

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 22  
2016  
№ 2

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ

Издательство "Новые технологии"

## СОДЕРЖАНИЕ

### МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Кухаренко Б. Г., Солнцева-Чалей М. О. Моделирование центроидов для пучков многомерных траекторий . . . . . 83
- Матвеев М. Г., Михайлов В. В., Сирота Е. А. Комбинированная прогностическая модель нестационарного многомерного временного ряда для построения пространственного профиля атмосферной температуры . . . . . 89
- Струченков В. И. Динамическое программирование в задачах планирования реализации частично возобновляемых ресурсов . . . . . 94
- Новикова Н. М., Борискин А. В. Математическая модель поиска влиятельных объектов социальной сети на основе априорной информации . . . . . 99

### ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Ахмедов М. З. Решение задачи прогнозирования с применением нечеткой импликации . . . . . 103
- Бу Вьет Тханг. Ускорение алгоритма кластеризации DBSCAN за счет использования алгоритма K-Means . . . . . 109

### СТРУКТУРНЫЙ СИНТЕЗ И ПРИНЯТИЕ РЕШЕНИЙ

- Захаров А. В., Рамазанова Р. Р., Курилова И. С. Управление системой, состоящей из набора типовых подсистем, с помощью системы показателей . . . . . 116

### ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

- Тарасов В. Н., Бахарева Н. Ф., Липилина Л. В. Математическая модель телеграфика на основе системы G/M/1 и результаты вычислительных экспериментов. . . 121

### СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

- Амирханов А. В., Гладких А. А., Глушко А. А., Макачук В. В., Шахнов В. А. Автоматизированная система формирования топологической информации для изготовления комплекта фотошаблонов сверхбольших интегральных схем . . . . . 127
- Гридин В. Н., Анисимов В. И., Дмитриевич Г. Д., Ларистов А. И., Аль-Шамери Я. М. Организация информационного обеспечения web-ориентированных схемотехнических САПР . . . . . 134
- Сурков А. В. Использование статического временного анализа для проектирования самосинхронных схем . . . . . 139

### ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

- Бершадский А. М., Бождай А. С., Мкртчян В. С. Принципы построения общедоступной самоадаптирующейся системы дистанционного обучения на основе модели изменчивости и сервис-ориентированной архитектуры . . . . . 146

### ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИМЕДИЦИНСКИХ СИСТЕМАХ

- Кухтичев А. А., Клёнов Е. А., Скородумов С. В. Разработка архитектуры информационной системы "ЦифроМед" цифровой медицины в авиации и космонавтике . . 154

Главный редактор:

СТЕМПКОВСКИЙ А. Л.,  
акад. РАН, д. т. н., проф.

Зам. главного редактора:

ИВАННИКОВ А. Д., д. т. н., проф.  
ФИЛИМОНОВ Н. Б., д. т. н., с.н.с.

Редакционный совет:

БЫЧКОВ И. В., акад. РАН, д. т. н.  
ЖУРАВЛЕВ Ю. И.,  
акад. РАН, д. ф.-м. н., проф.  
КУЛЕШОВ А. П.,  
акад. РАН, д. т. н., проф.  
ПОПКОВ Ю. С.,  
чл.-корр. РАН, д. т. н., проф.  
РУСАКОВ С. Г.,  
чл.-корр. РАН, д. т. н., проф.  
РЯБОВ Г. Г.,  
чл.-корр. РАН, д. т. н., проф.  
СОЙФЕР В. А.,  
чл.-корр. РАН, д. т. н., проф.  
СОКОЛОВ И. А., акад.  
РАН, д. т. н., проф.  
СУЕТИН Н. В., д. ф.-м. н., проф.  
ЧАПЛЫГИН Ю. А.,  
чл.-корр. РАН, д. т. н., проф.  
ШАХНОВ В. А.,  
чл.-корр. РАН, д. т. н., проф.  
ШОКИН Ю. И.,  
акад. РАН, д. т. н., проф.  
ЮСУПОВ Р. М.,  
чл.-корр. РАН, д. т. н., проф.

Редакционная коллегия:

АВДОШИН С. М., к. т. н., доц.  
АНТОНОВ Б. И.  
БАРСКИЙ А. Б., д. т. н., проф.  
ВАСЕНИН В.А., д. ф.-м. н., проф.  
ВИШНЕКОВ А. В., д. т. н., проф.  
ГАЛУШКИН А.И., д. т. н., проф.  
ДИМИТРИЕНКО Ю. И., д. ф.-м. н., проф.  
ДОМРАЧЕВ В.Г., д. т. н., проф.  
ЗАБОРОВСКИЙ Ю. С., д. т. н., проф.  
ЗАГИДУЛЛИН Р. Ш., к. т. н., доц.  
ЗАРУБИН В. С., д. т. н., проф.  
КАРПЕНКО А. П., д. ф.-м. н., проф.  
КОЛИН К. К., д. т. н., проф.  
КУЛАГИН В. П., д. т. н., проф.  
КУРЕЙЧИК В. М., д. т. н., проф.  
КУХАРЕНКО Б. Г., к. ф.-м. н., доц.  
ЛЬВОВИЧ Я. Е., д. т. н., проф.  
МИХАЙЛОВ Б. М., д. т. н., проф.  
НЕЧАЕВ В. В., к. т. н., проф.  
ПОЛЕЩУК О. М., д. т. н., проф.  
СОКОЛОВ Б. В., д. т. н., проф.  
ТИМОНИНА Е. Е., д. т. н., проф.  
УСКОВ В. Л., к. т. н. (США)  
ФОМИЧЕВ В. А., д. т. н., проф.  
ЧЕРМОШЕНЦЕВ С. Ф., д. т. н., проф.  
ШИЛОВ В. В., к. т. н., доц.

Редакция:

БЕЗМЕНОВА М. Ю.  
ГРИГОРИН-РЯБОВА Е. В.  
ЛЫСЕНКО А. В.  
ЧУГУНОВА А. В.

Информация о журнале доступна по сети Internet по адресу <http://novtex.ru/IT>.

Журнал включен в систему Российского индекса научного цитирования.

Журнал входит в Перечень научных журналов, в которых по рекомендации ВАК РФ должны быть опубликованы научные результаты диссертаций на соискание ученой степени доктора и кандидата наук.

### CONTENTS

#### MODELING AND OPTIMIZATION

- Kukhareno B. G., Solntseva-Chalei M. O.** Centroid Modeling for Multi Dimensional Trajectory Pencils . . . . . 83
- Matveev M. G., Mihailov V. V., Sirota E. A.** Combined Forecasting Model of Non-Stationary Multivariate Time Series for the Construction of the Spatial Profile of Atmospheric Temperature . . . . . 89
- Struchenkov V. I.** Dynamic Programming for Planning of the Partially Renewable Resources Implementation . . . . . 94
- Novikova N. M., Boriskin A. V.** Mathematical Model of the Powerful Object Search based on a Known Information . . . . . 99

#### INTELLIGENT SYSTEMS AND TECHNOLOGIES

- Ahmedov M. Z.** Handling Forecasting Problem with Applying Fuzzy Implication . . 103
- Vu Viet Thang.** Speedup Algorithm Clustering DBSCAN by Using Algorithm *K*-Means . . . . . 109

#### STRUCTURAL SYNTHESIS AND DECISION MAKING

- Zakharov A. V., Ramazanova R. R., Kurilova I. S.** Management of System Consisting of a Set of Typical Subsystems Using Scorecards . . . . . 116

#### COMPUTING SYSTEMS AND NETWORKS

- Tarasov V. N., Bahareva N. F., Lipilina L. V.** Mathematical Model of Teletraffic on the Based G/M/1 System and Results of Computational Experiment . . . . . 121

#### CAD-SYSTEMS

- Amirkhanov A. V., Gladkikh A. A., Makarchuk V. V., Shakhnov V. A.** Computer Aided Design System for Photomask Workspace Layout Generation of Very Large Scale Integrated Circuit . . . . . 127
- Gridin V. N., Anisimov V. I., Dmitrevich G. D., Laristov A. I., Al-Shameri Ya. M. A.** Organization of Information Support of Web-based Circuit Design CAD . . . . . 134
- Surkov A. V.** STA Algorithms for Developing Self-Timed Schemes . . . . . 139

#### INFORMATION TECHNOLOGIES IN EDUCATION

- Bershadsky A.M., Bozhday A. S., Mkrtychian V. S.** Construction Principles of Per- vading Self-Adapting Distance Learning System Based on Variability Model and Service-Oriented Architecture . . . . . 146

#### INFORMATION TECHNOLOGIES IN BIOMEDICAL SYSTEMS

- Kukhtichev A. A., Klenov E. A., Skorodumov S. V.** Development of Architecture of Digital Health Informational System 'CifroMed' in Aviation and Aerospace . . . 154

#### Editor-in-Chief:

Stempkovsky A. L., Member of RAS,  
Dr. Sci. (Tech.), Prof.

#### Deputy Editor-in-Chief:

Ivannikov A. D., Dr. Sci. (Tech.), Prof.  
Filimonov N. B., Dr. Sci. (Tech.), Prof.

#### Chairman:

Bychkov I. V., Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Zhuravljov Yu.I., Member of RAS,  
Dr. Sci. (Phys.-Math.), Prof.  
Kuleshov A. P., Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Popkov Yu.S., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Rusakov S. G., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Ryabov G. G., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Soifer V. A., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Sokolov I. A., Member of RAS,  
Dr. Sci. (Phys.-Math.), Prof.  
Suetin N. V.,  
Dr. Sci. (Phys.-Math.), Prof.  
Chaplygin Yu.A., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Shakhnov V. A., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Shokin Yu.I., Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Yusupov R. M., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.

#### Editorial Board Members:

Avdoshin S. M., Cand. Sci. (Tech.), Ass. Prof.  
Antonov B. I.  
Barsky A. B., Dr. Sci. (Tech.), Prof.  
Vasenin V. A., Dr. Sci. (Phys.-Math.), Prof.  
Vishnekov A. V., Dr. Sci. (Tech.), Prof.  
Galushkin A. I., Dr. Sci. (Tech.), Prof.  
Dimitrienko Yu. I., Dr. Sci. (Phys.-Math.), Prof.  
Domrachev V. G., Dr. Sci. (Tech.), Prof.  
Zaborovsky V. S., Dr. Sci. (Tech.), Prof.  
Zagidullin R. Sh., Cand. Sci. (Tech.), Ass. Prof.  
Zarubin V. S., Dr. Sci. (Tech.), Prof.  
Karpenko A. P., Dr. Sci. (Phys.-Math.), Prof.  
Kolin K. K., Dr. Sci. (Tech.)  
Kulagin V. P., Dr. Sci. (Tech.), Prof.  
Kureichik V. M., Dr. Sci. (Tech.), Prof.  
Kukhareno B. G., Cand. Sci. (Phys.-Math.)  
Ljvovich Ya.E., Dr. Sci. (Tech.), Prof.  
Mikhailov B. M., Dr. Sci. (Tech.), Prof.  
Nechaev V. V., Cand. Sci. (Tech.), Ass. Prof.  
Poleschuk O. M., Dr. Sci. (Tech.), Prof.  
Sokolov B. V., Dr. Sci. (Tech.)  
Timonina E. E., Dr. Sci. (Tech.), Prof.  
Uskov V. L. (USA), Dr. Sci. (Tech.)  
Fomichev V. A., Dr. Sci. (Tech.), Prof.  
Chermoshentsev S. F., Dr. Sci. (Tech.), Prof.  
Shilov V. V., Cand. Sci. (Tech.), Ass. Prof.

#### Editors:

Bezmenova M. Yu.  
Grigorin-Ryabova E. V.  
Lysenko A. V.  
Chugunova A. V.

Complete Internet version of the journal at site: <http://novtex.ru/IT>.

According to the decision of the Higher Certifying Commission of the Ministry of Education of Russian Federation, the journal is inscribed in "The List of the Leading Scientific Journals and Editions wherein Main Scientific Results of Theses for Doctor's or Candidate's Degrees Should Be Published"

# МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 51-74; 519.246

**Б. Г. Кухаренко**, канд. физ.-мат. наук, ст. науч. сотр., вед. науч. сотр.,  
Институт машиноведения РАН, г. Москва, e-mail: kukharenkobg@gmail.com,

**М. О. Солнцева-Чалей**, аспирант,  
Московский физико-технический институт (ГУ), e-mail: solnceva.chalei@gmail.com

## Моделирование центроидов для пучков многомерных траекторий

*Оптимизация целевой функции алгоритма  $k$ -средних с условием асимптотического схождения пучка позволяет определить центроиды для пучков пространственных траекторий. Для полного определения сходящихся пучков используются оценки их геометрических асимптот, как наиболее правдоподобные ортогональные линейные регрессии рассеянных данных траекторий. Пучки траекторий выделяются на основе меры косинуса до соответствующих асимптот, с последующим определением их центроидов. В качестве примера определены центроиды пучков траекторий посадки самолетов на полосы аэродрома.*

**Ключевые слова:** моделирование, оптимизация, пучки траекторий, центроиды, геометрические асимптоты, линейная регрессия, рассеянные данные

### Введение

Рассматриваются подходы к моделированию центроидов (оптимальных траекторий) для сходящихся пучков многомерных траекторий. Примером таких пучков являются наборы трехмерных пространственных траекторий посадки самолетов при заходе на заданную полосу аэродрома. Формально, для векторов  $\{\mathbf{x}[i] \in \mathbb{R}^{3 \times L}, i \in \mathbb{N}_k\}$  ( $L \gg 1$  — максимальная длина траектории), представляющих пучок траекторий  $\mathbb{N}_k, k = \overline{1, K_0}$  ( $K_0$  — эмпирический параметр), выполняется условие (асимптотического) схождения пучка

$$\forall (i, j) \in \mathbb{N}_k, \|\mathbf{x}[L_j; i] - \mathbf{x}[L_j; j]\|_2 < \varepsilon, \quad (1)$$

где ( $\forall i \in \mathbb{N}_k, \mathbf{x}[L_j; i]$  — координаты точек траекторий, которые почти совпадают, т. е. параметры  $L_j, i \in \mathbb{N}_k$ , подлежат определению;  $\|\cdot\|_2$  — Евклидова мера расстояния в трехмерном пространстве  $\mathbb{R}^3$ ;  $\varepsilon$  — порог (порядка ширины полосы в случае посадки самолетов). Пучки траекторий похожи на пучки функций и пучки решений дифференциальных уравнений (см. [1–3]). Траектории в анализируемых сходящихся пучках имеют типичный профиль и геометрическую асимптоту — линию в  $\mathbb{R}^3$ , удовлетворяющую условию (1) и касательную в окрестности конечных точек  $\forall i \in \mathbb{N}_k, \mathbf{x}[L_j; i]$  всех траекторий пучка (в рассматриваемом случае с порогом  $\varepsilon$  (1)) [4]. Тогда центроиды пучков  $\{\boldsymbol{\mu}[k] \in \mathbb{R}^{3 \times d}, k = \overline{1, K_0}\}$  также удовлетворяют условию типа (1) в виде

$$\forall k = \overline{1, K}, \forall i \in \mathbb{N}_k, \|\boldsymbol{\mu}[L_k; k] - \mathbf{x}[L_j; i]\|_2 < \varepsilon, \quad (2)$$

где параметры  $L_j, i \in \mathbb{N}_k$  и  $L_k, k = \overline{1, K}$  подлежат определению. Если считать, что пучки  $\mathbb{N}_k, k = \overline{1, K}$  определяются кластерами  $\mathbb{C}_k, k = \overline{1, K}$ , полученными в результате геометрического разбиения пространства векторов  $\{\mathbf{x}[n] \in \mathbb{R}^{3 \times d}, n = \overline{1, N}\}$ ,  $d \gg 1$  на ячейки в стиле диаграмм Воронова (*Voronoi diagram* [5]), то центроиды пучков определяются алгоритмом разбиения на заданное число кластеров (*K-means algorithm*) [6]. Ограничение этого подхода связано с тем, что: 1) выделяемые области пространства состояния  $\mathbb{R}^{3 \times d}$  имеют различное число граней; 2) Евклидова мера расстояния между траекториями (в целом) в пространстве состояний  $\mathbb{R}^{3 \times d}$  плохо отображает подобие (и различие) профилей траекторий в трехмерном пространстве  $\mathbb{R}^3$ ; 3) на практике в  $\mathbb{R}^3$  пучки траекторий пересекаются.

### 1. Методы моделирования центроидов для пучков многомерных траекторий

#### 1.1. Определение центроидов и пучков траекторий посредством алгоритма $K$ -средних

Для многомерных векторов  $\{\mathbf{x}[i] \in \mathbb{R}^{3 \times L}, i = \overline{1, N}\}$ ,  $L \gg 1$  с Евклидовой мерой расстояния алгоритм  $K$ -средних решает задачу идентификации центроидов  $\{\boldsymbol{\mu}[k] \in \mathbb{R}^{3 \times d}, k = \overline{1, K}\}$ , ассоциируемых с кластерами  $\mathbb{C}_k, k = \overline{1, K}$ . Для набора центроидов сумма квадратов Евклидовых расстояний до векторов в соответствующих кластерах  $\mathbb{C}_k, k = \overline{1, K}$ , является

минимальной. Введем назначения векторов кластерам посредством набора бинарных индикаторных переменных  $\{r[i; k] \in \{0, 1\}, k = \overline{1, K}, i = \overline{1, N}$  (то есть, если вектор  $\mathbf{x}[i]$  назначен кластеру  $k$ , то  $r[i; k] = 1$ , а в противном случае —  $r[i; k] = 0$  (т. е.  $C_k = \{\mathbf{x}[i] | r[i; k] = 1\}, k = \overline{1, K}$ ). Целевая функция алгоритма  $K$ -средних имеет вид

$$J = \sum_{i=1}^N \sum_{k=1}^K r[i; k] ((\mathbf{x}[i] - \boldsymbol{\mu}[k]) \cdot (\mathbf{x}[i] - \boldsymbol{\mu}[k])), \quad (3)$$

где " $\cdot$ " обозначает скалярное произведение векторов в пространстве состояний  $\mathbb{R}^{3 \times L}$ . Для определения кластеров, представляющих пучки траекторий, минимизацию целевой функции (3) необходимо проводить с учетом условия типа (2) в виде

$$\begin{aligned} \forall k = \overline{1, K}, \exists i \in \{j | r[j; k] = 1\}, \|\boldsymbol{\mu}[L; k] - \mathbf{x}[L; i]\|_2 < \varepsilon, \\ \forall k, l = \overline{1, K}, \|\boldsymbol{\mu}[L; k] - \boldsymbol{\mu}[L; l]\|_2 \gg \varepsilon, \end{aligned} \quad (4)$$

где  $\|\cdot\|_2$  — Евклидова мера расстояния в трехмерном пространстве  $\mathbb{R}^3$ . В алгоритме  $K$ -средних для инициализации  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  используются  $\{\mathbf{x}[i], i = \overline{1, N}\}$ . Минимизация  $J$  (3) осуществляется последовательными итерациями, состоящими из двух шагов: оценки  $\{r[n; k], n = \overline{1, N}, k = \overline{1, K}\}$  при фиксированных  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  в замкнутой форме

$$\begin{aligned} \langle r[n; k] \rangle = \\ = \begin{cases} 1, & \text{если } k = \arg \min_j ((\mathbf{x}[n] - \boldsymbol{\mu}[j]) \cdot (\mathbf{x}[n] - \boldsymbol{\mu}[j])) \\ 0, & \text{в противном случае} \end{cases} \end{aligned} \quad (5)$$

и оценки  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  при фиксированных  $\{r[n; k], n = \overline{1, N}, k = \overline{1, K}\}$

$$\begin{aligned} \langle \boldsymbol{\mu}[k] \rangle, k = \overline{1, K} = \\ = \arg \min_{\{\boldsymbol{\mu}[k], k = \overline{1, K}\}} \sum_{n=1}^N \sum_{k=1}^K r[n; k] ((\mathbf{x}[n] - \boldsymbol{\mu}[j]) \cdot (\mathbf{x}[n] - \boldsymbol{\mu}[j])) \end{aligned}$$

в замкнутой форме

$$\langle \boldsymbol{\mu}[k] \rangle = \frac{\sum_{n=1}^N r[n; k] \mathbf{x}[n]}{\sum_{n=1}^N r[n; k]} \quad (6)$$

до достижения сходимости. Поскольку каждый шаг уменьшает целевую функцию  $J$  (3), сходимость алгоритма  $K$ -средних гарантируется. Однако вследствие неудачной инициализации  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  он может сходиться к локальному, а не глобальному минимуму  $J$  при условии (4), гарантирующем, что

кластеры  $C_k, k = \overline{1, K}$ , представляют пучки траекторий  $\mathbb{N}_k, k = \overline{1, K}$ .

Непосредственная реализация алгоритма  $K$ -средних относительно медленная, поскольку на каждом шаге определения  $r[i; k], i = \overline{1, N}, k = \overline{1, K}$  (5) вычисляется Евклидово расстояние между каждым вектором  $\boldsymbol{\mu}[k], k = \overline{1, K}$  и каждым вектором  $\mathbf{x}[i], i = \overline{1, N}$ . Ускорение алгоритма  $K$ -средних обеспечивается предварительным построением дерева, в котором ближайшие вектора  $\{\mathbf{x}[i], i = \overline{1, N}\}$  находятся в одном поддереве [7, 8]. Используя неравенство треугольников для расстояний, также сокращают число вычислений расстояний [9, 10]. Использование в исходном пространстве состояний  $\mathbb{R}^{3 \times L}$  Евклидовой меры расстояния и представление о геометрическом разбиении пространства векторов  $\{\mathbf{x}[i] \in \mathbb{R}^{3 \times L}, i = \overline{1, N}\}, L \gg 1$ , на ячейки в стиле диаграмм Воронова не отражает характер пучков траекторий (посадки самолетов). Кроме того, определение центроидов оказывается неустойчивым к случайным отклонениям (*outliers*).

Первый способ адаптации условной задачи оптимизации (3)—(4) к анализу пучков пространственных траекторий состоит в использовании для центроидов  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  модели полиномиальной регрессии, которая решает задачу выравнивания  $\forall i \in \mathbb{N}_k, \mathbf{x}[L; i]$  — координаты точек траекторий пучка, которые почти совпадают [11, 12].

Второй способ состоит в использовании представлений векторов  $\{\mathbf{x}[i] \in \mathbb{R}^{3 \times L}, i = \overline{1, N}\}, L \gg 1$  в пространствах абстрактных характеристик  $\{\mathbf{y}[i] \in \mathbb{R}^M, i = \overline{1, N}\}$  с Евклидовой мерой расстояния [13—15]. Вектора, представляющие пучки траекторий, которые геометрически неразделимы в исходном пространстве состояний  $\mathbb{R}^{3 \times L}$ , в пространствах абстрактных характеристик становятся разделимыми, поэтому в этих пространствах используется Евклидова мера расстояния [16, 17]. При отображении в исходное пространство состояний эта метрика становится неевклидовой. Существуют примеры неевклидовой меры расстояния для векторов траекторий в пространстве состояний. Например, мера косинуса

$$\rho_{\cos}(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}') / (\sqrt{\mathbf{x} \cdot \mathbf{x}} \sqrt{\mathbf{x}' \cdot \mathbf{x}'}) \quad (7)$$

наиболее адекватно отражает близость векторов  $\{\mathbf{x}[i] \in \mathbb{R}^{3 \times L}, i = \overline{1, N}\}, L \gg 1$  в пространстве состояний, представляющих пучки траекторий определенного профиля [17]. Поэтому можно модифицировать задачу оптимизации (3)—(4), заменив в выражении (3) скалярное произведение на общую меру расстояния.

