0.989	4.373E-04
F30	35.722	5654.6867	5690.408	0.994	1.360E-03
F31	80.491	30083.811	30164.303	0.997	7.209E-03
F40	144.668	99875.772	100020.44	0.999	2.391E-02
F41	281.074	391926.894	392207.968	0.99928	9.374E-02
F50	487.273	1211412.527	1211899.800	0.9996	2.897E-01
F51	1026.322	5658006.139	5659032.460	0.99982	1.353E+00
F52	1635.996	14722518.097	14724154.093	0.99989	3.519E+00
ТЦ	720.468	3417114.935	3417835.403	0.99979	8.169E-01

Таблица 6. Кинетическая энергия для основных газодинамических параметров с учётом первого коэффициента при  $z = 10$  м,  $\psi = \pi/6$ 

Класс торнадо	$W_U$ МДж	$W_V$ МДж	$W$ МДж	$W_V/W$	$W$ кТн
F00	0.256	0.254	0.510	0.498	1.219E-07
F01	0.823	2.657	3.479	0.764	8.315E-07
F10	1.789	12.676	14.465	0.876	3.457E-06
F11	4.633	87.421	92.054	0.950	2.200E-05
F20	8.760	314.866	323.626	0.973	7.735E-05
F21	20.808	1808.514	1829.322	0.989	4.372E-04
F30	36.996	5652.194	5689.189	0.994	1.360E-03
F31	87.102	30070.520	30157.622	0.997	7.208E-03
F40	165.808	99831.592	99997.399	0.998	2.390E-02
F41	358.881	391753.338	392112.219	0.999	9.372E-02
F50	709.149	1210875.533	1211584.681	0.9994	2.896E-01
F51	1921.797	5655494.773	5657416.570	0.9996	1.352E+00
F52	3689.007	14715974.991	14719663.997	0.9997	3.518E+00
ТЦ	1395.974	3415603.866	3416999.839	0.99959	8.167E-01

Таблица 7. Кинетическая энергия для основных газодинамических параметров с учётом первого коэффициента при  $z = 50$  м,  $\psi = \pi/6$

Класс торнадо	$W_U$ МДж	$W_V$ МДж	$W$ МДж	$W_V/W$	$W$ кТн
F00	1.284	1.274	2.558	0.498	6.114E-07
F01	4.177	13.281	17.458	0.761	4.173E-06
F10	9.280	63.316	72.595	0.872	1.735E-05
F11	25.570	436.462	462.032	0.945	1.104E-04
F20	52.521	1571.790	1624.311	0.968	3.882E-04
F21	153.935	9027.162	9181.097	0.983	2.194E-03
F30	338.974	28212.008	28550.982	0.988	6.824E-03
F31	1230.185	150088.904	151319.089	0.992	3.617E-02
F40	3365.204	498278.564	501643.769	0.993	1.199E-01
F41	11117.974	1955305.447	1966423.421	0.994	4.699E-01
F50	30118.988	6043659.774	6073778.762	0.995	1.452E+00
F51	116815.281	28227323.352	28344138.634	0.996	6.774E+00
F52	264199.763	73449193.050	73713392.812	0.996	1.762E+01
ТЦ	87848.768	17047849.155	17135697.923	0.995	4.096E+00

кратное увеличение обеих составляющих кинетической энергии относительно расчёта при  $z = 10$  м.

И двукратное увеличение обеих компонент кинетической энергии для расчёта при  $z = 100$  м относительно расчёта при  $z = 50$  м.

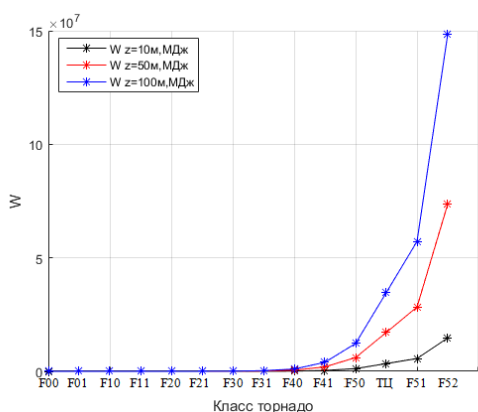


Рис. 14. Общая кинетическая энергия торнадо с учётом высоты

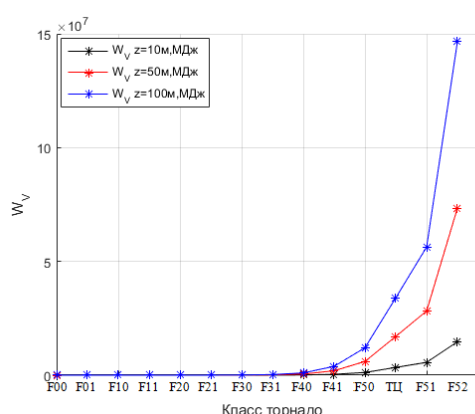


Рис. 15. Окружная кинетическая энергия торнадо с учётом высоты

На рисунках 14 – 17 изображены графики распределения кинетической энергии в соответствии с классами торнадо. На рис. 14 представлены значения

Таблица 8. Кинетическая энергия для основных газодинамических параметров с учётом первого коэффициента при  $z = 100$  м,  $\psi = \pi/6$ 

Класс торнадо	$W_U$ МДж	$W_V$ МДж	$W$ МДж	$W_V/W$	$W$ кТн
F00	2.603	2.581	5.185	0.498	1.239E-06
F01	8.785	26.637	35.422	0.752	8.466E-06
F10	20.704	126.652	147.356	0.860	3.522E-05
F11	66.255	871.794	938.049	0.929	2.242E-04
F20	159.606	3138.155	3297.761	0.952	7.882E-04
F21	619.033	18018.030	18637.063	0.967	4.454E-03
F30	1637.460	56305.596	57943.056	0.972	1.385E-02
F31	7408.369	299528.158	306936.527	0.976	7.336E-02
F40	22518.041	994377.498	1016895.538	0.978	2.430E-01
F41	80266.929	3902002.114	3982269.043	0.980	9.518E-01
F50	225622.667	12060623.514	12286246.181	0.982	2.937E+00
F51	900822.570	56329615.159	57230437.728	0.984	1.368E+01
F52	2057815.453	146572490.751	148630306.205	0.986	3.552E+01
ТЦ	678977.161	34020502.662	34699479.822	0.980	8.293E+00

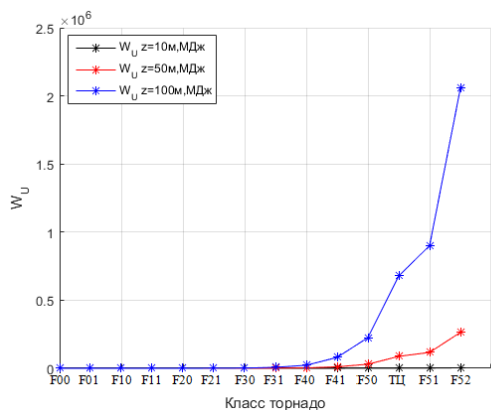


Рис. 16. Радиальная кинетическая энергия торнадо с учётом высоты

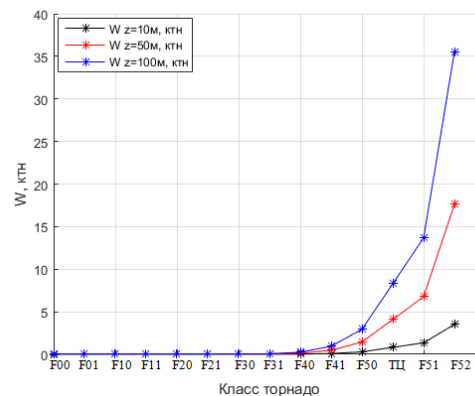


Рис. 17. Общая кинетическая энергия торнадо с учётом высоты в килотоннах

общей кинетической энергии в МДж, в килотоннах (рис. 17). По оси абсцисс расположены классы торнадо, по оси ординат — значение кинетической энергии: окружной составляющей (рис. 15), радиальной составляющей (рис. 16).

## 6. Выводы

- Основные изменения газодинамических параметров происходят вблизи радиуса стока  $r_0$ .
- С увеличением класса торнадо происходит увеличение кинетической энергии.
- С увеличением класса торнадо увеличивается значение как окружной компоненты кинетической энергии, так и радиальной её составляющей.
- Окружная кинетическая энергия составляет большую часть от общей кинетической энергии.
- При вычислении основных газодинамических параметров коэффициенты с первым индексом при  $z = 10$  м не дают значимых изменений относительно базового расчёта.
- При расчёте кинетической энергии виден значимый вклад, вносимый коэффициентами с первым индексом при сравнении расчётов при  $z = 50$  и  $z = 100$  м с базовым.

## Благодарности

Благодарю профессора С.П. Баутина и научного руководителя И.Ю. Крутова за полезные обсуждения.

## ЛИТЕРАТУРА

1. Баутин С.П. Торнадо и сила Кориолиса. Новосибирск : Наука, 2008. 96 с.
2. Баутин С.П., Обухов А.Г. Математическое моделирование разрушительных атмосферных вихрей. Новосибирск : Наука, 2012. 152 с.
3. Баутин С.П., Крутова И.Ю., Обухов А.Г., Баутин К.В. Разрушительные атмосферные вихри: теоремы, расчёты, эксперименты. Новосибирск: Наука, 2013. 216 с.
4. Баутин С.П., Дерябин С.Л., Крутова И.Ю., Обухов А.Г. Разрушительные атмосферные вихри и вращение Земли вокруг своей оси. Екатеринбург : УрГУПС, 2017. 355 с.
5. Казачинский А.О., Крутова И.Ю. Построение течений в придонной части восходящих закрученных потоков как решение одной характеристической задачи Коши. Препринт. Снежинск : издательство СФТИ НИЯУ МИФИ, 2016. 60 с.
6. Tatom F.B., Witton S.J. The transfer of energy from tornado into the ground Seismological // Research Letter. 2001. V. 72, no. 1. P. 12.
7. Факты о торнадо. Торнадо в море. URL: <https://yandex.ru/collections/card/58a5f2c1b1ab9dee05c144fa/> (дата обращения: 01.06.2018).
8. Крутова И.Ю., Опрышко О.В. Расчёт кинетической энергии течений в придонной части торнадо и тропического циклона. Препринт. Снежинск, СФТИ НИЯУ МИФИ, 2018. 45 с.

**CALCULATION OF THE PARAMETERS OF SPECIAL THREE-DIMENSIONAL STATIONARY FLOWS****O.V. Opryshko**

Postgraduate Student, e-mail: OpryshkoOV@gmail.com

Snezhinsk Physicotechnical Institute NRSU MEFhI, Snezhinsk, Russia

**Abstract.** Tornadoes and tropical cyclones are a natural phenomenon that is characterized by what arises from the ordinary, rising upward flow of air warmed by the sun. Under certain conditions, such a flow acquires high speeds, as a result of which, multiple disruptions occur. The original scheme of initiation of the natural ascending swirling flows in the form of tornadoes or tropical cyclones was put forward and substantiated by S. P. Bautin. The mathematic simulation of gas flow in a bottom part of a tornado is effectuated due to Fujita scale and enhanced Fujita scale which comprises systematized data of in situ observations of tornadoes of various intensity, namely, values of the width of damage path and maximum wind speed. For the system of equations of gas dynamics, one Cauchy problem is posed with the initial conditions given on an impenetrable horizontal plane  $z = 0$ . In the numerically-analytical method for solving this problem, the gas dynamic parameters are determined, including the kinetic energy of ascending swirling flows. Calculations were carried out for various classes of tornadoes and tropical cyclones from the enhanced Fujita scale for different heights of the considered flows.

**Keywords:** system of equations of gas dynamics, gas dynamic parameters, Fujita scale, kinetic energy of the flow.

*Дата поступления в редакцию: 07.06.2018*

## **АВТОКОРРЕЛЯЦИОННАЯ НЕЙРОННАЯ СЕТЬ ФАКТОРНОГО АНАЛИЗА**

**В.А. Шовин**

научный сотрудник, e-mail: v.shovin@mail.ru

Федеральное государственное бюджетное учреждение науки Институт математики  
им. С.Л. Соболева Сибирского отделения РАН (Омский филиал), Россия

**Аннотация.** Разработан алгоритм проведения факторного анализа на базе автокорреляционной нейронной сети. Нейронная сеть данного типа обладает способностью автокорреляции входного и выходного сигнала. В этой нейронной сети для осуществления обратного распространения ошибки от максимума корреляции входного и выходного сигнала добавляется дополнительный слой нейронов с весовыми коэффициентами равными значениям входного сигнала. Максимизация корреляции входного и выходного сигнала приводит к реализации факторного анализа и вычисления главных компонент в случае меньшего числа нейронов на выходном слое, чем на входном слое нейронной сети. Тестовые вычисления главных компонент по данному алгоритму на данных артериальной гипертензии показали идентичность значимых факторных нагрузок и факторных структур с другими методами поиска главных компонент.

**Ключевые слова:** факторный анализ, нейронная сеть, метод обратного распространения ошибки.

### **Введение**

Математические нейронные сети являются моделью реальных нейронных сетей. Такие сети известны своей автоматической способностью к обучению. Наиболее известной структурой математической нейронной сети является перцептрон Розенблатта. В данной структуре нейронной сети имеется несколько слоев нейронов, соединённых между собой связями с весовыми коэффициентами. Отдельные нейроны являются сумматорами входных сигналов, умноженных на весовые коэффициенты связей, пропущенными через передаточные функции нейронов. Такие передаточные функции приводят все суммы сигналов к диапазону значений  $[-1, 1]$  и могут вносить нелинейные монотонные искажения.

Главным принципом автокорреляционных нейронных сетей является максимизация значения автокорреляционной функции выходных сигналов данной нейронной сети и каких-либо других сигналов, например выходных сигналов другой сети. В данной работе предлагается максимизировать значение автокорреляционной функции выходных и входных сигналов нейронной сети при

меньшем числе нейронов выходного слоя, чем входного. Тем самым такая нейронная сеть осуществляет факторный анализ и поиск главных компонент.

Максимум корреляций значений главных компонент со значениями исходных переменных приводит к максимизации значений факторных нагрузок факторной структуры. Значения главных компонент в данной нейро-сетевой модели являются суммами исходных переменных, пропущенными через передаточные функции нейронов выходного слоя. Данная математическая модель является схожей с моделью факторного анализа, когда исходные переменные являются линейными комбинациями значений главных компонент. В случае линейной передаточной функции получаем идентичность математических структур моделей. Минимизация невязок модели факторного анализа в свою очередь приводит к максимизации автокорреляционной функции между значениями исходных переменных и главных компонент в виду фиксированных значений корреляций между исходными переменными, значений самих исходных переменных и оптимальных значений факторных нагрузок. Эти два критерия поиска главных компонент являются балансирующими между собой.

## 1. Модель факторного анализа

В теории структурных уравнений используются следующие типы матриц.

На значения параметров и значений латентных переменных могут накладываться дополнительные условия в виде равенств и неравенств.

Оптимальными значениями параметров и латентных переменных считаются те значения, которые минимизируют абсолютные значения невязок.

Линейная факторная модель описывается следующими уравнениями [1, 2]:

$$\begin{cases} z_{1t} = a_{11}p_{1t} + a_{12}p_{2t} + \dots + a_{1g}p_{gt} + \varepsilon_{1t}, \\ z_{2t} = a_{21}p_{1t} + a_{22}p_{2t} + \dots + a_{2g}p_{gt} + \varepsilon_{2t}, \\ \vdots \\ z_{mt} = a_{m1}p_{1t} + a_{m2}p_{2t} + \dots + a_{mg}p_{gt} + \varepsilon_{mt}, \end{cases} \quad (1)$$

где матрица  $Z \leftrightarrow z_{ij}$  — матрица значений измеряемых переменных у исследуемых объектов или состояний объекта размерности  $m \times n$ , где  $m$  — число измеряемых параметров,  $n$  — число объектов или состояний объекта (объём выборки).

Матрица  $P \leftrightarrow p_{ij}$  — матрица значений латентных переменных объектов размерности  $g \times n$ , где  $g$  — число латентных параметров.

Матрица  $A \leftrightarrow a_{ij}$  называется матрицей факторной структуры размерности  $m \times g$  весовых коэффициентов. Где  $m$  — число изучаемых параметров,  $g$  — число общих факторов;  $\varepsilon_{1t}, \varepsilon_{2t}, \dots, \varepsilon_{kt}$  — невязки модели для  $t$ -го объекта или состояния объекта.

Критерий оптимизации задаётся в следующем виде:

$$K = \sum_{t=1}^n \sum_{k=1}^m |\varepsilon_{kt}|. \quad (2)$$

Минимизация критерия  $K$  приводит к оптимальному решению для варьируемых значений элементов факторной структуры  $a_{ij}$  и факторов  $p_{ij}$ .

**Критерий автокорреляционной функции.** Рассмотрим  $i$ -ое уравнение факторной модели:

$$z_{it} = \sum_{j=1}^g a_{ij} p_{jt} + \varepsilon_{it}.$$

Умножим левую и правую часть уравнения на  $z_{st}$ , возьмём модуль и просуммируем по  $s$ :

$$|z_{it}| |z_{st}| \leq \left| \sum_{j=1}^g a_{ij} z_{st} p_{jt} \right| + |\varepsilon_{it}| |z_{st}|$$

$$\sum_{s=1}^m |z_{it}| |z_{st}| \leq \left| \sum_{s=1}^m \sum_{j=1}^g a_{ij} z_{st} p_{jt} \right| + \sum_{s=1}^m |\varepsilon_{it}| |z_{st}|.$$

Поскольку слагаемое  $\sum_{s=1}^m |z_{it}| |z_{st}| = const$ ,  $|z_{st}| = const$  и  $a_{ij} = const$ , т. к. являются оптимальными. Тогда минимизация  $|\varepsilon_{it}|$  приводит к максимизации  $\left| \sum_{s=1}^m \sum_{j=1}^g z_{st} p_{jt} \right|$ .

Критерий максимизации автокорреляционной функции  $F = \left| \sum_{s=1}^m \sum_{j=1}^g z_{st} p_{jt} \right| \rightarrow \max$  является балансирующим с критерием минимизации невязок  $\sum_{t=1}^n \sum_{i=1}^m |\varepsilon_{it}| \rightarrow \min$ .

Таким образом, можно сформулировать следующую теорему.

**Теорема 1.** Для линейной факторной модели (1) критерий максимизации автокорреляционной функции между исходными переменными и факторами  $F = \left| \sum_{s=1}^m \sum_{j=1}^g z_{st} p_{jt} \right| \rightarrow \max$  для каждого  $t$  является балансирующим с критерием минимизации невязок модели  $\sum_{t=1}^n \sum_{i=1}^m |\varepsilon_{it}| \rightarrow \min$ , т. е. позволяет определить неизвестные параметры модели.

## 2. Метод построения факторной модели

Обратной моделью факторного анализа является выражение факторов через линейные комбинации исходных переменных:

$$\begin{cases} p_{1t} = b_{11} z_{1t} + b_{12} z_{2t} + \dots + b_{1m} z_{mt} + \varepsilon_{1t}, \\ p_{2t} = b_{21} z_{1t} + b_{22} z_{2t} + \dots + b_{2m} z_{mt} + \varepsilon_{2t}, \\ \vdots \\ p_{gt} = b_{g1} z_{1t} + b_{g2} z_{2t} + \dots + b_{gm} z_{mt} + \varepsilon_{gt}. \end{cases}$$



Данную структурную модель можно построить на нейронной сети следующей конфигурации (рис. 1).

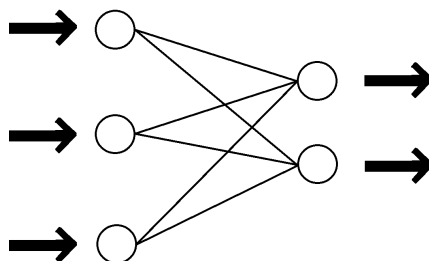


Рис. 1. Схема нейронной сети факторного анализа (число нейронов входного слоя равно числу переменных, число нейронов выходного слоя меньше числа нейронов входного слоя и равно числу факторов)

Такая конфигурация позволяет вычислить значения факторов как выходных сигналов нейронов выходного слоя при известных значениях исходных переменных — выходных сигналов нейронов входного слоя и весовых коэффициентов связей, поставленных в соответствие матрице  $B_{g \times m} \leftrightarrow b_{ij}$ .

Чтобы обеспечить максимум автокорреляционной функции между исходными переменными и факторами и использовать метод обратного распространения ошибки для осуществления оптимизации критерия, сеть расширяется до следующей конфигурации (рис. 2).

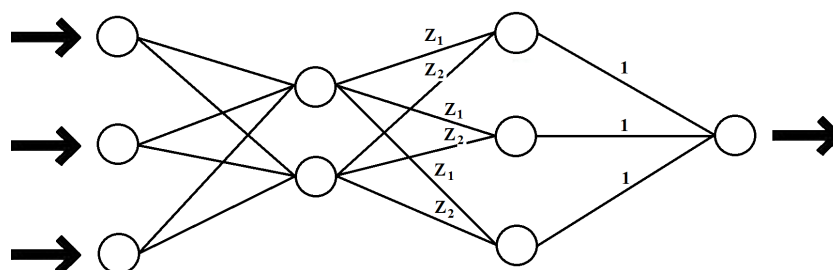


Рис. 2. Схема нейронной сети факторного анализа (два дополнительных слоя нейронов обеспечивают оптимизацию критерия автокорреляционной функции)

В данной конфигурации значение выходного сигнала с нейрона выходного слоя равно значению автокорреляционной функции, когда веса связей на выходном слое равны 1 и веса связей на предыдущем слое равны значениям исходных переменных.

Такая нейронная сеть обучается методом обратного распространения ошибки. Для каждого вектора исходных переменных целевым значением выходного сигнала сети является 1. Для вычисления разности между целевым значением выходного сигнала, равного 1, и самим выходным сигналом его значение берётся по модулю. Осуществляется расчёт ошибки выходного слоя в методе обратного распространения ошибки [3]. Происходит оптимизация критерия

максимума автокорреляционной функции и поиск оптимальных значений элементов матрицы  $B$ .

После вычисления оптимальной матрицы  $B$  матрица факторной структуры  $A$  вычисляется как матрица обратная  $B$  с использованием дополнительных ограничений на значения общностей исходных переменных, равных 1:

$$\sqrt{\sum_{k=1}^g a_{ik}^2} = 1.$$

Для расчёта матрицы  $A$  в работе использовался метод штрафов и метод конфигураций [4, 5]. Целевой функцией являлась сумма квадратов разностей элементов матрицы произведения матриц  $A$  и  $B$  и единичной матрицы  $E$ .

**Численный эксперимент.** В качестве исходных данных были взяты 15 биологических показателей для 131 лица с артериальной гипертензией начальной стадии [6]:

- 1) *вес*,
- 2) *индекс массы тела (ИМТ)*,
- 3) *частота дыхания (ЧД)*,
- 4) *сегментоядерные нейтрофилы (С)*,
- 5) *лимфоциты (Л)*,
- 6) *конечно-систолический размер левого желудочка (КСР)*,
- 7) *конечно-систолический объём левого желудочка (КСО)*,
- 8) *конечно-диастолический размер левого желудочка (КДР)*,
- 9) *конечно-диастолический объём левого желудочка (КДО)*,
- 10) *ударный объём (УО)*,
- 11) *минутный объём сердца (МОС)*,
- 12) *общее периферическое сосудистое сопротивление (ОПСС)*,
- 13) *индекс Хильдебрандта (ИХ)*,
- 14) *фракция выброса левого желудочка (ФВ)*,
- 15) *фракция укорочения левого желудочка (ФУ)*.

Расчёт матрицы факторной структуры по методу максимизации автокорреляционной функции между исходными переменными и факторами приведён в таблице 1.

Полученная матрица факторной структуры находится в хорошем соответствии с результатами, получаемыми другими методами [7, 8].

## Заключение

Предложен критерий максимизации автокорреляционной функции между исходными переменными и факторами, который является балансирующим с критерием минимальных невязок уравнений модели факторного анализа. Оптимизацию такого критерия предлагается осуществлять на нейронной сети специальной конфигурации методом обратного распространения ошибки. Тестовый

Таблица 1. Матрица факторной структуры.

	<i>F1</i>	<i>F2</i>	<i>F3</i>	<i>F4</i>	<i>F5</i>
<b>Вес</b>	-0,292	0,822	-0,309	0,205	<b>-0,315</b>
<b>ИМТ</b>	0,839	0,162	-0,057	-0,118	<b>-0,501</b>
<b>ЧД</b>	0,354	-0,589	-0,224	<b>0,655</b>	-0,218
<b>С</b>	-0,386	0,414	<b>-0,645</b>	0,496	0,124
<b>Л</b>	-0,333	0,378	<b>0,599</b>	0,100	-0,613
<b>КСР</b>	<b>0,659</b>	0,230	0,234	-0,526	0,423
<b>КСО</b>	<b>0,523</b>	-0,508	0,334	-0,352	0,480
<b>КДР</b>	<b>0,130</b>	-0,533	0,505	-0,335	0,574
<b>КДО</b>	<b>-0,668</b>	0,423	-0,315	0,523	-0,029
<b>УО</b>	<b>0,909</b>	-0,384	0,007	-0,109	-0,113
<b>МОС</b>	<b>0,669</b>	-0,276	0,596	-0,063	-0,339
<b>ОПСС</b>	<b>-0,853</b>	0,299	0,013	0,411	0,112
<b>ИХ</b>	0,173	-0,136	0,436	<b>-0,576</b>	0,654
<b>ФВ</b>	-0,501	<b>0,839</b>	-0,207	0,027	0,027
<b>ФУ</b>	-0,055	<b>-0,523</b>	0,525	-0,183	0,642

расчёт матрицы факторной структуры на данных артериальной гипертензии начальной стадии показал хорошее соответствие с результатами других методов.

## ЛИТЕРАТУРА

1. Иберла К. Факторный анализ. М. : Статистика, 1980.
2. Харман Г. Современный факторный анализ. М. : Статистика, 1972.
3. Шовин В.А., Гольдяпин В.В. Факторное моделирование с помощью нейронной сети // Математическое моделирование и численные методы. 2016. № 2. С. 85–103.
4. Банди Б. Методы оптимизации. Вводный курс. М. : Радио и связь, 1988. 128 с.
5. Кокуев А.Г. Оптимальное управление. Поиск экстремумов многомерных функций. Астрахань : АГТУ, 2011. 34 с.
6. Гольдяпин В.В., Шовин В.А. Косоугольная факторная модель артериальной гипертензии первой стадии // Вестник Омского университета. 2010. № 4. С. 120–128.
7. Шовин В.А. Конфирматорная факторная модель артериальной гипертензии // Компьютерные исследования и моделирование. 2012. Т. 4, № 4. С. 885–894.
8. Шовин В.А., Гольдяпин В.В. Методы вращения факторных структур // Математические структуры и моделирование. 2015. № 2(34). С. 75–84.

## AUTOCORRELATION NEURAL NETWORK OF FACTOR ANALYSIS

**V.A. Shovin**

Scientist Researcher, e-mail: v.shovin@mail.ru

The Federal State budget institution Science Institute of Mathematics S.L. Soboleva of Siberian Branch of RAS (Omsk Branch), Russia

**Abstract.** An algorithm for performing factor analysis based on an autocorrelation neural network is developed. The neural network of this type has the ability to autocorrelate the input and output signals. In this neural network, an additional layer of neurons with weight coefficients equal to the values of the input signal is added to carry out the back propagation of the error from the maximum of the correlation of the input and output signals. Maximizing the correlation of the input and output signals leads to the realization of factor analysis and calculation of the principal components in the case of a smaller number of neurons on the output layer than on the input layer of the neural network. Test calculations of the principal components of this algorithm on the data of arterial hypertension showed the identity of significant factor loads and factor structures with other methods of searching for the principal components.

**Keywords:** factor analysis, neural network, back propagation method.

*Дата поступления в редакцию: 06.09.2018*

## **КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ ИДЕНТИФИКАЦИИ ЛИЧНОСТИ ПО РАДУЖНОЙ ОБОЛОЧКЕ ГЛАЗА НА ОСНОВЕ OPENCV**

**Т.Ю. Назарова**

студент, e-mail: tanyanazarova-94@yandex.ru

**Д.Н. Лавров**

к.т.н., доцент, e-mail: lavrov@omsu.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

**Аннотация.** В статье представлены результаты конструирования алгоритма распознавания на основе использования библиотеки OpenCV. Представлены результаты первичных экспериментов по распознаванию на небольшой базе изображений.

**Ключевые слова:** радужная оболочка глаза, идентификация, распознавание.

### **Введение**

Целью нашей работы является построение системы компьютерного моделирования распознавания личности по радужной оболочке глаза. Работа является продолжением работ [1, 2]

На данный момент существуют различные методы распознавания образов. Все они разделяются на несколько этапов (рис. 1):

1. Получение биометрических данных.
2. Предварительная обработка.
3. Извлечение признаков из биометрических данных.
4. Классификация биометрических образцов (идентификация) или сравнение с образцом (верификация).

Перечислим основные приёмы, используемые на каждом этапе. Рассмотрение этапа получения изображений выходит за рамки данной статьи.

### **1. Предварительная обработка изображения глаза**

Этапы предварительной обработки изображения для поиска границ радужной оболочки глаза выглядят следующим образом: фильтрация изображения, улучшение контрастности, бинаризация изображения и выделение границ.

**Методы препроцессинга.** Фильтрация используется для уменьшения шума на изображении. Рассмотрим несколько наиболее часто используемых приёмов.

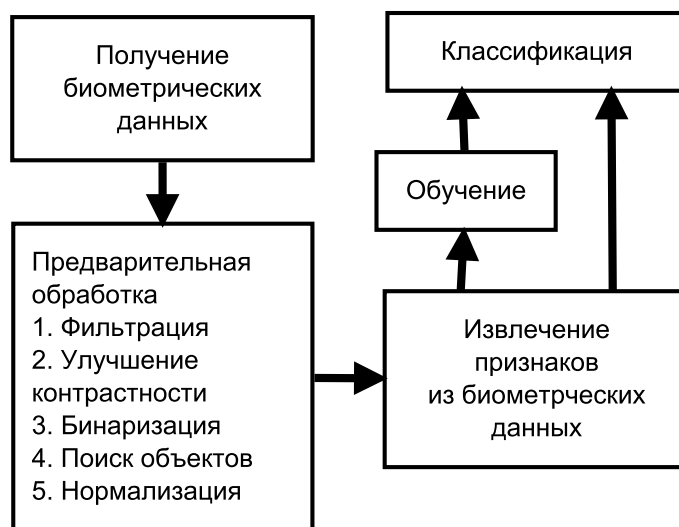


Рис. 1. Архитектура системы распознавания человека по радужной оболочке глаза

*Размытие Гаусса* — это фильтр, построенный на использовании нормального распределения для построения ядра свёртки  $G(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2+y^2}{2\sigma^2}}$ . При-

мер ядра свёртки  $3 \times 3$  для фильтра:  $K(x, y) = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$ ,  $x = -1, 0, 1$ ;  
 $y = -1, 0, 1$ .

Вычисление интенсивности пикселя производится по формуле вычисления свёртки  $I(x_i, y_i) = \sum_{x=-\lfloor n/2 \rfloor}^{\lfloor n/2 \rfloor} \sum_{y=-\lfloor n/2 \rfloor}^{\lfloor n/2 \rfloor} K(x, y) \cdot I(x_i + x, y_i + y)$ .

*Медианный фильтр.* Медиана рассчитывается путём сортировки всех значений пикселей окрестности рассматриваемого пикселя в порядке возрастания и выбора интенсивности в середине отсортированного списка. Затем интенсивность рассматриваемого пикселя заменяется на медианное значение соседей.

Для повышения контраста используется алгоритм *контрастного ограниченного адаптивного выравнивания гистограммы* (CLAHE) [7].

Последовательность действий этого алгоритма следующая: 1) вычисляется гистограмма  $H_{(x_i, y_i), k}(l)$  окрестности пикселя с координатами  $(x_i, y_i)$ , здесь  $l$  — интенсивность, а  $k \times k$  — размер рассматриваемой окрестности; 2) гистограмма обрезается по заданному порогу; 3) срезанные значения яркости равномерно перераспределяются (это можно сделать итеративно, пока не останется ни одного пикселя с интенсивностью выше ранее заданного порога); 4) по пересчитанной гистограмме  $\hat{H}_{(x_i, y_i), k}(l)$  вычисляется кумулятивная функция распределения:  $CDF_{(x_i, y_i), k}(l) = \sum_{i=0}^l \hat{H}_{(x_i, y_i), k}(i)$ ; 5) затем вычисляется интенсивность пикселя  $I'(x_i, y_i) = 255 \cdot CDF_{(x_i, y_i), k}(I(x_i, y_i))$  [6].

*Бинаризация* — алгоритм преобразования полноцветного изображения в монохромное. Бинаризация бывает двух видов — пороговая и адаптивная. Пороговая бинаризация — метод, при котором выбирается порог бинаризации, и, если выбранный пиксель выше этого порога, то он становится белым, ниже — чёрным. Адаптивные методы работают с участками изображения и используются при неоднородном освещении объектов [3].

*Преобразование Хафа* предназначено для поиска объектов (в данном случае окружностей), принадлежащих определённому классу фигур с использованием процедуры голосования. Процедура голосования применяется к пространству параметров, из которого и получаются объекты определённого класса фигур по локальному максимуму в аккумуляторном пространстве, которое строится при вычислении преобразования Хафа. Преобразование Хафа основывается на представлении искомого объекта в виде параметрического уравнения:  $(x - a)^2 + (y - b)^2 = r^2$ . Параметры этого уравнения представляют фазовое пространство. Перебираются все точки границ, и делается предположение, что точка принадлежит линии искомого объекта. Таким образом, для каждой точки изображения рассчитывается нужное уравнение и получаются необходимые параметры, которые сохраняются в пространстве Хафа. Финальным шагом является обход пространства Хафа и выбор максимальных значений, за которые «проголосовало» больше всего пикселей картинки, что и даёт нам параметры для уравнений искомого объекта [4].

*Нормализация* изображения нужна для того, чтобы каждое изображение имело единый размер. В противном случае возникают трудности при сравнении радужных оболочек глаза.

Даугман предлагает перевести изображение из декартовой системы координат  $(x, y)$  в *полярную систему координат*  $(\rho, \varphi)$ . Эта форма представления радужной оболочки (*развёртка*) имеет ряд преимуществ по сравнению с кольцом: поворот изображения заменяется циклическим сдвигом.

Для выделения существенных признаков на радужной оболочке используется *сингулярное разложение*. Представим, что развёртка является матрицей. Для любой вещественной  $(m \times m)$  матрицы  $A$  существуют две вещественные ортогональные матрицы  $U$  и  $V$  такие, что  $\Sigma = U^T A V$ , где диагональная матрица  $\Sigma$ . Матрицы  $U$  и  $V$  выбираются так, чтобы диагональные элементы матрицы  $\Sigma$  имели вид:  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$ , где  $r$  — ранг матрицы  $A$ . Следующим шагом необходимо удалить малозначимые (определяется по порогу) диагональные элементы матрицы  $\Sigma$  и обратным преобразованием восстановить исходное изображение, в котором останутся только главные компоненты.

Независимо от нашей разработки аналогичные подходы к предобработке были использованы в докторской диссертации Матвеева И.А. «Методы и алгоритмы автоматической обработки изображений радужной оболочки глаза». В работе исследованы характеристики изображений глаз человека, рассмотрены отдельные методы для выделения границ радужной оболочки, на основании полученных характеристик методы и алгоритмы были сведены в единую систему и протестированы [5].

## 2. Методы идентификации биометрических образцов

Классификация биометрических образцов происходит по следующему алгоритму:

1. Выделение признаков у биометрического образца.
2. Сравнение полученного вектора признаков с векторами, которые находятся в базе.
3. Подтверждение гипотезы о совпадении текущего биометрического образца с одним из образцов в базе.

Рассмотрим несколько алгоритмов, идентифицирующих биометрические образцы. Метод опорных векторов направлен на перевод исходного вектора в пространство более высокой размерности и поиск гиперплоскости, которая разделит два класса в этом пространстве. Размерность гиперплоскости на единицу меньше, чем размерность исходного пространства.

*Метод опорных векторов (SVM)* направлен на перевод исходного вектора в пространство более высокой размерности и поиск гиперплоскости, которая разделит два класса в этом пространстве. Размерность гиперплоскости на единицу меньше, чем размерность исходного пространства.

Рассмотрим бинарную задачу классификации. Пусть имеется обучающая выборка:  $(x_1, y_1), \dots, (x_m, y_m)$ ,  $x_i \in R^n$ ,  $y_i \in \{-1, 1\}$ . SVM строит функцию классификации  $F(x) = \text{sign}(\langle w, x \rangle + b)$ , где  $\langle w, x \rangle$  — скалярное произведение,  $w$  — вектор нормали к оптимальной гиперплоскости,  $b$  — дополнительный параметр сдвига. Если  $F(x) = 1$ , то объект попадает в один класс, а если  $F(x) = -1$ , то объект попадает в другой класс. Необходимо выбрать такие параметры  $w$  и  $b$ , которые расположат гиперплоскость на максимальное расстояние от двух классов. Предположим, что расстояние от каждого класса должно быть равным. Задача вычисления максимума эквивалентна задаче нахождения минимума, которую можно описать в виде следующих формул:

$$\begin{cases} \arg \min_{w,b} \|w\|^2 \\ y_i(\langle w, x \rangle + b) \geq 1, \quad i = 1 \dots m. \end{cases}$$

На практике довольно редко можно разделить два класса гиперплоскостью, поэтому важен случай линейной неразделимости.

В этом случае обучающую выборку переводят в пространство более высокой размерности с помощью следующего отображения:  $\varphi : R^n \rightarrow x$ . При этом отображение выбирается таким образом, чтобы разделить классы, выборка становится линейно разделима. Каждое скалярное произведение в приведённых выше формулах заменяется нелинейной функцией ядра. Ядром классификатора называется функция  $k(x, x') = \langle \varphi(x), \varphi(x') \rangle$ . Ядром может служить любая положительная симметричная функция от двух переменных. Она должна быть положительной, чтобы была возможность посчитать минимум  $\|w\|^2$  [6].

Следующий метод классификации — это *нейронные сети*. Они представляют собой структуру из нескольких слоев, содержащих вычислительные элементы (нейроны) и связи между ними. Искусственный нейрон работает по следующему принципу:



1. На каждый элемент входного слоя попадает входной сигнал  $x_i$ , который умножается на определённый вес  $w_i$ , соответствующий этому сигналу.
2. Все полученные результаты суммируются. Эта взвешенная сумма представляет степень возбуждения нейрона.
3. Взвешенная сумма передаётся в функцию активации  $\varphi$ , результат которой является выходным сигналом нейрона. Функция  $\varphi$  бывает разных типов:
  - пороговая функция, передающая на выход единичный сигнал;
  - сигмоидальная функция, которая преобразует входящие сигналы в значение в диапазоне  $\{0; 1\}$ ;
  - гиперболический тангенс, который позволяет получить значения разных знаков.

Для задач кластеризации и распознавания используются сети прямого пространства, в которых сигнал распространяется строго от входного слоя к выходному, сигнал в обратном направлении не распространяется.

Для распознавания человека по радужной оболочке глаза необходимо применять обучение нейронной сети на основе обучающей выборки, которая содержит в себе входные значения и правильные выходные значения. Для проверки правильности обучения также имеется тестовая выборка, по которой производится оценка качества сети.

*Метод Байеса* основан на предположении, что расположение интенсивностей на радужной оболочке глаза подчиняется нормальному распределению. Интенсивности  $I(x)$  одного образца представляют собой вектор.

Выбирается обучающая база, состоящая из  $N$ -векторов. Для этих векторов рассчитывается их среднее значение по формуле:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i.$$

На следующем шаге необходимо построить матрицу корреляции:

$$\hat{R} = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x}_i)(x_i - \bar{x}_i)^T.$$

Плотность нормального распределения в общем виде выглядит таким образом:

$$f(x) = \frac{1}{(2\pi)^n \det(R)} e^{-\frac{1}{2}(x_i - \bar{x}_i)^T R^{-1} (x_i - \bar{x}_i)}.$$

Эллипс, на котором показатель степени в последней формуле, а значит и плотность распределения, имеют постоянное значение, называется эллипсом равной плотности, ограниченная им область — эллипсом рассеивания,  $\bar{x}_i$  — центр рассеивания. После расчёта среднего значения и матрицы корреляции выбирается некоторый порог для определения, попадает ли наш исходный вектор  $x_i$  внутрь эллипса рассеивания. Значение  $P$  выбирается исходя из правила трёх сигм — практически все значения нормально распределённой случайной величины лежат в интервале  $(\bar{x} - \delta; \bar{x} + \delta)$ . Более строго — приблизительно с

вероятностью 0,9973 значение нормально распределённой случайной величины лежит в указанном интервале.

Метод, основанный на расстоянии Хэмминга, предполагает, что все биологические образцы можно представить в виде битовой последовательности. Для этого необходимо провести бинаризацию всего экземпляра при определённом пороге бинаризации (например, равном 127).

*Расстояние Хэмминга* — это число бит, в которых два битовых кода различны:

$$d = W(Z) = W(X \text{ xor } Y),$$

где  $X$  и  $Y$  — битовые коды. Если рассматривать два битовых кода, полученных с одной и той же радужной оболочки, то расстояние Хэмминга будет близко к 0.

В случае метода, основанного на использовании скалярного произведения, вектор представляет собой пиксели изображения в оттенках серого. Производится вычисление косинусов углов между входным вектором и векторами, находящимися в базе, по следующей формуле:

$$\cos(\widehat{a, b}) = \frac{\langle a, b \rangle}{|a| \cdot |b|}.$$

Если полученное значение равно 1, то векторы  $a$  и  $b$  находятся в линейной зависимости, следовательно, принадлежат одному объекту. Результатом алгоритма является вектор, косинус которого с исходным вектором ближе всего к 1.

Метод, основанный на коэффициенте *корреляции Спирмена*, является ранговым, то есть для оценки силы связи используются не численные значения, а соответствующие им ранги. Коэффициент инвариантен по отношению к любому монотонному преобразованию шкалы измерения.

Пусть заданы две выборки  $X = (x_1, \dots, x_n)$ ,  $Y = (y_1, \dots, y_n)$ . Коэффициент *корреляции Спирмена* вычисляется по формуле:

$$\rho = 1 - \frac{6}{n(n+1)} \sum_{i=1}^n (R_i - S_i)^2,$$

где  $R_i$  — ранг наблюдения  $x_i$  в ряду  $X$ ,  $S_i$  — ранг наблюдения  $y_i$  в ряду  $Y$ . Коэффициент  $\rho$  принимает значения из отрезка  $[-1; 1]$ . Равенство  $\rho = 1$  указывает на строгую прямую линейную зависимость,  $\rho = -1$  — на обратную зависимость. Ранги для выборки расставляются в порядке возрастания  $x_i$ , при совпадении  $x_i$  и  $x_j$  рангу присваивается значение среднего арифметического между совпадающими значениями.

### 3. Практическая реализация

Все алгоритмы препроцессинга были реализованы с помощью библиотек PIL, OpenCV, SciPy и NumPy с использованием языка программирования и интерпретатора Python. OpenCV — библиотека алгоритмов компьютерного зрения.

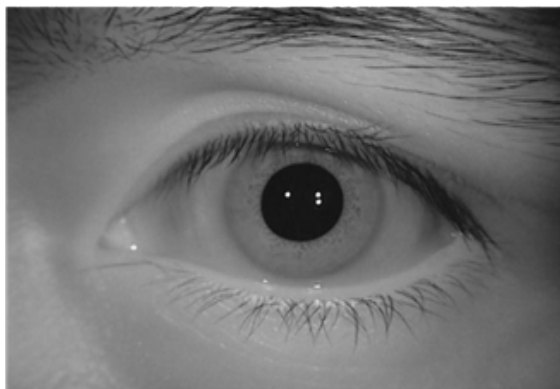


Рис. 2. Исходное изображение [8]

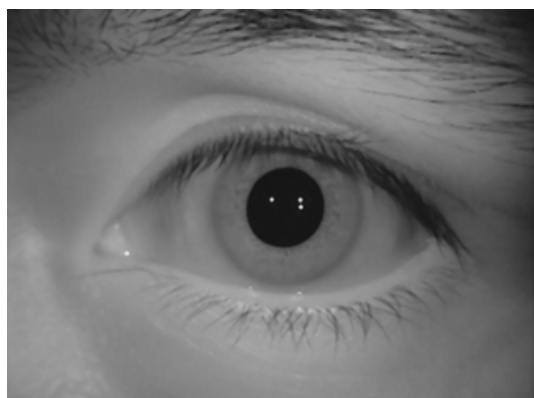


Рис. 3. Применение медианного фильтра к изображению

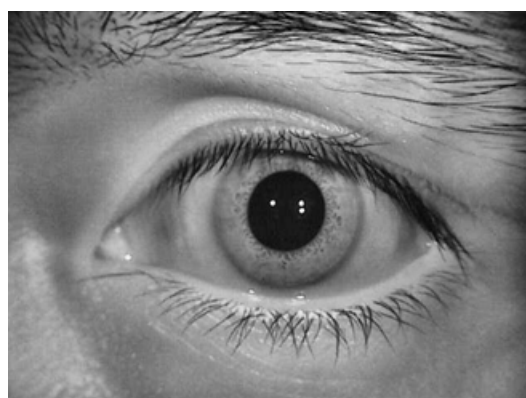


Рис. 4. Применение алгоритма CLAHE

**Реализация алгоритмов предобработки.** Для фильтрации изображения был использован медианный фильтр из библиотеки OpenCV. Медианный фильтр реализован с помощью функции `cv2.medianBlur(img,size)`, где `img` — исходное изображение, `size` — количество элементов для выборки.