### 1.2. Использование неевклидовой меры расстояния

Целевая функция (3) алгоритма  $K$ -средних обобщается введением общей меры расстояния  $\rho(\mathbf{x}, \mathbf{x}')$

между двумя векторами  $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^{3 \times L}$  и минимизацией целевой функции

$$J = \sum_{i=1}^N \sum_{k=1}^K r[i; k] \rho^2(\mathbf{x}[i], \boldsymbol{\mu}[k]) \quad (8)$$

(*K-medoids algorithm*) [6]. При заданных центроидах  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  шаг оценки  $\{r[i; k], i = \overline{1, N}, k = \overline{1, K}\}$  (как в стандартном алгоритме *K-средних*) включает назначение каждого вектора  $\mathbf{x}[i], i = \overline{1, N}$  кластеру  $\mathbb{C}_k, k = \overline{1, K}$ , для которого расстояние  $\rho(\mathbf{x}[i], \boldsymbol{\mu}[k])$  с соответствующим центроидом минимально:

$$\langle r[i; k] \rangle = \begin{cases} 1, & \text{если } k = \arg \min_j \rho(\mathbf{x}[i], \boldsymbol{\mu}[j]) \\ 0, & \text{в противном случае} \end{cases} \quad (9)$$

с оценкой вычислительной сложности  $O(K \cdot N)$ . Однако шаг оценки  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  является потенциально более сложным. При стандартном ограничении, что каждый центроид является одним из векторов, назначенных соответствующему кластеру, условие (4) выполняется автоматически. Это позволяет реализовать алгоритм для любого выбора меры расстояния  $\rho(\mathbf{x}[i], \boldsymbol{\mu}[k])$ , которая непосредственно вычисляется. Шаг определения  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  включает дискретный поиск по всем  $N_k$  векторам, назначенным этому кластеру  $\mathbb{C}_k, k = \overline{1, K}$ , и требует  $O(N_k^2)$  оценок меры расстояния  $\rho(\mathbf{x}[i], \boldsymbol{\mu}[k])$ .

Отметим, что взаимосвязь центроидов и кластеров

$$\boldsymbol{\mu}[k] \Leftrightarrow \mathbb{C}_k, k = \overline{1, K},$$

навязана логикой алгоритма *K-средних* и его обобщениями и, в принципе, центроиды  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  и кластеры  $\mathbb{C}_k, k = \overline{1, K}$ , представляющие пучки траекторий  $\mathbb{N}_k, k = \overline{1, K}$ , можно определять независимо.

### 1.3. Оценки центроидов и определение пучков траекторий

В качестве независимых методов оценки центроидов  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  используются скрытые компоненты линейных и нелинейных динамических моделей (см. [18, 19]), и скрытые последовательности марковских моделей (см. [20, 21]) при условии (4). После этого пучки траекторий определяются по схеме  $\boldsymbol{\mu}[k] \Rightarrow \mathbb{C}_k, k = \overline{1, K}$ , на основе однократного применения формулы (9) с мерой косинуса (7).

### 1.4. Оценка пучков траекторий и определение центроидов

Для оценки пучков траекторий можно использовать геометрические методы триангуляции [5, 22]. После этого центроиды  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  определя-

ются по схеме  $\mathbb{C}_k \Rightarrow \boldsymbol{\mu}[k], k = \overline{1, K}$  на основе однократного применения формулы

$$\begin{aligned} & \{\langle \boldsymbol{\mu}[k] \rangle, k = \overline{1, K}\} = \\ & = \arg \min_{\{\boldsymbol{\mu}[k], k = \overline{1, K}\}} \sum_{i=1}^N \sum_{k=1}^K r[i; k] \rho_{\cosin}^2(\mathbf{x}[i], \boldsymbol{\mu}[k]) \quad (10) \end{aligned}$$

при условии (4) с использованием квадрата меры косинуса (7). Оценка (10) эффективна при представлении векторов  $\{\mathbf{x}[i], i = \overline{1, N}\}$  и  $\{\boldsymbol{\mu}[k], k = \overline{1, K}\}$  в исходном пространстве состояний.

## 2. Линейная регрессия рассеянных данных алгоритмом RANSAC для оценки асимптот пучков траекторий

Как отмечается во введении, особенностью рассматриваемых пучков траекторий является то, что все они имеют характерную геометрическую асимптоту в области сходимости траекторий (1) [4]. Поскольку дискретные точки траекторий пучка плотно лежат в окрестности асимптоты, основа предлагаемого метода для оценки асимптот пучков траекторий состоит в том, что набор векторов кластера многомерных траекторий  $\{\mathbf{x}[i] \in \mathbb{R}^{3 \times L}, i = \overline{1, N}\}$  рассеивается во множество точек этих траекторий

$$\begin{aligned} & \{\mathbf{x}[i] \in \mathbb{R}^{3 \times L}, i = \overline{1, N}\} \Rightarrow \\ & \Rightarrow \{(x[j], i), y[j], i), z[j], i) \in \mathbb{R}^3, j = \overline{1, L}, i = \overline{1, N}\}. \quad (11) \end{aligned}$$

Множество точек (11) должно быть упорядочено по значениям одной из координат (в направлении возрастания — *ascend* или убывания — *descend*). При этом происходит упорядочение по остальным координатам всех точек, представляющих сходящийся пучок траекторий движения по определенному профилю. После этого для рассеянных трехмерных данных  $\{\mathbf{z}_i = (x_i, y_i, z_i), i = \overline{1, L \cdot N}\}$  (11) с помощью алгоритма RANSAC (Random Sample and Consensus — случайная выборка и консенсус) анализируются модели ортогональной линейной регрессии

$$\begin{aligned} \mathfrak{M}(\boldsymbol{\theta}) = \{(x, y, z) \in \mathbb{R}^3 | & (a_1 x + b_1 y + c_1 z = d_1) \wedge \\ & \wedge (a_2 x + b_2 y + c_2 z = d_2)\}, \quad (12) \end{aligned}$$

где  $\wedge$  — конъюнкция;  $\boldsymbol{\theta} = \{a_1, b_1, c_1, d_1, a_2, b_2, c_2, d_2\}$  — вектор параметров этих моделей при заданном пороге Евклидова расстояния  $\rho_{\perp}(\mathbf{z}, \mathfrak{M}(\boldsymbol{\theta}))$ , вычисляемого по ортогональной проекции точки  $\mathbf{z} = (x, y, z)$  из множества (11) на линию  $\mathfrak{M}(\boldsymbol{\theta})$ . Таким образом, модель (12) симметрична относительно координат  $x, y, z$ . Для выдвижения гипотезы относительно модели ортогональной линейной регрессии (12) достаточно любой пары точек из множества (11). Окончательная модель (12) подтверждается наибольшим относительным количеством (процентом) рассеянных данных  $\{\mathbf{z}_i = (x_i, y_i, z_i), i = \overline{1, L \cdot N}\}$  множества (11).

В настоящей работе используется алгоритм MLESAC (Maximum Likelihood Estimation Sample Consensus — консенсус выборок с оценкой по максимуму правдоподобия) — вероятностная версия алгоритма RANSAC [23, 24]. Этот алгоритм оценивает правдоподобие модели (12), представляя распределение расстояния рассеянных данных  $\{z_i = (x_i, y_i, z_i), i = \overline{1, L \cdot N}\}$  от модели  $\mathfrak{M}(\theta)$  (12), как смесь распределения данных, подтверждающих модель (12) (*inliers*), и распределения данных, отклоняющих эту модель (*outliers*). Считая, что рассеянные данные  $Z$  множества (11) независимые, получаем выражение для логарифма правдоподобия в виде

$$L(\rho_{\perp}(Z, \mathfrak{M}(\theta))|\theta) = \sum_{i=1}^{L \cdot N} \log(\gamma p(\rho_{\perp}(z_i, \mathfrak{M}(\theta))|z_i \text{ is } inlier) + (1 - \gamma)p(\rho_{\perp}(z_i, \mathfrak{M}(\theta))|z_i \text{ is } outlier)), \quad (13)$$

где  $\gamma$  — параметр смешивания. Распределение расстояний до данных, подтверждающих модель (12), представляется гауссовым распределением

$$p(\rho_{\perp}(z_i, \mathfrak{M}(\theta))|z_i \text{ is } inlier) \propto \exp\left(-\frac{(\rho_{\perp}(z_i, \mathfrak{M}(\theta)))^2}{2\sigma^2}\right), \quad (14)$$

где  $\sigma$  — стандартное отклонение. Распределения расстояний до данных, отклоняющих модель (12), описываются равномерным распределением

$$p(\rho_{\perp}(z_i, \mathfrak{M}(\theta))|z_i \text{ is } outlier) = \begin{cases} (2\rho_{\max})^{-1}, & \rho_{\perp}(z_i, \mathfrak{M}(\theta)) < \rho_{\max} \\ 0, & \rho_{\perp}(z_i, \mathfrak{M}(\theta)) \geq \rho_{\max} \end{cases} \quad (15)$$

где  $\rho_{\max}$  — наибольшее расстояние до данных (определяется контекстом). Минимизация логарифма правдоподобия (13) позволяет оценить вектор параметров  $\theta$  и параметр смешивания  $\gamma$ . Это обеспечивается итерациями алгоритма ожидания-максимизации правдоподобия [25].

Наиболее правдоподобная линейная регрессия рассеянных данных кластера траекторий определяет геометрическую асимптоту  $\mathfrak{M}(\theta)[k], k = \overline{1, K}$ , модели (12) одного из пучков кластера при условии (1). А эта асимптота играет ту же роль, что и центр тяжести  $\mu[k], k = \overline{1, K}$ , при условии (2), для оценки касательного ей пучка многомерных траекторий на основе однократного применения формулы (9) с мерой косинуса (7) (подраздел 1.3). После удаления из рассеянных данных (11) тех точек, которые представляют траектории выделенного пучка, повторяется процедура определения геометрической асимптоты и выделяется следующий пучок траекторий. Поскольку определение модели (12) должно быть симметричным относительно координат  $x, y, z$ , при формировании рассеянных данных оставшихся траекторий в (11) проводится сортировка по очередной пространственной координате по срав-

нению с использованной в (11) при определении предыдущей асимптоты модели (12). Возможная зависимость результата (12) от направления координат устраняется изменением направления сортировки в (11) с возрастания на убывание или наоборот. Анализ траекторий кластера завершается определением всех пучков в кластере.

### 3. Численный эксперимент

Используются траектории 117 самолетов, идущих на посадку в международном аэропорту и зарегистрированных радаром TRACON 1 января 2006 г. (данные в открытом доступе на сайте <https://c3.nasa.gov/dashlink/resources/132/>). Начало координат совпадает с положением радара, интервал времени между точками регистрации составляет около 5 с. В работе учитываются только 160 последних точек каждой траектории, что исключает случайные маневры самолетов перед заходом на посадку. Эти траектории в трехмерном пространстве представлены в работе [12]. Пять кластеров траекторий самолетов (на рис. 1 в проекциях на оси  $x, y$  и  $z$ , см. вторую сторону обложки) выделяются в результате применения метода полиномиальных регрессий [11, 12]. Распределение траекторий по кластерам следующее: 16 траекторий — в розовом кластере; 13 — в зеленом; 3, 37 и 38 — в синем, черном и красном кластерах соответственно.

Кластеры на рис. 1 (см. вторую сторону обложки) состоят из нескольких пучков траекторий, соответствующих определенным профилям посадки.

Определим пучки траекторий, составляющих красный кластер (см. рис. 1). Для структуры `trajs` размерности `size(trajs, 1) x size(trajs, 2) x size(trajs, 3)`, представляющей 38 трехмерных траекторий красного кластера (см. рис. 1), рассеянные данные `Data` (11) являются результатом кода на MATLAB (рис. 2).

```
% рассеяние данных
Data = zeros(size(trajs,2),size(trajs,3));
for k = 1:size(trajs,1),
    Data = [Data; squeeze(trajs(k,:,:))];
end;
% сортировка рассеянных данных
[~,Isort] = sort(Data(:,1),1);
Data = Data(Isort,:,:) ;
```

Рис. 2. Код на MATLAB для получения рассеянных данных траекторий

На рис. 3, а (см. вторую сторону обложки) показаны рассеянные данные `Data(:, 1:2)` (11) двумерных проекций `trajs(:, :, 1:2)` траекторий красного кластера (см. рис. 1) и результат их линейной регрессии с использованием алгоритма MLESAC, определяющий асимптоту первого пучка (голубая линия).

Траектории первого (голубого) пучка удаляются из красного кластера (см. рис. 1) на основе близо-

сти траекторий к голубой асимптоте (рис. 3, *a*, см. вторую сторону обложки) по мере косинуса (7). На рис. 3, *б* (см. вторую сторону обложки) показаны рассеянные данные  $Data(:, 1:2)$  двумерных проекций  $trajs(:, :, 1:2)$  траекторий красного кластера (см. рис. 1) за вычетом траекторий первого (голубого) пучка и результат их линейной регрессии (зеленая линия) с использованием алгоритма MLESAC, определяющий асимптоту второго (зеленого) пучка. Траектории второго (зеленого) пучка удаляются из красного кластера (см. рис. 1) на основе близости траекторий к зеленой асимптоте (см. рис. 3, *б*) по мере косинуса (7). В результате определяется третий (синий) пучок траекторий (см. рис. 3, *в*). Для трех выделенных пучков на рис. 3, *в*: голубого, зеленого и синего, их центроиды  $\{\mu[k], k = \overline{1, K}\}$  (10) показаны тремя толстыми красными линиями. Результат определения пучков в красном кластере тот же, что и в работе [17]. Потенциально посторонние траектории движения самолетов в зоне риска (выбросы) специально не определяли, потому что используется мера косинуса, а это значит, что выбросы уже определены в работе [17].

Пучки траекторий розового кластера (см. рис. 1) существенно пересекаются. Другой особенностью этих траекторий является присутствие практически линейных участков в их хвостах вдали от фокуса. Это препятствует непосредственному определению асимптот сходящихся пучков по методу раздела 2. Поскольку все траектории в кластере имеют одинаковое направление времени, при анализе траекторий розового кластера используются рассеянные данные сокращенных траекторий с частью точек  $quotum \approx 0,4$ , считая от фокусов пучков. На рис. 3, *a* (см. вторую сторону обложки) показаны рассеянные данные  $Data(:, 1:2)$  (11) двумерных проекций  $trajs(:, :, 1:2)$  полных траекторий розового кластера (см. рис. 1) и результат их линейной регрессии с использованием алгоритма MLESAC, определяющий асимптоту первого пучка (голубая линия), поскольку в этом случае результаты ортогональной линейной регрессии рассеянных данных сокращенных и полных траекторий совпадают.

Траектории первого (голубого) пучка удаляются из розового кластера (см. рис. 1) на основе близости траекторий к голубой асимптоте (рис. 4, *a*, см. вторую сторону обложки), по мере косинуса (7). На рис. 4, *б* (см. вторую сторону обложки) для наглядности показаны рассеянные данные  $Data(:, 1:2)$  двумерных проекций  $trajs(:, :, 1:2)$  полных траекторий розового кластера (см. рис. 1) за вычетом траекторий первого (голубого) пучка, но результат линейной регрессии (зеленая линия), определяющий асимптоту второго (зеленого) пучка, получен с использованием алгоритма MLESAC для рассеянных данных сокращенных траекторий. Траектории второго (зеленого) пучка удаляются из розового кластера (см. рис. 1) на основе близости траекторий к зеленой асимптоте (рис. 4, *б*) по мере косинуса (7). В результате определяется третий (синий) пучок траекторий (рис. 4, *в*, см. вторую сторону обложки). Для трех существенно пересекающихся пучков закрученных траекторий на рис. 4, *в*: голубого, зеленого и синего, их центроиды  $\{\mu[k], k = \overline{1, K}\}$  (10) показаны тремя толстыми красными линиями. Сравнение центроидов (рис. 4, *в*, см. вторую сторону обложки) с полиномиальной регрессией траекторий розового кластера (см. рис. 1) показывает, что результат алгоритма MLESAC на рис. 4, *в* является нетривиальным.

## Заключение

Рассмотрены подходы к моделированию центроидов для сходящихся пучков многомерных траекторий. Жесткая взаимозависимость при определении центроидов и соответствующих им пучков траекторий является следствием идиомы алгоритма  $K$ -средних. Поэтому существуют двухшаговые методы: сначала независимое определение сходящихся пучков и затем моделирование центроидов для уже определенных пучков траекторий. Например, сходящиеся пучки траекторий могут быть идентифицированы посредством определения касательных им геометрических асимптот в точках их фокусов. Показано, как для последовательного определения асимптот, касательных к сходящимся пучкам траекторий, используются наиболее правдоподобные ортогональные линейные регрессии рассеянных данных траекторий, полученные с использованием алгоритма MLESAC. После определения геометрических асимптот соответствующие им пучки траекторий отделяются на основе меры косинуса от траекторий до касательных им асимптот, и для этих пучков вычисляются центроиды. В качестве примера определены центроиды пучков траекторий посадки самолетов при заходе на полосы аэродрома.

## Список литературы

1. **Johnson W. W.** On singular solutions of differential equations of the first order // *Annals of Mathematics*. April 1887. Vol. 3, N. 2. P. 33–38.
2. **Griffiths P., Harris J.** *Principles of Algebraic Geometry*, New York: John Wiley & Sons, 1994.
3. **Eisenbud D., Harris J.** 3264 and All That Intersection Theory: A Second Course in Algebraic Geometry. Cambridge, UK: Cambridge University Press, 2015.
4. **Погорелов А. И.** *Дифференциальная геометрия*. 6-е изд. М.: Наука, 1974.
5. **de Berg M., Cheong O., van Kreveld M., Overmars M.** *Computational Geometry. Algorithms and Applications*. Third Edition. Berlin, Heidelberg: Springer-Verlag, 2008.
6. **MacQueen J.** Some methods for classification and analysis of multivariate observations / LeCam L. M., Neyman J., eds. // *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. University of California Press. 1967. Vol. 1. P. 281–297.
7. **Ramasubramanian V., Paliwal K. K.** A generalized optimization of the  $k$ - $d$  tree for fast nearest-neighbour search // *Proceedings Fourth IEEE Region 10 International Conference (TENCON'89)*. 1990. P. 565–568.

8. **Moore A. W.** The anchors hierarch: using the triangle inequality to survive high dimensional data // Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence. 2000. P. 397–405.
9. **Hodgson M. E.** Reducing computational requirements of the minimum-distance classifier // Remote Sensing of Environments. 1998. Vol. 25. P. 117–128.
10. **Elkan C.** Using the triangle inequality to accelerate  $k$ -means / Proceedings of the Twelfth International Conference on Machine Learning. AAAI. 2003. P. 147–153.
11. **Gaffney S., Smyth P.** Joint probabilistic curve clustering and alignment / Saul L., Weiss Y., Bottou L., eds. Proceedings of Neural Information Processing Systems (NIPS 2004). December 13–18, 2004, Vancouver, British Columbia, Canada. Advances in Neural Information Processing Systems. Vol. 17. Cambridge, MA: MIT Press. 2005. P. 473–480.
12. **Кухаренко Б. Г., Солнцева М. О.** Кластеризация управляемых объектов на основе сходства их многомерных траекторий // Информационные технологии. 2014. № 5. С. 3–7.
13. **Anjum N., Cavallaro A.** Trajectory clustering for scene context learning and outlier detection / Schonfeld D., Shan C., Tao D., Wang L., eds. Video Search and Mining. Berlin. Studies in Computational Intelligence. Vol. 287. Heidelberg: Springer, 2010. P. 33–51.
14. **Кухаренко Б. Г., Солнцева М. О.** Покомпонентный дискриминантный анализ результатов кластеризации многомерных траекторий // Информационные технологии. 2014. № 11. С. 3–7.
15. **Кухаренко Б. Г., Солнцева М. О.** Итеративная кластеризация траекторий управляемых объектов в многомерном пространстве характеристик // Информационные технологии. 2014. № 8. С. 11–16.
16. **Chen G., Lerman G.** Spectral curvature clustering (SCC) // International Journal on Computer Vision. 2009. Vol. 81, N. 3. P. 317–330.
17. **Кухаренко Б. Г., Солнцева-Чалей М. О.** Спектральный метод с использованием полярной кривизны для анализа результатов кластеризации многомерных траекторий // Информационные технологии. 2015. Т. 21, N. 12. С. 901–905.
18. **Кухаренко Б. Г., Солнцева М. О.** Анализ результатов кластеризации многомерных траекторий посредством моделей линейных динамических систем // Информационные технологии. 2015. Т. 21, № 2. С. 104–109.
19. **Кухаренко Б. Г., Солнцева-Чалей М. О.** Применение моделей нелинейных динамических систем для анализа результатов кластеризации многомерных траекторий // Информационные технологии. 2015. Т. 21, № 5. С. 341–345.
20. **Listgarten J., Neal R. M., Roweis S. T., Emili A.** Multiple alignment of continuous time series / Saul L. K., Weiss Y., Bottou L., eds. Proceedings of Neural Information Processing Systems (NIPS 2004). December 13–18, 2004, Vancouver British Columbia, Canada. Advances in Neural Information Processing Systems. Vol. 17. Cambridge, MA: MIT Press, 2005. P. 5–13.
21. **Кухаренко Б. Г., Солнцева-Чалей М. О.** Применение моделей непрерывного профиля для анализа результатов кластеризации многомерных траекторий // Информационные технологии. 2015. Т. 21, № 8. С. 585–590.
22. **Галанин М. П., Щеглов И. А.** Разработка и реализация алгоритмов трехмерной триангуляции сложных пространственных областей: прямые методы. Препринт. М.: Институт прикладной математики им. М.В. Келдыша РАН, 2006.
23. **Torr P. H. S., Zisserman A.** MLESAC: A new robust estimator with application to estimating image geometry // Journal of Computer Vision and Image Understanding. 2000. Vol. 78, N. 1. P. 138–156.
24. **Кухаренко Б. Г.** Алгоритмы анализа изображений для определения локальных особенностей и распознавания объектов и панорам // Информационные технологии. 2011. № 7. Приложение. С. 1–32.
25. **Dempster A., Laird N. M., Rubin D. B.** Maximum likelihood from incomplete data via the EM algorithm // Journal of the Royal Statistical Society B. 1977. Vol. 39, N. 1. P. 1–38.