После применения медианного фильтра изображение становится менее ярким, блики — более приглушёнными, удаляется часть шумов.

Следующая стадия — это повышение контрастности изображения, для этого были использованы две функции `clahe=cv2.createCLAHE(clipLimit, tileGridSize)`, где `clipLimit` — это специальный порог CLAHE, `tileGridSize` — размер области выравнивания, и `clahe.apply(img)`, где `img` — исходное изображение. Первая функция создаёт условия для работы алгоритма CLAHE, а вторая производит преобразования изображения.

После применения метода линии на изображениях становятся более чёткими, что в дальнейшем поможет выделить границы на изображении.

Для проведения процесса бинаризации переводим наши изображения в оттенки серого функцией `cv2.cvtColor(img,cv2.COLOR_GRAY2BGR)`. Далее

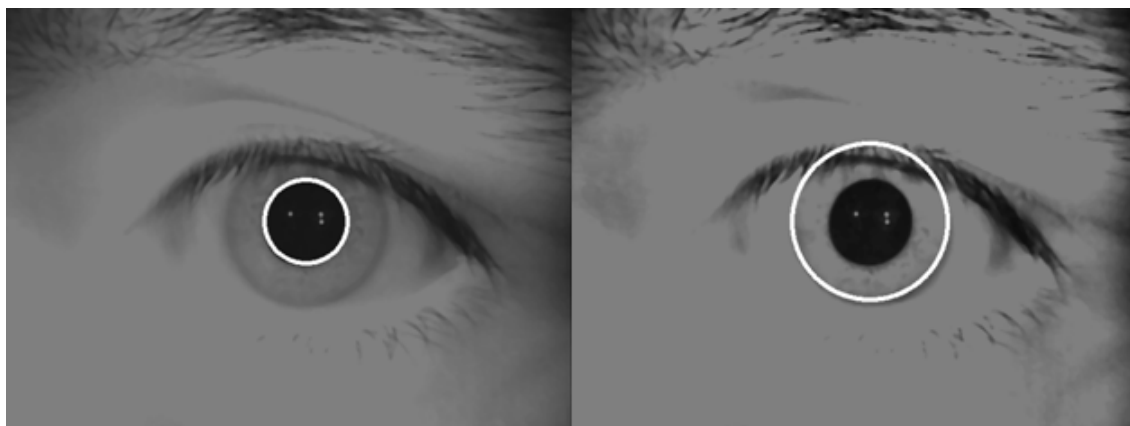


Рис. 5. Результат применения преобразования Хафа

проводим самую бинаризацию с помощью функции `cv2.threshold(img, thresh, maxValue, cv2.THRESH_TRUNC)`, где `img` — исходное изображение, `thresh` — порог бинаризации, `maxValue` — значение, которое получает пиксель выше порога, `cv2.THRESH_TRUNC` — тип бинаризации.

Для выполнения преобразования Хафа воспользуемся функцией `cv2.HoughCircles(img, cv2.CV_HOUGH_GRADIENT, dp, min_dist, param1, param2, minRadius, maxRadius)`, где `img` — это исходное изображение, `cv2.CV_HOUGH_GRADIENT` — используемый метод, `dp` — разрешение сумматора, используемое для детектирования центров кругов, `min_dist` — минимальная дистанция между центрами детектируемых кругов, `param1` — первый параметр (в зависимости от метода трансформации), `param2` — второй параметр (в зависимости от метода трансформации), `min_radius` минимальный радиус круга, `max_radius` — максимальный радиус круга.

Для нахождения зрачка глаза устанавливаем `minRadius=0` и `maxRadius=75`, для нахождения внешней границы радужной оболочки — `minRadius=85` и `maxRadius=130`.

После нахождения внешней границы радужной оболочки глаза обрезается изображение для удобства дальнейшей обработки. Необходимо найти посторонние объекты, которые могли попасть на радужку глаза (веки, ресницы). Для этого пользуемся детектором границ Canny. Для применения метода Canny берём функцию `cv2.Canny(img, threshold1, threshold2)`, где `img` — исходное изображение, `threshold1` и `threshold2` — левый и правый порог процедуры.

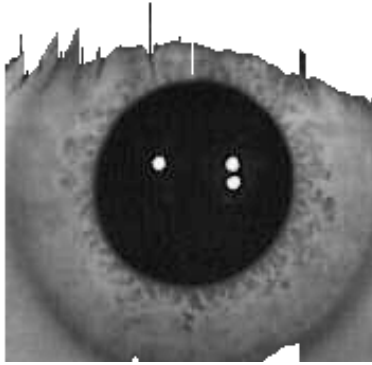


Рис. 6. Удаление лишних элементов на изображении



Рис. 7. Выделение общей границы

Для окончательного выделения границ накладываем полученные границы с помощью преобразования Хафа и детектора Саппу, и проводим общую границу.

Следующим шагом после нахождения границ является развёртка кольца оболочки. Для этой процедуры необходимо перевести координаты точек  $(x, y)$  из декартовой плоскости в координаты  $(\rho, \varphi)$  полярной по формулам

$$\begin{cases} \rho = \sqrt{x^2 + y^2}; \\ \varphi = \text{atan2}\left(\frac{y}{x}\right). \end{cases}$$

Расчёт значений начинается с пикселей, расположенных на внешней границе радужной оболочке глаза.

Вследствие перевода из декартовых координат в полярные теряются некоторые пиксели развёртки изображения. Для восстановления этих данных используется линейная интерполяция.

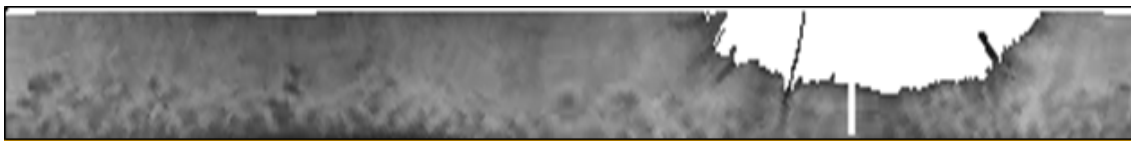


Рис. 8. Развёртка радужной оболочки

**Заполнение базы шаблонов.** Для сравнения образцов изображение необходимо привести к размеру 256x32 пикселя. `Image.resize(size, resample)`, где `size` — размер необходимого изображения, `resample` — тип преобразования (`Image.BICUBIC` — двойная сплайн-интерполяция).

**Идентификация заданного образца.** Для идентификации заданного образца реализуем несколько алгоритмов. Пусть на вход после проведения всей предварительной подготовки подан вектор  $x$  размером 8192 байта, каждая координата которого представлена значением от 0 до 255.

Алгоритм идентификации с помощью расстояния Хэмминга представлен в следующем виде.

Таблица 1. Результаты первого эксперимента (в процентах)

Метод	Процент правильных ответов
Расстояние Хэмминга	62
Скалярное произведение векторов	70
Коэффициент корреляции Спирмена	69

1. Посчитать расстояние Хэмминга для всех векторов, представленных в базе шаблонов.
2. Подать на выход номер объекта, расстояние которого было наименьшим. Этот объект и является результатом работы алгоритма.

Алгоритм идентификации с использованием скалярного произведения между векторами:

1. Посчитать косинус угла между заданным вектором и всеми векторами объектов в базе.
2. Подать на выход номер объекта, косинус которого был наибольшим. Этот объект и является результатом работы алгоритма.

Для улучшения алгоритма необходимо посчитать значения косинуса, сменяя в цикле значения векторов в базе на 32 бита. После всех подсчётов выбрать максимальное значение косинуса для классификации.

Алгоритм идентификации с использованием коэффициента корреляции Спирмена:

1. Вычислить ранги у заданного вектора и всех векторов объектов в базе.
2. Посчитать значение коэффициента корреляции Спирмена для заданного вектора и каждого вектора из базы.
3. Подать на выход номер объекта, коэффициент которого был наибольшим. Этот объект и является результатом работы алгоритма.

#### 4. Эксперимент

Для проверки работы метода использовалась база изображений глаз человека Casia-IrisV4 [8]. Были взяты изображения 10 человек, для каждого человека было дано 5 изображений глаза.

Было проведено два эксперимента для проверки алгоритма. В первом эксперименте алгоритму необходимо было решить проблему идентификации объекта, найти, с каким объектом у входного вектора больше всего совпадений.

Во втором эксперименте входной вектор сравнивался со всеми векторами базы, и в ответе показывался результат по каждому вектору базы (верификация объекта).

Для второго эксперимента необходимо вычислить значение порога  $p$ , при котором определяется принадлежность класса. Для этого воспользуемся следующими метриками:

1. TruePositive (TP) — количество образцов тестовой выборки, на которых

алгоритм выдаёт принадлежность к определённому объекту и достоверно этот образец принадлежит этому классу.

2. TrueNegative (TN) — количество образцов тестовой выборки, на которых алгоритм выдаёт непринадлежность к определённому объекту и достоверно этот образец не принадлежит этому классу.
3. FalsePositive (FP) — количество образцов тестовой выборки, на которых алгоритм выдаёт принадлежность к определённому объекту и достоверно этот образец не принадлежит этому объекту.
4. FalseNegative (FN) — количество образцов тестовой выборки, на которых алгоритм выдаёт непринадлежность к определённому объекту и достоверно этот образец принадлежит этому объекту.
5. Доля правильных ответов, выданных системой, на основе выше описанных параметров, вычисляется по формуле

$$accuracy = \frac{TP + TN}{TP + TN + FN + FP}.$$

6. Доля объектов, названных системой положительными и при этом действительно являющихся положительными:

$$precision = \frac{TP}{TP + FP}.$$

7. Доля объектов положительного класса, которые определила система, из всех объектов положительного класса:

$$recall = \frac{TP}{TP + FN}.$$

В таблицах 2, 3, 4 представлены результаты, полученные на основе тестов.

Выберем метод, который дал наилучший результат в первом эксперименте, и применим к нему сингулярное разложение, оставив только 16 первых компонент. Доля правильных ответов для первого эксперимента составляет 72 %.

Для второго эксперимента результат отражён в таблице 5.

На основе проведённых экспериментов можно сказать, что реализованные методы дают лучшие показатели при решении задачи идентификации.

## Заключение

В статье сделан обзор методов, применяемых для систем идентификации человека по радужной оболочке глаза: методы предварительной обработки изображения (фильтр Гаусса, медианный фильтр, бинаризация изображения, преобразование Хафа) и методы классификации объектов (метод опорных векторов, нейронные сети, метод Байеса, метод с использованием расстояния Хэмминга и т. д.). Реализованы методы предварительной обработки и методы идентификации объекта, основанные на расстоянии Хэмминга, скалярном произведении векторов и коэффициенте корреляции Спирмена, а также проведена их оценка.

В заключении можно сказать, что алгоритмы, реализованные в данной работе, можно усовершенствовать за счёт улучшения методов препроцессинга. Кроме того, для выделения признаков можно также использовать нейросети.

Таблица 2. Расстояние Хэмминга. Результаты эксперимента при  $p = 0,4$   
(в количестве выбранных объектов)

Объект	TP	TN	FP	FN	accuracy	precession	recall
1	0	38	187	20	0,15	0,07	0
2	3	48	177	17	0,2	0,01	0,145
3	16	3	222	4	0,07	0,06	0,8
4	2	69	156	18	0,28	0,01	0,1
5	11	50	175	9	0,24	0,05	0,55
6	14	15	210	6	0,11	0,06	0,7
7	8	47	178	12	0,22	0,04	0,4
8	18	13	212	2	0,12	0,07	0,9
9	7	69	156	13	0,31	0,04	0,35
10	14	9	216	6	0,09	0,06	0,7

Таблица 3. Скалярное произведение векторов. Результаты эксперимента при  $p = 0,55$   
(в количестве выбранных объектов)

Объект	TP	TN	FP	FN	accuracy	precession	recall
1	20	82	143	0	0,41	0,12	1
2	20	107	118	0	0,51	0,14	1
3	17	147	78	3	0,66	0,17	0,85
4	20	36	189	0	0,22	0,09	0,85
5	16	75	150	4	0,37	0,09	0,8
6	16	132	93	4	0,6	0,14	0,8
7	20	67	158	0	0,35	0,11	0,8
8	14	141	84	6	0,63	0,14	0,7
9	20	81	144	0	0,41	0,12	0,7
10	14	137	88	6	0,61	0,13	0,7



Таблица 4. Коэффициент корреляции Спирмена. Результаты при  $p = 0,5$   
(в количестве выбранных объектов)

Объект	TP	TN	FP	FN	accuracy	precession	recall
1	20	205	20	0	0,91	0,21	1
2	11	191	34	9	0,82	0,24	0,55
3	0	225	0	20	0,91	0,24	0,55
4	16	181	44	4	0,8	0,26	0,8
5	8	196	29	12	0,83	0,21	0,4
6	2	218	7	18	0,89	0,22	0,1
7	19	194	31	10	0,83	0,24	0,1
8	2	219	6	18	0,9	0,25	0,1
9	13	180	45	7	0,78	0,22	0,65
10	6	221	4	14	0,92	0,6	0,3

Таблица 5. Скалярное произведение векторов. Результат эксперимента при  $p = 0,55$   
(в количестве выбранных объектов)

Объект	TP	TN	FP	FN	accuracy	precession	recall
1	20	74	0	0	0,38	0,11	1
2	20	100	125	0	0,48	0,13	1
3	18	140	85	2	0,64	0,17	0,9
4	20	30	195	0	0,20	0,09	0,9
5	15	68	157	5	0,33	0,08	0,75
6	15	123	102	5	0,56	0,12	0,75
7	20	59	166	0	0,32	0,107	0,75
8	14	133	92	6	0,6	0,132	0,7
9	20	75	150	0	0,38	0,117	0,7
10	17	128	97	3	0,59	0,14	0,85

## Благодарности

Выражаем признательность Илье Дмитриевичу Сиганову за обсуждение результатов и ценные советы.

## ЛИТЕРАТУРА

1. Лавров Д.Н., Назарова Т.Ю. Алгоритмы предварительной обработки изображения для поиска границ радужной оболочки глаза // Молодёжь третьего тысячелетия: сборник научных статей. Омск : Изд-во Ом. гос. ун-та, 2017. С. 828–832.
2. Грищенко Н.П., Лавров Д.Н. Обзор методов идентификации человека по радужной оболочке глаза // Математические структуры и моделирование. 2014. №1(29). С. 43–64.
3. Бинаризация изображений. Распознавание образов для программистов. URL: <http://recog.ru/blog/applied/15.html> (дата обращения: 14.05.2016).
4. Дегтерева А., Вежнев В. Преобразование Хафа (Houghtransform) // Сетевой журнал «Компьютерная графика и Мультимедиа». URL: <http://sgm.computergraphics.ru/content/view/36> (дата обращения: 14.05.2018).
5. Матвеев И.А. Методы и алгоритмы автоматической обработки изображений радужной оболочки глаза: дис. ... канд. техн. наук. М. : ВЦ РАН, 2014.
6. Lopez P. SVM versus a monkey. Makeyourbets. URL: <https://quantdare.com/svm-versus-a-monkey> (дата обращения: 20.05.2017).
7. Zuiderveld K. Contrast limited adaptive histogram equalization. Graphics gems IV. Academic Press Professional, Inc., 1994. P. 474–485.
8. Biometrics Ideal Test. CASIA-IrisV4. URL: <http://biometrics.idealtest.org/dbDetailForUser.do?id=4> (дата обращения: 20.11.2015).

## COMPUTER MODELING OF IDENTIFICATION BY THE IRIS OF THE EYE BASED ON OPENCV

**T.Yu. Nazarova**

Student, e-mail: [tanyanazarova-94@yandex.ru](mailto:tanyanazarova-94@yandex.ru)

**D.N. Lavrov**

Ph.D. (Eng.), Associate Professor, e-mail: [lavrov@omsu.ru](mailto:lavrov@omsu.ru)

Dostoevsky Omsk State University, Omsk, Russia

**Abstract.** The results of recognition algorithm construction based on the use of the OpenCV library are given. The results of primary experiments on image recognition on a small base are presented.

**Keywords:** iris, identification, recognition.

*Дата поступления в редакцию: 09.09.2018*

## **FUZZY IDEAS EXPLAIN A COMPLEX HEURISTIC ALGORITHM FOR GAUGING PAVEMENT CONDITIONS**

**Edgar Daniel Rodriguez Velasquez**<sup>1,2</sup>

Instructor, e-mail: edgar.rodriguez@udep.pe, edrodriguezvelasquez@miners.utep.edu

**Carlos M. Chang Albitres**<sup>2</sup>

Ph.D. (Engr.), Associate Professor, e-mail: cchangalbitres2@utep.edu

**Vladik Kreinovich**<sup>2</sup>

Ph.D. (Phys.-Math.), Professor, e-mail: vladik@utep.edu

<sup>1</sup>Universidad de Piura in Peru (UDEP)

<sup>2</sup>University of Texas at El Paso, El Paso, Texas 79968, USA

**Abstract.** To gauge pavement conditions, researchers have come up with a complex heuristic algorithm that combines several expert estimates of pavement characteristics into a single index — which is well correlated with the pavement's durability and other physical characteristics. While empirically, this algorithm works well, it lacks physical or mathematical justification beyond being a good fit for the available data. This lack of justification decreases our confidence in the algorithm's results — since it is known that often, empirically successful heuristic algorithms need change when the conditions change. To increase the practitioners' confidence in the resulting pavement condition estimates, it is therefore desirable to come up with a theoretical justification for this algorithm. In this paper, we show that by using fuzzy techniques, it is possible to come up with the desired justification.

**Keywords:** gauging pavement conditions, fuzzy logic.

### **1. Formulation of the Problem**

**It is important to gauge pavement conditions.** Most roads are heavily used. Heavy traffic stresses the pavement. As a result, after several years, it is necessary to maintain — or sometimes even repair — the roads.

Roads repairs are expensive. It is therefore important to adequately gauge pavement conditions — so that we will be able to correctly decide which road segments need maintenance or repair, and which can wait a few more years.

This is especially important since it is known that a proper maintenance can make the road last much longer and thus, drastically decrease the need for expensive road repairs.

**How pavement conditions are gauged now.** One of the most frequently used technique for gauging pavement conditions is based on visual inspection of the pavement.

Visual inspection enables the inspectors to detect different types of problems — known as *distresses*. We can have buckling, we can have potholes, we can have cracks, etc. For each type of distress, inspectors:

- measure the area affected by this type of distress (or the length, for linear distresses like linear cracks), and
- use the results of these measurement to evaluate the severity of the corresponding distress.

The resulting data is then combined into a single *pavement condition index* (PCI).

The combination rules used in the computation of the PCI are selected so as to provide the most accurate prediction of the pavement durability. To improve the predictive quality, more and more complex algorithms are used; see, e.g., the latest international standard [1].

**Problem.** The problem is that the existing algorithm for gauging the pavement condition is heuristic. This algorithm has been selected purely empirically, it does not have any physical or mathematical justification — beyond being a good fit for the available data.

In general, heuristic methods often work well, but they are usually less reliable than theoretically justified algorithms — since they rely solely on the past experiences and when the situations change, we may need to change the algorithms as well. To increase the user's confidence in the PCI algorithm, it is thus desirable to come up with a theoretical justification for this algorithm.

**What we do in this paper.** In this paper, we provide the desired theoretical justification for the current state-of-the-art complex heuristic algorithm for gauging pavement conditions.

In this justification, we take into account the fact that this algorithm combines — somewhat subjective — inspector observations, observations which include information described not in numerical terms, but rather in terms of imprecise (“fuzzy”) words from natural language, such as “high”, “low”, and “medium”. Thus, to analyze this problem, it is reasonable to use techniques specifically designed for translating such knowledge into precise numbers — namely, *fuzzy* techniques; see, e.g., [3, 8–10, 12, 13].

These techniques are what we will use in our justification.

## 2. The Current State-of-the-Art Algorithm for Gauging Pavement Conditions: A Brief Reminder

**What we start with.** For each road segment, this algorithm starts with the numbers  $x_1, \dots, x_n$  that describe the relative areas (or relative lengths) of the distresses within this segment.

**First step: a non-linear transformation.** First, an appropriate non-linear transformation  $f_i(x_i)$  is applied to each value  $x_i$ , resulting in so-called *deduct values*  $s_i = f_i(x_i)$  ranging from 0 to 100 (or, equivalently, from 0 to 1). These non-

linear transformations  $f_i(x_i)$  are selected so that the resulting PCI have the largest correlation with the pavement's durability.

The deduct values are selected in such a way that larger values of the scores correspond to more severe distresses:

- the value 100 (or 1) corresponds to the most severe distress, while
- the value 0 corresponds to the absence of distress.

**Second step: sorting the deduct values.** The deduct values corresponding to distresses of different types are then sorted in the decreasing order, from the most severe to the least severe:  $s_{(1)} \geq s_{(2)} \geq \dots$

**Third step: deciding how many deduct values to use.** Based on the largest deduct value, we then decide how many deduct values to use. This number  $m$  of used deduct values is found from a formula

$$m = 1 + \frac{9}{98} \cdot (100 - s_{(1)}). \quad (1)$$

We then use only the values  $s_{(1)} \geq s_{(2)} \geq \dots \geq s_{(m)}$ .