**B. G. Kukhareno**, Leading Research Scientist, e-mail: kukhareno.bg@gmail.com,  
Blagonravov Mechanical Engineering Research Institute of the RAS,  
**M. O. Solntseva-Chalei**, Postgraduate Student, e-mail: solntseva.chalei@gmail.com,  
Moscow Institute of Physics and Technology (SU)

## Centroid Modeling for Multi Dimensional Trajectory Pencils

*In this paper we describe how prevalent flight routes can be found automatically out of a comprehensive set of arrival flight trajectories. As example  $k$ -means clustering of the actual arrival routes is a fundamental step directing to the determination of the airport's prevalent arrival trajectory pencils. For each arrival trajectory pencils then a typical arrival route can be computed, describing the pencil specific mean arrival route (centroid). The polynomial regression model of centroid is useful in this case. As interrelation of centroid and respective trajectory pencils is a sequence of  $k$ -means idiom, independent centroid modeling methods for multi dimensional trajectory pencils are under study in present paper. With respect to independent centroid modeling methods, linear and nonlinear dynamical models and Markov models are mentioned. Next, to estimate converging trajectory pencil geometric asymptotes, maximum likelihood orthogonal linear regression models of the trajectory scattered data are in use. The models are obtained by MLESAC algorithm, which is a probabilistic version of RANSAC where the distance of scattered data points from the linear regression model is assumed as distributed according to a mixture of a Gaussian and a uniform distribution. After determining pencil geometric asymptotes sequentially, the trajectory pencils are selected based on trajectory cosine measure to respective asymptotes, and their centroids are defined. As example, trajectory pencil centroids are determined of specific airplane final descending on airport landing-strips of different directions.*

**Keywords:** modeling, optimization, trajectory pencil, centroids, geometric asymptotes, linear regressions, scattered data

### References

- Johnson W. W.** On singular solutions of differential equations of the first order, *Annals of Mathematics*, april 1887, vol. 3, no. 2, pp. 33–38.
- Griffiths P., Harris J.** *Principles of Algebraic Geometry*, New York: John Wiley & Sons, 1994.
- Eisenbud D., Harris J.** *3264 and All That Intersection Theory: A Second Course in Algebraic Geometry*, Cambridge, UK: Cambridge University Press, 2015.
- Pogorelov A. V.** *Differential Geometry*. Translated from the first Russian ed. by Boron L. F. Groningen: P. Noordhoff, N. V., 1959.
- de Berg M., Cheong O., van Kreveld M., Overmars M.** *Computational Geometry. Algorithms and Applications*. Third ed. Berlin, Heidelberg: Springer-Verlag, 2008.

6. **MacQueen J.** Some methods for classification and analysis of multivariate observations, LeCam L. M., Neyman J., eds. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. University of California Press, vol. 1, 1967. P. 281–297.
7. **Ramasubramanian V., Paliwal K. K.** A generalized optimization of the k-d tree for fast nearest – neighbour search, *Proceedings Fourth IEEE Region 10 International Conference (TENCON'89)*, 1990. P. 565–568.
8. **Moore A. W.** The anchors hierarch: using the triangle inequality to survive high dimensional data, *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence*, 2000, pp. 397–405.
9. **Hodgson M. E.** Reducing computational requirements of the minimum-distance classifier, *Remote Sensing of Environments*, 1998, vol. 25, pp. 117–128.
10. **Elkan C.** Using the triangle inequality to accelerate k-means, *Proceedings of the Twelfth International Conference on Machine Learning*, AAAI, 2003, pp. 147–153.
11. **Gaffney S., Smyth P.** Joint probabilistic curve clustering and alignment / Saul L., Weiss Y., Bottou L., eds. *Proceedings of Neural Information Processing Systems (NIPS 2004), December 13–18, 2004, Vancouver, British Columbia, Canada*. Advances in Neural Information Processing Systems. V. 17. Cambridge, MA: MIT Press, 2005, pp. 473–480.
12. **Kukhareno B. G., Solntseva M. O.** Klasterizacia upravlyayemykh objektov na osnove shodstva ih mnogomernykh trajektoriy, *Informacionnye tehnologii*, 2014, no. 5. P. 3–7.
13. **Anjum N., Cavallaro A.** Trajectory clustering for scene context learning and outlier detection, Schonfeld D., Shan C., Tao D., Wang L., eds., *Video Search and Mining*, Berlin, Studies in Computational Intelligence. V. 287. Heidelberg: Springer, 2010, pp. 33–51.
14. **Kukhareno B. G., Solntseva M. O.** Pokomponentnyi discriminantnyi analiz rezultatov klasterizatsii mnogomernykh trajektoriy, *Informacionnye tehnologii*, 2014, no. 11, pp. 3–7.
15. **Kukhareno B. G., Solntseva M. O.** Iterativnaya klasterizatsiya trajektoriy upravlyayemykh objektov v mnogomernom prostranstve harakteristik, *Informacionnye tehnologii*, 2014, no. 8, pp. 11–16.
16. **Chen G., Lerman G.** Spectral curvature clustering (SCC), *International Journal on Computer Vision*, 2009, vol. 81, no. 3, pp. 317–330.
17. **Kukhareno B. G., Solntseva-Chalei M. O.** Spectralnyi metod s ispolzovaniem polyarnoi krivizny dlya analiza rezultatov klasterizatsii mnogomernykh trajektoriy, *Informacionnye tehnologii*, 2015, vol. 21, no. 12, pp. 901–905.
18. **Kukhareno B. G., Solntseva M. O.** Analiz rezultatov klasterizatsii mnogomernykh trajektoriy posredstvom modelei lineinykh dinamicheskikh sistem, *Informacionnye tehnologii*, 2015, no. 2, pp. 104–109.
19. **Kukhareno B. G., Solntseva-Chalei M. O.** Primenenie modelei nelineinykh dinamicheskikh sistem dlya analiza rezultatov klasterizatsii mnogomernykh trajektoriy, *Informacionnye tehnologii*, 2015, vol. 21, no. 5, pp. 341–345.
20. **Listgarten J., Neal R. M., Roweis S. T., Emili A.** Multiple alignment of continuous time series, Saul L. K., Weiss Y., Bottou L., eds. *Proceedings of Neural Information Processing Systems (NIPS 2004), December 13–18, 2004, Vancouver, British Columbia, Canada*. Advances in Neural Information Processing Systems. V. 17. Cambridge, MA: MIT Press, 2005. P. 5–13.
21. **Kukhareno B. G., Solntseva-Chalei M. O.** Primenenie modelei nepreryvnogo profilya dlya analiza rezultatov klasterizatsii mnogomernykh trajektoriy, *Informacionnye tehnologii*, 2015, vol. 21, no. 8, pp. 585–590.
22. **Galanin M. P., Scheglov I. A.** *Razrabotka i realizatsiya algoritmov trehmernoi triangulyatsii slognykh prostranstvennykh oblastei: pryamye metody*, Preprint, Moscow M. V. Keldysh Institute of Applied Mathematics of the RAS, 2006.
23. **Torr P. H. S., Zisserman A.** MLESAC: A new robust estimator with application to estimating image geometry, *Journal of Computer Vision and Image Understanding*, 2000, vol. 78, no. 1, pp. 138–156.
24. **Kukhareno B. G.** Algoritmy analiza izobrazheniy dlya opredeleniya lokalnykh osobennosti i raspoznavaniya objektov i ponoram, *Informacionnye tehnologii*, 2011, no. 7, Prilogenie, pp. 1–32.
25. **Dempster A., Laird N. M., Rubin D. B.** Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society B*, 1977, vol. 39, no. 1, pp. 1–38.

УДК 330.43

**М. Г. Матвеев**, д-р техн. наук, проф., зав. каф., e-mail: mgmatveev@yandex.ru,

Воронежский государственный университет,

**В. В. Михайлов**, д-р техн. наук, проф., нач. факультета, e-mail: VladimirMihailov36@gmail.com,  
ВУНЦ ВВС "Военно-воздушная академия им. профессора Н. Е. Жуковского и Ю. А. Гагарина",

**Е. А. Сирота**, канд. физ.-мат. наук, доц. каф. цифровых технологий, e-mail: atoris@list.ru,

Воронежский государственный университет

## **Комбинированная прогностическая модель нестационарного многомерного временного ряда для построения пространственного профиля атмосферной температуры**

*Предлагается обоснование, построение и анализ комбинированной прогностической модели нестационарного многомерного временного ряда в целях построения пространственного профиля атмосферной температуры. Результаты моделирования показывают, что использование комбинированной модели с выделением кластеров однородной статистики для построения прогноза профиля атмосферной температуры может обеспечить приемлемые характеристики прогноза, а также может служить для построения вертикальных температурных профилей и, в конечном итоге, для составления рекомендаций при выполнении полетов авиации.*

**Ключевые слова:** нестационарный многомерный временной ряд, прогностическая модель, метеорология, векторная авторегрессия, профиль атмосферной температуры, моделирование, классы однородной статистики

## Введение и постановка задачи

Большинство количественных прогнозов в метеорологии, экономике, социологии, медицине основывается на моделях временных рядов, отражающих динамику переменных соответствующей предметной системы. Временные ряды, как правило, нестационарны, при этом нестационарность нельзя представить устойчивым систематическим изменением (линейным или нелинейным трендом) математического ожидания уровней ряда в течение длительного периода. Изменение характера тренда принято называть структурными разрывами [1] или разладкой [2]. Возникновение структурного разрыва может происходить скачкообразно или относительно медленно с явным переходным процессом, и обуславливается трудно контролируемым изменением состояния системы. Если механизм возникновения тренда имеет стохастический характер, т. е. рассматриваемый стохастический процесс относится к DS-процессам [1] и для моделирования используются "безфакторные" модели типа ARMA, то проблема построения прогностической модели становится нетривиальной.

Для решения этой проблемы в последние годы сложился подход, основанный на комбинировании моделей различных типов, что дает возможность компенсировать недостатки одних моделей достоинствами других, и повысить, в конечном счете, возможности прогнозирования. Одной из первых работ в этом направлении является работа [3], где предлагается выделять однородные в определенном смысле группы (кластеры) данных временного ряда и строить модели прогнозирования для каждой группы отдельно, что существенно повышает точность прогноза. Обзор различных комбинаций моделей предлагается в работе [4], где отмечается, что часто для кластеризации используют искусственные нейронные сети (ИНС), а для построения модели внутри кластера — модели типа ARMA. Действительно, авторегрессионные (AR) модели, при простоте и удобстве идентификации, не могут отображать нелинейные и нестационарные эффекты поведения временного ряда. Модели ИНС, наоборот, прекрасно справляются с отображением различных нелинейностей. От комбинации этих моделей можно ожидать хороших результатов.

Конкретная структура комбинированной модели определяется особенностями анализируемой предметной области и всегда является результатом соответствующего исследования. Цель предлагаемой статьи — построение комбинированной прогностической модели многомерного временного ряда атмосферной температуры для построения прогноза пространственного температурного профиля в горизонтальных слоях атмосферы. Динамика изменения атмосферной температуры во многом определяется состоянием атмосферы, характеризующимся множеством плохо контролируемых факторов и способным резко изменяться, обуславливая возникновение структурных разрывов.

Базовая идея построения комбинированной модели предполагает, что адекватное описание динамики температурных изменений можно представить модификацией модели векторной авторегрессии с переменными параметрами, характеризующими изменения состояний атмосферы. Нелинейные закономерности изменения параметров векторной авторегрессии могут быть описаны ИНС, входами которых являются какие-либо индикаторы состояний атмосферы.

Необходимая статистика обеспечивается данными реанализа, находящимися в открытом доступе в сети Интернет [5].

## 1. Основные допущения и методика построения комбинированной модели

В соответствии с данными реанализа [5] представим схему измерения температуры в виде сетки с нумерацией узлов  $i = 1, \dots, n; j = 1, \dots, m$ , показанной на рис. 1.

Временные ряды температур в узлах сетки коррелированы между собой. Корреляционный анализ показывает, что для большинства узлов статистически значимые оценки коэффициентов корреляции имеют временные ряды только в смежных узлах [6, 7]. Такие ряды целесообразно рассматривать как многомерный временной ряд, представленный в виде вектора температур смежных узлов сетки в момент времени  $t$ , например,  $y(t) = (y_{i-1,j}(t); y_{i+1,j}(t); y_{i,j}(t); y_{i,j-1}(t); y_{i,j+1}(t))$ . Тогда временной ряд температур в узле  $ij$  сетки взаимодействует со смежными рядами в узлах:  $i-1, j; i+1, j; i, j-1; i, j+1$ . В таком случае прогнозное значение температуры в  $(t+1)$ -й момент времени —  $y_{ij}^m(t+1)$  можно представить как взвешенное среднее компонент вектора  $y(t)$  [8], т. е. для произвольного, но не граничного узла  $ij, i = 2, \dots, n-1; j = 2, \dots, m-1$ , можно записать следующее выражение:

$$y_{ij}(t+1) = y_{ij}^m(t+1) + \xi_{ij}(t+1) = a_{ij}y_{ij}(t) + a_{i-1,j}y_{i-1,j}(t) + a_{i+1,j}y_{i+1,j}(t) + a_{i,j-1}y_{i,j-1}(t) + a_{i,j+1}y_{i,j+1}(t) + \xi_{ij}(t+1), \quad (1)$$

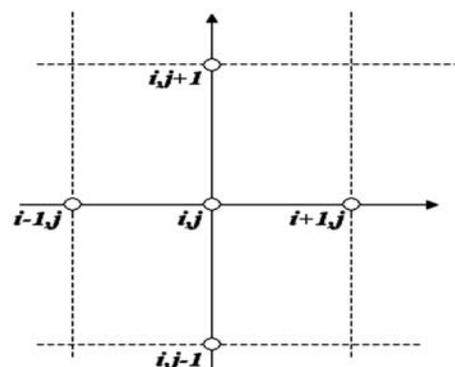


Рис. 1. Фрагмент схемы измерения метеорологических показателей (шаг сетки по параллелям и меридианам составляет  $2,5^\circ$ )

где  $a_{ij}$  — весовые коэффициенты;  $\xi_{ij}(t+1)$  — серийно некоррелированные ошибки; предполагается, что они имеют нулевое среднее и ковариацию  $\sigma_{\xi\xi}$ .

Уравнения вида (1) можно записать для  $d = n \times m - 2(n + m - 2)$  узлов сетки. Нетрудно заметить, что эти уравнения не образуют векторной авторегрессии в классическом понимании, они описываются следующим матричным уравнением:  $y_{d \times 1}(t+1) = a_{d \times 1} + A_{d \times s} y_{s \times 1}(t)$ , где  $s = n \times m$ .

Наличие нестационарности и структурных разрывов будем учитывать, рассматривая в модели (1) коэффициенты  $a_{ij}$ ,  $\forall i, j$  как переменные, изменяющиеся со временем  $t$  по некоторой изменяющейся закономерности. При этом нестационарность должна отражаться как закономерность изменения коэффициентов, а структурные разрывы — как изменения этой закономерности. Для того чтобы оценить эти закономерности необходимо на множестве статистических температурных данных научиться строить сегменты квазиоднородных температур, соответствующие определенному квазипостоянному состоянию атмосферы. В каждом из этих сегментов будет определяться свой тип нестационарности, задаваемый соответствующими постоянными значениями  $a_{ij}$ .

Для сегментации уровней ряда необходим, по крайней мере, один измеримый информационный индикатор, рассматриваемый как характеристика состояния атмосферы. Будем считать, что состояние атмосферы характеризуется динамикой ее переходных процессов, это очевидно находит отражение в динамике изменения атмосферной температуры. Тогда, в качестве информационного индикатора или критерия однородности можно принять изменения температуры по координатам сетки, вычисляемые как вектор центральных разностей в каждом узле пространственной сетки  $\Delta_{ij} = (\delta_i; \delta_j)$ , где  $\delta_i = y_{i+1,j} - y_{i-1,j}$ ;  $\delta_j = y_{i,j+1} - y_{i,j-1}$ . Задача состоит в выделении таких сегментов временного ряда (1), у которых значения критерия  $\Delta$  примерно равны, а значит сохраняются основные динамические закономерности в атмосфере. Такая задача относится к классу задач апостериорной сегментации [9]. Каждой измеренной температуре  $y_{ij}(t)$  в однозначное соответствие ставится значение критерия  $\Delta_{ij}(t)$ . Будем понимать под качественной однородностью совокупности данных сходство всех элементов этой совокупности по какому-либо признаку и несходство по всем остальным. Тогда можно говорить, что уровни  $y_{ij}(t_n)$  и  $y_{kl}(t_m)$  качественно однородны, если им соответствуют близкие значения  $\Delta_{ij}(t_n)$  и  $\Delta_{kl}(t_m)$ .

На длинных рядах температурная динамика претерпевает изменения с многократными повторами состояний. Сегменты, соответствующие таким повторам, могут объединяться в однородные кластеры. Необходимо учитывать, что четких границ у сегментов быть не может в силу приблизительности наших представлений о неизменности закономер-

ностей динамики, следовательно априори неизвестно и число сегментов. Поэтому от задачи сегментации, где время выполняет функцию упорядочения данных в узлах сетки и присутствует в явном виде, мы переходим к задаче построения кластеров температур неупорядоченных как по времени, так и по узлам сетки и однородных в смысле критерия  $\Delta$ , на множестве измерений в различные моменты времени  $t$  во всех узлах регулярной сетки. Однако в каждом узле и в каждый момент времени соответствия, определяемые уравнением (1), запоминаются, т. е. запоминается соответствие

$$S: \{y_{ij}(t+1)\} \rightarrow \{(y_{ij}(t); y_{i-1,j}(t); y_{i+1,j}(t); y_{i,j-1}(t); y_{i,j+1}(t))\}.$$

Для построения кластеров воспользуемся алгоритмом многомерной классификации [10], основанном на выделении областей с локальной "плотностью" точек признакового пространства  $\Delta$ . В качестве меры "локальной плотности" предлагается функция принадлежности  $f(\Delta_{ij}, \Delta, R) = |\eta(\Delta_{ij}, \Delta, R)|$  — число точек, попавших в гиперсферу заданного радиуса  $R$  с центром в точке  $\Delta_{ij}$ ;  $\eta$  — множество точек гиперсферы заданного радиуса  $R$  с центром в точке  $\Delta_{ij}$ . В соответствии с алгоритмом [8] определяются локальные максимумы функции принадлежности и соответствующие точки объявляются модами классов локальной "плотности". Полученное разбиение является оптимальным в том смысле, что выделяются все унимодальные множества и полностью реализуется их разделение.

Каждый кластер с номером  $k$  можно рассматривать как статистическую совокупность значений температур  $y_i^k$ ,  $i = 1, \dots, n_k$ , где  $n_k$  — объем совокупности с качественной однородностью, наличием вариаций и возможностью нахождения статистических оценок. В частности, можно найти оценку среднего совокупности  $\bar{y}^k = \frac{\sum y_i^k}{n_k}$  и ввести центрированную случайную переменную

$$x^k = \{x_i^k\} = \{y_i^k - \bar{y}^k\} \quad (2)$$

с нулевым средним.

На основании запомненного соответствия  $S$  и замены переменных (2) для каждого кластера  $k$  можно записать новые соответствия

$$SC: \{x_{ij}^k(t+1)\} \rightarrow \{(x_{ij}^k(t); x_{i-1,j}^k(t); x_{i+1,j}^k(t); x_{i,j-1}^k(t); x_{i,j+1}^k(t))\}$$

между центрированными переменными в виде следующих уравнений:

$$x_{ij}^k(t+1) = b_1^k x_{ij}^q(t) + b_2^k x_{i-1,j}^r(t) + b_3^k x_{i+1,j}^p(t) + b_4^k x_{i,j-1}^s(t) + b_5^k x_{i,j+1}^d(t) + \xi_{ij}^k(t+1), \quad (3)$$

**Оценки параметров модели (3)  
для кластеров с однородной статистикой**

Но- мер клас- тера	$b_1^k$	$b_2^k$	$b_3^k$	$b_4^k$	$b_5^k$	$R^2$
$K_1$	-9,29824	1,01641	0,0991023	5,34475	3,8321	0,890
$K_2$	-7,1782	1,03939	-0,0143984	4,25046	2,90878	0,901
$K_3$	-2,80172	0,840422	0,026325	2,37757	0,554852	0,911
$K_4$	-5,95311	1,05258	0,146581	3,20532	2,54854	0,899
$K_5$	-8,37952	1,03208	0,141075	4,8547	3,34873	0,888
$K_6$	-5,92641	0,916967	0,0381898	3,57287	2,40352	0,908
$K_7$	-4,68736	0,851404	0,0612615	3,14575	1,62743	0,911
$K_8$	-8,69499	0,702887	0,149402	5,07142	3,77392	0,880
$K_9$	-5,28913	0,970231	0,0810225	3,13149	2,1009	0,919
$K_{10}$	-6,07191	0,925191	0,0958082	3,62117	2,42331	0,910

где  $x_{ij}^k(t) = y_{ij}^k(t) - \bar{y}^k$ ,  $k = 1, \dots, m$ ;  $m$  — число кластеров;  $k, q, r, p \in N$  — множество номеров кластеров. Следует обратить внимание на то, что центрированные уровни ряда в правой части уравнения (3), вообще говоря, необязательно принадлежат кластеру с номером  $k$ .

Оценки коэффициентов  $b^k$  в уравнении (3) можно находить с помощью метода наименьших квадратов:  $b^k = (X^{kT} X^k)^{-1} X^{kT} x^k$ , где  $X^k$  — матрица наблюдений факторов, которая формируется на основе запомненных соответствий SC;  $x^k$  — вектор наблюдений объясняемой переменной.

Каждому кластеру в функциональное соответствие можно поставить вектор коэффициентов уравнения (3)  $b^k = (b_1^k; b_2^k; b_3^k; b_4^k; b_5^k)$ ,  $k = 1, \dots, m$ :

$$F: \{\bar{\Delta}^k(x^k)\} \rightarrow \{b^k\}, \quad (4)$$

где  $\{\bar{\Delta}^k(x^k)\}$  — среднее значение классификационного критерия в  $k$ -м кластере. Дискретное соответствие  $F$  в каждом кластере можно аппроксимировать непрерывной зависимостью  $b(x) = (b_1, \dots, b_5)$  и построить систему моделей

$$x_{ij}(t+1) = b_1(x)x_{ij}(t) + b_2(x)x_{i-1,j}(t) + b_3(x)x_{i+1,j}(t) + b_4(x)x_{i,j-1}(t) + b_5(x)x_{i,j+1}(t) + \xi_{ij}^k(t+1), \\ i = 2, \dots, n-1; j = 2, \dots, m-1, \quad (5)$$

описывающую многомерный нестационарный временной ряд со структурными разрывами. Если в качестве аппроксиматора принять искусственную нейронную сеть (ИНС)  $F_{\text{ИНС}} = b(x)$ , то получим искомую комбинированную модель.

## 2. Экспериментальное исследование комбинированной модели

Для построения и экспериментальной апробации комбинированной модели использовали статистические данные реанализа параметров атмосферы [5], приведенные к среднесуточным значениям. Рассматривали процессы изменения температуры при геопотенциале 300 ГПа в узлах регулярной сетки от 0 до 180° восточной долготы и от 40 до 70° северной широты с шагом 2,5°.

Согласно алгоритму кластеризации было выделено 1800 кластеров однородных данных в смысле критерия  $\Delta$ , 1118 из которых оказались репрезентативны. Основная сложность формирования кластеров заключалась в построении и запоминании взаимных соответствий внутри каждой пары точек на большом объеме входных данных (284 284 точки). Для осуществления таких объемов вычислений использовали 24-процессорный параллельный компьютерный кластер высокой производительности.

Параметры модели (5) определяли с использованием МНК. Было построено 1118 авторегрессионных уравнений, номера которых соответствуют номерам кластеров. Оценки соответствующих векторов параметров  $b^k$ , например, для первых 10 кластеров представлены в таблице, где кластеры обозначены номерами  $K_1$ — $K_{10}$ ,  $R^2$  — коэффициент детерминации.

Оценку характера изменения параметров отдельно для каждого класса выполнял универсальный аппроксиматор — многослойный перцептрон, с вектором входа  $x = (x_{i-1,j}; x_{i+1,j}; x_{j-1}; x_{j+1})$  и вектором выхода  $b(x)$ . Для обучения был использован трехслойный перцептрон с четырьмя нейронами входного слоя, четырьмя нейронами скрытого слоя и пятью нейронами выходного слоя. Для обучения сети использовали инструментальную среду MATLAB 10. В качестве функций активации входного и скрытого слоев использовалась сигмоидальная функция "logsig", для выходного слоя — линейная функция "purelin". Обучение ИНС проводилось с помощью алгоритма Levenberg-Marquardt ("trainlm").

Результаты моделирования оценивали на контрольной выборке температурных данных. На рис. 2 (см. третью сторону обложки) представлены результаты работы нейронной сети — график прогноза и реальных температурных наблюдений.

По оси абсцисс — номера 800 узлов, расположенные в линейном порядке, по оси ординат — значения прогноза и реальных температурных наблюдений.

Если расположить те же точки в соответствии с их географическими координатами, получим графики поверхностей реальных температурных показаний (рис. 3, см. третью сторону обложки) и результатов моделирования (рис. 4, см. третью сторону обложки). На рис. 3 плоскость  $XOY$  соответствует значениям географических координат, по оси  $Z$  отложены значения реальных температурных показаний (°C).

На рис. 4 плоскость  $XOY$  соответствует значениям географических координат, по оси  $Z$  отложены

значения результатов моделирования температурных показаний ( $^{\circ}\text{C}$ ).

В качестве показателя эффективности модели использовали коэффициент детерминации  $R^2$  и показатель процентной погрешности прогноза MAPE:

$$\text{MAPE} = \frac{1}{T} \sum_{t=1}^T \left| \frac{x(t) - \hat{x}(t)}{x(t)} \right| \cdot 100 \%, \quad (6)$$

где  $x(t)$ ,  $\hat{x}(t)$  — фактические и расчетные значения уровней ряда соответственно. MAPE не должен превышать 10—12 %.

В нашем случае  $R^2$  получился равным 0,9105, MAPE — 3,4 %.

### Заключение

Результаты моделирования показали, что использование комбинированной модели с выделением кластеров однородной статистики для построения прогноза профиля атмосферной температуры может обеспечить приемлемые характеристики прогноза. Ранее проведенные исследования при моделировании поведения временного ряда в одном узле, результаты сравнительного анализа которых представлены в работе [7], позволяют сделать вывод о возможности построения прогноза температурного профиля по множеству узлов пространственной сетки с точностью не ниже точности прогнозирования атмосферной температуры в одном, отдельно моделируемом узле. Полученные результаты могут быть использованы для построения вертикальных

температурных профилей и, в конечном итоге, для составления рекомендаций при выполнении полетов авиации.

### Список литературы

1. Гребенюк Е. А. Методы анализа нестационарных временных рядов с неявными изменениями свойств // Автоматика и телемеханика. 2005. № 12. С. 3—29.
2. Орлов Ю. Н., Шагов Д. О. Индикативные статистики для нестационарных временных рядов / Препринты ИПМ им. М. В. Келдыша. 2011. № 53. 20 с.
3. Fogler H. R. A pattern recognition model for forecasting // Management science. 1974. N. 8. P. 1178—1189.
4. Alfarcos H. K., Nazeeruddin M. Electric load forecasting: literature survey and classification of methods // International Journal of Systems Science. 2002. Vol. 33. P. 23—34.
5. NCEP/DOE AMIP II Reanalysis [Электронный ресурс]. URL: <http://www.cdc.noaa.gov/cdc/data.ncep.reanalysis2.html>.
6. Матвеев М. Г., Михайлов В. В., Семенов М. Е., Сирота Е. А. Модель анализа динамики векторного метеорологического процесса // Вестник ВГУ. Серия "Системный анализ и информационные технологии". 2013. № 1. С. 89—94.
7. Матвеев М. Г., Сирота Е. А. Разработка и исследование статистических моделей нестационарного многомерного временного ряда атмосферных температур в условиях неоднородности // Информационные технологии. 2014. № 12. С. 20—24.
8. Лукашин Ю. П. Адаптивные методы краткосрочного прогнозирования временных рядов. М.: Финансы и статистика, 2003. 415 с.
9. Тищенко А. К., Плисс И. П. Сегментация многомерных нестационарных временных рядов с помощью метода нечеткой кластеризации // Восточно-Европейский журнал передовых технологий. 2012. Т. 4. № 4 (58). С. 22—25.
10. Буховец А. Г. Последовательное применение алгоритмов многомерной классификации. Многомерный анализ социологических данных (методические указания, алгоритмы и описание программ). М.: ИСИ АН СССР, 1981. С. 24—73.

M. G. Matveev, Ph. D., Professor, Voronezh State University;

V. V. Mihailov, Ph. D., Professor, Air Force Academy

named after Professor N. E. Zhukovsky and U. A. Gagarin,

E. A. Sirota, Ph. D. Associate Professor, [atoris@list.ru](mailto:atoris@list.ru), Voronezh State University

## Combined Forecasting Model of Non-Stationary Multivariate Time Series for the Construction of the Spatial Profile of Atmospheric Temperature

*In this paper we study, construction and analysis of a combined forecast model of non-stationary multivariate time series in order to build the spatial profile of atmospheric temperature.*

*The basic idea of constructing a combined model suggests that an adequate description of the dynamics of changes in temperature can be represented by modifying vector autoregression model with variable parameters characterizing changes in the state of the atmosphere. Non-linear patterns of change in the parameters of vector autoregression can be described by the INS, the inputs of which are any indication states of the atmosphere.*

*The simulation results showed that the use of combined model with the release of clusters of homogeneous statistics for the forecast profile of atmospheric temperature can provide acceptable performance forecast.*

*Earlier studies in modeling the behavior of the time series in one unit suggest the possibility of constructing the forecast temperature profile across multiple nodes of spatial grid with an accuracy of not less than the prediction of atmospheric temperature in one, separately simulated node. The results can be used to construct vertical temperature profiles and, ultimately, to make recommendations in the performance of aviation.*

**Keywords:** multi-dimensional non-stationary time series forecasting model, meteorology, vector autoregression, profile atmospheric temperature, modeling classes homogeneous statistics

## References

1. Grebenyuk E. A. Metody analiza nestacionarnykh vremennykh ryadov s neyavnymi izmeneniyami svoystv, *Avtomatika i telemekhanika*, 2005, no. 12, pp. 3–29.
2. Orlov Yu. N., Shagov D. O. *Indikativnye statistiki dlya nestacionarnykh vremennykh ryadov*, Preprinty IPM im. M. V. Keldysha, 2011, no. 53, 20 p.
3. Fogler H. R. A pattern recognition model for forecasting, *Management science*, 1974, no. 8, pp. 1178–1189.
4. Alfares U. K., Nazeeruddin M. Electric load forecasting: literature survey and classification of methods, *International Journal of Systems Science*, 2002, vol. 33, pp. 23–34.
5. NCEP/DOE AMOP II Reanalysis [Elektronnyy resurs], URL: <http://www.cdc.noaa.gov/cdc/data.ncep.reanalysis2.html>.
6. Matveev M. G., Mihajlov V. V., Semenov M. E., Sirota E. A. Model' analiza dinamiki vektornogo meteorologicheskogo processa *Vestnik VGU, seriya "Sistemnyy analiz i informacionnye tekhnologii"*, 2013, no. 1, pp. 89–94.
7. Matveev M. G., Sirota E. A. Pazpabotka i issledovanie statisticheskikh modelej nestacionapnogo mnogomernogo vremennoy ryada atmosferynykh temperatuy v usloviyah neodnopolnosti, *Informacionnye tekhnologii*, 2014, no. 12, pp. 20–24.
8. Lukashin Yu. P. Adaptivnye metody kratkosrochnogo prognozirovaniya vremennykh ryadov, Moscow, Finansy i statistika, 2003, 415 p.
9. Tishchenko A. K., Pliss I. P. Segmentatsiya mnogomernykh nestacionarnykh vremennykh ryadov s pomoshch'yu metoda nechetkoj klasterizatsii, *Vostochno-Evropejskiy zhurnal peredovykh tekhnologii*, 2012, vol. 4, no. 4 (58). P. 22–25.
10. Buhovec A. G. *Posledovatel'noe primenenie algoritmov mnogomernoy klassifikatsii, Mnogomernyy analiz sociologicheskikh dannykh (metodicheskie ukazaniya, algoritmy i opisaniya programm)*, 1981, Moscow, ISI AN SSSR, pp. 24–73.

УДК 519.857

**В. И. Струченков**, д-р техн. наук, проф., e-mail: str1942@mail.ru,  
Московский государственный университет информационных технологий,  
радиотехники и электроники (МИРЭА)

## Динамическое программирование в задачах планирования реализации частично возобновляемых ресурсов

*Рассматривается задача составления оптимального поэтапного плана использования частично возобновляемого однородного ресурса в течение заданного времени. Анализируются различные модели целевой функции как разности дохода от использования ресурса или его части и сопутствующих затрат. Для простых моделей на основе анализа решения задачи с помощью метода динамического программирования получены расчетные формулы, что позволяет избежать перебора вариантов пошаговых решений.*

**Ключевые слова:** ресурс, целевая функция, множество состояний, динамическое программирование, оптимальный путь

### Введение

Для классической задачи оптимального распределения заданного количества некоторого однородного ресурса [1] известны метод динамического программирования [2] и его новые реализации, усовершенствованные за счет использования множеств Парето [3] и комбинации динамического программирования с методом ветвей и границ [4]. При этом заданное начальное количество ресурса на каждом этапе могло только уменьшаться.

Однако динамическое программирование можно применить и в задачах распределения частично возобновляемых ресурсов. Например, промысловая рыба, морские животные и вообще "население" лесов, морей, озер и рек являются частично возобновляемыми природными ресурсами. Возникает задача оптимального использования таких ресурсов, т. е. планирования их добычи на ряд лет вперед, с тем чтобы обеспечить максимальную прибыль за все время планирования и сохранить минимально необходимый объем ресурса. Если в начале планируемого периода объем ресурса несущественно превышает заданный минимум, то решение задачи может показаться тривиальным: ежегодно брать у

природы только прирост ресурса и не более того. Однако это решение не очевидно, если учесть, что при сохранении неприкосновенности ресурса через несколько лет можно взять больше, чем при ежегодном изъятии прироста.

Кроме возобновляемых природных ресурсов возможны и другие процессы, в которых ресурс не только расходуется, но и частично возобновляется.

Цель настоящей статьи состоит в анализе некоторых моделей подобной задачи и алгоритмов ее решения по методу динамического программирования.

### Постановка задачи

Задача состоит в следующем: в начальный момент времени имеется заданное количество частично возобновляемого ресурса  $b_1$ .

Заданный период планирования  $T$  (например,  $T = 10$  лет) разбивается на этапы, например годы или месяцы. Для определенности примем годы и будем считать, что планирование начинается с года номер один. Тогда период планирования, номер последнего года и число лет — это одно и то же число  $T$ .

При реализации  $x$  единиц ресурса в течение года доход равен  $d(x)$ , а сопутствующие затраты в тече-

ние года есть функция  $c(x, b)$  от количества реализуемого ресурса  $x$  и количества ресурса на начало года  $b$ . Ресурс частично возобновляемый, и при наличии на начало произвольного года  $t$  ( $t = 1, 2, \dots, T$ )  $b_t$  единиц ресурса и реализации в течение этого года  $x_t$  его единиц на начало следующего года будет  $b_{t+1} = p(b_t - x_t)$  единиц ресурса. Коэффициент  $p$  считается известным. В начале последнего года  $T$  при рассмотрении вариантов реализации ресурса в этот год считается, что оставшееся на  $(T + 1)$ -й год количество ресурса уже не имеет значения. Поэтому в начале последнего года  $T$  задача принятия решения существенно упрощается.

Требуется построить такой план реализации ресурса, чтобы в течение  $T$  лет суммарный количественный показатель качества плана (целевая функция) принял максимальное значение. При этом могут использоваться различные целевые функции, например, суммарная прибыль, экономическая эффективность и др.

Если целевая функция — это суммарная прибыль, то задача сводится к следующему формальному представлению.

Найти вектор  $\mathbf{x}(x_1, x_2, \dots, x_T)$ , при котором достигает максимума

$$\sum_{t=1}^T (d(x_t) - c(x_t, b_t)) \text{ и } 0 \leq x_t \leq b_t; b_{t+1} = p(b_t - x_t).$$

Для решения конкретной задачи применительно к построению оптимального плана вылова форели в [5] предлагается классическая схема динамического программирования. Ключевое для метода динамического программирования понятие "состояние системы" формализуется как количество имеющегося ресурса. Соответственно "траектория" (или "путь") — это последовательность состояний, т. е. значений имеющегося ресурса в начале каждого года. В начале года  $t$  множество состояний — это множество возможных значений ресурса  $b_t$ . Процесс рассматривается от "конца к началу", т. е. начиная с множества состояний  $b_T$  в начале последнего года. Заметим, что число состояний  $b_T$  в начале последнего года  $T$ , т. е.  $b_T = b_1 p^{T-1} + 1$ , может быть велико при реальных  $b_1$ ,  $p$  и  $T$ , поэтому усовершенствования классической схемы динамического программирования актуальны с точки зрения объема вычислений.

Далее, **для каждого состояния**  $b_T$ , т. е. для всех возможных значений  $b_T$  (от 0 до  $b_1 p^{T-1}$ ), определяется "оптимальный путь до конца", т. е. оптимальное количество ресурса, реализуемого в последний год ( $x_T^*$ ), и вычисляется соответствующее ему значение целевой функции  $f_T(b_T)$ :

$$f_T(b_T) = \max_{x_T} \{d(x_T) - c(x_T, b_T)\},$$

где  $0 \leq x_T \leq b_T$ .

Далее рассматриваются два последних года и **для каждого состояния**  $b_{T-1}$  перебором всех возможных переходов в состояния  $b_T$  в начале послед-

него года определяется  $x_{T-1}^*$  и соответствующее ему значение целевой функции  $f_{T-1}(b_{T-1})$ :

$$f_{T-1}(b_{T-1}) = \max_{x_{T-1}} \{d(x_{T-1}) - c(x_{T-1}, b_{T-1}) + f_T(b_T)\},$$

где  $0 \leq x_{T-1} \leq b_{T-1}$ .

И так до тех пор, пока не будет найден  $x_1^*$  при заданном  $b_1$ . Значения  $x_t^*$ , при которых достигается максимум, запоминаются. Другими словами, рекомендуется двигаться по сетке состояний в обратном направлении в предположении, что в начале очередного года возможно любое количество ресурса от нуля до максимального, получаемого при полном отсутствии реализации ресурса.

Далее, зная  $x_1^*$ , вычисляем соответствующее ему  $b_2^* = p(b_1 - x_1^*)$ . Ему соответствует  $x_2^*$  и т. д. до восстановления оптимальной траектории, т. е. построения плана реализации ресурса. Заметим, что в данной задаче промежуточные значения  $b_{t+1} = p(b_t - x_t)$  при целых значениях исходного количества ресурса  $b_1$  и последующих назначаемых  $x_t$  не обязаны быть целыми при не целом  $p$ .

По поводу выбора порядка рассмотрения этапов в [5] утверждается: "В большинстве приложений динамическое программирование получает оптимальное решение путем движения в обратном направлении — от конца задачи к началу". Не располагая данными обо всех приложениях динамического программирования, мы не будем опровергать это утверждение, однако отметим, что именно в данной задаче при ее численном решении удобно и более эффективно двигаться от начала к концу, а аналитические выкладки, наоборот, удобнее выполнять при рассмотрении задачи в обратном направлении, т. е. от последнего этапа к первому.

Конкретный вид функций  $d(x)$  — доход и  $c(x, b)$  — затраты в [5] не приводятся. Приводятся только  $b_1 = 10\,000$ ,  $p = 1,2$  и  $T = 10$ .

Далее будем рассматривать задачу в общем виде.

Для начала отметим, что для простых моделей можно, используя принцип оптимальности Р. Беллмана [1, 2], получать последовательно, начиная с  $t = T$  аналитические выражения оптимального количества реализуемого ресурса  $x_t^*$  как функции  $b_t$ , не прибегая к численным методам и не разбивая сетку варьирования вообще. Рассмотрим две такие модели, различающиеся функциями дохода  $d(x)$  и затрат  $c(x, b)$ .

**Первая модель:**  $d(x) = ax$ , где  $a$  — доход от реализации единицы ресурса. Вид функции дохода вполне естественен. Для затрат  $c(x, b)$  можно предположить, что они возрастают с ростом  $x$  и убывают при заданном  $x$  с ростом  $b$  — количества имеющегося ресурса. Действительно, например, применительно к разведению форели выловить 100 форелей легче, если их много, скажем 10 000, чем, если их всего 100.

Для первой модели примем  $c(x, b) = (k/b)x$ , где  $k$  — заданное число. Прибыль при реализации  $x$  единиц ресурса из имеющихся  $b$  единиц составит

$ax - (k/b)x$ . Здесь  $k/b$  — затраты при реализации единицы ресурса из имеющихся  $b$  единиц. Соответственно  $k$  — затраты при полной реализации ресурса. Выражение  $ax - (k/b)x$  только условно можно назвать прибылью при наличии затрат, не зависящих от количества реализуемого ресурса. Теоретически наличие в целевой функции постоянного слагаемого несущественно, так как оно не влияет на точку экстремума, т. е. на искомое решение. Мы не будем учитывать наличие постоянной составляющей затрат на реализацию ресурса. Однако формальный оптимум может соответствовать решению, при котором в каком-нибудь году  $ax - (k/b)x$  меньше постоянной составляющей годовых затрат на реализацию ресурса, поэтому годовая прибыль отрицательна, хотя суммарно за все годы прибыль максимальна. Если отрицательная годовая прибыль недопустима, то на каждом этапе появляется ограничение на минимальное количество реализации ресурса. Это ограничение несколько усложняет алгоритм поиска. Для простоты изложения вначале допустим возможность реализации любого количества имеющегося ресурса, т. е. будем рассматривать только переменную часть целевой функции. В дальнейшем внесем необходимые уточнения.

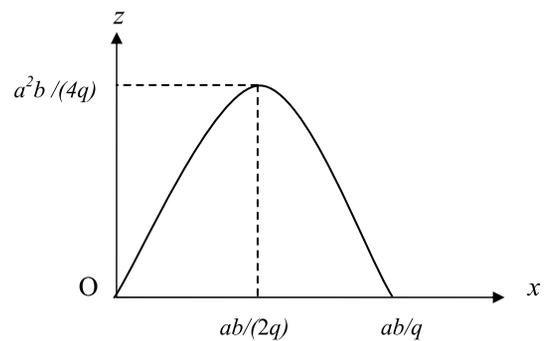
Рассмотрим последний год  $T$  и решим вопрос, сколько нужно реализовать ресурса из имеющихся  $b_T$  единиц, чтобы получить максимальную прибыль. Ответ очевиден: если  $a > k/b_T$ , нужно реализовать ресурс полностью ( $x_T^* = b_T$ ), иначе — не использовать его вообще ( $x_T^* = 0$ ). Мы нашли  $x_T^*$  — количество реализуемого ресурса в год  $T$  как функцию  $b_T$  ( $x_T^* = 0$  или  $x_T^* = b_T$ ).

Переходим к предпоследнему шагу  $T - 1$ . Суммарная прибыль за два последних года линейно зависит от  $x_{T-1}$ , так как  $b_T = p(b_{T-1} - x_{T-1})$ . Линейная функция достигает экстремальных значений на концах рассматриваемого интервала  $0 \leq x_{T-1} \leq b_{T-1}$ . Нужно взять  $x_{T-1} = 0$ , так как в этом случае  $b_T > b_{T-1}$  и суммарная прибыль за два года больше, если вообще имеет смысл реализация ресурса. Рассуждая аналогично применительно ко всем годам, для линейной модели приходим к выводу: оптимальный план состоит в том, чтобы вообще не расходовать ресурс до последнего года и только в последний год реализовать его полностью, если  $a > k/(b_1 p^{T-1})$ , или не заниматься реализацией ресурса вообще в противном случае, который не реален.

Если потребность в реализации ресурса не позволяет накапливать его до последнего года, то следует ограничиться минимально необходимой его реализацией.

Может показаться, что этот же вывод о сохранении ресурса до последнего года справедлив не только для линейной, но и для любой монотонно возрастающей целевой функции. Однако это не так, как будет ясно из дальнейшего изложения.

**Вторая модель.** В первой модели затраты на полную реализацию ресурса ( $k$ ) не зависят от его



Зависимость прибыли  $z(x)$  от количества реализуемого ресурса  $x$

количества. Если оставить ту же функцию дохода  $d(x)$  и рассмотреть новую функцию затрат вида  $c(x, b) = (q/b)x^2$ , то для переменной части прибыли  $z$  получается функция  $z(x, b) = ax - (q/b)x^2$  (см. рисунок), где  $a$  и  $q > 0$  заданы. В этом случае  $a$  — это по-прежнему доход при реализации единицы ресурса, а  $q$  — затраты на реализацию единицы ресурса в случае его полной реализации. Другими словами,  $a/q$  — отношение дохода к затратам при полной реализации ресурса. Очевидно неравенство  $a > q$ . Если в  $c(x, b)$  есть линейный член, то его можно объединить с  $d(x)$ .

График зависимости  $z(x)$  показывает, что при  $a/(2q) < 1$  точка максимума  $x_{\max} = ab/(2q) < b$  и нет смысла рассматривать все  $x > x_{\max}$ , так как при  $x = x_{\max}$  больше прибыли и остается больше ресурса, чем при  $x > x_{\max}$ . Если же  $a/(2q) \geq 1$ , то нужно принять  $x_{\max} = b$ , так как  $0 \leq x \leq b$ .

Рассмотрим последний год  $T$  и решим вопрос, сколько ( $x_T^*$ ) нужно реализовать ресурса из имеющихся  $b_T$  единиц, чтобы получить максимальную прибыль? Максимум прибыли  $ax_T - (q/b_T)x_T^2$  достигается в точке  $x_T^* = ab_T/(2q)$ .

Поскольку должно выполняться условие  $0 \leq x \leq b$ , то при  $a/(2q) > 1$  следует принять  $x_T^* = b_T$ . Таким образом,  $x_T^* = \gamma_T b_T$ , где  $\gamma_T = \min(1, a/(2q))$ .

Это означает, что если при полной реализации ресурса доход вдвое и более превышает затраты, то нужно реализовать весь ресурс, иначе — его  $a/(2q)$  часть.

Нецелесообразность полной реализации ресурса в последнем году, т. е. неравенство  $a/(2q) < 1$ , представляется нереальной. Однако мы будем рассматривать общий случай.

Получили результат: **за последний год оптимальный размер реализации ресурса линейно зависит от его количества в начале года.** Коэффициент пропорциональности  $\gamma_T = a/(2q)$  при  $a/(2q) < 1$  и  $\gamma_T = 1$  в противном случае. Прибыль  $z_t$  за любой **один** год  $t$  при  $x_t^* = \gamma_t b_t$  равна

$$z_t^* = a\gamma_t b_t - q/b_t (\gamma_t b_t)^2 = \gamma_t (a - \gamma_t q) b_t = \lambda_t b_t \quad (1)$$

где  $\lambda_t = \gamma_t (a - \gamma_t q)$ .

Другими словами, **годовая** прибыль линейно зависит от имеющегося ресурса  $b_t$  при линейной зависимости количества реализации от  $b_t$ . На этом основаны все дальнейшие построения.

Для последнего года все определено: при любом  $b_T$  надо реализовать  $\gamma_T b_T$  единиц ресурса и получить прибыль  $\lambda_T b_T$ .

За год  $T$  прибыль  $z_T = a^2 b_T / (4q)$  при  $\gamma_T = a / (2q)$  и  $x_T = (a - q) b_T$  при  $\gamma_T = 1$ .

Будем обозначать годовую прибыль за год с номером  $r$  через  $z_r$ , а суммарную прибыль за все оставшиеся годы, начиная с  $r$ -го, через  $Z_r$ .