**Final step: combining deduct values.** To combine the values  $s_{(1)}, \dots, s_{(m)}$ , we do the following:

- first, we compute the sum of the largest deduct value  $s_{(1)}$  and of  $m - 1$  small values (equal to 2); we apply an appropriate non-linear transformation to transform this sum into the interval  $[0, 100]$ ; thus, we get the first combined deduct value  $c_1$ ;
- then, we compute the sum of the two largest deduct values and of  $m - 1$  2s — and apply a different non-linear transformation to the resulting sum; thus, we get the second combined deduct value  $c_2$ ;
- after that, we compute the sum of 3 largest deduct values and  $m - 2$  2s, and apply a yet different non-linear transformation to the resulting sum; thus, we get the third combined deduct value  $c_3$ ;
- then we repeat the same procedure for 4 largest deduct values, for 5 largest deduct values, etc., until we are combine all  $m$  deduct values.

As a result, we get  $m$  combined deduct values  $c_1, c_2, \dots, c_m$ .

After that, we take the largest of the resulting combined distress values  $c \stackrel{\text{def}}{=} \max_i c_i$ . The PCI is simply 100 minus this largest value:  $\text{PCI} \stackrel{\text{def}}{=} 100 - c$ .

**The resulting combination of somewhat subjective estimates is indeed well-correlated with physical properties.** The algorithm has been selected so as to provide the largest correlation with the pavement durability and other physical characteristic. For example, it has been shown that PCI is strongly correlated with the International Roughness Index that measures the passing vehicle's vibrations caused by the pavement's imperfection; see, e.g., [8].

**Towards reformulating the final step.** Our ultimate goal is to decide when a road segment needs maintenance or repair. This decision is made by comparing the PCI estimated for this segment with a certain threshold  $t_0$ . The condition that

$$100 - c \geq t_0$$

is equivalent to  $c \leq 100 - t_0$ . In its turn, the condition that  $c = \max_i c_i \leq 100 - t_0$  is equivalent to requiring that  $c_i \leq 100 - t_0$  for each  $i$ .

Each value  $c_i$  is obtained from the sum  $s_{(1)} + \dots + s_{(i)}$  :

- by adding  $(m - i)$  values of 2 and
- by applying an appropriate non-linear transformation to the resulting sum.

Thus, the condition  $c_i \leq 100 - t_0$  is equivalent to requiring that the sum  $s_{(1)} + \dots + s_{(i)}$  is greater than or equal to some threshold  $t_i$ . Thus, we can reformulate the final step as follows.

**Reformulation of the final step.** To decide whether the given road segment needs repairs or maintenance, we check, for each  $i$  from 1 to  $m$ , whether

$$s_{(1)} + s_{(2)} + \dots + s_{(i)} \geq t_i$$

for the corresponding threshold  $t_i$ .

**What needs explanation.** Natural questions are:

- Why should we use sum and not any other combination function?
- Why should we consider the sum of a few largest distress values and not of all these values?
- Why should we consider several sums instead of just one?
- Where does the formula for the number  $m$  of considered deduct values come from?

There can be many other questions, since the above procedure, with its emphasis on sorting and maxima, does not look like any physical formula — physics formulas very rarely use maxima.

### 3. Why Should We Use Sum and Not Any Other Combination Function: An Explanation

**Let us start analyzing the problem.** The road segment is good if there are not too many distresses of each type, i.e., if there are:

- few distresses of the first type *and*
- few distresses of the second type, etc.

In other words, the pavement is good if:

- the first value  $x_1$  is small *and*
- the second value  $x_2$  is small, etc.

This looks like a typical phrase to be analyzed by fuzzy techniques. Namely, phrase is an “and”-combination of simpler phrases like “the value  $x_1$  is small”, “the value  $x_2$  is small”, etc. To assign a numerical value to the validity of this phrase, it makes sense:

- first, to estimate the degree to which each simple statement “ $x_i$  is small” is true, and then
- combine these degree of confidence into a single degree.

This is exactly what we will do.

**We need different membership functions for different  $i$ .** In accordance with the usual fuzzy techniques, for each  $i$  and for each  $x_i$ , we need to come with a number  $d_i$  describing to what extent the given value  $x_i$  is small. Let us denote this number by  $\mu_i(x_i)$ . In fuzzy techniques, the corresponding function  $\mu_i(x_i)$  is known as the *membership function* corresponding to the notion “small”.

In the traditional application of fuzzy techniques, when we have several occurrences of the same word like “small”, we use the same membership function. However, most fuzzy textbooks emphasize that this is not necessarily the case: for example, then transforming the size in meters into a number, “small” means two different things when referring to cats or to people — a cat the size of a small human being is, by cats’ standards, a giant.

This is exactly the case here. For example, if  $x_1$  describes the relative area of severe distress, then  $x_1$  should really be small for this distress to be acceptable and not requiring any maintenance. However, for low severity distress  $x_2$ , even if this distress takes a significant part of the road segment, by itself, this may not necessarily trigger any need for maintenance. Thus, in our case, we need different membership functions  $\mu_i(x_i)$  for different  $i$ .

**How to combine the degrees.** In general, the problem of combining the degrees is as follows:

- we know the degrees  $a$  and  $b$  to which statements  $A$  and  $B$  are true, and
- we want to use these values  $a$  and  $b$  to estimate the degree to which a composite statement  $A \& B$  is true.

In fuzzy logic, the corresponding estimate is called an “*and*”-operation (or, for historical reasons, a *t-norm*); let us denote it by  $f_{\&}(a, b)$ .

In these terms, the desired degree of confidence that the road segment is good is equal to

$$f_{\&}(\mu_1(x_1), \mu_2(x_2), \dots). \quad (2)$$

**Natural conditions on an “and”-operation.** The “and”-operation should satisfy several conditions. First, since  $A \& B$  and  $B \& A$  mean the same, it is reasonable to expect that the corresponding estimates for their degrees should be the same, i.e., that we should have  $f_{\&}(a, b) = f_{\&}(b, a)$  for all  $a$  and  $b$ . In other words, the “and”-operation should be *commutative*.

Similarly, since  $A \& (A \& C)$  and  $(A \& B) \& C$  means the same, we expect that the estimates of the degree of these two statement should be the same, i.e., that for all  $a$ ,  $b$ , and  $c$ , we should have  $f_{\&}(a, f_{\&}(b, c)) = f_{\&}(f_{\&}(a, b), c)$ . In other words, an “and”-operation should be *associative*.

There are several other reasonable properties; see, e.g., [3, 8–10, 12, 13]. An “and”-operation that satisfies all these properties is usually what is called a *t-norm*.

**Structure of a generic t-norm.** Some t-norms have the form

$$f_{\&}(a, b) = g^{-1}(g(a) + g(b)) \quad (3)$$

for some increasing function  $g(z)$ , where  $g^{-1}(z)$  indicates an inverse function, for which  $g^{-1}(g(z)) = z$ . Such t-norms are known as *Archimedean*.

For example, for the probability-inspired operation  $f_{\&}(a, b) = a \cdot b$ , we get this form with  $g(z) = -\ln(z)$ . A more traditional way of representing Archimedean t-norms is by reducing them to the product, as  $f_{\&}(a, b) = h^{-1}(h(a) \cdot h(b))$ ; this can be reduced to the above sum-based representation if we take  $g(a) = h(-\ln(a))$ .

It is known (see, e.g., [5]) that for every t-norm  $f_{\&}(a, b)$  and for every  $\varepsilon > 0$ , there exists an Archimedean t-norm  $f'_{\&}(a, b)$  which is  $\varepsilon$ -close to  $f_{\&}(a, b)$ , i.e., for which

$$|f'_{\&}(a, b) - f_{\&}(a, b)| \leq \varepsilon$$

for all  $a$  and  $b$ . Since the expert's degrees of confidence are always approximate, and  $\varepsilon$  can be arbitrary small, in practice, we can safely replace the original t-norm with an  $\varepsilon$ -close Archimedean one — as long as  $\varepsilon$  is small enough. Thus, without losing generality, we can safely assume that the t-norm  $f_{\&}(a, b)$  is Archimedean.

**This explains why in gauging pavement conditions, we use sum.** Indeed, the degree of confidence that the road segment is good is determined by the formula (2). As we have discussed, we can safely assume that the corresponding t-norm is Archimedean, i.e., that it is described by the formula (3).

Substituting the expression (3) into the formula (2), we conclude that the desired degree  $d$  has the form  $d = g^{-1}(g(\mu_1(x)) + g(\mu_2(x_2)) + \dots)$ , i.e., equivalently, the form  $d = g^{-1}(s)$ , where  $s = s_1 + s_2 + \dots$ ,  $s_i = f_i(x_i)$ , and  $f_i(z) \stackrel{\text{def}}{=} g(\mu_i(z))$ .

In particular, since the function  $g(z)$  is increasing, the condition that road is good enough, i.e., that  $d \geq d_0$  for some threshold  $d_0$ , can be equivalently reformulated as  $s \geq t_0 \stackrel{\text{def}}{=} g(d_0)$ . In other words, we get  $s_1 + s_2 + \dots \geq t_0$ . This is exactly the sum-based formula used to estimate the desired degree — which is thus explained by fuzzy ideas.

#### 4. Why Should We Consider the Sum of a Few Largest Distress Values And Not of All These Values?

**Analysis of the problem: analyzed road segments are reasonably good.** The whole procedure makes sense when roads are reasonably well maintained and are in reasonable condition. If the road is in a clearly bad condition, there is no need to accurately gauge its quality, we just need to repair it.

The need for an accurate estimate of the road's quality occurs when we have several segments of reasonably good quality, and we need to find the way to maintain them and making them even better.

In such situations, most distress values  $x_i$  are small. When a distress value is very small, it does not affect the overall quality of a road segment.

**Computational consequences of this analysis.** Since small distress values do not affect the quality of a road segment, taking them into account would be a waste of computational resources.



To avoid this waste, it makes sense to ignore these very small values, and consider only a few largest distress values. This is exactly what is usually done: instead of taking the sum of all the values  $s_1 + s_2 + \dots$ , we only consider the sum of the  $m$  largest values  $s_{(1)} + s_{(2)} + \dots + s_{(m)}$ . This is exactly what practitioners do.

## 5. Why Should We Consider Several Sums Instead of Just One?

**General idea.** If, based on the largest distress, we know that the road segment need repair or maintenance, there is no need to consider all other distresses. In this case, taking other distresses into account would be a waste of computational resources.

If, based on the first distress, we cannot make a definite conclusion, it is reasonable to also consider the second distress, etc.

Thus, instead of always taking all  $m$  distresses into account, it makes sense to first check just the largest distress, then two largest, then three largest. etc.

This is exactly what is done in practice.

**This is a fuzzy analog of lazy logical operations.** In classical 2-valued logic, if we want to find the truth value of a statement  $A \& B$  and we know that  $A$  is false, there is no need to find the truth value of  $B$  — we can already conclude that the composite statement  $A \& B$  is also false.

This simple observation saves us computation time. The corresponding operation is known as a *lazy* “and”. This is the most commonly used “and”-operation in programming languages such as C or Java.

What we are describing here is the fuzzy analogue of such lazy “and”-operations. Indeed, when the first values  $s_{(1)}, s_{(2)}, \dots$  are already large — corresponding to close-to-false (0) values of the corresponding degrees  $\mu_i(x_i)$  — then there is no need to compute any further terms, we know that the road segment needs repair or maintenance.

## 6. Where Does the Formula for the Number $m$ of Considered Deduct Values Come From?

**Analysis of the problem.** Suppose that we know the largest distress  $s_{(1)}$ . Let us denote, by  $S_0$ , the overall distress level after which the road segment needs repairs or maintenance.

Let us denote, by  $s_0$ , the smallest value of an individual distress that is still worth taking into account, so that values smaller than  $s_0$  can be safely set to 0. Then, if, in addition to the largest distress, we take into account  $m - 1$  other non-zero distresses, we get the overall value  $s_{(1)} + (m - 1) \cdot s_0$ . If this value is already larger than or equal to the threshold  $S_0$ , this means that there is no need to consider any additional distresses — we already know that the road segment needs repairs or maintenance.

On the other hand, if among the  $m$  largest distresses, the smallest is already below  $s_0$  — and can hence be safely ignored — this means that all smaller distresses can also be ignored. So, considering more than  $m$  distresses also does not make sense.

Thus, in all possible cases, the largest number of distresses to be continued is the smallest  $m$  for which  $s_{(1)} + s_0 \cdot (m - 1) \geq S_0$ . In terms of  $m$ , this inequality can be reformulated in the equivalent form

$$m \geq 1 + \frac{1}{s_0} \cdot (S_0 - s_{(1)}).$$

So, the smallest possible value  $m$  that satisfies this property has the form

$$m = 1 + \frac{1}{s_0} \cdot (S_0 - s_{(1)}). \quad (4)$$

**This analysis explains the formula for the number  $m$  of considered deduct values.** Indeed, (4) is exactly the formula used to estimate how many deduct values we need to take into account.

## Acknowledgments

This work was supported in part by the National Science Foundation grant HRD-1242122 (Cyber-ShARE Center of Excellence).

## REFERENCES

1. ASTM International. Standard Practice for Roads and Parking Lots Pavement Condition Index Surveys. International Standard D6433-18.
2. Belohlavek R., Dauben J.W., Klir G.J. Fuzzy Logic and Mathematics: A Historical Perspective. Oxford University Press, New York, 2017.
3. Klir G., Yuan B. Fuzzy Sets and Fuzzy Logic. Prentice Hall, Upper Saddle River, New Jersey, 1995.
4. Mendel J.M. Uncertain Rule-Based Fuzzy Systems: Introduction and New Directions. Springer, Cham, Switzerland, 2017.
5. Nguyen H.T., Kreinovich V., Wojciechowski P. Strict Archimedean t-norms and t-conorms as universal approximators // International Journal of Approximate Reasoning. 1998, V. 18, Nos. 3–4. P. 239–249.
6. Nguyen H.T., Walker E.A. A First Course in Fuzzy Logic. Chapman and Hall/CRC, Boca Raton, Florida, 2006.
7. Novák V., Perfilieva I., Močkoř J. Mathematical Principles of Fuzzy Logic. Kluwer, Boston, Dordrecht, 1999.
8. Park K., Thomas N.E., Lee K.W. Applicability of the International Roughness Index as a predictor of asphalt pavement condition // Journal of Transportation Engineering. 2007. V. 133, No. 12. P. 706–709.
9. Zadeh L.A. Fuzzy sets // Information and Control. 1965. V. 8. P. 338–353.

## НЕЧЁТКИЕ ИДЕИ ОБЪЯСНЯЮТ СЛОЖНЫЙ ЭВРИСТИЧЕСКИЙ АЛГОРИТМ ДЛЯ ОПРЕДЕЛЕНИЯ УСЛОВИЙ ДОРОЖНОГО ПОКРЫТИЯ

**Эдгар Даниэль Родригес Веласкес**<sup>1,2</sup>

преподаватель, e-mail: @utep.edu

**Карлос М. Чанг Альбитрес**<sup>2</sup>

к.т.н, доцент, e-mail: vladik@utep.edu

**В. Крейнович**<sup>2</sup>

к.ф.-м.н., профессор, e-mail: vladik@utep.edu

<sup>1</sup>Университет Пиуры в Перу (UDEP)

<sup>2</sup>Техасский университет в Эль Пасо, США

**Аннотация.** Чтобы оценить условия дорожного покрытия, исследователи придумали сложный эвристический алгоритм, который объединяет несколько экспертных оценок характеристик дорожного покрытия в единый индекс, хорошо коррелирующий с долговечностью покрытия и другими физическими характеристиками. Эмпирически этот алгоритм работает хорошо, но ему не хватает физического или математического обоснования, он просто хорошо подходит для доступных данных. Это отсутствие обоснования уменьшает нашу уверенность в результатах алгоритма — известно, что эмпирически успешные эвристические алгоритмы часто нуждаются в изменении, когда меняются условия. Поэтому, чтобы повысить уверенность практиков в оценках состояния дорожного покрытия, желательно придумать теоретическое обоснование этого алгоритма. В этой статье мы показываем, что, используя нечёткие методы, можно придумать желаемое обоснование.

**Ключевые слова:** оценка условий дорожного покрытия, нечёткая логика.

*Дата поступления в редакцию: 30.06.2018*

## **SOFT COMPUTING IDEAS CAN HELP EARTHQUAKE GEOPHYSICS**

**Solymar Ayala**<sup>1</sup>

Doctoral Student, e-mail: sayalacortez@miners.utep.edu

**Aaron Velasco**<sup>1</sup>

Ph.D. (Geosciences), Professor, e-mail: aavelasco@utep.edu

**Vladik Kreinovich**<sup>2</sup>

Ph.D. (Phys.-Math.), Professor, e-mail: vladik@utep.edu

<sup>1</sup>Department of Geological Sciences

<sup>2</sup>Department of Computer Science

University of Texas at El Paso, Texas 79968, USA

**Abstract.** Earthquakes can be devastating, thus it is important to gain a good understanding of the corresponding geophysical processes. One of the challenges in geophysics is that we cannot directly measure the corresponding deep-earth quantities, we have to rely on expert knowledge, knowledge which often comes in terms of imprecise (“fuzzy”) words from natural language. To formalize this knowledge, it is reasonable to use techniques that were specifically designed for such a formalization — namely, fuzzy techniques. In this paper, we formulate the problem of optimally representing such knowledge. By solving the corresponding optimization problem, we conclude that the optimal representation involves using piecewise-constant functions. For geophysics applications, this means that we need to go beyond tectonic plates to explicitly consider parts of the plates that move during the earthquake. We argue that such an analysis will lead to a better understanding of earthquake-related geophysics.

**Keywords:** earthquake geophysics, soft computing, Haar wavelets.

### **1. Specifics of Data Processing in Earthquake Analysis (and in Geophysics in General)**

**Earthquake analysis is important.** Earthquakes can be devastating. It is therefore important to gain as much understanding about the corresponding geophysical processes as possible; see, e.g., [1, 7].

**Usual approach to data processing.** A good understanding means that we know, for each location  $(x, y)$  and for each depth  $z$ , what is the density  $\rho$  at this location and this depth, what are the mechanical properties of the material at this 3-D location  $u = (x, y, z)$ , what are the stresses at this 3-D location. In other words, we need to find the corresponding functions like  $\rho(u) = \rho(x, y, z)$ .

How can we describe a function in a computer? A usual way is to select a *basis* of functions

$$e_1(u), e_2(u), \dots, \quad (1)$$

so that each desired function  $f(u)$  can be represented as a linear combination of the basis functions

$$f(u) = c_1 \cdot e_1(u) + c_2 \cdot e_2(u) + \dots,$$

and then represent the desired function  $f(x)$  by the corresponding coefficients

$$(c_1, c_2, \dots). \quad (2)$$

In principle, we can consider different bases, but it is usually convenient to *orthonormalize* them, i.e., to consider linear combinations

$$e_i^{on}(x) = \sum_{j=1}^i c_{ij} \cdot e_j(u)$$

for which, for all  $i$  and  $j$ , we have

$$\int (e_i^{on}(u))^2 du = 1$$

and

$$\int e_i^{on}(u) \cdot e_j^{on}(u) du = 0 \text{ when } i \neq j.$$

In this case, the desired coefficients  $c_i$  can be obtained by using a simple formula

$$c_i = \int f(u) \cdot e_i^{on}(u) du. \quad (3)$$

Thus, without losing generality, we can safely assume that the basis (1) is orthonormal.

The most widely used examples of such bases are:

- sines and cosines, and
- wavelets; see, e.g., [2, 4, 6, 11].

For sines and cosines, the expansion into the corresponding basis is known as *Fourier transform*. For wavelets, the transformation from the original function  $f(u)$  to the coefficients  $c_i$  is known as the *wavelet transform*.

**It is important to select an appropriate basis.** It is known that selecting an appropriate basis can drastically improve the quality of the data processing results.

For example, in many cases, wavelet analysis has led to interesting discoveries that were not possible when Fourier analysis was used to process the corresponding data.

It is therefore very important, in each practical situations, to select the most appropriate basis.

**What we plan to do in this paper: main idea.** In this paper, we provide arguments for selecting the most appropriate basis for earthquake-related analysis. In this analysis, we use the specific features of the geophysical data processing.

**Specifics of geophysical data processing.** In comparison with most other data processing situations, geophysical analysis has two important specifics.

First, in most data processing situations, we have continuous functions. For example, when we control a vehicle, its location continuously depends on time. In contrast, in geophysics, there are clear discontinuities:

- as we go deeper,
- we have an abrupt transition between different layers.

The second difference is that in most other data processing situations, we can determine the ground truth, i.e., the actual values of the corresponding quantities. In geophysics, our ability to get the ground truth is very limited: up to a certain depth, we can drill a borehole and find out what are the actual properties, but at larger depths, this is not practically possible.

**Why soft computing.** Since we cannot determine the actual values to check different models, we have to rely on expert knowledge to decide which model works better.

Expert knowledge rarely comes in precise terms, it usually comes in terms of imprecise (“fuzzy”) words. To describe the corresponding knowledge in precise terms, it is therefore reasonable to use techniques specifically designed to handle such knowledge — namely, the techniques of fuzzy logic; see, e.g., [3, 5, 8–10, 12, 13].

## 2. Analysis of the Problem

**Main idea.** Since the values  $f(u)$  comes from expert estimates, they come with a fuzzy uncertainty. In other words, for every  $u$ , we have fuzzy information about the difference  $\Delta f(u) \stackrel{\text{def}}{=} \tilde{f}(u) - f(u)$  between:

- the expert estimate  $\tilde{f}(u)$  and
- the actual (unknown) value  $f(u)$ .

In precise terms, this means that:

- we do not know the probabilities of different possible values of  $\Delta f(u)$ , but
- we have a membership function  $\mu(\Delta f)$  that describes, for each possible value  $\Delta f$ , the degree to which this value is possible.

Since we have no reason to assume that the estimation errors are positive or negative, it is reasonable to assume that the degree of possibility of each value  $\Delta f$  does not depend on its sign:  $\mu(-\Delta f) = \mu(\Delta f)$ .

Once we have selected the basis  $e_i(u)$ , we will then transform the estimate for  $f(u)$  into the sequence of the corresponding coefficients  $c_i$ .

- Since the values  $f(u)$  are known with uncertainty,
- as a result, we can only determine the coefficients  $c_i$  — and thus, the corresponding term  $c_i \cdot e_i(u)$  — with uncertainty.

A reasonable idea is to select the basis  $e_i(u)$  for which this uncertainty in the term  $c_i \cdot e_i(u)$  is the smallest possible.

**Let us describe this idea in precise terms.** When we process the expert estimates  $\tilde{f}(u)$ , we get the following estimates  $\tilde{c}_i$  for the coefficients  $c_i$ :

$$\tilde{c}_i = \int \tilde{f}(u) \cdot e_i(u) du. \quad (4)$$

The actual (unknown) value  $c_i$  of the corresponding coefficient can be obtained if we apply the same procedure to the actual (unknown) function  $f(u)$ :

$$c_i = \int f(u) \cdot e_i(u) du. \quad (5)$$

If we subtract (5) from (4) and take into account that the integral of the difference is equal to the difference of the integrals, we get the following formula for the inaccuracy  $\Delta c_i \stackrel{\text{def}}{=} \tilde{c}_i - c_i$ :

$$\Delta c_i = \int \Delta f(u) \cdot e_i(u) du. \quad (6)$$

The inaccuracy in the product  $c_i \cdot e_i(u)$  is equal to the product  $\Delta c_i \cdot e_i(u)$ . This value depends on the location  $u$ :

- for some locations  $u$ , the value  $|e_i(u)|$  is larger, so the inaccuracy is larger;
- for other locations  $u$ , the value  $|e_i(u)|$  is smaller, so the inaccuracy is smaller.

It is reasonable to minimize the worst-case inaccuracy

$$\Delta c_i \cdot \max_u |e_i(u)| = \max_u |e_i(u)| \cdot \int \Delta f(u) \cdot e_i(u) du. \quad (7)$$

Here, each value  $\Delta f(u)$  is a fuzzy number, so  $\Delta c_i$  is also a fuzzy number. In fuzzy logic, this fuzzy number is determined by the *Zadeh's extension principle*.

It is known that in general, computing the result  $Y = f(X_1, \dots, X_n)$  of applying a function  $f(x_1, \dots, x_n)$  to  $n$  fuzzy numbers  $X_1, \dots, X_n$  can be described as follows: for each  $\alpha \in (0, 1]$ , the  $\alpha$ -cut  $Y(\alpha) \stackrel{\text{def}}{=} \{y : \mu(y) \geq \alpha\}$  is equal to the range of the function  $f(x_1, \dots, x_n)$  when each  $x_i$  takes values from the corresponding  $\alpha$ -cut  $X_i(\alpha) \stackrel{\text{def}}{=} \{x_i : \mu_i(x_i) \geq \alpha\}$ . In precise terms, we have

$$Y(\alpha) = \{f(x_1, \dots, x_n) : x_1 \in X_1(\alpha), \dots, x_n \in X_n(\alpha)\}.$$

Let us apply this general result to our formula (7). Since the membership function  $\mu(\Delta f)$  does not change if we change the sign of the difference  $\Delta f$ , for each  $\alpha$ , the corresponding  $\alpha$ -cut is a symmetric interval. Let us denote this interval by

$$[-\Delta(\alpha), \Delta(\alpha)].$$

The expression (7) is a linear combination of all the values  $\Delta f(u)$ :

- when  $e_i(u) > 0$ , this function is increasing in  $\Delta f(u)$ ;

- when  $e_i(u) < 0$ , this function is decreasing in  $\Delta f(u)$ .

Thus, when each value  $\Delta f(u)$  takes all possible values from the interval  $[-\Delta(\alpha), \Delta(\alpha)]$ , the largest possible value of the expression (6) is attained when:

- for those  $u$  for which  $e_i(u) > 0$ , the value  $\Delta f(u)$  is the largest possible, i.e.,  $\Delta f(u) = \Delta(\alpha)$ , and
- for those  $u$  for which  $e_i(u) < 0$ , the value  $\Delta f(u)$  is the smallest possible, i.e.,  $\Delta f(u) = -\Delta(\alpha)$ .

In both cases, the largest possible value of the product  $\Delta f(u) \cdot e_i(u)$  is equal to  $\Delta(\alpha) \cdot |e_i(u)|$ . Thus, the largest possible value of the integral (6) is equal to

$$\int \Delta(\alpha) \cdot |e_i(u)| \, du = \Delta(\alpha) \cdot \int |e_i(u)| \, du.$$

Hence, the largest possible value of the integral (7) is equal to

$$\Delta(\alpha) \cdot \max_u |e_i(u)| \cdot \int |e_i(u)| \, du.$$

Similarly, we can show that the smallest possible value of the expression (7) is equal to

$$-\Delta(\alpha) \cdot \max_u |e_i(u)| \cdot \int |e_i(u)| \, du.$$

Thus,  $Y(\alpha) = [-y(\alpha), y(\alpha)]$ , where

$$y(\alpha) \stackrel{\text{def}}{=} \Delta(\alpha) \cdot \max_u |e_i(u)| \cdot \int |e_i(u)| \, du.$$

The estimate for  $c_i$  is the most accurate when this interval is the narrowest possible, i.e., when the value

$$\Delta(\alpha) \cdot \max_u |e_i(u)| \cdot \int |e_i(u)| \, du$$

is the smallest possible.

In this product, the factor  $\Delta(\alpha)$  is given. So, the smallest possible value of the above product is attained when the product

$$\max_u |e_i(u)| \cdot \int |e_i(u)| \, du \tag{9}$$

attains its smallest possible value. Hence, we arrive at the following optimization problem.

**Resulting optimization problem.** Among all possible functions  $e_i(u)$  for which  $\int e_i^2(u) \, du = 1$ , we need to find the function with the smallest possible value of the product (9).

**Analysis of the resulting optimization problem.** We always have  $e_i^2(u) = |e_i(u)|^2$ . Thus,  $\int |e_i(u)|^2 \, du = \int e_i^2(u) \, du = 1$ .



Also, for every  $u$ , we have  $e_i(u) \leq \max_v |e_i(v)|$ . Hence,

$$|e_i(u)|^2 \leq \left( \max_v |e_i(v)| \right) \cdot |e_i(u)|. \quad (10)$$

Integrating both parts of this inequality, we conclude that

$$1 = \int |e_i(u)|^2 du \leq \max_v |e_i(v)| \cdot \int |e_i(u)| du. \quad (11)$$

Thus, the product (9) that we want to minimize cannot be smaller than 1. One can easily check that when  $|e_i(u)| = \text{const}$  for all  $u$  from the given region, we get exact equality in the formula (1) and thus, in formula (11).

So, when the absolute value  $|e_i(u)|$  is constant, we attain the smallest possible value 1 of the desired product (9).

Vice versa, if at least for one value  $u$ , we have strict inequality in (10), we will have strict inequality in (11) as well. So, to attain the smallest possible value of the product (9), we must always have equality in the formula (1), i.e., we must always have the following equality:

$$|e_i(u)|^2 = \left( \max_v |e_i(v)| \right) \cdot |e_i(u)|. \quad (12)$$

When  $e_i(u) \neq 0$ , we can divide both sides of this equality by  $|e_i(u)|$  and conclude that  $|e_i(u)| = \max_v |e_i(v)|$ . In other words, for every  $u$ :

- we either have  $e_i(u) = 0$
- or we have  $|e_i(u)|$  equal to the largest possible value  $m \stackrel{\text{def}}{=} \max_v |e_i(v)|$ .

So, we arrive at the following conclusion:

**Resulting solution.** Each function  $e_i(u)$  from the geophysically optimal basis take only three values: 0,  $m$ , and  $-m$ , for some real number  $m > 0$ .

This means, in particular, that all the optimal basis functions are piecewise-constant.

*Comment.* Let us consider the geophysical meaning of this result.

### 3. Geophysical Meaning of Our Result

**What does our result means in terms of earthquake analysis.** An earthquake leads to a spatial shift at different locations. For catastrophic earthquakes, this shift can be in meters; for smaller earthquakes, we can have centimeters-size shift.

In general, we have a shift  $s(x, y)$  as a function of 2-D spatial coordinates. Our optimization result shows that the optimal way to analyze the empirical data about this shift is to represent it as a linear combination  $\sum c_i \cdot e_i(u)$  of piece-wise constant functions  $e_i(u)$ . Such a linear combination is also piece-wise constant. Thus, what we need to do is to divide the whole area into several zones, in each of which the shift is fixed.

In geometric terms, this means that instead of considering each spatial location  $(x, y)$  by itself, we divide the whole region into parts, each of which moves as a whole (i.e., as a solid body).

**How do we transform the observed shifts into this piece-wise constant presentation: an algorithm.** When a function is piece-wise constant, it means that it attains finitely many different values. Let us sort these values into an increasing order:  $s_1 < s_2 < \dots < s_m$ .

Suppose at first that these values are given. In this case, we want to approximate the original function  $f(u)$  by a piece-wise constant function  $a(u)$  that takes values  $s_i$ . For each  $u$ , the value  $a(u)$  is equal to one of the values  $s_1, \dots, s_m$ . Thus, describing the function  $a(u)$  is equivalent to describing, for each  $i$  from 1 to  $m$ , the set  $S_i$  of all the locations  $u$  for which  $a(u) = s_i$ . These  $m$  sets should form a partition of the original domain  $S$ , i.e., we should have  $S_1 \cup \dots \cup S_m = S$  and  $S_i \cap S_j = \emptyset$  for all  $i \neq j$ .

A natural idea is to use the Least Squares approach, i.e., to find such a function  $a(u)$  for which the integral  $\int (f(u) - a(u))^2 du$  attains the smallest possible value. One can easily check that the integral attains the smallest possible value if and only if for each  $u$ , we select the value  $a(u) \in \{s_1, \dots, s_m\}$  for which the value  $(f(u) - a(u))^2$  is the smallest possible. In other words, for each location  $u$ , as  $a(u)$ , we take the value  $s_i$  which is the closest to the original value  $f(u)$ . In other words:

- we select  $a(u) = s_1$  if  $f(u) \leq \frac{s_1 + s_1}{2}$ ;
- we select  $a(u) = s_2$  if  $\frac{s_1 + s_2}{2} \leq f(u) \leq \frac{s_2 + s_3}{2}$ ;
- ...
- for each  $i = 2, \dots, m - 1$ , we select  $a(u) = s_i$  if  $\frac{s_{i-1} + s_i}{2} \leq f(u) \leq \frac{s_i + s_{i+1}}{2}$ ;
- ...
- finally, we select  $a(u) = s_m$  if  $\frac{s_{m-1} + s_m}{2} \leq f(u)$ .

We can repeat this procedure for different tuples  $s = (s_1, \dots, s_m)$ . For each such tuple, we find the resulting mean square error

$$\int \min_i (f(u) - s_i)^2 du.$$

We then select a tuple  $s$  for which this mean square error attains the smallest possible value.

**This immediately brings to mind tectonic plates.** The above piece-wise description bring to mind the geophysical idea that the earth's surface consists of tectonic plates, solid bodies that move in relation to each other.

So, in the first approximation, our mathematical result leads to the very well-known plate tectonics idea.

**Our result goes beyond plate tectonics.** In the first approximation, our result simply leads to a well-known idea of plate tectonics. In this first approximation, the whole plate moves as a whole, the shift is exactly the same on all locations from this plate.

In practice, the shift is somewhat different in different locations on the same tectonic plate. To capture this difference and thus, provide a more accurate description of the corresponding geophysics, we therefore need to divide each affected plate into two (or more) different parts, with different shifts in each part.

**This idea has geophysical sense.** It is known that the major earthquakes are caused by the interaction of tectonic plates — that move relative to each other. As a result, all major earthquakes — and the vast majority of smaller earthquakes — happen at the boundaries between tectonic plates. Specifically, they happen at the *convergent* boundaries, where the plates move towards each other, accumulating a stress. This stress is released by an earthquake.

The above description is a first crude approximation to the corresponding geophysics, in which we can consider the whole plate as a solid body, in which all parts move the same way. In reality, different parts of the plate may accumulate the stress differently and move differently. As a result, some earthquakes only involve a part of the boundary between the plates. Depending on the size of this part, we can get earthquakes of different magnitudes.

**Beyond piece-wise constant functions: geophysics-motivated idea.** Solid bodies do not just shift, they can also rotate. So, a natural idea is to consider not only shifts, but also rotations of the parts of the plate.

In this case:

- instead of approximating the measured values  $f(u)$  by a piece-wise constant function,
- we approximate it by a piece-wise linear functions corresponding to shifts and rotations of different parts of each tectonic plate.

**This can help in earthquake studies.** In view of the above, to get a better understanding of the earthquake geophysics, it is important to analyze which parts of the plate are involved in different earthquakes, which parts have accumulated more stress and in which part, the stress has been released.

**This idea is challenging.** From the computational viewpoint, our idea is very challenging:

- while we can relatively easily identify the boundary between the plates, where the big motion occurs,
- it is much more challenging to identify the parts of the plate that are involved in an earthquake.

The reason why this identification is not easy is because we are interested in geophysical processes far away from the boundaries, where the earthquake-related motion is much smaller in amplitude and thus, much more difficult to detect.

**We all need to work together to overcome these challenges.** As of now, what we have are ideas and models.

Our preliminary results show that these ideas are promising, and we will continue working on them.

However, we think that it will be beneficial to publicize these ideas so that others can implement them, use them, improve them if needed — and thus, help

to get a better understanding of earthquake-related geophysics.

## **Acknowledgments**

This work was supported in part by the CONACYT Fellowship for Solymar Ayala Cortez and by the US National Science Foundation grant HRD-1242122.

The authors are greatly thankful to Dr. Throne Lay from the Earth and Planetary Sciences Department of the University of California at Santa Cruz and to Dr. Steven Harder from the Department of Geological Sciences of the University of Texas at El Paso for valuable discussions and encouragement.

## **REFERENCES**

1. Aki K., Richard P. *Quantitative Seismology, Theory and Methods*. University Science Books, Sausalito, California, 2009.
2. Addison P.S. *The Illustrated Wavelet Transform Handbook: Introductory Theory and Applications in Science, Engineering, Medicine and Finance*. CRC Press, Boca Raton, Florida, 2017.
3. Belohlavek R., Dauben J.W., Klir G.J. *Fuzzy Logic and Mathematics: A Historical Perspective*. Oxford University Press, New York, 2017.
4. Chandrasekhar E., Dimri V.P., Gadre V.M. (eds.) *Wavelets and Fractals in Earth System Sciences*. CRC Press, Boca Raton, Florida, 2014.
5. Hajan A., Styles P. *Application of Soft Computing and Intelligent Methods in Geophysics*. Springer, Cham, Switzerland, 2018.
6. Han B. *Framelets and Wavelets: Algorithms, Analysis, and Applications*. Springer, Cham, Switzerland, 2017.
7. Havskov J., Ottemoller L. *Routine Data Processing in Earthquake Seismology: With Sample Data, Exercises and Software*. Springer, Dordrecht, Heidelberg, London, New York, 2010.
8. Klir G., Yuan B. *Fuzzy Sets and Fuzzy Logic*. Prentice Hall, Upper Saddle River, New Jersey, 1995.
9. Mendel J.M. *Uncertain Rule-Based Fuzzy Systems: Introduction and New Directions*. Springer, Cham, Switzerland, 2017.
10. Nguyen H.T., Walker E.A. *A First Course in Fuzzy Logic*. Chapman and Hall/CRC, Boca Raton, Florida, 2006.
11. Nicholas P. *Framelets and Wavelets: Algorithms, Analysis, and Applications*. Cambridge University Press, Cambridge, UK, 2017.
12. Novák V., Perfilieva I., Močkoř J. *Mathematical Principles of Fuzzy Logic*. Kluwer, Boston, Dordrecht, 1999.
13. Zadeh L.A. Fuzzy sets // *Information and Control*. 1965. V. 8. P. 338–353.

## ИДЕИ МЯГКИХ ВЫЧИСЛЕНИЙ МОГУТ ПОМОЧЬ ГЕОФИЗИКЕ ЗЕМЛЕТРЯСЕНИЯ

**Солимар Айяла**<sup>1</sup>

аспирант, e-mail: sayalacortez@miners.utep.edu

**Аарон Веласко**<sup>1</sup>

к.г.-м.н., профессор, e-mail: aavelasco@utep.edu

**В. Крейнович**<sup>2</sup>

к.ф.-м.н., профессор, e-mail: vladik@utep.edu

<sup>1</sup>факультет геологических наук

<sup>2</sup>факультет компьютерных наук

Техасский университет в Эль Пасо, США

**Аннотация.** Землетрясения могут быть разрушительными, поэтому важно получить хорошее представление о соответствующих геофизических процессах. Одна из проблем в геофизике заключается в том, что мы не можем непосредственно измерять соответствующие глубинные величины, мы должны полагаться на экспертные знания, знания, которые часто выражены неточными («нечёткими») словами естественного языка. Чтобы формализовать эти знания, разумно использовать методы, специально предназначенные для такой формализации, а именно, нечёткие методы. В этой статье мы формулируем задачу оптимального представления такого знания. Решая соответствующую задачу оптимизации, мы заключаем, что оптимальное представление включает использование кусочно-постоянных функций. Для приложений геофизики это означает, что нам нужно выйти за пределы тектонических плит, чтобы подробно рассмотреть части плит, которые движутся во время землетрясения. Мы утверждаем, что такой анализ приведёт к лучшему пониманию геофизики, связанной с землетрясением.

**Ключевые слова:** сейсмическая геофизика, мягкие вычисления, вейвлеты Хаара.

*Дата поступления в редакцию: 30.06.2018*

## **HOW TO SELECT THE BEST PAPER: TOWARDS JUSTIFICATION (AND POSSIBLE ENHANCEMENT) OF CURRENT SEMI-HEURISTIC PROCEDURES**

**Francisco Zapata**

Ph.D. (Phys.-Math.), Instructor, e-mail: fazg74@gmail.com

**Olga Kosheleva**

Ph.D. (Phys.-Math.), Associate Professor, e-mail: olgak@utep.edu

**Vladik Kreinovich**

Ph.D. (Phys.-Math.), Professor, e-mail: vladik@utep.edu

University of Texas at El Paso, El Paso, Texas 79968, USA

**Abstract.** To select the best paper at a conference or in a journal, people use reasonably standard semi-heuristic procedures like averaging scores. These procedures usually work well, but sometimes, new situations appear for which the existing procedures are not automatically applicable. Since the existing procedures are heuristic, it is often not clear how to extend them to new situations. In this paper, we provide a possible explanation for the existing procedures. This explanation enables us to naturally generalize these procedures to possible new situations.

**Keywords:** selecting the best paper, probabilistic approach.

### **1. How Best Papers Are Selected Now: Description and Challenges**

**How best papers are usually selected.** Many conferences and journals have a tradition of selecting the best paper; for journals, it is usually the best paper published in a certain time period. To select the best paper, the conference or the journal asks several respected researchers to form a committee. Members of the committee state their opinions. These opinions are then combined into a single score, and the paper with the largest score is proclaimed the winner.

**Sometimes, experts provide numerical estimates of the papers' quality.** In some cases, committee members are asked to evaluate each paper by a number on a given scale – e.g., on a scale from 0 to 10. As a result, for each expert  $i$  and for each expert  $j$ , we get a number  $v_{ij}$  describing the  $j$ -th expert's estimate of the quality of the  $i$ -th paper.

**How these estimates are usually combined?** Usually, for each paper, we simply add the experts' estimates of this paper – or, which is equivalent in terms of the resulting ordering of papers, we take the arithmetic average of different scores.

**Sometimes, experts are asked to rank the papers.** In some cases, instead of asking the experts to evaluate each paper on a scale, the conference simply asks each expert to rank all the candidate papers. Then, if there are  $n$  candidate papers, for each expert, the top paper gets  $n$  points, second best gets  $n - 1$  points, etc., all the way to the worst paper that gets 1 point. These points are then averaged, and the paper with the largest overall score is proclaimed the best.

**Challenges.** In most cases, the above procedure works well, but sometimes, unusual situations occur. For example, sometimes, one of the experts, instead of ranking all the papers, just selects a paper which is the best according to him/her, and does not provide the ranking of all the other papers. How do we take the opinion of this expert into account?

This is not an easy question to answer because the existing strategy is semi-heuristic, it is not based on any well-defined set of principles, so it is not clear how to extend the existing strategies to such new cases.

As a result, for each such new case, people have to invent new ideas. It would be therefore nice to come up with a theoretical explanation of the existing strategies, an explanation that would enable us to automatically generalize these existing strategies to new situations.

In this paper, we provide such an explanation.

## 2. How to Justify the Current Semi-Heuristic Methods for Selecting Best Papers

**Why arithmetic average.** The first procedure that we described above was taking an arithmetic average of estimates provided by several experts. A natural first question is: why arithmetic average? Why not mean squared value? Why not geometric average? To answer this question, let us analyze this situation in detail.

For each paper  $j$ , have several estimates  $v_{1j}, v_{2j}, \dots, v_{nj}$  provided by different experts. We want to combine them into a single estimate. A natural idea is to assume that for each paper, there is the actual objective value  $e_j$ , and expert estimates  $v_{ij}$  are approximations to this desired value. In other words, we want to find the value  $e_j$  for which  $v_{1j} \approx e_j, v_{2j} \approx e_j$ , etc.

This is a typical situation in data processing, when we have several results  $v_{1j}, v_{2j}$ , etc., of measuring the desired quantity  $e_j$ , and we want to combine these measurement results into a single – more accurate – estimate of this quantity.

There are usually many different (and reasonable independent) factors due to which the measurement result differs from the actual value. The difference  $v_{ij} - e_j$  is the result of the joint effect of all these factors. It is known that, under reasonable condition, the probability distribution of the sum of a large number of independent similar-size random variables is close to Gaussian (normal); see, e.g., [10]. Thus, it makes sense to assume that the differences  $v_{ij} - e_j$  are normally distributed, with some standard deviation  $\sigma$ .

This means that for each value  $e_j$  and for each expert  $i$ , the probability of

having the estimate  $v_{ij}$  is proportional to  $\exp\left(-\frac{(v_{ij} - e_j)^2}{2\sigma^2}\right)$ . The experts are usually assumed to be independent. So, the probability to have all the given estimates is equal to the product of the product of the corresponding probabilities:

$$\prod_{i=1}^n \exp\left(-\frac{(v_{ij} - e_j)^2}{2\sigma^2}\right). \tag{1}$$

Out of all possible values  $e_j$ , it is reasonable to select the *most probable* value, i.e., the value for which the probability (1) is the largest. Maximizing the expression (1) is equivalent to minimizing its negative logarithm, i.e., equivalently, minimizing the sum

$$\sum_{i=1}^n (v_{ij} - e_j)^2.$$

Differentiating this expression with respect to  $e_j$  and equating the derivative to 0, we get

$$e_j = \frac{1}{n} \cdot \sum_{i=1}^n v_{ij},$$

i.e., we get a justification of the usual arithmetic average.