Переходим к предпоследнему году  $T - 1$ . Найдем  $x_{T-1}^*$ , для которого максимальна суммарная прибыль за два последних года. В терминах динамического программирования это означает: из состояния  $b_{T-1}$  найдем такой переход в новое состояние  $b_T$ , чтобы суммарная прибыль  $Z_{T-1}$  за последние два года (этапа) была максимальна. Используя формулу (1) применительно к году  $T$ , получаем для прибыли за два последних года формулу

$$Z_{T-1} = ax_{T-1} - (q/b_{T-1})x_{T-1}^2 + \gamma_T(a - \gamma_T q)(b_{T-1} - x_{T-1})p,$$

так как  $b_T = (b_{T-1} - x_{T-1})p$ .  $Z_{T-1}$ , как функция  $x_{T-1}$ , максимальна при  $x_{T-1}^* = (a(1 - p\gamma_T) + qp\gamma_T^2) \times b_{T-1} / (2q)$ . В силу  $x_{T-1} \leq b_{T-1}$  имеем

$$x_{T-1}^* = \gamma_{T-1} b_{T-1},$$

где  $\gamma_{T-1} = \min(1, (a(1 - p\gamma_T) + qp\gamma_T^2) / (2q))$ .

При  $\gamma_T = a / (2q) \leq 1$  имеем

$$(a(1 - p\gamma_T) + qp\gamma_T^2) / (2q) = a / (2q) - a^2 p / (8q^2) < \gamma_T \leq 1.$$

При  $a / (2q) > 1$

$$\gamma_T = 1, x_{T-1}^* = (a(1 - p) + qp) / (2q) b_{T-1}; \\ \gamma_{T-1} = a(1 - p) / (2q) + p/2.$$

Так как  $p > 1$ , то и в этом случае  $\gamma_{T-1} < \gamma_T = 1$ . Действительно, предположим противное, т. е.  $a(1 - p) / (2q) + p/2 > 1$ . Это неравенство равносильно неравенству  $(p - 1)a/q < (p - 2)$ , которое не может выполняться при  $p > 1$  и  $a / (2q) > 1$ .

Поэтому можно записать  $\gamma_{T-1} = (a(1 - p\gamma_T) + qp\gamma_T^2) / (2q)$ , опуская операцию взятия минимума. Полученное неравенство  $\gamma_{T-1} < 1$  означает нецелесообразность полной реализации ресурса в предпоследнем году, что вполне естественно. Отсюда следует и нецелесообразность полной реализации ресурса и в любой предшествующий год, так как, допуская полную реализацию в следующем за ним году, получаем лучшее решение.

Интересно отметить, что  $\gamma_{T-1}$  может оказаться отрицательным. Это означает, что  $x_{T-1}^* = 0$  и реализация ресурса в  $(T - 1)$ -м году нецелесообразна вообще.

При  $a / (2q) < 1$   $\gamma_{T-1} = a / (2q)(1 - a / (2q)p/2)$ . И  $\gamma_{T-1} < 0$  при  $2/p < a / (2q) < 1$ .

Если это неравенство не выполнено, то  $0 < \gamma_{T-1} < 1$  и целесообразна реализация ресурса при любом его

количестве. Например, при  $p = 2$  и  $a / (2q) = 0,8$  получаем  $\gamma_{T-1} = 0,16$ .

При  $a / (2q) \geq 1$   $\gamma_T = 1$  и  $\gamma_{T-1} = a(1 - p) / (2q) + p/2$ . Условие  $\gamma_{T-1} < 0$  выполнено при  $a / (2q) \geq p / (2p - 2)$ . Это условие может быть выполнено как при  $p / (2p - 2) > 1$ , так и при  $p / (2p - 2) < 1$ . Так, при  $p = 1,2$  должно быть  $a / (2q) \geq 3$ , а при  $p = 2,2$  получаем  $a / (2q) \geq 11/12$ , т. е. любое  $a / (2q) \geq 1$ . При  $a / (2q) \geq 1$  прибыль за один год — это монотонно возрастающая функция использованного ресурса и даже в этом случае при соответствующих значениях  $a / (2q)$  и  $p$  в предпоследний год целесообразна реализация значительной части ресурса независимо от его количества. Например, при  $p = 1,2$  и  $a / (2q) = 1,5$  эта часть, т. е.  $\gamma_{T-1}$ , составляет 30 %.

Далее, прибыль за один  $(T - 1)$ -й год равна  $\lambda_{T-1} b_{T-1}$ , где  $\lambda_{T-1} = \gamma_{T-1}(a - \gamma_{T-1}q)$ , что следует из формулы (1) для произвольного года при подстановке  $\gamma_{T-1} b_{T-1}$  вместо  $\gamma_r b_r$ .

Итак, мы получили, что и оптимальное годовое количество реализации ресурса и прибыль за каждый год при оптимальном плане реализации в расчете на максимум прибыли за два оставшихся года линейно зависят от количества ресурса в начале каждого года.

Используя принцип математической индукции, докажем, что эта **линейная** зависимость оптимального годового количества реализации и годовой прибыли от количества ресурса в начале года сохраняется для любого числа лет при максимизации суммарной прибыли за все оставшиеся годы.

Итак, пусть это утверждение верно для лет с номерами  $s, s + 1, s + 2, \dots, T$ . Нам нужно доказать, что это утверждение верно и для всех оставшихся лет, если начинать рассмотрение с  $(s - 1)$ -го года. Тогда будет доказано, что это утверждение верно для любого года, так как для последнего (да и предпоследнего года) это утверждение уже доказано.

Наше предположение означает следующее:

$$Z_s = \lambda_s b_s + \lambda_{s+1} b_{s+1} + \dots + \lambda_T b_T \\ \text{и } x_r^* = \gamma_r b_r \text{ для всех } s \leq r \leq T.$$

Тогда суммарная прибыль за все оставшиеся годы, начиная с  $(s - 1)$ -го, составит:

$$Z_{s-1} = ax_{s-1} - (q/b_{s-1})x_{s-1}^2 + \lambda_s b_s + \lambda_{s+1} b_{s+1} + \dots + \lambda_T b_T.$$

Поскольку для любого  $b_r$  при  $r > s - 1$   $b_{r+1} = p(b_r - x_r) = p(1 - \gamma_r)b_r$ , то

$$Z_{s-1} = ax_{s-1} - (q/b_{s-1})x_{s-1}^2 + b_s \{ \lambda_s + \lambda_{s+1} p(1 - \gamma_s) + \lambda_{s+2} p^2(1 - \gamma_s)(1 - \gamma_{s+1}) + \dots + \lambda_T p^{T-s}(1 - \gamma_s)(1 - \gamma_{s+1}) \dots (1 - \gamma_{T-1}) \}.$$

Обозначив выражение в фигурных скобках через  $D$ :

$$D_{s-1} = \lambda_s + \lambda_{s+1} p(1 - \gamma_s) + \lambda_{s+2} p^2(1 - \gamma_s)(1 - \gamma_{s+1}) + \dots + \lambda_T p^{T-s}(1 - \gamma_s)(1 - \gamma_{s+1}) \dots (1 - \gamma_{T-1}), \quad (2)$$

получим

$$Z_{s-1} = ax_{s-1} - (q/b_{s-1})x_{s-1}^2 + p(b_{s-1} - x_{s-1})D_{s-1}.$$

Максимум  $Z_{s-1}$  достигается при  $x_{s-1}^* = (a - pD_{s-1})b_{s-1}/(2q)$ . Или  $x_{s-1}^* = \gamma_{s-1}b_{s-1}$ , где

$$\gamma_{s-1} = (a - pD_{s-1})/(2q). \quad (3)$$

Как установлено выше,  $\gamma_s - 1 < 1$ , так как  $\gamma_{s-1} = 1$  означает полную реализацию ресурса, что не является оптимальным решением во все годы, кроме последнего.

В соответствии с формулой (1)  $z_{s-1} = \lambda_{s-1}b_{s-1} = \gamma_{s-1}(a - \gamma_{s-1}q)b_{s-1}$ .

Получили, что и  $x_{s-1}^*$  и  $z_{s-1}$  также линейно зависят от  $b_{s-1}$ , что и требовалось доказать.

Из формулы (2) следует рекуррентное соотношение  $D_{s-1} = \lambda_s + D_s p(1 - \gamma_s)$ ;  $s = T, T-1, \dots, 2$ . (4)

В итоге получаем следующий алгоритм для построения оптимального плана реализации ресурса.

1. Вычисляем  $\gamma_T = \min(1, a/(2q))$  и  $\lambda_T = \gamma_T(a - \gamma_T q)$ . Полагаем  $D_T = 0$ .

2. Последовательно применяя формулы (4), (3), (1), вычисляем  $D_{T-1}, \gamma_{T-1}, \lambda_{T-1}$ .

3. Аналогично вычисляем все  $D_r, \gamma_r$  и  $\lambda_r$  для всех  $1 \leq r < T-1$ , начиная с  $r = T-2$ . На каждом шаге запоминаем  $\gamma_r, \lambda_r$  и последнее из вычисленных  $D_r$ .

4. Обратным разворотом, используя заданное  $b_1$  и вычисленные  $\gamma_r$  и  $\lambda_r$ , последовательно находим все  $x_i^*, b_{i+1}^*$  и  $z_i^*$  ( $i = 1, 2, \dots, T$ ).

Естественно, при получении нецелых значений для  $x_i^*$  их следует округлять до целых значений. Суммарную прибыль получим, суммируя  $z_i^*$ .

Определяемая по формуле (3) величина  $\gamma_{s-1}$  на некотором шаге может стать меньше нуля. Если это произойдет, она заменяется нулем. Это означает, что в соответствующий и все предшествующие годы нет реализации ресурса.

Возвращаясь к вопросу о влиянии постоянной составляющей ежегодных затрат, отметим следующую корректировку алгоритма.

1. При обратном развороте после вычисления на очередном этапе с номером  $r$  величин  $b_r$  и  $z_r = \lambda_r b_r$ , соответствующих оптимальному плану, вычитаем из  $z_r$  постоянную составляющую годовых затрат.

2. Если полученная разность (годовая прибыль) положительна, переходим к следующему этапу. В противном случае можно увеличить реализацию, т. е. увеличить  $\gamma_{r-1}$  и соответственно прибыль до требуемого размера. Однако при такой корректировке оптимальной траектории возможно досрочное исчерпание ресурса, что, как уже отмечалось, не соответствует оптимуму. Поэтому представляется предпочтительным при возникновении подобной ситуации с недопустимо малой прибылью уменьшить реализацию ресурса в предшествующие годы, рассматривая их последовательно, или вообще отказаться от реализации ресурса в  $r$ -м году, т. е. принять  $\gamma_{r-1} = 0$  и тем самым перейти в состояние следующего этапа с ресурсом  $b'_r$  ( $b'_r > b_r$ ).

3. После корректировки состояния  $b_r$ , процесс в обратном направлении продолжается с вычисленными ранее величинами  $\gamma_i$  и  $\lambda_i$  ( $i = t, t+1, \dots, T$ ). Эти величины для оптимальной траектории не зависят от количества ресурса на данном шаге, они зависят только от номера шага.

В данной модели легко учесть рост цен (величины  $a, q$  год от года умножаются на соответствующие коэффициенты) и желание получить прибыль поскорее, так как прибыль сегодня не эквивалентна такой же прибыли через несколько лет.

В итоге рассмотрения простых моделей целевой функции можно констатировать, что в данной задаче метод динамического программирования можно использовать не как метод пошагового поиска оптимальной траектории с анализом всех промежуточных состояний, но как метод анализа задачи и построения алгоритма, который сводится к последовательному вычислению оптимальных пошаговых решений по готовым формулам.

Для более сложных моделей трудно рассчитывать на получение расчетных формул по аналогии с приведенными выше. Приходится использовать другую реализацию метода динамического программирования. Например:

1. Начиная с первого года, имея  $b_1$  единиц ресурса, рассматриваем все возможные количества его реализации  $0 \leq x_1 \leq b_1$ . Для каждого из них вычисляем и запоминаем ответствующие значения целевой функции и множество состояний с  $b_2 = (b_1 - x_1)p$ . Переходы с недопустимо малой прибылью отбраковываются.

Тем самым завершается формирование множества состояний после первого года (этапа).

2. Аналогично формируем состояния каждого из последующих этапов, вычисляя суммарные значения целевой функции и оставшиеся количества ресурса. Во избежание полного перебора при достижении одного состояния разными путями в соответствии с принципом оптимальности Р. Беллмана [1, 2] оставляем только путь, по которому это состояние достигается с большим значением целевой функции, и запоминаем соответствующее ему состояние предыдущего этапа (связь).

3. Завершив последний этап, имеем оптимальное значение целевой функции и обратным разворотом по цепочке связей, начиная с оптимального конечного состояния ( $b_{T+1}$ ), восстанавливаем оптимальную последовательность состояний и соответствующие величины реализации ресурса на каждом этапе, т. е. в каждом году.

Неприятная особенность такого алгоритма состоит в необходимости округлений получаемых количеств оставшегося ресурса до целых значений.

При наличии дополнительных требований, например, не оставлять ресурса меньше заданного числа или не рассматривать реализацию ресурса меньше заданной величины легко скорректировать построение множества состояний на каждом этапе.

## Заключение

Для практического использования рассмотренных моделей и алгоритмов необходимо исследование реальных процессов и получение соответствующих реальных данных из практики. Однако возможность применения динамического программирования для составления оптимальных планов реализации частично возобновляемых ресурсов сомнений не вызывает.

В данной задаче возможно существенное сокращение объема вычислений при использовании более совершенной реализации классического алго-

ритма динамического программирования за счет использования множеств Парето [3, 4].

## Список литературы

1. **Беллман Р.** Динамическое программирование. М.: ИЛ, 1960.
2. **Беллман Р., Дрейфус С.** Прикладные задачи динамического программирования. М.: Наука, 1965. 458 с.
3. **Струченков В. И.** Устаревшие стереотипы и новые алгоритмы решения прикладных задач дискретной оптимизации // Информационные технологии. 2012. № 5.
4. **Struchenkov V. I.** Combined Algorithms of Optimal Resource Allocation // Applied Mathematics. Scientific Research. 2012. Vol. 3, N. 1.
5. **Косоруков О. А., Мищенко А. В.** Исследование операций: Учеб. для вузов. М.: Экзамен, 2003. 270 с.

**V. I. Struchenkov**, Professor, e-mail: str1942@mail.ru

Moscow State University of Radio Engineering, Electronics and Automation, Moscow, Russia

## Dynamic Programming for Planning of the Partially Renewable Resources Implementation

*Under study is the problem of optimal planning of the partially renewable resources implementation, such as the commercial breeding of fish, animals and so on.*

*The aim of this article is to study the opportunity of the optimal plan calculation using dynamic programming.*

*The predetermined scheduling period is divided into a number of stages, such as months or years. At each stage, the key concept of dynamic programming "system status" is formalized as the amount of available resources.*

*Two models of the objective function are considered. The first model is linear, and the second model is quadratic. On the basis of the dynamic programming method built computational algorithms that do not require busting options step by step. Instead of these classical algorithms of dynamical programming the recurrence formulas for optimal plan calculation are received.*

**Keywords:** resource, the objective function, the set of states, dynamic programming, the optimal path

## References

1. **Bellman R.** *Dinamicheskoe programirovanie* (Dynamic Programming), Moscow, Foreign Literature, 1960. 234 p. (in Russian).
2. **Bellman R., Drejfus S.** *Prikladnye zadachi dinamicheskogo programirovanija* (Applied Problems of Dynamic Programming), Moscow, Science, 1965, 458 p. (in Russian).

3. **Struchenkov V. I.** Ustarevshie stereotipy i novye algoritmy reshenija prikladnyh zadach diskretnoi optimizacii, *Informacionnye tehnologii*, 2012, no. 5, pp. 20–29 (in Russian).

4. **Struchenkov V. I.** Combined Algorithms of Optimal Resource Allocation. *Applied Mathematics. Scientific Research*, 2012, vol. 3, no. 1.

5. **Kosorukov O. A., Mishenko A. V.** *Issledovanie operacij: Uchebnik dlja studentov vuzov*. Moscow: Exam, 2003, 270 p. (in Russian).

УДК 004.62

**Н. М. Новикова**, д-р техн. наук, проф., e-mail: nov.nelly@gmail.com,

**А. В. Борискин**, аспирант, email: boriskinpost@gmail.com,

Воронежский государственный университет

## Математическая модель поиска влиятельных объектов социальной сети на основе априорной информации

*Для определенного подмножества объектов социальной сети (на примере Vkontakte) некоторые объекты обладают повышенным влиянием на другие объекты. Рассматривается задача поиска остальных влиятельных объектов на основе данной информации. Предложен алгоритм решения, приведены иллюстрации работы разработанного программного обеспечения.*

**Ключевые слова:** виртуальные социальные сети, интеллектуальный анализ данных, нечеткие графы, методы оптимизации

## Введение

В 1992 г. эксперт по бизнес-анализу Григорий Пятецкий-Шапиро ввел термин "data mining" [1]. Дословный перевод "добыча данных" вряд ли следует считать удачным. Наиболее приемлемая версия звучит как "интеллектуальный анализ данных". Под ним скрывается процесс обнаружения в "сырых" данных ранее неизвестных, нетривиальных, практически полезных, доступных интерпретаций знаний, необходимых для принятия решений в различных сферах человеческой деятельности [2]. В современном информационном мире "сырых" данных встречается достаточно много. Наиболее богатыми источниками являются виртуальные социальные сети. В них участники выкладывают на стены записи, фотографии, видео, снабжают их комментариями и оценками, помечают друг друга как друзья. Происходит практически полноценное общение, только в виртуальном формате. Целью данной статьи является обработка этих данных с последующим получением полезных результатов.

## Постановка задачи

Пусть есть множество пользователей социальной сети, содержащее людей разных профессий. В нем могут быть подмножества врачей, программистов, писателей, кинокритиков, фотографов и многих других с иными интересами. Представим, что необходимо распространить информацию среди определенной группы пользователей, например, информацию о выходе нового объектива, который позволяет запечатлеть самые сложные моменты спортивных событий. Очевидно, что эти данные не представляют никакого интереса для людей, которые занимаются ремонтом автомобилей или строительством небоскребов. Данная информация должна дойти до некоторой группы фотографов, которая интересуется съемкой именно спортивных событий. В большинстве случаев в подобных группах существуют люди, которые пользуются авторитетом, к мнению которых прислушиваются. Через таких людей можно начинать распространять информацию. В дальнейшем будем называть подобные объекты *влиятельными*. Задача ставится следующим образом: располагая данными о некоторых влиятельных объектах, найти иные объекты, которые также пользуются повышенным влиянием.

## Теоретический анализ

Социальную сеть принято представлять в виде графа, узлами которого являются люди [3]. Если объекты как-то связаны между собой (состоят в друзьях либо переписываются), то связь между этими объектами обозначается в виде дуги. Полученный в этом случае граф обладает существенным недостатком: неважно, насколько близко знакомы люди друг с другом, важно лишь, существует ли

между ними хоть какая-то связь. В конечно счете он приравняет отношения между случайными знаковыми к отношениям между людьми, которые тесно общаются уже не один десяток лет. Необходимо иметь модель, которая позволяла бы дифференцировать подобные случаи.

Для решения данной задачи может быть использовано понятие нечеткого графа [4]. В нашем случае под графом, представляющим модель данных, понимается пара  $(X, A)$ , где  $X$  — четкое множество вершин, представляющих людей в сети,  $A$  — нечеткое множество отношений между людьми. Более строго,  $A$  представляет собой множество пар  $(x_i, x_j)$ , для которых задана функция принадлежности  $\mu_A(x, y)$ , указывающая, в какой степени объект  $x$  напрямую влияет на объект  $y$ .

Следует заметить, что это понятие *влиятельности* не сводится только к числу связей с другими объектами. Должны учитываться степени влияния объектов, с которыми связан целевой объект (объект, для которого считается степень влияния). Таким образом, нужно так формализовать понятие влиятельности, чтобы учитывались как число связей, так и влиятельность объектов, с которыми связан целевой объект.

В статье [5] представлен один из алгоритмов решения данной задачи, который обладает вышеуказанным недостатком — бинарностью в отношениях. Ниже представлена модель, которая позволяет обойти это. Пусть  $\mu_A(x, y)$  — функция принадлежности, указывающая, в какой степени объект  $x$  напрямую влияет на объект  $y$ . Принимает значения из отрезка  $[0, 1]$ .

В [5] вводится понятие *итерированной силы объекта  $i$  порядка  $k$*  (обозначается  $p^i(k)$ ) через матрицу смежности графа [6]. Вместо матрицы смежности будем использовать функцию принадлежности. Получим

$$\begin{cases} p^i(k) = \sum_{j=1}^n \mu_A(x_i, x_j) p^j(k-1), i = 1, \dots, n, k \geq 1; \\ p^i(0) = 1, i = 1, \dots, n. \end{cases} \quad (1)$$

Заметим, что итерированная сила первого порядка объекта  $i$  есть сумма значений функции принадлежности данного объекта с другими. Она еще не учитывает влиятельность других объектов. Начиная со второго порядка, в данную сумму включается влиятельность иных объектов.

Возникает вопрос: до какого порядка считать вектор итерированных сил? Покажем, что из практических соображений достаточно считать до второго либо до третьего порядка. Предполагается, что итерированная сила объекта  $i$  порядка  $k$  выражает степень влияния объекта  $i$  с учетом того, что он свое влияние может распространить не более чем в радиусе  $k$ . Это вытекает из определения итерированной силы. Например, если вы просите своего друга попросить кого-то сделать что-то для вас, то

это влияние в радиусе 2, и выражается она итерированной силой второго порядка. Если учитываются цепи влияния с еще одним промежуточным участником, то нужно считать вектор итерированных сил 3-го порядка. Считать итерированные силы больших порядков кажется нецелесообразным ввиду ничтожно малой вероятности возникновения столь длинных цепочек в реальной жизни.

Таким образом, если анализируется влияние объектов в масштабе крупного города и выше, то нужно считать до 3-го порядка. Для меньших масштабов (в том числе если необходимо провести анализ в какой-либо группе по профессиональному интересу) целесообразно использовать 2-й порядок.

Следует заметить, что важно не числовое значение итерированной силы, а то, как силы для разных объектов соотносятся между собой. Поэтому после расчета вектора итерированной силы очередного порядка целесообразно нормировать данный вектор.

Остался открытым вопрос о вычислении функции принадлежности  $\mu_A(x, y)$ . Ее можно определить по-разному в зависимости от количества предоставляемой социальной сетью информации. Ключевым моментом при определении должно стать следующее правило: чем выше значение функции, тем большее прямое влияние оказывает объект  $x$  на объект  $y$ .

Социальная сеть VK (Vkontakte) предоставляет следующую информацию о взаимодействии объектов  $x$  и  $y$ :

- являются ли объекты  $x$  и  $y$  друзьями;
- число комментариев, которые пользователь  $y$  оставил на фотографиях пользователя  $x$ ;
- число комментариев, которые пользователь  $y$  оставил на видеозаписях пользователя  $x$ ;
- число комментариев, которые пользователь  $y$  оставил на заметках пользователя  $x$ ;
- число лайков, которые пользователь  $y$  оставил на фотографиях/видеозаписях/заметках пользователя  $x$ ;

и др.