**Why  $n$  points for the top rank,  $n - 1$  points for next ranked paper, etc.**

The above subsection describes how to combine the numerical grades, but what if instead of numerical grades, experts only provide rankings of different papers? How do we convert a ranking into numerical grades?

Let us assume that all these grades are from some interval. Without losing generality, we can safely assume that this interval is the interval  $[0, 1]$ . We want to assign, to each paper, a number so that paper ranked better would get the higher number. If we denote the value assigned to the worst paper by  $v_1$ , the values assigned to the second worst paper by  $v_2$ , all the way to the best of  $n$  paper to which we assign the value  $v_n$ , then these  $n$  numbers must satisfy the inequality

$$0 \leq v_1 < v_2 < \dots < v_n \leq 1. \tag{2}$$

There are many different tuples  $(v_1, \dots, v_n)$  with this property. We have no reason to assume that one of these tuples is preferable, so it makes sense to assume that all these tuples are equally probable. In precise terms, this means that we assume that we have a uniform distribution on the set of all the tuples that satisfy the inequality (2).

We have different possible tuples, we need to combine all these different possible tuples into a single tuple. Similarly to the previous section, we can argue that the most appropriate combination is taking the arithmetic average of all these tuples, i.e., in mathematical terms, taking the mean values  $m_1, \dots, m_n$  of the corresponding components  $v_1, \dots, v_n$  – mean values in the sense of the above uniform distribution.

It turns out that these mean values have a single form  $m_i = \frac{i}{n + 1}$ ; see, e.g., [1–9]. Thus:



- the amount of points assigned to the most highly ranked paper, with  $i = n$ , is proportional to  $n$ ;
- the next paper gets the number of points proportional to  $n - 1$ ,
- etc.
- until we get to the lowest ranked paper, to which we assign the number of points proportional to 1.

This is exactly how this assignment is usually done, which means that this usual assignment has also been justified.

### 3. How to Use These Justifications to Handle Possible Challenges

The ideas behind the above justifications can be used to provide a recommendation on what to do in other situations as well. For example, let us consider the above challenge when one of the experts, instead of ranking all  $n$  papers, just selects the best one.

In this case, no matter how we rank other papers, the selected paper get the value  $n$ . Depending on how we rank all other papers, other papers get values from 1 to  $n - 1$ . Since we do not have any reason to assume that one of the rankings is preferable, it is reasonable to conclude that all these rankings are equally probable. Thus, similarly to the previous session, the number assigned to each of the remaining  $n - 1$  papers should be equal to the arithmetic average – i.e., to the mean – of all the values assigned to this paper according to different possible rankings.

To compute these averages, there is no need to consider all  $(n - 1)!$  possible rankings: it is sufficient to take into account that, first, the number assigned to each of the  $n - 1$  papers is the same, and second, since the sum of numbers  $m$  assigned to all  $n - 1$  papers is always equal to

$$1 + 2 + \dots + (n - 1) = \frac{(n - 1) \cdot n}{2},$$

the mean values should also add up to the same sum:

$$m + \dots + m \text{ (} n - 1 \text{ times)} = (n - 1) \cdot m = \frac{(n - 1) \cdot n}{2}.$$

Thus,  $m = n/2$ . So, in this situation:

- to the highest ranked paper, we assign the value  $n$ , and
- to every other paper, we assign the value  $n/2$ .

### Acknowledgments

This work was supported in part by the National Science Foundation grant HRD-1242122 (Cyber-ShARE Center of Excellence).

## REFERENCES

1. Ahsanullah M., Nevzorov V.B., Shakil M. An Introduction to Order Statistics. Atlantis Press, Paris, 2013.
2. Arnold B.C., Balakrishnan N., Nagaraja H.N. A First Course in Order Statistics. Society of Industrial and Applied Mathematics (SIAM), Philadelphia, Pennsylvania, 2008.
3. David H.A., Nagaraja H.N. Order Statistics. Wiley, New York, 2003.
4. Kosheleva O., Kreinovich V., Lorkowski J., Osegueda M. How to Transform Partial Order Between Degrees into Numerical Values // Proceedings of the 2016 IEEE International Conferences on Systems, Man, and Cybernetics SMC'2016, Budapest, Hungary, October 9–12, 2016.
5. Kosheleva O., Kreinovich V., Osegueda Escobar M., Kato K. Towards the Most Robust Way of Assigning Numerical Degrees to Ordered Labels, With Possible Applications to Dark Matter and Dark Energy // Proceedings of the 2016 Annual Conference of the North American Fuzzy Information Processing Society NAFIPS'2016, El Paso, Texas, October 31 – November 4, 2016.
6. Lorkowski J., Kreinovich V. Interval and Symmetry Approaches to Uncertainty – Pioneered by Wiener – Help Explain Seemingly Irrational Human Behavior: A Case Study // Proceedings of the 2014 Annual Conference of the North American Fuzzy Information Processing Society NAFIPS'2014, Boston, Massachusetts, June 24–26, 2014.
7. Lorkowski J., Kreinovich V. Likert-type fuzzy uncertainty from a traditional decision making viewpoint: how symmetry helps explain human decision making (including seemingly irrational behavior) // Applied and Computational Mathematics. 2014. V. 13, No. 3. P. 275–298.
8. Lorkowski J., Kreinovich V. Granularity Helps Explain Seemingly Irrational Features of Human Decision Making / W. Pedrycz and S.-M. Chen (eds.) // Granular Computing and Decision-Making: Interactive and Iterative Approaches. Springer Verlag, Cham, Switzerland, 2015. P. 1–31.
9. Lorkowski J., Kreinovich V. Fuzzy Logic Ideas Can Help in Explaining Kahneman and Tversky's Empirical Decision Weights / L. Zadeh et al. (Eds.) // Recent Developments and New Direction in Soft-Computing Foundations and Applications. Springer Verlag, 2016. P. 89–98.
10. Sheskin D.J. Handbook of Parametric and Nonparametric Statistical Procedures. Chapman and Hall/CRC, Boca Raton, Florida, 2011.

**КАК ВЫБРАТЬ ЛУЧШУЮ СТАТЬЮ: НА ПУТИ К ОБОСНОВАНИЮ  
(И ВОЗМОЖНОМУ УЛУЧШЕНИЮ) ТЕКУЩИХ ПОЛУЭВРИСТИЧЕСКИХ  
ПРОЦЕДУР**

**Ф. Запата**

к.ф.-м.н., преподаватель, e-mail: fazg74@gmail.com

**О. Кошелева**

к.ф.-м.н., доцент, e-mail: olgak@utep.edu

**В. Крейнович**

к.ф.-м.н., профессор, e-mail: vladik@utep.edu

Техасский университет в Эль Пасо, США

**Аннотация.** Чтобы выбрать лучшую статью на конференции или в журнале, люди используют достаточно стандартные полуэвристические процедуры, такие как средние оценки. Эти процедуры обычно работают хорошо, но иногда появляются новые ситуации, для которых существующие процедуры не применяются автоматически. Поскольку существующие процедуры являются эвристическими, часто неясно, как их распространить на новые ситуации. В этой статье мы приводим возможное объяснение существующих процедур. Эти объяснения позволяют естественно обобщить эти процедуры на возможные новые ситуации.

**Ключевые слова:** выбор лучшей бумаги, вероятностный подход.

*Дата поступления в редакцию: 19.08.2018*

## **ПРОГРАММНОЕ ПРИЛОЖЕНИЕ ДЛЯ ОБУЧЕНИЯ ДЕТЕЙ МЛАДШИХ КЛАССОВ ПРИЁМАМ СКОРОЧТЕНИЯ**

**Л.А. Володченкова**

доцент, к.б.н., e-mail: VolodchenkovaLA@omsu.ru

**С.И. Королёв**

студент, e-mail: korolev131093@gmail.com

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

**Аннотация.** В статье описывается программное приложение, с помощью которого можно автоматизировать процесс обучения детей быстрому чтению.

**Ключевые слова:** быстрое чтение, программное приложение, методики быстрого чтения.

### **Введение**

Эффективное обучение детей чтению — одна из важнейших задач педагогики. Если ребёнок в начальной школе не научился быстро и осознанно читать, то в будущем у него возникнут трудности с восприятием, запоминанием, и, следовательно, с усвоением материала по всем предметам школьного курса.

Существует множество методик для обучения детей быстрому чтению. Раньше все они использовались с применением подручных средств и инструментов, что сейчас не является актуальным и рациональным.

Применение электронных средств обучения, компьютерных обучающих игр и тренажёров способствует повышению мотивации у детей, познавательного интереса, желания учиться. Использование компьютера позволяет разнообразить процесс обучения, представить учебный материал более наглядным и доступным для восприятия учащимися, превратить занятие в увлекательную игру.

Многосторонняя, правильно организованная работа с компьютерными тренажёрами позволяет активизировать процессы восприятия, памяти, отработать навыки чтения. При этом отпадает необходимость использования каких-то подручных средств, обучающий процесс реализуется намного быстрее, удобнее и увлекательнее. Поэтому разработка электронных обучающих комплексов, тренажёров, игр является крайне актуальной задачей современного процесса обучения.

Целью статьи является представление компьютерного комплекса для обучения чтению первоклассников, отработки у них навыков быстрого и осознанного чтения, включающего различные методики, тренажёры и игры.

## **1. Технологии обучения скорочтению учащихся младших классов**

Скорость осознанного чтения — один из главных факторов, влияющих на успеваемость учеников. В арсенале педагогов младших классов немало методик по развитию техники чтения. Память человеческая устроена таким образом, что запоминается не то, что постоянно перед глазами, а то, что мелькает: то есть, то нет. Именно это создаёт раздражение и запоминается. Поэтому педагоги младших классов рекомендуют ежедневно, через определённые промежутки времени проводить с учащимися начальной школы небольшие по объёму тренировочные упражнения, в которых предлагаются к чтению и запоминанию слова, предложения, объём и время экспозиции которых требуется регулировать. В век информационных технологий это легко сделать при помощи небольших компьютерных программ, которые также будут автоматически оценивать правильность выполнения заданий учащимися младших классов. Согласно исследованиям, у каждого ребёнка есть своё «поле чтения»: один схватывает взглядом одно слово, другой — три, а некоторые — пять-шесть слов. Для расширения поля чтения рекомендуется записывать на слайд четыре-пять слов из нового текста и проецировать на экран в течение одной-двух секунд. Не все дети смогут сразу назвать слова, но постепенно учащиеся будут не только быстро прочитывать их, но и произносить в предложенном порядке. Такие несложные упражнения развивают память детей.

Слаборазвитая память у детей младших классов тормозит развитие техники чтения.

Схема работы над формированием умения быстро воспринимать текст и отрабатывать навыки письма включает 18 диктантов-наборов. Учащимся младших классов нужно запоминать не только те слова, которые даны в рамках одного предложения, но и сами короткие предложения в рамках всего набора. В результате ежедневной работы по данной программе дети: 1) легко осмысливают различные тексты; 2) заинтересовываются чтением; 3) быстрее овладевают навыком аналитического чтения и грамотного письма; 4) после отработки каждого комплекта предложений прибавляют в скорости чтения в среднем по 44 слова.

Андреев О.А. и Хромов Л.Н. в своей книге «Учитесь быстро читать» [1] рекомендуют для увеличения поля зрения использовать тестовые и цифровые таблицы Шульте, названные по имени их создателя немецкого психиатра и психотерапевта Вальтера Шульте. Каждая таблица Шульте представляет собой разграфлённый на ячейки квадрат с вписанными в ячейки в хаотичном порядке числами или буквами. Таблицы Шульте — это фактически модель страницы текста. Главное при работе с таблицами Шульте — это расширение поля зрения. Перед началом работы с таблицей взгляд фиксируется в её центре, при этом ставится задача видеть всю таблицу целиком. При стремлении охватить взором всю таблицу Шульте раздражаются периферические зоны сетчатки глаза и формируются нейронные ансамбли, обеспечивающие съём и обработку информации с резервных зон поля зрения. Фиксируя взгляд в центре таблицы,

ученик как бы отправляет в матрицу оперативной памяти мозга панораму всей таблицы. И дальнейшая деятельность представляет собой уже не поиск цифр, а как бы вспоминание уже известного их расположения в клетках таблицы. Занятия с таблицами Шульте также благотворно влияют на развитие памяти, концентрации, расширение поля зрения, что является необходимыми условиями развития навыков скорочтения.

Другом упражнением, способствующим увеличению поля зрения, является работа с числовой пирамидой. Числовая пирамида для расширения поля зрения состоит из двух равномерно расходящихся сверху вниз рядов случайных двузначных чисел и ряда последовательных чисел (от 1 до 12), расположенных посередине. Осевые центральные числа ориентируют учеников во время выполнения упражнения. Тренировка заключается в том, что, фиксируя взгляд на числах центральной от взора линии, необходимо распознавать одновременно и числа, расположенные в расходящихся рядах.

Числовая пирамида для расширения поля зрения способствует так называемому чтению текста вертикальным движением глаз при наличии у читателя широкого поля зрения. Скользя по тексту вертикальным взглядом, человек с широким полем зрения может хорошо понять текст, т. е. нет необходимости скользить глазами по каждой строчке текста.

Все методики скорочтения направлены на устранение повторного перечитывания текста; на выполнение упражнений для расширения поля зрительного восприятия, развитие периферического зрения; скорости чтения; выработку навыка распознавания слов прямо по их визуальному типографическому образу; концентрацию на тексте и осмысливание его сразу во время чтения. Залог успешного обучения скорочтению, как и любому другому навыку, — постоянные тренировки и контроль процесса обучения.

Проведённые исследования позволяет выделить два уровня освоения быстрого чтения. Первый уровень состоит из преодоления двух задач: чтение без проговаривания и расширение угла зрения до границ, определяемых строением глаза. Второй уровень определяется особенностями работы мозга, который человек использует менее чем на 10 %. Обычно информация, представленная в текстовом виде, носит избыточный характер. Из всей массы информации только от 10 % до 30 % — это информация в «чистом виде». При определённой тренировке мозг человека может воспринимать только «чистую» информацию, сжимая временные рамки чтения.

## **2. Обзор приложений для обучения скорочтению**

Разработчики программного обеспечения создали множество программ, помогающих учащимся младших классов освоить рекомендуемые для них педагогами принципы скорочтения.

Образовательная программа AceReader относится к профессиональным программам в области обучения скорочтению. Программа AceReader отличается простотой и комфортом использования, удобным интерфейсом, большим числом настроек, широтой охвата используемых методик. Разработчики внедрили

в программу AceReader большое количество упражнений, способствующих развитию скорочтения.

Программа RocketReader ориентирована на обучение скорочтению и развитие памяти, помогает быстро устранить такие вредные привычки чтения, как проговаривание текста и регрессии. Программа RocketReader содержит множество различных упражнений, при желании можно создать тексты на русском языке и включить их в программу.

Отечественная программа Reader32 предназначена для тренировки ускорения чтения текстовых файлов за счёт устранения эффекта внутреннего проговаривания, который присутствует при обычном чтении. С 2000 года программа Reader32 находится в стадии постоянного тестирования и доработки. В каждой новой версии программы происходит её качественное улучшение. Программа Reader32 работает с обычными текстовыми файлами в формате DOC и RTF, буфером обмена, а также способна читать HTML-файлы и файлы форматов SHML и PHP. Для устранения внутреннего проговаривания текста в программе Reader32 слова подаются на экран одно за другим, практически не оставляя пользователю времени на проговаривание слов. Целью пользователя является увеличение скорости чтения и понимание текста при этом.

Бесплатная программа SST QuickRead предназначена для быстрого чтения текстов. Тексты для чтения можно загружать из системного буфера. Метод, использованный в программе SST QuickRead, заключается в последовательном показе слов крупным шрифтом в центре экрана. Этот метод позволяет заметно увеличить скорость чтения за счёт исключения движения глаз по строке. Особенностью программы SST QuickRead является то, что одновременно с быстрым пословным чтением можно видеть и весь текст с выделенным словом, показанным в данный момент на экране. Программа обеспечивает размещение любого числа закладок в тексте, запоминание местоположения файла и открытие его с того места, где было закончено чтение.

Программа «Скорочей» предназначена для чтения текстов под ритм метронома методом фокусирования взгляда на начале и конце строки (в основе программы — одно из упражнений методики обучения скорочтению М.А. Зиганова [2]) и, в конечном счёте, формирует навыки чтения без регрессий. При каждом втором ударе метронома происходит установка мишени подсветки в начало строки, где необходимо зафиксировать взгляд, затем со вторым ударом метронома нужно успеть пробежать взглядом до конца строки, куда переместится мишень. Следующий удар заставит пользователя вновь переместить взгляд в начало строки и т.д. Таким образом, благодаря перемещению мишени подсветки обеспечивается быстрое ритмичное чтение текста.

### **3. Выбор средств разработки и языков программирования**

Обычно в программах по скорочтению используются одна–две методики обучения приёмам скорочтения, этого достаточно, чтоб овладеть тем или иным приёмом.

Главным отличием предлагаемого приложения от уже известных нам является то, что мы объединили несколько разных методик скорочтения. Это позволило более детально и глубоко изучить приёмы и навыки скорочтения. Методики в приложении представлены как *тренажёры*.

Интерфейс программы построен таким образом, чтобы позволить найти индивидуальный подход к каждому обучаемому благодаря расширенной настройке параметров тренажёра, ориентированности на пользователя как с высоким, так и с низким уровнем владения компьютерной техникой. Также программный комплекс оснащён функцией, которая предупреждает пользователя об ошибках и сбоях работы программы при некорректных действиях.

В программу входят тренинги по развитию периферийного зрения; упражнения на фиксацию взгляда, на подавление проговаривания прочитанного текста и др. Также реализована система мониторинга для контроля успехов пользователя.

Средствами для достижения целей и задач проекта явились среда программирования Delphi, где создаётся приложение, СУБД Access, в которой спроектирована база данных (БД). База данных связана с приложением посредством механизма ADO и архитектуры файл-сервер, к достоинствам которой относятся относительная простота её создания и обслуживания, наличие собственного интерпретатора языка SQL, а также высокая скорость доступа к данным. С помощью технологии OLE-Automation организуется связь с приложением MSExcel, в котором созданы отчёты.

Основные особенности использования технологии ADO не зависят от архитектуры БД: эта технология характерна не только для файл-серверных БД, но также и для клиент-серверных и трёхзвенных БД.

Основным достоинством технологии ADO является её естественная ориентация на создание «облегчённого» клиента. В рамках этой технологии на машине разработчика БД устанавливаются базовые объекты MS ADO и соответствующие компоненты Delphi, обеспечивающие использование технологии ADO (эти установки осуществляются автоматически при развёртывании Delphi). На машине сервера данных (это может быть файловый сервер в рамках файл-серверной технологии или машина с сервером данных — в технологии клиент-сервер) устанавливается так называемый провайдер данных — некоторая надстройка над специальной технологией OLE DB, «понимающая» запросы объектов ADO и «умеющая» переводить эти запросы в нужные действия с данными. Взаимодействие компонентов ADO и провайдера осуществляется на основе универсальной для Windows технологии ActiveX, причём провайдер реализуется как COM-сервер, а ADO-компоненты — как COM-клиенты.

При помощи Delphi можно создать приложения посредством интерактивного выбора и перетягивания на форму компонентов из Component Palette. При этом он сам создаст весь необходимый код. Ещё одно важное достоинство — наличие быстрого и эффективного компилятора. Это лучший «помощник», который извещает об ошибках и даёт подсказки, необходимые для создания более понятного кода. Вывод сообщений сразу о нескольких ошибках существенно экономит время, требуемое для их устранения.



Важным подспорьем являются уже существующие модули. Расширяемость Delphi даёт возможность интегрировать в среду различные компоненты и подпрограммы. Можно расширить их функциональность, дополнить своими пунктами меню и т. д. Delphi имеет ориентацию на работу с базами данных. Это позволяет с лёгкостью управлять информацией, создавая программы, работающие с базами данных.

## 4. Программная реализация электронного обучающего комплекса для отработки навыков скорочтения

### 4.1. Описание инструментов среды разработки

Рассмотрим особенности использования ADO технологии. На машине сервера создаётся и размещается источник данных. В случае файл-серверных систем отдельные таблицы типа dBASE, FoxPro, Paradox и т. п. должны управляться соответствующим ODBC-драйвером, а в роли провайдера используется Microsoft OLE DB Provider for ODBC drivers (рис. 1).

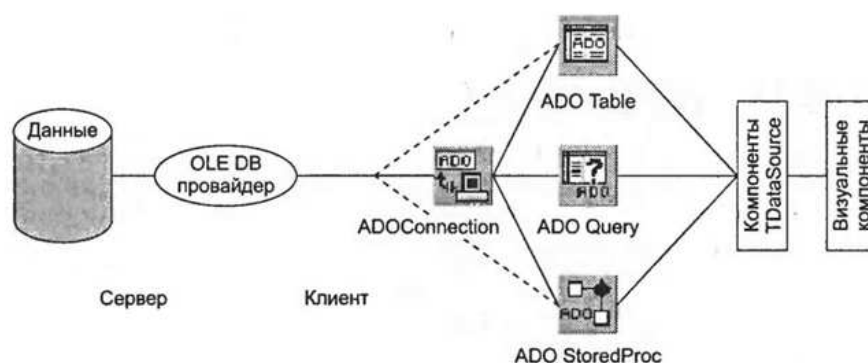


Рис. 1. Реализация технологии ADO в Delphi

На машине клиента располагаются связанные компоненты TADOConnection и компоненты-наборы данных TADOTable, TADOQuery, TADOStoredProc. Компонент TADOConnection играет роль концентратора соединений с источником данных компонентов-наборов, и в этом смысле он подобен компоненту TDatabase в традиционной архитектуре с BDE.

Технология OLE (Object Linking and Embedding — связывание и внедрение объектов) является стандартным для Windows средством обмена данными между независимыми приложениями. В соответствии с этой технологией одна программа (клиент) может содержать в своём составе данные (объекты), которые созданы (или по её требованию создаются) другой программой (сервером). Типичным примером могут служить таблицы, созданные и обслуживаемые табличным процессором MS Excel и вставленные в документ, подготовленный текстовым процессором MSWord.

База данных (БД) — это поименованная совокупность структурированных данных, относящихся к определённой предметной области.



Рис. 2. Логическая структура обучающего комплекса

Необходимо создать базу данных слов и словосочетаний, обучающихся скорочтению, исследовать заданную предметную область, построить логическую модель, установить связи между объектами.

В реляционных системах управления базами данных отношения представлены в виде таблиц, строки которых соответствуют кортежам или записям, а столбцы — атрибутам отношений, доменам, полям.

Поле, каждое значение которого однозначно определяет соответствующую запись, называется простым ключом (ключевым полем).

Чтобы связать две реляционные таблицы, необходимо ключ первой таблицы ввести в состав ключа второй таблицы.

Для разработки базы данных в проекте использовалась СУБД MSAccess. В данном проекте использовалось несколько связанных между собой таблиц в отношении один-ко-многим. Соответствующая логическая модель приведена на рис. 1.

#### 4.2. Описание этапов разработки

Работа по созданию программы проводилась в несколько этапов.

1. Сбор и анализ необходимой информации.

2. Разработка логической структуры программы и определение путей навигации по её разделам.
3. Подключение базы данных к проекту.
4. Алгоритмизация и программирование.
5. Тестирование и отладка программы.

Вся необходимая информация по предметной области, а именно методика эффективного обучения скорочтению взята из различных информационных ресурсов.

Логическую структуру программы можно изобразить в виде схемы, представленной на рис. 2.

Для подключения базы данных к проекту Delphi необходимо создать модуль данных, в котором размещаются ADO компоненты для связи базы данных и приложения (рис. 3).

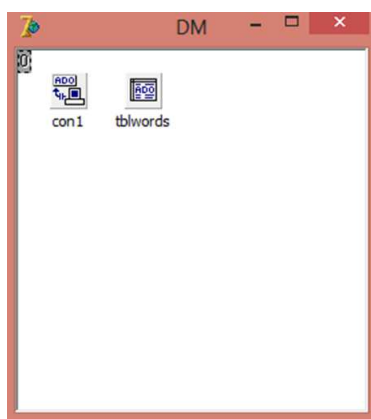


Рис. 3. Разработка модуля данных

После размещения в окне компонента ADOConnection(ADODB) прописываем строку соединения (рис. 4).



Рис. 4. Строка соединения с базой данных

В свойствах каждой ADO таблицы указываем имя и связываемую с ней таблицу базы данных. Это свойства Name и TableName. Для каждой таблицы также размещаем компоненты DataSource. Добавляем в таблицы объекты-поля (рис. 5).

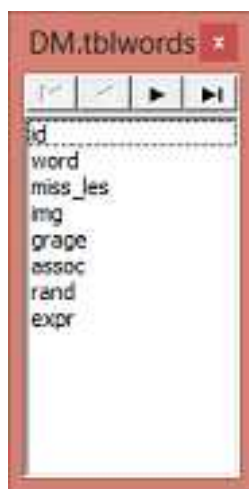


Рис. 5. Объекты-поля

Далее программируем основные модули программы. Опишем алгоритмы их работы.

## 5. Описание алгоритмов работы приложения

### 5.1. Пользовательский интерфейс и руководство пользователя для работы с программой

При запуске программы появляется главное окно программы (см. рис. 6), на котором размещено меню и панель инструментов для быстрого доступа. При выборе пункта «Пирамиды» откроется окно тренажёра (рис. 7), из которого можно перейти на тот или иной тренажёр, посмотреть методические рекомендации, отредактировать учебные данные, а также получить справку по использованию комплекса и правила пользования тем или иным тренажёром.

В окне тренажёра «Таблицы Шульте» (см. рис. 8) каждая таблица представляет собой разграфлённый на ячейки квадрат с вписанными в ячейки в хаотичном порядке числами или буквами.

Таблицы Шульте — это фактически модель страницы текста. Главное при работе с таблицами Шульте — расширение поля зрения.

Перед началом работы с таблицей взгляд фиксируется в её центре, при этом ставится задача видеть всю таблицу целиком.

При стремлении охватить взором всю таблицу Шульте раздражаются периферические зоны сетчатки глаза и формируются нейронные ансамбли, обеспечивающие съём и обработку информации с резервных зон поля зрения. Фикси-



Рис. 6. Главное меню программы

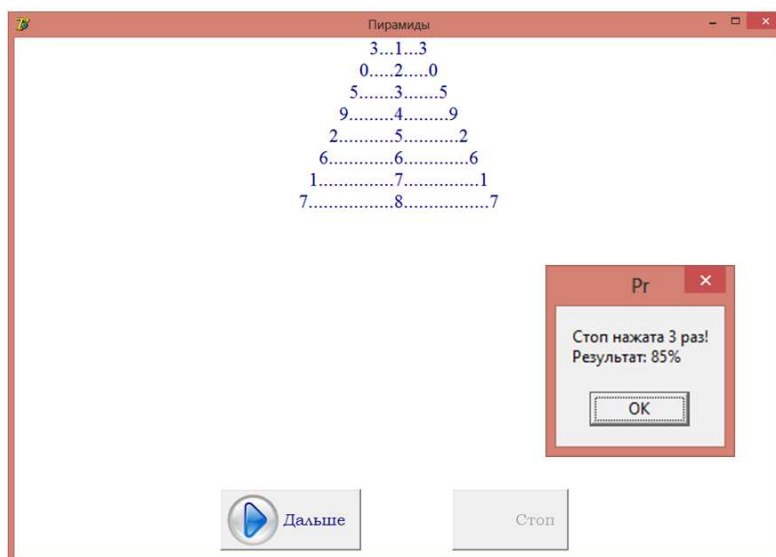


Рис. 7. Окно тренажёра «Пирамида»



Рис. 8. Окно тренажёра «Таблицы Шульте»

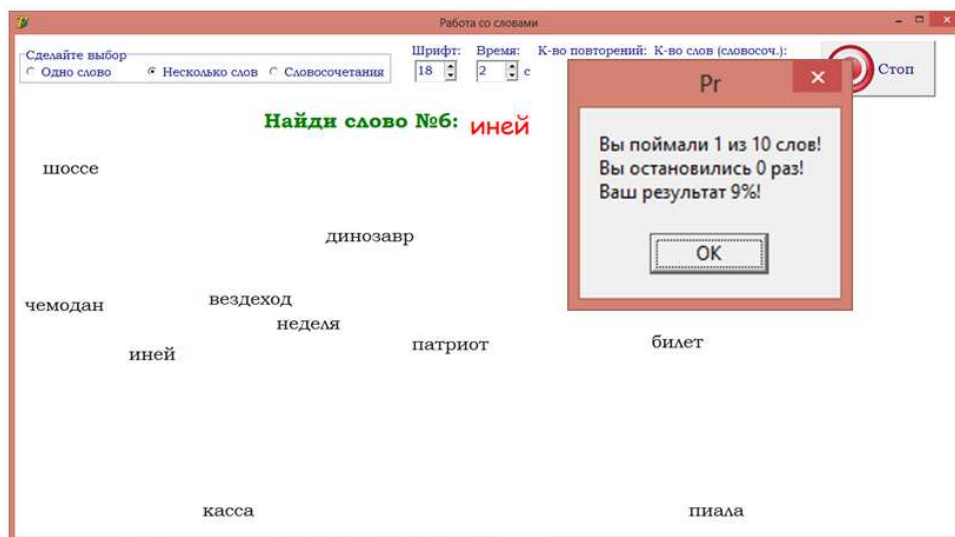


Рис. 9. Окно тренажёра «Работа со словами»

руя взгляд в центре таблицы, ученик как бы отправляет в матрицу оперативной памяти мозга панораму всей таблицы.

В окне «Работа со словами» (см. рис. 9) реализована работа со словами. У пользователя есть несколько вариантов выбора: работать с одним словом или с двумя, или приступить к выполнению задания со словосочетаниями. Данная методика скорочтения направлена на устранение повторного прочитывания текста; выполнение упражнений способствует развитию периферического зрения; увеличению скорости чтения; выработке навыка распознавания слов прямо по их визуальному типографическому образу. Залог успешного обучения скорочтению, как и любому другому навыку, — постоянные тренировки и контроль процесса обучения.

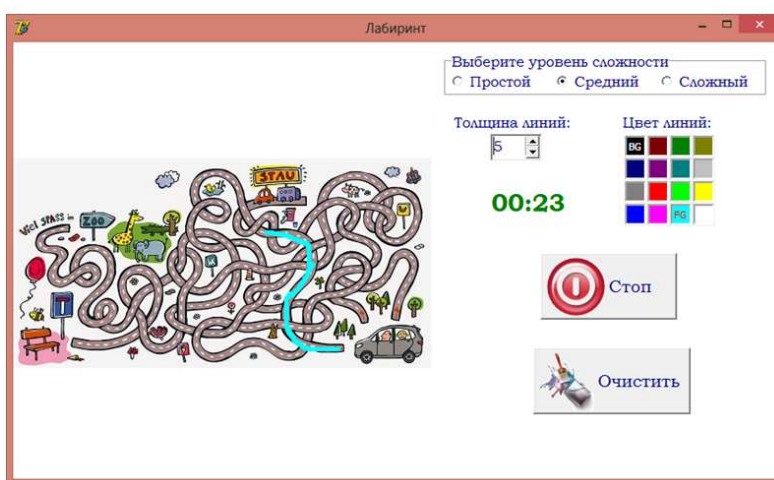


Рис. 10. Окно тренажёра «Лабиринт»

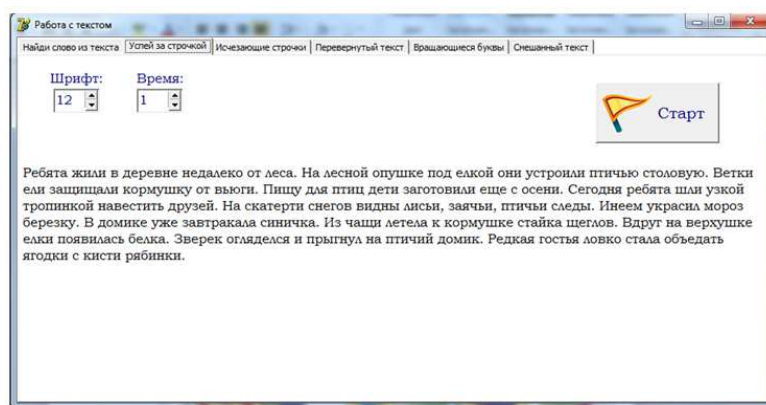


Рис. 11. Окно тренажёра «Работа с текстом»

Тренажёр «Лабиринт» (рис. 10) предназначен для расширения угла зрения, что позволяет заметно увеличить скорость чтения за счёт исключения движе-

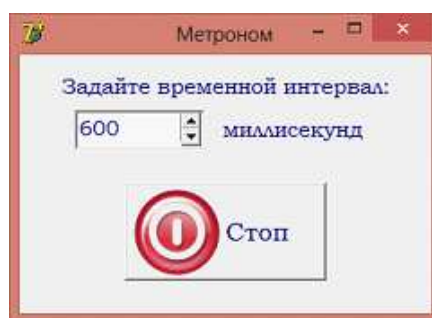


Рис. 12. Окно «Метроном»

ния глаз по строке.

Для устранения внутреннего проговаривания текста в программе слова подаются на экран одно за другим, практически не оставляя пользователю времени на проговаривание слов.

В приложении можно задать временной интервал метронома. Это способствует тому, что данный программный комплекс допускает индивидуальный подход к пользователю (рис. 12).

## 6. Заключение

Представленный в статье проект призван повысить эффективность обучения школьников скорочтению. В наше время школа требует от учителей введения и владения инновационными технологиями. Одной из таких технологий является автоматизация процесса обучения.

Обучающий электронный комплекс, разработанный в рамках данного проекта, поможет школьникам отработать практические навыки быстрого чтения. Приложение способно повысить интерес и мотивацию к данным занятиям, а также разнообразит занятия и поможет учителю автоматизировать процесс преподавания.

К особенностям программы, разработанной в проекте, относится возможность редактирования и создания базы данных учащихся и материалов к тренировкам.

К техническим характеристикам программы относятся:

1. Надёжность, которая включает предупреждение ошибок и сбоев работы программы при некорректных действиях пользователя.
2. Оперативность — высокая скорость обработки информации.
3. Функциональность — наличие исчерпывающих функций для проведения учителем тестирования по предмету.
4. Дружелюбный интерфейс и удобство работы.
5. Ориентированность на пользователя как с высоким, так и с низким уровнем владения компьютерной техникой.

В ходе работы над проектом были проанализированы теоретические вопросы



по методике обучения быстрому чтению, аспекты применения автоматизированных систем. Используются алгоритмические структуры и технологии языка программирования Delphi, а также технология OLE Automation для обмена данными между Delphi и табличным процессором Excel.

#### ЛИТЕРАТУРА

1. Андреев О.А., Хромов Л.Н. Учитесь быстро читать. URL: <https://profilib.net/chtenie/41474/oleg-andreev-uchites-bystro-chitat.php> (дата обращения: 01.07.2018).
2. Зиганова М. Скорочтение. М., 2013.

#### SOFTWARE FOR TEACHING CHILDREN OF LOWER GRADES FAST READING

**L.A. Volodchenkova**

Ph.D. (Biology), Associate Professor, e-mail: VolodchenkovaLA@omsu.ru

**S.I. Korolev**

Student, e-mail: korolev131093@gmail.com

Dostoevsky Omsk State University, Omsk, Russia

**Abstract.** The article the software application for automatization of the process of children teaching to fast reading.

**Keywords:** fast reading, software application, techniques of rapid reading.

*Дата поступления в редакцию: 14.07.2018*

## ПОЛУАВТОМАТИЧЕСКОЕ УПРАВЛЕНИЕ ПАМЯТЬЮ ПРИЛОЖЕНИЯ

**А.Д. Городецкий**

аспирант, e-mail: overln2@gmail.com

**Д.Н. Лавров**

к.т.н., доцент, e-mail: lavrov@omsu.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

**Аннотация.** В статье представлен подход к управлению памятью приложения на основе полуавтоматического управления, который сочетает в себе удобства автоматического «сборщика мусора» и полностью ручного управления, как это принято в языке С. Это достаточно гибкий подход, который может менять виды менеджеров памяти во время выполнения программы и/или даже в зависимости от платформы.

**Ключевые слова:** сборщик мусора, память приложения, алгоритмы управления памятью.

### Введение

В работах [1, 2] рассматривается более 100 алгоритмов менеджмента динамической памяти программ. Каждый из этих алгоритмов имеет свои достоинства и недостатки. Есть высокая вероятность, что менеджер памяти, работающий максимально эффективно на любых сценариях, так и не будет создан.

По этой причине языки системного уровня (С / С++ / Rust), позволяющие манипулировать сырой памятью программ, до сих пор широко применимы. Языки, предназначенные для создания прикладных программ, предлагают автоматическое управление памятью по какому-то одному алгоритму. Некоторые среды, например Oracle HotSpot VM, позволяют выбрать один из нескольких алгоритмов на этапе старта программы.

В итоге мы имеем 2 полярных метода — полностью ручное управление памятью, что приводит к необходимости контроля корректности каждой строки кода, или же абсолютно автоматическое управление (пусть хоть и одним из нескольких способов).

На самом деле прикладные программы работают иначе. Любая программа — это совокупность подпрограмм (подсистем), алгоритм работы каждой из которых подразумевает использование какого-то конкретного менеджера памяти, максимально эффективного в данном случае. Более того, часто эти подсистемы фактически имеют изолированные сегменты памяти.

Ручное управление памятью максимально эффективно с точки зрения машинного времени, но максимально неэффективно в качестве использования

временных затрат на разработку. Автоматическое — наоборот. Стоит ли нам рассмотреть гибридные схемы? Схемы ручного управления памятью в *подсистемах* программ, а не в строчках кода. Внутри же самой подсистемы память будет управляться автоматически. Назовём такую схему управления *полуавтоматической*.

## 1. Полуавтоматическое управление памятью

Реализация подразумевает как поддержку на этапе компиляции, так и на этапе исполнения программы. На этапе компиляции в код программы должны быть добавлены вызовы указателей на функции сборщика мусора. А на этапе исполнения должна быть возможность эти функции подменить для переключения блоков кода (подсистем) между разными сборщиками мусора.

Также на этапе компиляции должна быть организована проверка корректности управления памятью на уровне блоков в соответствии с двумя правилами.

1. Контроль времени жизни — переменные, выделенные в одном блоке (менеджере памяти) не могут его пережить.
2. Блоки кода могут разделять переменные только на чтение.

Вопросы корректности требуют отдельного рассмотрения и выходят за рамки данной статьи.

## 2. Обобщённый интерфейс сборщика мусора

Алгоритмы менеджмента памяти можно разделить на 3 категории: трассирующие, подсчёт ссылок, арены. Кратко их рассмотрим.

*Трассирующие сборщики* — в классическом варианте освобождают память на этапе запроса новой памяти. Реально собирают не мусор, а достижимые объекты. В большинстве случаев осуществляют дефрагментацию памяти путём копирования достижимых объектов в новый регион.

*Подсчёт ссылок* — дополнительно для каждого объекта хранят счётчик ссылок на этот объект. При каждой перезаписи переменных или полей объектов счётчики ссылок обновляются.

*Арена* — изначально выделяет регион памяти достаточно большого размера. В момент запроса на создание объекта отрезает от региона фрагмент, равный размеру объекта. Реально сборку мусора не осуществляет. Все объекты удаляются во время освобождения области памяти, выделенной ареной.

Для возможности использования этих трёх типов менеджеров памяти компилятор должен делегировать менеджеру памяти следующие вызовы:

1. Создание объекта.
2. Выход объекта из области видимости.
3. Запись одного объекта в поле другого; или одной переменной в другую.

Так как трассирующие сборщики мусора используют корни (gc roots) [1] как стартовые узлы для обхода графов объектов, то нам следует явно разграничить вызовы для создания (выхода из области видимости) локальных переменных (alloc root) и неименованных объектов (alloc). Функции, необходимые

для реализации менеджера памяти, в зависимости от типа менеджера указаны в таблице 1.

Менеджерам памяти нужна информация о типах объектов. В простейшем случае (C malloc) в функцию выделения памяти передаётся размер объекта. Однако в общем случае также необходима информация о выравнивании, типе объекта, ссылках на другие объекты, деструкторах.

Определим структуру `tinfo`

```
typedef void (*walker_routine)
            (void* ctx, void* obj, tinfo* info);
typedef void (*walker)
            (void* ctx, walker_routine callback);
typedef void (*destructor)
            (void* obj);

struct tinfo {
    const char* tname;
    sizeof size;
    sizeof align;
    destructor destroy;
    walker walk;
}
```

Указатель на функцию `walker` позволяет рекурсивно обойти все дочерние поля объекта и совершить над ними какую-либо конкретную операцию, например, уничтожить, вызвав деструктор. Выразим общий интерфейс менеджера памяти на языке C:

```
void* alloc(tinfo* info)
void* alloc_root(tinfo* info)
void free(tinfo* info, void* obj)
void free_root(tinfo* info, void* obj)
void store(tinfo* info, void* dest, void* src)
```

В случае, когда менеджеру памяти не нужна какая-то часть вызовов, используются заглушки.

### 3. Поддержка на этапе исполнения

На этапе исполнения нам нужна возможность подменять указатели на функции менеджера памяти. Однако нужно не допустить конфликтов на уровне потоков исполнения команд (системных потоков). Для этого глобальные переменные указателей на функции должны быть объявлены как `thread_local[]`. Thread local переменные широко поддерживаются линковщиками, и на многих архитектурах (например x86-64) не вносят дополнительных расходов.

Алгоритм замены менеджера памяти:

Таблица 1. Функции, необходимые для реализации менеджера памяти

	<b>trace gc</b>	<b>ref count</b>	<b>arena</b>
<b>alloc</b>	+	+	+
<b>alloc root</b>	+	=alloc	=alloc
<b>free</b>	stub	+	stub
<b>free root</b>	stub	=free	stub
<b>store</b>	stub	+	stub

1. Сохранить указатели на функции родительского менеджера памяти.
2. Записать свои указатели на функции.
3. Для получения памяти самому необходимо использовать функции *родительского* менеджера памяти.
4. При деактивации вернуть указатели на функции родительского менеджера памяти.

```
# pool
def withPool = self: MemPool,
    action: () -> None do
  oldAlloc = loadAlloc() # состояние старого аллокатора
  storeAlloc(
poolAlloc,
poolRootAlloc,
poolFree,
poolRootFree,
poolStore) # подменяем на себя
  action() # вызываем пользовательский код под пулом
  storeAlloc(oldAlloc) # возвращаем родительский менеджер

# использование
pool = mkMemPool(conf)
pool withPool lambda
  doSomeCrazyStuff()
```

Делегирование собственных запросов на память родительскому менеджеру позволяет выстраивать иерархии менеджеров памяти с полным контролем лимитов на память.

## 4. Заключение

Данная схема позволяет получить контроль над управлением памятью в программе, достаточный для требуемого уровня производительности. Чем требуемый уровень производительности выше, тем меньшими блоками кода может

понадобиться управлять. Мы можем очень гибко варьировать производительность итоговой программы, внося дополнительное ручное управление памятью.

Главным преимуществом по сравнению с ручным управлением является то, что для изменения схемы управлением памятью в каком-то блоке кода (рекурсивно на все вызовы) нет необходимости переписывать *каждую строчку кода*. Достаточно лишь переключить блок кода на новый менеджер и согласовать вход и выход данных. Один и тот же библиотечный код может работать на *разных алгоритмах менеджмента памяти* без каких-либо изменений.

Главным преимуществом по сравнению с автоматическим управлением является то, что вы получаете контроль над памятью, и можете получить дополнительную производительность на этапе *оптимизации*, если вам будет это необходимо.

Данная реализация достаточно проста и близка к libc API для работы с динамической памятью. Накладные расходы на вызов заглушек (stub) достаточно малы, что позволяет иметь скорость, сравнимую с C при использовании арен.

## Благодарности

Выражаем признательность Елене Анатольевне Костюшиной и Евгению Александровичу Тюменцеву за обсуждение результатов и конструктивную критику.

## ЛИТЕРАТУРА

1. Jones R., Hosking A., Moss E. The Garbage Collection Handbook: The Art of Automatic Memory Management // Applied Algorithms and Data Structures series. Chapman & Hall/CRC, 2011. 511 p.
2. Bartlett J. Inside memory management: The choices, tradeoffs, and implementations of dynamic allocation // IBM. Developer Works. 2004. URL: <https://www.ibm.com/developerworks/linux/library/l-memory/> (дата обращения: 03.09.2018).