Пусть  $f_i(x, y)$  характеризует степень влияния  $i$ -го фактора объекта  $x$  на объект  $y$ . Например, в случае  $i = 1$  это может быть число месяцев, прошедших с момента, когда объект  $x$  пометил объект  $y$  как друга. В случае  $i = 2$  это будет число фотографий, которые объект  $x$  пометил у объекта  $y$  как "мне нравится" и т. д. Обозначим число факторов через  $m$ . Тогда функцию принадлежности можно определить следующим образом:

$$\mu_A(x, y) = \sum_{i=1}^m c_i F_i(x, y),$$

где  $F_i(x, y) = 1 - e^{-f(x, y)}$ , а  $c_i$  — некоторые вещественные коэффициенты, которые удовлетворяют следующим ограничениям:

$$0 \leq \sum_i c_i \leq 1, \quad 0 \leq c_i \leq 1.$$

Следует заметить, что при фиксированных  $x$  и  $y$  функция принадлежности будет представлять собой линейную функцию относительно коэффициентов  $c_i$ . Принимая во внимание формулу (1), получим, что итерированная сила  $k$ -го порядка является членом  $k$ -го порядка относительно данных коэффициентов.

Пусть из  $n$  объектов социальной сети нам известны объекты, которые должны обладать значительным влиянием. Обозначим множество таких объектов через  $A$ . Необходимо таким образом подобрать коэффициенты  $c_i$ , чтобы максимизировать их суммарную влияние в сети, что сводится к следующей задаче:

$$\sum_{j \in A} p^j(c, 2) \rightarrow \max;$$

$$\begin{cases} 0 \leq \sum_i c_i \leq 1; \\ 0 \leq c_i \leq 1. \end{cases}$$

Это задача условной оптимизации, которая может быть решена с помощью метода внутренней штрафной функции [7]. Его суть заключается в преобразовании задачи с ограничениями в последовательность задач безусловной оптимизации некоторых вспомогательных функций. Последние получаются путем модификации целевой функции с помощью функций-ограничений таким образом, чтобы данные ограничения не фигурировали в задаче оптимизации в явном виде. В нашем случае штрафной функцией могла бы быть

$$\Phi(c, a) = a \left[ \ln \left( \sum_i c_i \right) + \ln \left( 1 - \sum_i c_i \right) + \sum_i \ln(c_i) + \sum_i \ln(1 - c_i) \right],$$

где  $a$  — положительный параметр, определяющий масштаб штрафа.

Алгоритм решения состоит в следующем. В качестве начальной точки  $c[0]$  выбирается произвольная точка допустимого множества (например,  $c[0]_i = 1/m$ ). Задается некоторая монотонно убывающая к нулю последовательность  $\{a_k\}$ ,  $k = 1, 2, \dots$ . Берется первый элемент этой последовательности и решается задача безусловной оптимизации (например, по методу наискорейшего спуска)

$$F(c, a) = \sum_{j \in A} p^j(c, 2) + \Phi(c, a) \rightarrow \max.$$

Решением этой задачи будет точка  $c[a_1]$ . Эта точка используется в качестве начальной для решения задачи поиска максимума  $F(c, a_2)$ , где  $a_2 \leq a_1$ . Последовательность полученных таким образом то-

чек  $c[a_k]$  сходится к оптимальному решению задачи. Вычисления прекращают при выполнении условий:

$$\left| \sum_{j \in A} p^j(c[k], 2) - \sum_{j \in A} p^j(c[k-1], 2) \right| \leq \varepsilon;$$
$$\|c[k] - c[k-1]\| \leq \beta,$$

где  $\varepsilon, \beta$  — числа, определяющие точность вычислений.

После нахождения индексов функция принадлежности задана полностью. На последнем шаге необходимо рассчитать вектор итерированных сил 2-го порядка согласно формулам (1), (2). Отсортированный по убыванию список итерированных сил будет представлять наиболее влиятельные объекты в вершине данного списка.

### Методика эксперимента

Для проведения эксперимента использовалась группа социальной сети "Vkontakte" известного Интернет-портала. Приложение выводит 100 объектов данной группы, обладающих наибольшим влиянием. В качестве априорной информации были выбраны пять человек, которые являются авторами известных статей.

Анализ результатов показал, что данные пять человек находятся в первой двадцатке. Приложение предоставило большое число людей, которые постоянно и регулярно публикуют новые статьи, в результате чего пользуются повышенным вниманием других пользователей.

### Выводы

В работе предложена математическая модель поиска наиболее влиятельных объектов социальной сети на основе априорной информации. Поставлена задача, разработан алгоритм ее решения, продемонстрировано применение данного алгоритма к известной социальной сети "ВКонтакте". Анализ результатов показывает, что расчет влиятельностей объектов в крупных социальных сетях может предоставить информацию, которая была бы полезной в различных сферах человеческой деятельности.

### Список литературы

1. **Frawley W. J., Piatetsky-Shapiro G., Matheus C. J.** Knowledge Discovery in Databases: An Overview // *AI Magazine*. 1992. Vol. 13, N. 3. P. 57–70.
2. **Интеллектуальный анализ данных** [сайт]. URL: [http://www.machinelearning.ru/wiki/index.php?title=Data\\_Mining](http://www.machinelearning.ru/wiki/index.php?title=Data_Mining) (дата обращения 22.07.2015).
3. **Давыдов А. А.** Системная социология. М.: ИС РАН, 2009. 192 с.
4. **Ибрагимов В. А.** Элементы нечеткой математики. Баку: АГНА, 2010. 394 с.
5. **Web и Social Mining**: [сайт]. URL: [http://www.basegroup.ru/library/web\\_mining/](http://www.basegroup.ru/library/web_mining/) (дата обращения 23.06.2015).
6. **Берж К.** Теория графов и ее применения. М.: Изд. иностр. лит., 1962. 318 с.
7. **Васильев Ф. П.** Методы оптимизации. М.: Факториал Пресс, 2002. 824 с.

**N. M. Novikova**, Professor, e-mail: [nov.nelly@gmail.com](mailto:nov.nelly@gmail.com),  
**A. V. Boriskin**, Graduate Student, e-mail: [boriskinpost@gmail.com](mailto:boriskinpost@gmail.com)  
Voronezh State University, Voronezh, Russia

## Mathematical Model of the Powerful Object Search Based on a Known Information

*The modern virtual social networks contain a lot of information. Most of this data is provided in a raw view. It's reasonable to process it for finding the information that can be considered as useful. For example, information about powerful objects can be used by many PR companies. The problem can be described this way: there is a subset of virtual social network objects. It's known that some of them can be considered as powerful objects in the group. The goal of the article is to use this information for searching other powerful objects. The algorithm of solving the problem is provided. The authors consider other approaches and compare the current one with them. The screenshot of the implemented software is shown.*

**Keywords:** virtual social networks, data mining, fuzzy graphs, optimization methods

### References

1. **Frawley W. J., Piatetsky-Shapiro G., Matheus C. J.** Knowledge Discovery in Databases: An Overview, *AI Magazine*, 1992, vol. 13, no. 3, pp. 57–70.
2. **Интеллектуальный анализ данных** [web-site]. (URL: [http://www.machinelearning.ru/wiki/index.php?title=Data\\_Mining](http://www.machinelearning.ru/wiki/index.php?title=Data_Mining) (access date 07/22/2015).
3. **Davydov A. A.** *Sistemnaja sociologija*. Moscow: IS RAN, 2009, 192 p.
4. **Ibragimov V. A.** *Jelementy nechetkoj matematiki*. Baku: AGNA, 2010. 394 p.
5. **Web i Social Mining**: [web-site]. URL: [http://www.basegroup.ru/library/web\\_mining/](http://www.basegroup.ru/library/web_mining/) (access date 06/23/2015).
6. **Berzh K.** *Teorija grafov i ee primenenija*. Moscow: Izd-vo inostr. lit., 1962, 318 p.
7. **Vasil'ev F. P.** *Metody optimizacii*. Moscow: Faktorial Press, 2002, 824 p.

УДК 338.27

**М. З. Ахмедов**, ст. препод., e-mail: m\_axmedov@mail.ru,  
Мингечевирский государственный университет, г. Мингечевир, республика Азербайджан

## Решение задачи прогнозирования с применением нечеткой импликации

*Изложен новый метод решения задачи прогнозирования на основе нечетких временных рядов с применением нечетких импликаций. Предложенный метод базируется на применении уравнения с нечетким отношением и излагается на примере решения тестовой задачи. На основе сравнительного анализа результатов, полученных с помощью других методов прогнозирования, установлено, что предложенный метод имеет меньшую усредненную погрешность прогнозирования (AFER).*

**Ключевые слова:** нечеткое множество, нечеткие временные ряды, нечеткое логическое отношение, нечеткая импликация

### Введение

Как известно, прогнозирование играет очень важную роль в повседневной человеческой деятельности и принятии решений в различных областях экономики. В качестве примеров можно привести прогноз погоды, предсказание землетрясений, планирование продаж товаров и т. п., которые оказывают существенное влияние на поведение людей в будущем. Для решения задачи прогнозирования довольно широко используют последовательности упорядоченных во времени данных, и эти данные — временной ряд (*time series*) — рассматривают для прогнозирования будущих значений характеристик на основании наблюдений и измерений, относящихся к настоящим и прошлым моментам времени. Все или почти все известные качественные и количественные методы прогнозирования ставят своей целью увеличение точности прогнозирования.

Традиционные методы нельзя применять к задачам, в которых исторические данные представлены не в привычном числовом виде, а в вербальной (словесной) форме. Нечеткие временные ряды (*fuzzy time series*) позволяют преодолеть эту проблему, открывая реальную перспективу возможности обработки как лингвистических, так и числовых данных. Таким образом, предложения модели нечетких временных рядов было связано с появлением нового направления — нечеткой математики, с помощью которого стало возможным решение задачи прогнозирования при нечеткой исходной информации или в нечеткой среде.

До настоящего времени для решения задачи прогнозирования на основе нечетких временных рядов были предложены различные методы (см., напри-

мер, [1—3, 6, 7, 10]). В основном эти методы были применены для решения тестовой задачи прогнозирования зачислений в университет Алабамы. В предложенных методах прогнозирования на основе нечетких временных рядов как универсальное множество были использованы или исторические данные, или разности исторических данных, без какой-либо предварительной обработки и/или получения дополнительной информации.

Данная статья посвящена методу решения задачи прогнозирования на основе нечетких временных рядов с применением нечеткой импликации. В предложенном методе прогнозирования была использована *time-invariant*-модель прогнозирования, и было установлено, что усредненная погрешность прогнозирования предложенного метода для тестовой задачи меньше по сравнению с известными методами.

### Предварительные сведения

Целесообразно дать краткую информацию о нечетких временных рядах. Предположим, что  $U$  — универсальное множество и было задано следующим образом:  $U = \{u_1, u_2, \dots, u_n\}$ , здесь  $u_i$  ( $i = 1, n$ ) интервалы одинаковой длины [2, 7].

**Определение 1.** Предположим  $Y(t)$  ( $\dots 0, 1, 2, 3, \dots$ ) является подмножеством множества  $R^1$ . Во множестве  $Y(t)$  определяется нечеткое множество  $\mu_t(t)$  ( $t = 1, 2, \dots$ ) и  $Y(t)$  является универсальным множеством для этого нечеткого множества. Определим множество  $F(t)$ , которое состоит из  $\{\mu_t(t) (t = 1, 2, \dots)\}$ , т. е.  $F(t)$  является множеством нечетких множеств. Тогда  $F(t)$  является нечетким временным рядом в универсальном множестве  $Y(t)$  ( $t = \dots 0, 1, 2, \dots$ ).

Понятно что, если мы принимаем  $F(t)$  как лингвистическую переменную, то  $\mu_A(t)$  будет приниматься как возможное лингвистическое значение  $F(t)$ . Кроме того, видно, что функция  $F(t)$  зависит от времени, и это значит что в различное время  $t$  функция  $F(t)$  может принимать различные значения.

**Определение 2.** Если существует такое отношение  $R(t, t - 1)$ , при котором  $F(t) = F(t - 1) \circ R(t, t - 1)$ , мы говорим, что значение  $F(t)$  обусловлено значением  $F(t - 1)$ . Здесь " $\circ$ " — оператор композиции  $\text{Maxmin}$ . Отношение между  $F(t)$  и  $F(t - 1)$  может быть представлено символически как  $F(t - 1) \rightarrow F(t)$ , а  $R(t, t - 1)$  называется моделью нечеткого временного ряда первого порядка.

**Определение 3.** Если для любого момента времени  $t$  временной ряд  $F(t)$  зависит только от  $F(t - 1)$ , т. е. такая связь может быть представлена в виде  $F(t) = F(t - 1) \circ R(t, t - 1)$ , где нечеткое отношение  $R$  инвариантно относительно времени  $t$ , то  $F(t)$  называется стационарным (*time-invariant*) нечетким временным рядом, в противном случае, он называется переменным по времени (*time-variant*) нечетким временным рядом.

Для решения проблемы прогнозирования с помощью нечетких временных рядов было предложено уравнение с неизвестным нечетким отношением [6, 7]:

$$A_i = A_{i-1} \circ R. \quad (1)$$

Здесь  $A_i$  — исторические данные  $i$ -го года, представленные в виде нечетких множеств (результаты прогнозирования);  $A_{i-1}$  — исторические данные  $(i - 1)$ -го года, представленные в виде нечетких множеств (фактические данные);  $\circ$  — символ  $\text{Maxmin}$ -го оператора композиции;  $R$  — нечеткая матрица отношений.

Q. Song и B. S. Chissom [5—7], предлагая метод для решения проблемы прогнозирования, сделали прогноз зачисления в университет Алабамы. Метод, который они предложили, вкратце можно описать следующим образом:

1-й шаг. Определить универсальное множество  $U$ .

2-й шаг. Разделить универсальное множество  $U$  на несколько интервалов, равной длины.

3-й шаг. Определить лингвистическую переменную и некоторые лингвистические термы как возможные значения этой переменной.

4-й шаг. Фаззифицировать исходные данные.

5-й шаг. На основе фаззифицированных исходных данных формировать нечеткие логические правила.

6-й шаг. Используя созданные нечеткие логические правила, вычислить соответствующую матрицу нечетких отношений и на основе этих матриц формировать итоговую матрицу нечетких отношений.

7-й шаг. Используя фаззифицированные исходные данные, прогнозировать зачисления на следующий год.

8-й шаг. Дефаззифицировать полученный результат.

Надо отметить, что для формирования матриц нечетких отношений на основе нечетких логических правил, было использовано правило условного логического вывода:  $P = \text{ЕСЛИ } x \text{ есть } A, \text{ ТО } y \text{ есть } B$ , и оператор Мамдани, а для фаззификации исходных данных — ступенчатая функция принадлежности [6].

В предложенном методе прогнозирования для формирования матриц нечетких отношений используются нечеткие импликации, удовлетворяющие указанным ниже принципам [4].

**Определение 4.** Нечеткой импликацией является функция, определенная как  $[0, 1]^2 \rightarrow [0, 1]$  и удовлетворяющая следующим принципам:

I.1. Если  $a_1 \leq a_3$ , тогда  $I(a_1, a_2) \geq I(a_3, a_2)$ , для любой  $a_1, a_2, a_3 \in [0, 1]$ .

I.2. Если  $a_2 \leq a_3$ , тогда  $I(a_1, a_2) \leq I(a_1, a_3)$ , для любой  $a_1, a_2, a_3 \in [0, 1]$ .

I.3.  $I(0, a_2) = 1$  для любой  $a_2 \in [0, 1]$  (falsity implies anything).

I.4.  $I(a_1, 1) = 1$  для любой  $a_1 \in [0, 1]$  (anything implies tautology).

I.5.  $I(1, 0) = 0$  (Booleanity).

Нечеткие импликации Zadeh, Lukasewich, Goguen, ALI-1, ALI-2, ALI-3 удовлетворяют вышеуказанным принципам. В предложенном методе прогнозирования были использованы указанные выше нечеткие импликации и следующие правила условного логического вывода для формирования матрицы нечетких отношений [9]:

$P = \text{ЕСЛИ } x \text{ есть } A, \text{ ТО } y \text{ есть } B \text{ Иначе } y \text{ есть } C.$

На основе этого правила условного логического вывода для формирования матриц нечетких отношений использованные нечеткие импликации и соответствующие формулы представлены в табл. 1 [9].

Надо отметить, что в экспериментальных исследованиях для фаззификации исходных данных были выбраны треугольная и колоколообразная функции принадлежности.

Отмечая то, что цель предложенного в статье метода прогнозирования — увеличить точность полученных результатов прогнозирования по сравнению с известными методами с использованием нечетких временных рядов, алгоритм предложенного метода можно описать следующим образом.

Шаги 1—5 выполняются аналогично.

6-й шаг. Используя созданные нечеткие логические правила, вычислить соответствующую матрицу нечетких отношений с учетом правила условного логического вывода, ЕСЛИ ... ТО ... ИНАЧЕ ... и с применением выбранных нечетких импликаций.

7-й шаг. Применяется модель (1) вычисления прогнозных значений для каждой матрицы нечетких отношений.

Нечеткие импликации для формирования матрицы нечетких отношений

Нечеткая импликация	Формула	Операция
Zadeh	$a \rightarrow b = \min \left\{ \begin{array}{l} \max[\min(a, b), 1 - a], \\ \max[\min(1 - a, 1 - b), a] \end{array} \right\}$	Конъюнкция
Lukasewich	$a \rightarrow b = \min\{1, \min[(1 - a + b), (1 + a - b)]\}$	Конъюнкция
Goguen	$a \rightarrow b = \begin{cases} \min(1, b/a), & \text{if } a = 1 \\ \min\left(1, \frac{1-b}{2-a}\right), & \text{others} \end{cases}$	$P \wedge Q = \begin{cases} 0, & \text{if } p = 0 \\ \max\left(0, \frac{p+q-1}{p}\right), & \text{if } p > 0 \end{cases}$
ALI-1	$a \rightarrow b = \begin{cases} 1 - b, & \text{if } a < b \\ 1, & \text{if } a = b \\ b, & \text{if } a > b \end{cases}$	$P \wedge Q = \begin{cases} p, & \text{if } p + q < 1 \\ 0, & \text{if } p + q = 1 \\ q, & \text{if } p + q > 1 \end{cases}$
ALI-2	$a \rightarrow b = \begin{cases} \min(a, 1 - b), & \text{if } a < b \\ 1, & \text{if } a = b \\ \min(1 - a, b), & \text{if } a > b \end{cases}$	$P \wedge Q = \begin{cases} 0, & \text{if } p + q \leq 1 \\ \max(p, q), & \text{if } p + q > 1 \end{cases}$
ALI-3	$a \rightarrow b = \begin{cases} \frac{1-b}{1-a+b}, & \text{if } a < b \\ 1, & \text{if } a = b \\ \frac{b}{[a+(1-b)]}, & \text{if } a > b \end{cases}$	$P \wedge Q = \begin{cases} (1-q)/(1-p+q), & \text{if } p < q \\ 1, & \text{if } p = q \\ q/(p+1-q), & \text{if } p > q \end{cases}$

Таблица 2

Сравнительный анализ средних квадратических отклонений

Мощность $N$	Функции принадлежности	Нечеткие импликации					
		Zadeh	Lukasewich	Goguen	ALI-1	ALI-2	ALI-3
5	Треугольная	1605,5	1467,9	***	1488,4	672,3	724,8
	Колоколообразная	1542,4	1338,3	***	1229,5	595,9	635,5
7	Треугольная	1580,2	1361,2	***	1333,8	672,6	505,8
	Колоколообразная	1456,7	1339,1	800,4	1371,1	720,4	534,5
9	Треугольная	1562,3	1329,7	***	1227,5	***	629,7
	Колоколообразная	1445,1	1326,4	904,6	1232,6	555,8	610,9
11	Треугольная	1537,6	1305,8	***	1129,1	638,3	600,7
	Колоколообразная	1434,6	1315,3	799,7	1199,5	628,3	700,6

8-й шаг. Используя результаты вычисления 7-го шага, с применением для выбранной нечеткой импликации соответствующей формулы конъюнкции вычислить конечные прогнозные значения.

9-й шаг. Дефазифицировать полученные результаты.

В результате проведенных экспериментальных исследований с различными значениями мощности нечетких множеств стало известно, что не все нечеткие импликации в универсальном множестве дают корректное решение. Значит, требуется проведение экспериментальных исследований выбора нечетких импликаций для формирования матриц нечетких отношений и выбора функций принадлежности для фаззификации исходных данных. Результаты

проведенных экспериментальных исследований для выбранных нечетких импликаций и сравнительный анализ средних квадратических отклонений результатов прогнозирования для тестовой задачи представлены в табл. 2. Из этой таблицы видно, что самое меньшее среднее квадратичное отклонение получается с применением нечеткой импликации ALI-3, при мощности  $N = 7$  и с выбором треугольной функции принадлежности.

### Описание метода прогнозирования

Формальное описание алгоритма реализации предложенного метода с применением нечеткой импликации ALI-3 для решения тестовой задачи

прогнозирования на базе нечетких временных рядов состоит из следующих шагов:

**Шаг 1** с использованием исторических данных зачислений в университет Алабамы определяется универсум [6,7]:  $U = [13000, 20000]$ .

**Шаг 2.** Это универсальное множество делится на семь интервалов одинаковой длины. Эти интервалы представлены ниже:

$$\begin{aligned} u_1 &= [13000, 14000], u_2 = [14000, 15000], \\ u_3 &= [15000, 16000], u_4 = [16000, 17000], \\ u_5 &= [17000, 18000], u_6 = [18000, 19000], \\ u_7 &= [19000, 20000]. \end{aligned}$$

**Шаг 3.** На универсуме  $U$  надо определить нечеткие множества, которые соответствуют определенным значениям лингвистической переменной [8] "Зачисления студентов". С учетом числа интервалов можно определить следующие лингвистические термы:

$$\begin{aligned} A_1 &= (\text{not many}), A_2 = (\text{not too many}), A_3 = (\text{many}), \\ A_4 &= (\text{many many}), A_5 = (\text{very many}), \\ A_6 &= (\text{too many}), A_7 = (\text{too many many}). \end{aligned}$$

**Шаг 4.** Для фаззификации исходных данных и лингвистических термов в предложенном методе используется треугольная функция принадлежности

$$\mu(x) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0 & c \leq x. \end{cases} \quad (2)$$

Здесь  $a, c$  — диапазон изменения переменной;  $b$  — наиболее возможное значение переменной;  $x$  — вектор, для координат которого необходимо рассчитать степени принадлежности.

После фаззификации полученные нечеткие множества представлены следующим образом:

$$\begin{aligned} A_1 &= \{u_1/1,0, u_2/0,85, u_3/0,69, u_4/0,54, \\ &u_5/0,38, u_6/0,23, u_7/0,08\}; \\ A_2 &= \{u_1/0,33, u_2/1,00, u_3/0,82, u_4/0,64, \\ &u_5/0,45, u_6/0,27, u_7/0,09\}; \\ A_3 &= \{u_1/0,20, u_2/0,60, u_3/1,00, u_4/0,78, \\ &u_5/0,56, u_6/0,33, u_7/0,11\}; \\ A_4 &= \{u_1/0,14, u_2/0,43, u_3/0,71, u_4/1,00, \\ &u_5/0,71, u_6/0,43, u_7/0,14\}; \\ A_5 &= \{u_1/0,11, u_2/0,33, u_3/0,56, u_4/0,78, \\ &u_5/1,00, u_6/0,60, u_7/0,20\}; \\ A_6 &= \{u_1/0,09, u_2/0,27, u_3/0,45, u_4/0,64, \\ &u_5/0,82, u_6/1,00, u_7/0,33\}; \\ A_7 &= \{u_1/0,08, u_2/0,23, u_3/0,38, u_4/0,54, \\ &u_5/0,69, u_6/0,85, u_7/1,00\}. \end{aligned} \quad (3)$$

Здесь  $u_i (i = \overline{1,7})$  — интервалы равной длины;  $A_i (i = \overline{1,7})$  — нечеткие множества.

**Шаг 5.** После фаззификации исходных данных для формирования матриц нечетких отношений сформированные нечеткие логические отношения представлены в табл. 3.

**Шаг 6.** Проводится вычисление матриц нечетких отношений. Например, для нечеткого логического отношения  $A_1 \rightarrow A_1$  сформированная на основе нечеткой импликации ALI-3 матрица нечетких отношений представлена в табл. 4.

**Шаг 7.** С применением Махmin-й композиционной операции проводится процесс прогнозирования для каждого нечетких отношений с использованием фаззифицированных исторических данных [1–3, 6, 7]. Для примера на основе фактического показателя 1971 г., результат Махmin-й операции для 1972 г. представлен в табл. 5. Указанная операция над другими историческими данными проводится аналогичным образом.

**Шаг 8.** На этом этапе проводится операция конъюнкции над полученными нечеткими множествами шага 7 с применением формулы соответствующей выбранной нечеткой импликации ALI-3 [9].

**Шаг 9.** Проводится дефаззификация над полученными результатами шага 8 для получения реальных прогнозных значений. Дефаззификация результатов прогнозирования проводится методом центра тяжести [6, 7]. Результаты прогнозирования с применением нечеткой импликации ALI-3 представлены в табл. 6.

Таблица 3

Нечеткие логические отношения

$A_1 \rightarrow A_1$	$A_1 \rightarrow A_2$	$A_2 \rightarrow A_3$	$A_3 \rightarrow A_3$	$A_3 \rightarrow A_4$
$A_4 \rightarrow A_4$	$A_4 \rightarrow A_3$	$A_4 \rightarrow A_6$	$A_6 \rightarrow A_6$	$A_6 \rightarrow A_7$

Таблица 4

Нечеткая матрица отношения R1

	1,00	0,85	0,69	0,54	0,38	0,23	0,08
1,00	1	0,73	0,53	0,37	0,23	0,13	0,04
0,85	0	1	0,60	0,41	0,26	0,14	0,04
0,69	0	0,13	1	0,47	0,29	0,16	0,05
0,54	0	0,12	0,27	1	0,33	0,18	0,05
0,38	0	0,11	0,24	0,40	1	0,20	0,06
0,23	0	0,10	0,21	0,35	0,53	1	0,07
0,08	0	0,09	0,19	0,32	0,47	0,67	1

Таблица 5

Результат Махmin-й операции для матрицы R1

0,99	0,73	0,53	0,37	0,24	0,13	0,04
0	0,04	0,04	0,04	0,04	0,04	0,04
0	0,02	0,02	0,02	0,02	0,02	0,02
0	0,02	0,02	0,02	0,02	0,02	0,02
0	0,01	0,01	0,01	0,01	0,01	0,01
0	0,01	0,01	0,01	0,01	0,01	0,01
0	0,01	0,01	0,01	0,01	0,01	0,01
<b>0,99</b>	<b>0,73</b>	<b>0,53</b>	<b>0,37</b>	<b>0,24</b>	<b>0,13</b>	<b>0,04</b>

Результаты прогнозирования с применением ALI-3

Годы	Факт	Нечеткие множества								Прогноз	Погрешность, %
1971	13055									13995	3,18
1972	13563	1,00	0,98	0,00	0,00	0,00	0,00	0,00	0,00	13997	0,94
1973	13867	1,00	0,99	0,00	0,00	0,00	0,00	0,00	0,00	14479	1,47
1974	14696	0,99	0,99	0,93	0,00	0,00	0,00	0,00	0,00	14987	3,06
1975	15460	0,98	1,00	0,99	0,95	0,00	0,00	0,00	0,00	15508	1,29
1976	15311	0,95	1,00	1,00	1,00	0,97	0,00	0,00	0,00	15490	0,73
1977	15603	0,96	1,00	1,00	0,99	0,94	0,00	0,00	0,00	15992	0,83
1978	15861	0,00	1,00	1,00	1,00	0,98	0,00	0,00	0,00	16486	1,91
1979	16807	0,00	0,99	1,00	1,00	0,99	0,96	0,00	0,00	16504	2,45
1980	16919	0,00	0,97	0,99	1,00	1,00	0,99	0,98	0,00	17019	3,85
1981	16388	0,00	0,95	0,99	1,00	1,00	1,00	0,99	0,00	16500	6,91
1982	15433	0,00	0,99	1,00	1,00	1,00	0,99	0,00	0,00	15502	0,03
1983	15497	0,95	1,00	1,00	0,99	0,96	0,00	0,00	0,00	15508	2,40
1984	15145	0,95	1,00	1,00	1,00	0,97	0,00	0,00	0,00	15015	0,97
1985	15163	0,95	1,00	1,00	0,99	0,00	0,00	0,00	0,00	15011	6,08
1986	15984	0,96	1,00	1,00	0,99	0,00	0,00	0,00	0,00	16490	2,19
1987	16859	0,00	0,99	1,00	1,00	0,99	0,97	0,00	0,00	17013	6,27
1988	18150	0,00	0,96	0,99	1,00	1,00	0,99	0,99	0,00	18503	2,46
1989	18970	0,00	0,00	0,00	0,00	0,99	1,00	1,00	0,00	19003	1,68
1990	19328	0,00	0,00	0,00	0,00	0,00	0,99	1,00	0,00	19005	1,72
1991	19337	0,00	0,00	0,00	0,00	0,00	0,97	0,99	0,00		
1992	18876										
Не прогнозировано											
Усредненная погрешность прогнозирования											
2,52											

Таблица 7

Сравнительный анализ прогнозных значений

Годы	Факт, значение	Song	Chen	Mamdani	Goguen	Zadeh	Lukasewich	ALI-1	ALI-2	ALI-3
1971	13055									
1972	13563	14000	14000	14000	13500	14500	13500	14000	14000	13995
1973	13867	14000	14000	14000	13500	15000	14000	14000	14500	13997
1974	14696	14000	14000	14000	13500	15000	14000	14000	14500	14479
1975	15460	15500	15500	15500	14500	16500	16000	15500	15000	14987
1976	15311	16000	16000	16000	17250	16500	16000	16500	16000	15508
1977	15603	16000	16000	16000	16000	16500	16000	16500	16000	15490
1978	15861	16000	16000	16000	16500	16500	16000	16500	16000	15992
1979	16807	16000	16000	16000	16500	16500	17000	16500	16000	16486
1980	16919	16813	16833	16833	17500	16500	17000	17500	17000	16504
1981	16388	16813	16833	16833	17500	16500	17000	17500	17000	17019
1982	15433	16789	16833	16833	17500	16500	17000	17500	17000	16500
1983	15497	16000	16000	16000	16500	16500	16000	16500	16000	15502
1984	15145	16000	16000	16000	17500	16500	16000	16500	16000	15508
1985	15163	16000	16000	16000	15500	16500	16000	16500	16000	15015
1986	15984	16000	16000	16000	15500	16500	16000	16500	16000	15011
1987	16859	16000	16000	16833	16500	16500	17000	17000	16000	16490
1988	18150	16813	16833	16833	17500	16500	17000	17500	17000	17013
1989	18970	19000	19000	19000	19500	19500	18500	19500	18500	18503
1990	19328	19000	19000	19000	19500	19500	19500	19500	18500	19003
1991	19337	19000	19000	19000	19500	19500	19500	18500	19000	19005
1992	18876	Not Forecasted								
<b>AFER, %</b>		<b>3,22</b>	<b>3,23</b>	<b>2,99</b>	<b>4,98</b>	<b>4,83</b>	<b>2,95</b>	<b>4,60</b>	<b>3,68</b>	<b>2,52</b>

На рисунке показан результат прогнозирования тестовой задачи с применением предложенного метода.

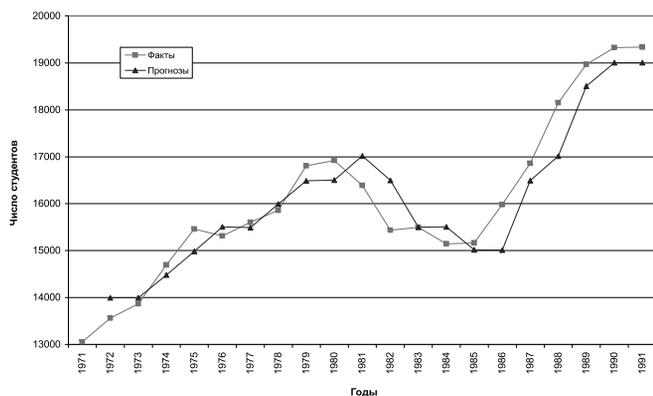
В табл. 7 приведены данные для сравнительного анализа результатов прогнозирования, полученных с помощью предложенного метода, с результатами, полученными с помощью других методов и с применением других нечетких импликаций.

Для сравнения была использована усредненная погрешность прогнозирования (*average forecasting error rate (AFER)*), определяемая следующим образом:

$$AFER = \frac{\sum_{i=1}^n |A_i - F_i| / A_i}{n} \cdot 100 \%, \quad (4)$$

где  $A_i$  — актуальные значения исторических данных;  $F_i$  — прогнозные значения;  $n$  — число наблюдений.

Зачисления студентов



### Результат прогнозирования

Таким образом, на основе анализа и сравнения полученных результатов с лучшими результатами предыдущих методов можно сказать, что предложенный метод дает меньшую усредненную погрешность. На основе алгоритма предложенного метода была разработана программа в среде программирования C++Builder 6.0.

### Заключение

В настоящей работе предложен метод прогнозирования на основе модели нечетких временных рядов с применением нечетких импликаций. В частности, описывается алгоритм предложенного

метода для прогнозирования тестовой задачи на примере использования правила условного логического вывода, ЕСЛИ...ТО...ИНАЧЕ... для формирования матрицы нечеткого отношения и с применением нечеткой импликации АЛИ-З. Сравнение полученных результатов прогнозирования с результатами других методов показывает, что предложенный метод имеет меньшую усредненную погрешность прогнозирования (AFER).

### Список литературы

1. **Chen S. M., Hsu C. C.** A new method to forecasting enrollments using fuzzy time series // *International Journal of Applied Science and Engineering*. 2004. N. 2, 3. P. 234—244.
2. **Chen S. M.** Forecasting enrollments based on fuzzy time series // *Fuzzy Sets and Systems*. 1996. Vol. 81. P. 311—319.
3. **Hsu C. C., Chen S. M.** A new method for forecasting enrollments based on fuzzy time series // *Proceedings of the Seventh Conference on Artificial Intelligence and Applications*, Taichung, Taiwan, Republic of China, 2004. P. 17—22.
4. **Rutkowski L., Spalka K.** Flexible Neuro-Fuzzy Systems // *IEEE Transactions on Neural Networks*. May 2003. Vol. 14, N. 3.
5. **Song Q., Chissom B. S.** Fuzzy time series and its models // *Fuzzy Sets and Systems*. 1993. Vol. 54. P. 269—277.
6. **Song Q., Chissom B. S.** Forecasting enrollments with fuzzy time series. Part I. // *Fuzzy Sets and Systems*. 1993. Vol. 54. P. 1—9.
7. **Song Q., Chissom B. S.** Forecasting enrollments with fuzzy time series. Part I. // *Fuzzy Sets and Systems*. 1994. Vol. 62. P. 1—8.
8. **Zadeh L. A.** The concept of a linguistic variable and its application to approximate reasoning. Part I—III // *Inf. Sci.* 1975. N. 8. P. 199—249; 1975. N. 8. P. 301—357; 1975. N. 9. P. 43—80.
9. **Алиев Р. А., Алиев Р. Р.** Теория интеллектуальных систем. Баку: Чашыоглы, 2001. 719 с.
10. **Ахмедов М. З.** Новый вариант решения проблемы прогнозирования с помощью нечетких временных рядов // *Ученые записки НАН Азербайджана*, № 3. Баку, 2001.

**M. Z. Ahmedov**, Senior Teacher, m\_axmedov@mail.ru,  
Mingechevir Government University, Mingechevir city, Azerbaijan republic

## Handling Forecasting Problem with Applying Fuzzy Implication

*In this paper, the new forecasting method based fuzzy time series, with applying fuzzy implication described, with which help the best result in the decision of forecasting problems received. The suggested method based on relational equations approach, have shown better results of fuzzy time series forecasting. For the proof the known task — forecasting university enrollments of Alabama used. In the paper the comparative analysis of the received results on a background of results received by the above named authors is given. In result, minimal known, that average forecasting error (AFER) in all bases of forecasting in comparison with other methods.*

**Keywords:** Fuzzy set, Fuzzy time series, Fuzzy logical relation, Fuzzy implication

### References

1. **Chen S. M., Hsu C. C.** A new method to forecasting enrollments using fuzzy time series, *International Journal of Applied Science and Engineering*, 2004, vol. 2, 3, pp. 234—244.
2. **Chen S. M.** Forecasting enrollments based on fuzzy time series, *Fuzzy Sets and Systems*, 1996, vol. 81, pp. 311—319.
3. **Hsu C. C., Chen S. M.** A new method for forecasting enrollments based on fuzzy time series, *Proceedings of the Seventh Conference on Artificial Intelligence and Applications*, Taichung, Taiwan, Republic of China, 2004, pp. 17—22.
4. **Rutkowski L., Spalka K.** Flexible Neuro-Fuzzy Systems. *IEEE Transactions on Neural Networks*, May 2003, vol. 14, no. 3.
5. **Song Q., Chissom B. S.** Fuzzy time series and its models, *Fuzzy Sets and Systems*, 1993, vol. 54, pp. 269—277.
6. **Song Q., Chissom B. S.** Forecasting enrollments with fuzzy time series Part I. *Fuzzy Sets and Systems*, 1993, vol. 54, pp. 1—9.
7. **Song Q., Chissom B. S.** Forecasting enrollments with fuzzy time series Part II. *Fuzzy Sets and Systems*, 1994, vol. 62, pp. 1—8.
8. **Zadeh L. A.** The concept of a linguistic variable and its application to approximate reasoning, part I—III, *Inf. Sci.*, 1975, no. 8, pp. 199—249; 1975, no. 8, pp. 301—357; 1975, no. 9, pp. 43—80.
9. **Aliiev R. A., Aliiev R. R.** *Teorija intellektual'nyh system* (Theory of the intellectual systems), Baku, Chashyogly, 2001. 719 p.
10. **Ahmedov M. Z.** Novyj variant reshenija problemy prognozirovaniya s pomoshh'ju nechetkih vremennyh rjadov (New variant of the decision of forecasting problems using fuzzy time series), *Uchenye zapiski*, no. 3, NAN Azerbaijan, Baku, 2001, pp. 96—103.

Ву Вьет Тханг, аспирант, thangvuviet84@gmail.com,  
Московский физико-технический институт (ГУ)

## Ускорение алгоритма кластеризации DBSCAN за счет использования алгоритма K-means

*Кластеризация это одна из самых важных задач интеллектуального анализа данных (DataMining). Хотя существует много исследованных способов кластеризации таких как K-means, Fuzzy C-means и др., но существует проблема повышения точности и ускорения алгоритмов кластеризации, вследствие того, что в течение 10 последних лет количество обрабатываемых данных существенно выросло. В данной работе представлен новый подход для ускорения алгоритма кластеризации на основе плотности DBSCAN (Density Based Spatial Clustering of Applications with Noise) [1]. Практические исследования показывают, что скорость кластеризации предложенного алгоритма выше при сохранении точности.*

**Ключевые слова:** кластеризация, DBSCAN, K-means

### Введение

Кластеризация это процесс разбиения множества с  $N$  элементами  $x_1, x_2, \dots, x_n$  ( $x_i$  имеет размерность  $m$ ) на  $K$  кластеров, так, чтобы в каждом кластере все элементы были схожи в каком-то смысле. Элементы  $x_i$  могут быть числовыми, категориальными или смешанными данными. Одним из важных направлений DataMining являются методы кластерного анализа. Было предложено несколько методов, таких как K-means (1956), метод объединения (1960), метод графов (1973), метод нечетких C-Means (1981), спектральный метод кластеризации (1990), плотностной метод DBSCAN (1996) и др.

В данной работе сделан акцент на ускорении алгоритма DBSCAN. Предложен новый алгоритм FastDBSCAN (Fast Density based Clustering), основанный на алгоритме кластеризация K-means и выборе примеров таких, при которых пропорция плотности между кластерами не изменялась.

### 1. Метод кластерного анализа DBSCAN

Алгоритм DBSCAN — плотностной алгоритм для кластеризации пространственных данных с присутствием шума был предложен М. Эстер, Г.-П. Кригел и их коллегами в 1996 г. как решение проблемы разбиения (изначально пространственных) данных на кластеры произвольной формы [1, 2]. Большинство алгоритмов, проводящих плоское разбиение, создают кластеры, по форме близкие к сферическим, так как минимизируют расстояние точки до центра кластера.

Авторы DBSCAN экспериментально показали, что их алгоритм способен распознавать кластеры различной формы, например, как на рис. 1.

Идея, положенная в основу алгоритма, заключается в том, что внутри каждого кластера плотность точек (объектов) заметно выше, чем плотность снаружи кластера, а также плотность в областях с шумом ниже плотности любого из кластеров. Еще

точнее, для каждой точки кластера ее окрестность в диапазоне заданного радиуса должна содержать не менее некоторого числа точек, которое задается пороговым значением. Перед изложением алгоритма дадим необходимые определения.

**Определение 1.** *Eps-соседство* точки  $p$ , обозначаемое как  $N_{Eps}(p)$ , определяется как множество точек, находящихся от точки  $p$  на расстоянии не более  $Eps$ :  $N_{Eps}(p) = \{q \in D \mid dist(p, q) \leq Eps\}$ , где  $D$  — множество заданных точек;  $dist(p, q)$  — Евклидово расстояние между двумя точками  $p$  и  $q$ . Поиска точек, чье  $N_{Eps}(p)$  содержит хотя бы минимальное число точек ( $MinPt$ ) недостаточно, так как точки бывают двух видов: ядровые и граничные.

**Определение 2.** Точка  $p$  непосредственно плотно достижима из точки  $q$  (при заданных  $Eps$  и  $MinPt$ ), если  $p \in N_{Eps}(q) \mid N_{Eps}(q) \geq MinPt$  (рис. 2).

**Определение 3.** Точка  $p$  плотно достижима из точки  $q$  (при заданных  $Eps$  и  $MinPt$ ), если существует последовательность точек  $q = p_1, p_2, \dots, p_n$ , где  $p_{i+1}$  непосредственно плотно достижима из  $p_i$ . Это отношение транзитивно, но не симметрично в общем случае, однако симметрично для двух ядровых точек.



Рис. 1. Примеры кластеров произвольной формы, распознанных DBSCAN



Рис. 2. Пример точек, находящихся в отношении непосредственно плотной достижимости

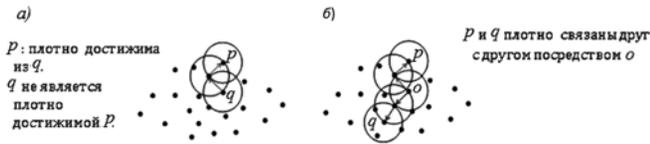


Рис. 3. Пример точек, находящихся в отношении плотной связанности

**Определение 4.** Точка  $p$  плотно связана с точкой  $q$  (при заданных  $Eps$  и  $MinPt$ ), если существует точка  $o$ :  $p$  и  $q$  плотно достижимы из  $o$  (при заданных  $Eps$  и  $MinPt$ ) (рис. 3).

Теперь можно дать определения кластеру и шуму.

**Определение 5.** Кластер  $C_j$  (при заданных  $Eps$  и  $MinPt$ ) — это непустое подмножество точек, удовлетворяющее следующим условиям:

- а)  $\forall p, q$ : если  $p \in C_j$  и  $q$  плотно достижима из  $p$  (при заданных  $Eps$  и  $MinPt$ ), то  $q \in C_j$ ;
- б)  $\forall p, q \in C_j$ :  $p$  плотно связана с  $q$  (при заданных  $Eps$  и  $MinPt$ ).

Итак, кластер — это множество плотно связанных точек. В каждом кластере содержится хотя бы  $MinPt$  точек.

Шум — это подмножество точек, которые не принадлежат ни одному кластеру:  $\{p \in D | \forall j: p \notin C_j, j = \overline{1, K}\}$ .

Алгоритм DBSCAN для заданных значений параметров  $Eps$  и  $MinPt$  кластеризует множество точек следующим образом: сначала выбирает случайную точку, являющуюся ядровой, помещает в кластер саму эту точку и все точки, плотно достижимые из нее.

Ниже приведен алгоритм DBSCAN в общем виде [4].

**Вход:** множество точек  $D$ ,  $Eps$ ,  $MinPt$ .

- Шаг 1. Установить всем элементам множества  $D$  флаг «не посещен». Присвоить текущему кластеру  $C_j$  нулевой номер,  $j := 0$ . Множество шумовых точек  $Noise := \emptyset$ .
- Шаг 2. Для каждого  $d_i \in D$  такого, что флаг  $(d_i)$  = «не посещен», выполнить:
- Шаг 3. флаг  $(d_i)$  := «посещен»;
- Шаг 4.  $N_i := N_{Eps}(d_i) = \{q \in D | dist(d_i, q) \leq Eps\}$
- Шаг 5. Если  $|N_i| \geq MinPt$ , то  $Noise := Noise + \{d_i\}$  иначе номер следующего кластера  $j := j + 1$ ;
- EXPANDCLUSTER( $d_i, N_i, C_j, Eps, MinPt$ );

**Выход:** множество кластеров  $C = \{C_j\}$ .

EXPANDCLUSTER:

**Вход:** текущая точка  $d_i$ , его eps-соседство  $N_i$ , текущий кластер  $C_j$  и  $Eps, MinPt$ .

- Шаг 1.  $C_j := C_j + \{d_i\}$ ;
  - Шаг 2. Для всех документов  $d_k \in N_i$ :
  - Шаг 3. Если флаг  $(d_k)$  = «не посещен», то
  - Шаг 4. флаг  $(d_k)$  := «посещен»;
  - Шаг 5.  $N_{ik} := N_{Eps}(d_k)$ ;
  - Шаг 6. Если  $N_{ik} \geq MinPt$ , то  $N_i := N_i + N_{ik}$ ;
  - Шаг 7. Если не  $\exists r: d_k \in C_r, r = \overline{1, |C|}$ , то  $C_j := C_j + \{d_k\}$ ;
- Выход.** Кластер  $C_j$ .

В общем случае алгоритм DBSCAN имеет вследствие поиска  $Eps$ -соседства квадратичную вычислительную сложность, равную  $O(N^2)$ . Однако авторы алгоритма использовали для этой цели специальную структуру данных —  $R^*$ -деревья, в результате

вычислительная сложность поиска  $Eps$ -соседства для одной точки равна  $O(\log n)$ . Общая вычислительная сложность DBSCAN —  $O(n \log n)$ .

Используя алгоритм DBSCAN, можно разделить данные на кластеры и шум. Этот алгоритм можно применить во многих классах задач, таких, например, как распознавание лиц или обнаружение атак в IDS (Intrusion Detection System), когда шум можно считать аномалией.

## 2. Ускорение алгоритма DBSCAN за счет использования алгоритма K-means

Одно из достоинств алгоритма DBSCAN — он хорошо работает со множествами данных произвольной формы, примеры даны на рис. 4.