## SEMI-AUTOMATIC APPLICATION MEMORY MANAGEMENT

**A.D. Gorodetsky**

Postgraduate Student, e-mail: overln2@gmail.com

**D.N. Lavrov**

Ph.D.(Eng.), Associate Professor, e-mail: lavrov@omsu.ru

Dostoevsky Omsk State University, Omsk, Russia

**Abstract.** The article presents an approach to memory management of an application based on semi-automatic control that combines the convenience of an automatic "garbage collector" and completely manual control, as is customary in the C language. This is a fairly flexible approach that can change the types of memory managers during program execution and / or even depending on the platform.

**Keywords:** garbage collector, application memory, memory management algorithms.

*Дата поступления в редакцию: 01.09.2018*

## **ПРОБЛЕМЫ ПОСТРОЕНИЯ ПОЛИТИКИ БЕЗОПАСНОСТИ ПРИ ОБЪЕДИНЕНИИ ИНФОРМАЦИОННЫХ СИСТЕМ**

**С.В. Белим**

профессор, д.ф.-м.н., профессор кафедры информационной безопасности,  
e-mail: sbelim@mail.ru

**С.Ю. Белим**

доцент, к.п.н., доцент кафедры информационной безопасности,  
e-mail: svbelim@gmail.com

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

**Аннотация.** В статье рассмотрена проблема построения политики безопасности при объединении двух информационных систем как оптимизационная задача. В данной задаче выделены два параметра — конфиденциальность и доступность информации. Записана целевая функция. Рассмотрены варианты объединения систем с одинаковыми моделями безопасности. Исследована оптимальность предложенных решений.

**Ключевые слова:** политика безопасности, дискреционное разграничение доступа, мандатное разграничение доступа.

### **Введение**

Проблема объединения информационных систем с самостоятельными политиками безопасности возникает в большом количестве прикладных задач. Прежде всего такая задача является актуальной при объединении сетей двух независимых организаций с требованием непрерывной работы всех систем. Идеальной будет ситуация полного пересмотра всех правил разделения доступа и создание единой логически выстроенной политики безопасности. Однако в большом количестве случаев ставится требование обеспечения непрерывности всех процессов, что делает невозможной полную пересборку системы.

Существует ряд ситуаций в локальных системах, когда также возникает ситуация двух политик безопасности. Прежде всего необходимость выполнения двух политик безопасности возникает при установке в рамках операционной системы прикладного программного обеспечения с собственными правилами разграничения доступа. К таким прикладным системам можно отнести виртуальные машины, системы управления базами данных, комплексные системы защиты информации и т. д. Например, при подключении к базе данных пользователя СУБД обеспечивает разграничение доступа к табличным пространствам и таблицам, как правило, в рамках ролевой политики безопасности. С другой

стороны СУБД осуществляет хранение таблиц в файлах, к которым пользователь может получить доступ средствами операционной системы. Несмотря на то, что СУБД, как правило, защищает свои файлы от внесения изменений другими процессами, пользователь, обладая определённым набором прав, может это сделать. Аналогичная ситуация с виртуальными машинами, виртуальный жёсткий диск которых для операционной системы представляет собой обычный файл. В связи с развитием технологии виртуализации распространённым явлением стали базы данных, установленные на виртуальном сервере. Для обеспечения безопасности таких систем необходим учёт сразу трёх политик безопасности. Также следует упомянуть о часто встречающейся задаче объединения баз данных.

До настоящего времени проблема построения объединённой политики безопасности решалась преимущественно для специализированных систем [1–4].

Целью данной статьи ставится построение формальной модели объединения двух и более политик безопасности в рамках субъектно-объектной модели.

## 1. Объединение информационных систем

Рассмотрим процесс объединения информационных систем с точки зрения формального субъектно-объектного подхода построения политики безопасности. Пусть множество объектов первой системы  $O_1$ , а второй —  $O_2$ . Соответственно, множества субъектов  $S_1$  и  $S_2$ . В новой объединённой информационной системе множество объектов  $O = O_1 \cup O_2$ , а множество субъектов —  $S = S_1 \cup S_2$ . Пусть в первой системе действует политика безопасности  $P_1$ , а во второй —  $P_2$ . Необходимо построить политику безопасности  $P$  для новой объединённой информационной системы. Все вопросы безопасности решаются рассмотрением доступов субъектов к объектам. Политика безопасности  $P_1$  определяет легитимность доступов субъектов из множества  $S_1$  к объектам из множества  $O_1$ . Политика безопасности  $P_2$  определяет легитимность доступов субъектов из множества  $S_2$  к объектам из множества  $O_2$ . При объединении необходимо разграничить доступ субъектов из множества  $S_1$  к объектам из множества  $O_2$  и субъектов из множества  $S_2$  к объектам из множества  $O_1$ . Каждая политика безопасности использует определённые дополнительные структуры данных, на основе которых определяются права доступа. В дискреционной политике безопасности такой структурой является матрица доступов, в мандатной политике безопасности — метки безопасности, в ролевой политике безопасности — списки полномочий ролей. При расширении множеств объектов и субъектов структуры данных политик безопасности требуют расширения с учётом новых элементов. Так структуры данных политики безопасности  $P_1$  должны быть доопределены для элементов множеств  $S_2$  и  $O_2$ , а структуры данных политики безопасности  $P_2$  должны быть доопределены для элементов множеств  $S_1$  и  $O_1$ . При этом может возникнуть проблема, связанная с тем, что одни и те же доступы разграничиваются двумя независимыми политиками безопасности. Две политики безопасности могут принимать различные решения для одного и того же доступа. В дальнейшем эту ситуацию будем называть проблемой



согласования политик безопасности.

При решении проблемы согласования политик возможны два различных подхода. Первый подход согласования политик безопасности, будем называть его «жестким», состоит в том, что доступ разрешён тогда и только тогда, когда он разрешён обеими политиками безопасности. В этом случае не появляется дополнительных разрешённых доступов, однако могут появиться дополнительные запреты на доступ, что может приводить к нежелательным последствиям в вопросах доступности информации и снижать производительность системы. Второй подход, будем называть его «мягким», состоит в том, что доступ разрешён, если он разрешён хотя бы одной из политик безопасности. В этом случае не появляется новых запретов на доступ, что положительно сказывается на доступности информации, однако могут возникнуть дополнительные разрешённые доступы, которые являются потенциальными каналами утечки информации. Таким образом, задачу согласования политик безопасности можно рассматривать как оптимизационную с двумя параметрами — конфиденциальность и доступность.

Кроме рассмотренных двух крайних вариантов согласования, жесткого и мягкого, возможен и промежуточный вариант, в котором для каждого из доступов выбирается доминирующая политика безопасности. В этом случае система принимает решение, основываясь на политике безопасности  $P_1$  или  $P_2$  в зависимости от параметров доступа и текущего состояния системы. Данный вариант согласования политик безопасности будем называть «полужестким».

Определим целевую функцию оптимизационной задачи согласования политик безопасности. Необходимо, чтобы минимальным было как количество новых запрещённых доступов  $D$ , так и новых разрешённых доступов  $A$ . Используем метод весовых коэффициентов, тогда целевая функция будет иметь вид:

$$F = k_1 D + k_2 A,$$

где  $k_1, k_2$  — весовые коэффициенты, показывающие относительную важность доступности и конфиденциальности в конкретной задаче, причём  $k_1 + k_2 = 1$ .

Для жесткого согласования политик безопасности  $F = k_1 D$ , для мягкого —  $F = k_2 A$ . Для полужесткого варианта согласования политик безопасности необходимо выбирать алгоритм принятия решения о доступе, для которого  $F \rightarrow \min$ . Следует учитывать, что всегда  $F \geq 0$ . Если какое-то решение обеспечивает нулевое значение целевой функции, то оно является оптимальным.

Проблема согласования политик безопасности существенно зависит от их типа. В различных ситуациях возможны различные проблемы и решения. На сегодняшний день распространены три принципиально разных типа политик безопасности — дискреционная, мандатная и ролевая. Возможно объединение двух однотипных или разнотипных политик безопасности. Всего существует шесть различных вариантов объединения.

Ряд вопросов объединения политик безопасности уже был рассмотрен авторами в более ранних статьях. Проблемы совмещения ролевой и мандатной политик безопасности были рассмотрены в статье [5]. В работе [6] представлен подход к объединению двух мандатных политик безопасности. Объединение

двух ролевых моделей безопасности рассмотрено в статье [7]. Все эти работы связаны с жёстким вариантом совмещения политик безопасности. Полужёсткие варианты совмещения двух политик безопасности рассмотрены в статье [8].

## 2. Примеры объединения информационных систем

Рассмотрим два примера объединения двух информационных систем с одинаковыми моделями безопасности. Простейшим случаем является объединение двух систем с дискреционной политикой безопасности, права доступа в которых определяются матрицами доступов. Второй случай состоит в объединении двух мандатных политик безопасности, задаваемых различными решётками ценностей. В обоих случаях будем использовать классическое решение и проверять его на оптимальность.

### 2.1. Объединение двух дискреционных политик безопасности

Рассмотрим случай, когда обе политики безопасности являются дискреционными. Пусть политика безопасности первой системы  $P_1$  определяется матрицей доступов  $M_1$ , а политика безопасности второй системы  $P_2$  определяется матрицей доступов  $M_2$ . Разграничение доступа в объединённой системе может быть сведено к дискреционной политике безопасности, матрица доступов  $M$  которой будет иметь вид:

$M_1$	$M_3$
$M_4$	$M_2$

Подматрицы  $M_3$  и  $M_4$  определяются произвольно администратором независимо от  $M_1$  и  $M_2$ . При таком объединении не возникает ни новых разрешённых доступов ( $A = 0$ ), ни новых запретов на доступ ( $D = 0$ ). Следовательно, целевая функция  $F = 0$  и решение является оптимальным.

### 2.2. Объединение двух мандатных политик безопасности

Пусть для первой системы выполняется политика безопасности  $P_1$  на основе решётки ценностей  $L_1$ , а для второй системы —  $P_2$  на основе решётки ценностей  $L_2$ . Традиционное решение состоит в построении объединённой политики безопасности  $P$  с решёткой ценностей, определяемой как декартово произведение исходных решёток  $L = L_1 \times L_2$ . Рассмотрим оптимальность такого решения. Очевидно, что «лишних» доступов в этом случае не возникает, поэтому  $A = 0$ . Однако у каждого объекта возникают метки безопасности от «чужой» решётки ценностей, что может приводить к запрету доступов, существовавших ранее. В связи с этим в общем случае  $D > 0$ , откуда следует  $F > 0$  и решение не является оптимальным. Оптимальности можно добиться, используя «мягкие» методы объединения политик безопасности.

## Заключение

В заключении выделим основные проблемы, возникающие при объединении двух и более политик безопасности.

1. Приведение структур данных политик безопасности различного типа к универсальному виду.
2. Трудности проведения формального анализа безопасности вследствие большого объёма и сложности структуры данных политики безопасности.
3. Трудности анализа рисков, связанные с большим количеством прав, привилегий и активных сущностей системы.
4. Выбор политики безопасности, обеспечивающей минимальные угрозы конфиденциальности при минимальном снижении доступности данных.

Все эти проблемы не могут быть решены в едином ключе и требуют рассмотрения большого количества частных случаев.

## ЛИТЕРАТУРА

1. Bonatti P., de Capitani di Vimercati S., Samarati P. A modular approach to composing access control policies // Proceedings of the 7th ACM conference on Computer and communications security, Athens, Greece, (CCS '00). 2000. P. 164–173.
2. Filippopolitis A., Gelenbe E. A distributed decision support system for building evacuation // Proceedings of the 2nd conference on Human System Interactions (HSI'09). 2009. P. 320–327.
3. Flechais I., Mascolo C., Sasse M.A. Integrating security and usability into the requirements and design process // International Journal of Electronic Security and Digital Forensics. 2007. V. 1(1). P. 12–26.
4. Rezaeibagha F., Mu Y. Access Control Policy Combination from Similarity Analysis for Secure Privacy-Preserved EHR Systems // IEEE Trustcom/BigDataSE/ICSS. 2017. P. 386–393.
5. Белим С.В., Богаченко Н.Ф., Ракицкий Ю.С. Теоретико-графовый подход к проблеме совмещения ролевой и мандатной политик безопасности // Проблемы информационной безопасности. Компьютерные системы. 2010. № 2. С. 9–17.
6. Ракицкий Ю.С., Белим С.В. Модель совмещения двух мандатных политик безопасности // Безопасность информационных технологий. 2011. Вып. 1. С. 125–126.
7. Белим С.В., Богаченко Н.Ф. Построение ролевой политики безопасности на произвольном ориентированном графе // Проблемы информационной безопасности. Компьютерные системы. 2009. Вып. 3. С. 7–17.
8. Белим С.В., Богаченко Н.Ф., Ракицкий Ю.С. Совмещение политик безопасности, основанное на алгоритмах поддержки принятия решений // Информационно-управляющие системы. 2016. № 5(84). С. 66–72.

**PROBLEMS OF SECURITY POLICY CREATION AT INFORMATION SYSTEMS  
ASSOCIATION**

**S.V. Belim**

Dr.Sc. (Phys.-Math.), Professor, e-mail: sbelim@mail.ru

**S.Yu. Belim**

Ph.D. (Ped.), Associate Professor, e-mail: svbelim@gmail.com

Dostoevsky Omsk State University, Omsk, Russia

**Abstract.** In article the problem of security policy creation at two information systems association as an optimizing task is considered. In this task two parameters — confidentiality and availability of information are allocated. Criterion function is make up. Options of systems association with identical security models are considered. The optimality of the proposed solutions is investigated.

**Keywords:** security policy, discretionary access control, mandatory access control.

*Дата поступления в редакцию: 14.03.2018*

## Авторам

### Предоставляемые данные и документы

Автор предоставляет в редакцию:

- рукопись статьи в формате  $\LaTeX$  (см. требования к оформлению);
- список из трёх экспертов по тематике статьи, давших согласие написать рецензию на представленную работу<sup>1</sup>;
- экспертное заключение о возможности открытого опубликования.

### Лицензирование

Согласно ГК РФ ст. 1286 лицензионный договор с автором для публикации в периодических изданиях может быть заключён в устной форме. Сам факт получения рукописи статьи редколлегией журнала «Математические структуры и моделирование» является акцептом (принятием) лицензионного договора.

Все статьи в журнале «Математические структуры и моделирование» публикуются под лицензией Creative Commons Attribution 4.0 International (CC-BY). Текст лицензии находится по адресу <https://creativecommons.org/licenses/by/4.0/legalcode>.

### Требования к оформлению рукописи

К публикации принимаются рукописи объёмом не более 16 страниц.

Авторам необходимо предоставить следующую информацию на русском и английском языках:

- название статьи;
- список авторов с указанием
  - фамилии, имени и отчества,
  - учёного звания,
  - учёной степени,
  - должности,
  - места работы или учёбы,
  - действующего адреса электронной почты;
- аннотация (абстракт) объёмом от 100 до 250 слов;
- список ключевых слов.

Автор также указывает УДК (универсальный десятичный код) статьи. Его можно подобрать по тематике статьи в справочнике <http://msm.univer.omsk.su/udc/>.

Библиографические ссылки оформляются согласно ГОСТ 7.0.5–2008.

Рукопись статьи представляется в редакцию по электронной почте в двух форматах pdf и tex. Статья должна быть набрана с использованием макропакета  $\LaTeX$ и стиля msmb.cls, предоставляемого редакцией <http://msm.univer.omsk.su/files/msmb.zip>. Рекомендуется установить компилятор  $\text{MiKTeX}$ , так как именно им пользуются в редакции.

Отклонения в оформлении рукописи от приведённых правил позволяют редколлегии принять решение о снятии статьи с публикации. Статья может быть отклонена по причинам несоответствия тематике журнала или в связи с низким уровнем качества научного исследования.

В статье запрещается переопределять стандартные команды и окружения.

Нумеруемые формулы необходимо выделять в отдельную строку.

Нумерация только арабскими цифрами в порядке возрастания с единицы. Нумеровать следует только те формулы, на которые в тексте имеются ссылки.

---

<sup>1</sup>Необходимы полные данные экспертов (место работы, учёная степень, должность), с указанием способа связи с ними (e-mail, телефон). Редколлегия может обратиться к одному из экспертов из предложенного списка с просьбой написать рецензию или может назначить рецензента из собственного списка.

Запрещается использовать в формулах буквы русского алфавита. Если без них никак не обойтись, то следует использовать команду `\mbox{...}`.

Все рисунки и таблицы должны иметь подпись, оформленную с помощью команды `\caption{...}`.

Файлы с рисунками необходимо представить в формате PDF или EPS (использовать редакторы векторной графики типа InkScape, Adobe Illustrator или Corel Draw).

Используйте стандартные команды переключения на готический, каллиграфический и ажурный шрифты: `\mathfrak`, `\mathcal` и `\mathbb`.

Не допускается заканчивать статью рисунком или таблицей.

В списке литературы обязательно указание следующих данных: для книг — фамилии и инициалы авторов, название книги, место издания, издательство, год издания, количество страниц; для статей — фамилии и инициалы авторов, название статьи, название журнала, год издания, том, номер (выпуск), страницы начала и конца статьи (для депонированных статей обязательно указать номер регистрации).

Кавычки в русском тексте («абвгд») должны быть угловыми, в английском — прямыми верхними кавычками ("abcdeг" или "abcdeг").

Обязательна расшифровка сокращений при первом вхождении термина. Например: ... искусственный интеллект (ИИ)...

## Порядок рецензирования

Первичная экспертиза проводится главным редактором (заместителем главного редактора). При первичной экспертизе оценивается соответствие статьи тематике журнала, правилам оформления и требованиям, установленным редакцией журнала к научным публикациям.

Все статьи, поступившие в редакцию научного журнала «Математические структуры и моделирование», проходят через институт рецензирования.

Рецензент выбирается главным редактором журнала из числа членов редколлегии или ведущих специалистов по профилю данной работы.

Рецензенты уведомляются о том, что присланные им рукописи являются частной собственностью авторов и относятся к сведениям, не подлежащим разглашению. Рецензентам не разрешается делать копии статей для своих нужд.

Срок для написания рецензии устанавливается по согласованию с рецензентом.

Рецензия должна раскрывать актуальность представленного материала, степень научной новизны исследования, определять соответствие предлагаемого к публикации текста общему профилю издания и стиль изложения.

Рецензент выносит заключение о возможности опубликования статьи: «рекомендуется», «рекомендуется с учётом исправления замечаний, отмеченных рецензентом» или «не рекомендуется». В случае отрицательной рецензии редакция направляет автору мотивированный отказ, заверенный главным редактором или его заместителем.

В случае несогласия с мнением рецензента автор статьи имеет право предоставить аргументированный ответ в редакцию журнала. Статья может быть направлена на повторное рецензирование, либо на согласование в редакционную коллегию.

При наличии в рецензии рекомендаций по исправлению и доработке статьи автору направляется текст рецензии с предложением учесть их при подготовке нового варианта статьи или аргументированно (частично или полностью) их опровергнуть. Доработанная (переработанная) автором статья повторно направляется на рецензирование и рассматривается в общем порядке. В этом случае датой поступления в редакцию считается дата возвращения доработанной статьи.

После принятия редколлегией решения о допуске статьи к публикации автор информируется об этом и указываются сроки публикации.

Оригиналы рецензий хранятся в редакции в течение пяти лет.

### **Авторская этика**

Авторы публикаций должны гарантировать, что в список авторов включены только лица, соответствующие критериям авторства (лица, внесшие значительный вклад в работу), и что заслуживающие авторства исследователи не исключены из списка авторов.

Должны работать вместе с редакторами или издателями для скорейшего исправления своих работ в случае обнаружения в них ошибок или упущений после публикации.

Обязаны незамедлительно уведомлять редакцию в случае обнаружения ошибки в любой поданной ими на публикацию, принятой для публикации или уже опубликованной работе.

Не вправе копировать из других публикаций ссылки на работы, с которыми они сами не ознакомились; цитаты и ссылки на другие работы должны быть точными и оформленными в соответствии с предъявляемыми требованиями.

Должны ссылаться максимально правильно и точно на имеющие отношение к публикации предыдущие работы как других исследователей, так и самих авторов, обращаясь, прежде всего к первоисточнику; дословное воспроизведение собственных работ и их перефразирование неприемлемы, они могут быть использованы лишь в качестве основы для новых выводов.

Необходимо указывать авторство данных, текста, рисунков и идей, которые автор получил из других источников — они не должны представляться, как принадлежащие автору публикации; прямые цитаты из работ других исследователей должны выделяться кавычками и соответствующей ссылкой.

Должны соблюдать нормы законодательства о защите авторских прав; материалы, защищённые авторским правом (например, таблицы, цифры или крупные цитаты), могут воспроизводиться только с разрешения их владельцев.

### **Памятка для перевода должностей, учёных степеней и званий на английский язык**

Профессор = Professor

Доцент = Associate Professor

Старший преподаватель = Assistant Professor

Преподаватель = Instructor

Ассистент = Instructor

Аспирант = Postgraduate Student или Ph.D. Student

Соискатель = Ph.D. Doctoral Candidate

Магистрант = Master's Degree Student

Студент = Student

д.ф.-м.н. = Dr.Sc. (Phys.-Math.)

к.ф.-м.н. = Ph.D. (Phys.-Math.)

д.т.н. = Dr.Sc. (Eng.)

к.т.н. = Ph.D. (Eng.)

Инженер-программист = Software Engineer

Старший/младший научный сотрудник = Senior/Junior Scientist Researcher

### **Электронная почта для отправки статей**

lavrov@omsu.ru — зам. главного редактора (ответственный за выпуск) Д.Н. Лавров.

Научный журнал

# Математические структуры И моделирование

№3(47)

Главный редактор  
**А.К. Гуц**

Зам. глав. ред., выпускающий редактор  
**Д.Н. Лавров**

Зам. глав. ред., технический редактор  
**Н.Ф. Богаченко**

Корректор:  
**И.Н. Баловнева**

Проверка корректности перевода:  
**А.Н. Кабанов**

---

#### Адрес научной редакции

644077, Омская обл., г. Омск, пр-т Мира, д. 55а,  
Омский государственный университет

E-mail: [guts@omsu.ru](mailto:guts@omsu.ru), [lavrov@omsu.ru](mailto:lavrov@omsu.ru)

Электронная версия журнала:

<http://msm.univer.omsk.su>

<http://msm.omsu.ru>



---

Подписано в печать 21.09.2018. Формат 60 × 84 1/8.  
Усл. печ. л. 15,7. Тираж 100 экз. Заказ № 234.

---

Отпечатано на полиграфической базе издательства ОмГУ им. Ф.М. Достоевского  
644077, Омская обл., г. Омск, пр-т Мира, д. 55а



ISSN 2222-8772



9 772222 877005



18047 >