Видно, что для первого набора данных оба алгоритма, K-means и DBSCAN, хорошо проводят кластеризацию, так как множество данных имеет круглую форму. Но для второго набора данных K-means работает хуже, чем DBSCAN, так как у второго набора форма намного сложнее, чем у первого.

Алгоритм K-means имеет низкую вычислительную сложность  $O(K \cdot N)$  — это его основное достоинство и он хорошо работает с большим количеством данных [3], а DBSCAN довольно медленно работает с большим количеством данных. Поэтому для ускорения DBSCAN будем использовать алгоритм K-means. K-means применяется для разбиения множества данных  $D$  на  $K$  кластеров ( $K$  — достаточно большое, чтобы покрыть все множество данных). После этого выбирается случайно  $t$  % данных из каждого кластера и получается новое множество  $E$ . Так делается для того, чтобы относительная плотность между регионами множества  $D$  не изменилась.

Пусть задано множество  $D$  с 2000 точек (рис. 5), к нему применяется алгоритм K-means. После этого выбирается 40 % от каждого кластера и получается новое множество  $E$  с 813 точками (рис. 6).

Из рис. 6 видно, что относительная плотность регионов множеств  $E$  и  $D$  почти не отличается.

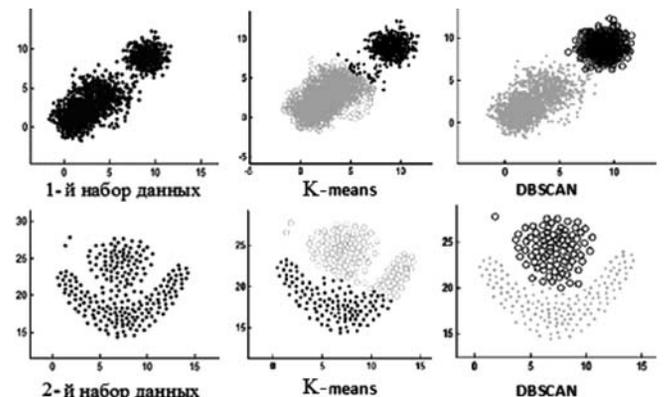


Рис. 4. Примеры кластеризации данных алгоритмом DBSCAN и K-means

После применения алгоритма K-means для поиска промежуточных кластеров, используется алгоритм DBSCAN на множестве  $E$ . Получаем ускоренный алгоритм FastDBSCAN.

Алгоритм FastDBSCAN

**Вход.** Множество  $D$ , число промежуточных кластеров для K-means  $K$ , доля  $t$ .

**Выход.** Кластеры и аномалии.

Шаг 1. Инициализируем  $K$  центров случайно.

Шаг 2. Реализуем алгоритм K-means с  $K$  центрами.

Шаг 3. Взяв  $100 \cdot t$  процентов каждого полученного кластера, получаем новое множество  $E$ .

Шаг 4. Выполняем алгоритм DBSCAN с множеством  $E$ .

Шаг 5. Отображаем обратно результаты, чтобы получить кластеры и аномалии для множества  $D$ .

Вычислительная сложность FastDBSCAN определяется вычислительной сложностью алгоритмов K-means и DBSCAN. В общем случае вычислительная сложность K-means равна  $O(K \cdot N)$ , где  $K$  — число кластеров;  $N$  — количество данных. Сложность DBSCAN равна  $O((t \cdot N)^2)$ , где  $t$  — заданная доля. Общая вычислительная сложность FastDBSCAN равна  $O(K \cdot N + (t \cdot N)^2)$ .

### 3. Экспериментальная оценка результатов кластеризации с DBSCAN и FastDBSCAN

Эксперименты проводили на ПЭВМ со следующими характеристиками:

- процессор: Intel® Core™ i5-4460S CPU® 2.90GHz × 64;
- оперативная память: 4.00GB;
- тип системы: 64-разрядная операционная система Windows 8.

Выполним алгоритмы DBSCAN и FastDBSCAN с четырьмя популярными множествами D31, t1.2k, t5.8k, t8.8k произвольной формы, с разным количеством данных [7], показанных на рис. 7 и в табл. 1.

В табл. 2—5 показаны параметры и результаты выполнения этих алгоритмов.

Результаты экспериментов показывают, что для алгоритма FastDBSCAN оптимальные параметры ( $Esp$ ,  $Minpts$ ) отличаются от оптимальных параметров алгоритма DBSCAN. Это связано с тем, что алгоритм FastDBSCAN работает с редуцированным множеством данных  $E$ , в котором, в зависимости от параметра  $t$  %, может меняться относительная плотность кластеров. Влияние параметра  $t$  % на точность кластеризации будет исследовано в дальнейшем.

Из приведенных в табл. 6 результатов следует, что средняя точность кластеризации методом FastDBSCAN снизилась незначительно (например D31 на 0,42 %, t5.8k на 0,68 %, t8.8k на 0,52 %). Максимальная точность может даже увеличиться (например, D31 на 0,22 %).

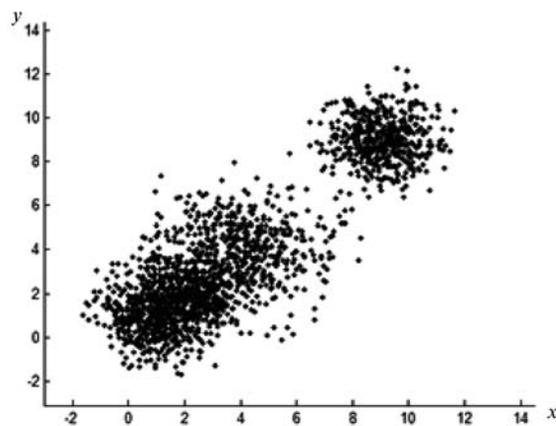


Рис. 5. Оригинальное множество точек  $D$

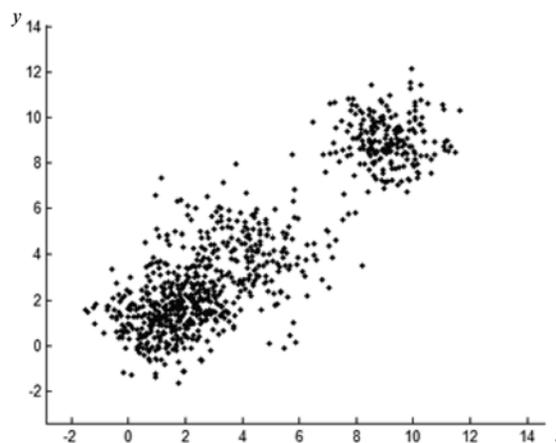


Рис. 6. Новое множество  $E$ , состоящее из  $t = 40$  % данных из каждого кластера

Таблица 1

Название набора данных	Размерность набора данных	Число записей
D31	2	3100
t1.2k	2	2000
t5.8k	2	8000
t8.8k	2	8000

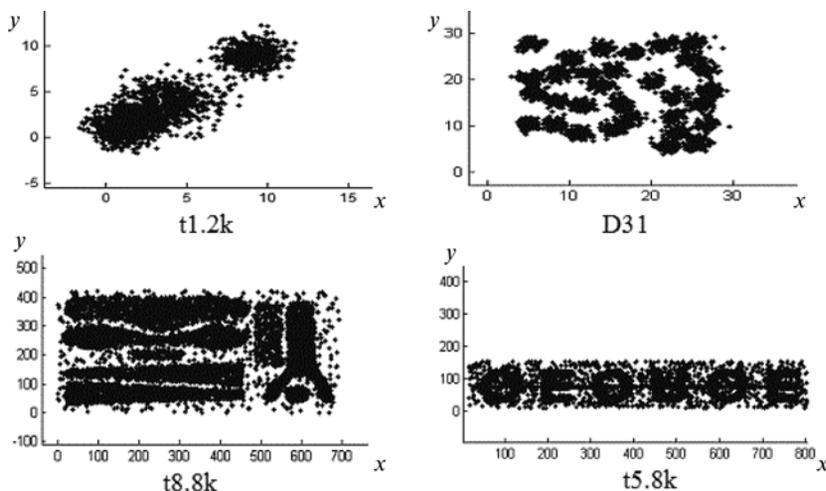


Рис. 7. Тестовые множества данных

Таблица 2

D31							
DBSCAN				FastDBSCAN			
Esp	Minpts	Время, с	Точность, %	Esp	Minpts	Время, с	Точность, %
2,00	38,00	10,17	72,34	0,90	24,00	5,49	78,79
0,60	34,00	11,73	79,58	2,10	24,00	1,83	81,49
1,80	33,00	9,78	81,50	2,10	23,00	1,88	82,35
1,80	35,00	9,98	84,61	2,10	26,00	1,89	83,40
1,80	37,00	9,91	86,45	2,10	21,00	1,90	85,59
1,60	38,00	10,45	87,88	0,90	23,00	6,32	86,95
1,40	32,00	12,47	88,72	1,90	21,00	1,96	87,17
1,40	35,00	10,94	89,09	1,70	22,00	1,89	88,99
1,40	37,00	10,86	91,74	1,90	26,00	1,87	89,73
1,40	36,00	10,95	92,35	1,90	27,00	1,89	91,53
1,20	32,00	11,36	94,18	1,70	26,00	1,88	92,36
1,20	35,00	11,77	96,00	1,90	24,00	1,86	93,13
1,20	37,00	12,12	96,62	0,90	22,00	5,31	94,28
1,20	38,00	11,97	97,26	1,10	27,00	6,10	95,07
0,80	37,00	18,70	97,41	1,50	21,00	1,86	96,82
0,80	36,00	18,48	97,62	1,30	27,00	3,11	97,94
0,80	35,00	18,37	97,73	1,10	23,00	4,21	98,05
1,00	34,00	13,65	98,64	1,50	25,00	1,88	98,16
0,80	32,00	17,89	98,64	1,30	26,00	1,91	98,22
1,00	35,00	13,83	98,64	1,30	23,00	1,84	98,62
1,00	37,00	14,17	99,65	1,10	22,00	2,76	98,76
1,00	38,00	14,61	99,67	1,50	27,00	1,87	99,89

Таблица 3

t1							
DBSCAN				FastDBSCAN			
Esp	Minpts	Время, с	Точность, %	Esp	Minpts	Время, с	Точность, %
0,20	9,00	8,18	42,55	0,60	8,00	3,80	46,21
0,20	10,00	7,84	43,75	0,60	9,00	3,58	47,03
0,20	7,00	8,34	44,56	0,80	13,00	3,74	50,04
0,20	8,00	8,33	44,67	0,80	12,00	3,99	50,30
0,20	11,00	7,57	45,23	0,60	7,00	2,79	51,29
0,20	6,00	8,24	47,46	0,00	12,00	0,31	62,48
0,20	5,00	7,96	49,78	0,80	7,00	3,38	64,46
0,00	5,00	4,50	62,48	0,80	11,00	3,87	68,60
0,40	10,00	6,67	83,18	0,80	9,00	4,08	70,82
0,40	11,00	7,46	83,70	0,80	8,00	4,32	74,52
0,40	9,00	6,78	86,64	0,80	10,00	3,94	74,83
0,40	8,00	6,66	91,59	1,20	10,00	0,97	75,88
0,40	7,00	5,82	93,53	1,20	12,00	2,24	75,89
0,40	5,00	5,53	93,81	1,00	11,00	2,30	75,94
0,40	6,00	5,71	95,80	1,00	8,00	2,02	79,04
0,60	9,00	5,65	97,03	1,00	10,00	2,07	81,90
0,60	7,00	5,44	97,05	1,20	11,00	1,22	83,13
0,60	6,00	5,30	97,42	1,20	7,00	1,14	85,61
0,60	5,00	5,22	97,49	1,00	13,00	4,09	93,34
0,60	8,00	5,37	97,61	1,20	13,00	2,68	96,46
0,80	10,00	5,27	98,79	1,00	12,00	3,39	97,24
0,80	9,00	5,14	98,86	1,60	13,00	1,53	97,73
0,80	5,00	4,99	99,40	1,40	13,00	1,42	98,12
0,60	10,00	5,87	99,60	2,00	12,00	0,35	98,22
0,80	11,00	5,42	99,70	1,40	11,00	1,65	98,71
1,20	9,00	4,36	99,90	1,40	8,00	0,63	98,81
1,00	10,00	4,36	100,00	1,40	9,00	0,87	98,91
				1,00	7,00	1,95	99,01
				2,00	7,00	0,16	99,30
				1,60	8,00	0,53	99,40
				1,20	9,00	1,92	99,50
				1,80	13,00	0,45	99,60
				2,00	8,00	0,27	99,70
				1,80	8,00	0,17	99,90
				1,80	11,00	0,27	100,00

Таблица 4

t5.8k							
DBSCAN				FastDBSCAN			
Esp	Minpts	Время, с	Точность, %	Esp	Minpts	Время, с	Точность, %
11,20	10,00	74,71	60,32	16,40	6,00	12,20	67,35
12,40	10,00	69,00	65,98	16,40	7,00	12,95	71,33
13,00	12,00	74,79	66,15	15,40	7,00	10,87	72,80
11,00	10,00	76,68	76,09	6,40	13,00	21,56	75,24
11,00	11,00	76,72	81,33	7,40	14,00	18,91	75,50
12,00	10,00	73,80	83,32	7,40	11,00	14,58	76,31
11,80	11,00	73,10	84,33	7,40	10,00	13,97	77,06
12,00	11,00	75,06	84,33	7,40	9,00	15,27	77,85
11,60	10,00	76,52	84,35	15,40	6,00	11,10	78,99
12,00	12,00	75,90	85,18	9,40	13,00	17,86	79,83
11,40	11,00	79,44	85,21	8,40	9,00	15,57	80,79
12,40	12,00	69,05	85,22	7,40	8,00	19,72	81,85
12,80	13,00	73,25	85,24	9,40	14,00	15,96	82,10
13,00	13,00	77,72	86,24	11,40	7,00	14,11	83,75
11,40	13,00	75,27	86,37	7,40	7,00	17,04	84,54
11,20	11,00	73,85	86,48	9,40	11,00	17,31	85,88
12,00	13,00	74,56	89,07	10,40	14,00	14,52	86,25
12,20	13,00	73,26	89,11	11,40	14,00	10,70	87,95
11,00	12,00	75,16	89,24	9,40	10,00	11,91	88,57
11,80	13,00	79,53	89,26	11,40	8,00	14,44	88,72
11,40	12,00	75,80	89,36	11,40	13,00	12,50	88,81
11,20	12,00	75,32	89,38	11,40	10,00	11,89	88,89
11,60	13,00	78,01	90,68	14,40	12,00	11,31	88,98
				12,40	6,00	12,53	89,07
				15,40	9,00	14,07	89,20
				15,40	12,00	14,70	89,34
				11,40	11,00	12,47	89,58

Таблица 5

t8.8k							
DBSCAN				FastDBSCAN			
Esp	Minpts	Время, с	Точность, %	Esp	Minpts	Время, с	Точность, %
16,50	1,00	56,48	73,00	15,80	1,00	10,05	72,40
16,00	1,00	56,19	75,08	15,20	1,00	11,28	75,90
16,00	3,00	59,29	79,37	14,20	9,00	20,48	76,49
16,50	4,00	58,32	80,40	14,00	8,00	10,36	77,54
16,50	7,00	59,19	81,41	14,80	9,00	13,72	78,57
16,50	3,00	58,16	82,42	15,20	9,00	10,43	79,75
17,00	5,00	59,98	83,44	14,40	7,00	13,46	80,94
15,50	1,00	56,33	84,10	15,80	8,00	10,20	81,97
15,50	5,00	59,64	84,21	15,80	3,00	10,34	82,92
14,50	1,00	56,56	84,50	15,00	2,00	11,32	83,29
14,50	2,00	57,61	84,74	16,00	7,00	10,21	83,96
14,50	3,00	58,72	85,00	14,00	7,00	10,58	84,95
15,50	7,00	59,96	85,96	14,80	6,00	10,57	85,74
15,00	3,00	58,38	86,06	15,20	5,00	11,03	86,95
15,00	4,00	59,41	86,07	14,60	4,00	13,65	87,28
14,00	1,00	56,63	86,44	15,40	6,00	10,72	87,82
13,50	1,00	64,51	86,48	14,60	2,00	10,70	87,93
14,00	2,00	58,21	86,58	16,00	5,00	11,36	87,96
13,50	2,00	61,71	86,88	14,40	3,00	14,65	88,13
14,50	4,00	59,48	87,87	14,40	2,00	14,35	88,70
15,00	5,00	59,52	88,91	15,00	3,00	11,94	88,99
15,00	7,00	60,36	89,51	14,20	3,00	10,26	89,04
14,50	6,00	60,00	89,79	15,40	2,00	10,20	89,36

Таблица 6

Название набора данных	Средняя точность DBSCAN, %	Средняя точность Fast-DBSCAN, %	Максимальная точность DBSCAN, %	Максимальная точность Fast-DBSCAN, %
D31	92,11	91,69	99,67	99,89
t1.2k	81,17	82,63	100	100
t5.8k	83,14	82,46	90,68	89,58
t8.8k	84,28	83,76	89,99	89,36

В целом, точность кластеризации методом Fast-DBSCAN незначительно отличается от точности, достигаемой DBSCAN (рис. 8).

Сравнительный анализ трех алгоритмов DBSCAN, FastDBSCAN и K-means выполнен экспериментальной работой и результат показан в табл. 7. Из табл. 7 следует, что с набором данных, у которых сложные формы, DBSCAN и FastDBSCAN лучше работают, чем алгоритм K-means.

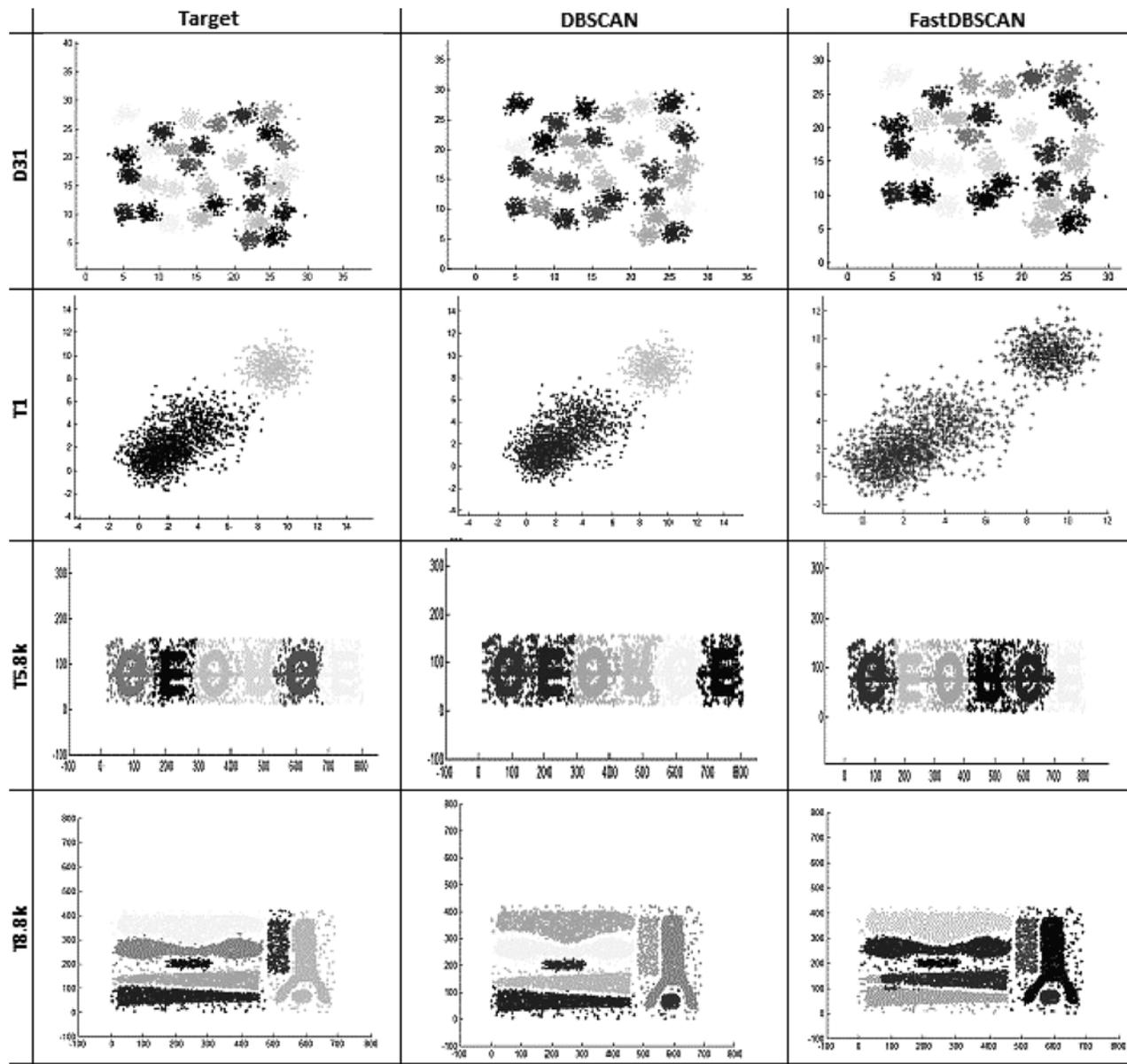


Рис. 8. Результаты кластеризации алгоритмов DBSCAN и FastDBSCAN

Таблица 7

Название набора данных	DBSCAN			FastDBSCAN			K-means	
	<i>Esp</i>	<i>Minpts</i>	Точность, %	<i>Esp</i>	<i>Minpts</i>	Точность, %	Число кластеров	Точность, %
D31	1,00	38,00	99,67	1,50	27,00	99,89	31	95,14
t1.2k	1,00	10,00	100	1,80	11,00	100	2	98,85
t5.8k	11,60	13,00	90,68	11,40	11,00	89,58	6	78,25
t8.8k	14,50	6,00	89,99	15,40	2,00	89,36	8	75,36

Для измерения точности мы используем Ранд-статистики (*Randstatistic*) [9, 10], с помощью которых измеряют аналогичность двух наборов кластеров  $X$  и  $Y$ .

Заданы два множества  $X = \{X_1, X_2, \dots, X_n\}$  и  $Y = \{Y_1, Y_2, \dots, Y_n\}$  для сравнения аналогичности, дадим определение следующих величин:

- $a$  — это число пар объектов, отнесенных к одному и тому же кластеру в обоих множествах  $X$  и  $Y$ ;
- $b$  — это число пар объектов, отнесенных к различным кластерам в  $X$  и в  $Y$ ;
- $c$  — это число пар объектов, отнесенных к одному и тому же кластеру в  $X$  и к различным кластерам в  $Y$ ;
- $d$  — это число пар объектов, отнесенных к различным кластерам в  $X$  и к одному и тому же кластеру в  $Y$ .

Ранд-статистики обозначены  $R$ ,

$$R = \frac{a+b}{a+b+c+d} = \frac{a+b}{\frac{n(n-1)}{2}} = \frac{2(a+b)}{n(n-1)},$$

где  $a + b + c + d$  — это число возможных разных пар  $n$  точек.

Интуитивно,  $a + b$  можно рассматривать как число соглашений между множествами  $X$  и  $Y$ ;  $c + d$  — как число разногласий между  $X$  и  $Y$ .

Значение Ранд-статистики находится в диапазоне  $[0, 1]$ , где 0 означает, что два кластера не ана-

логичны при любых парах точек и 1 означает, что два кластера аналогичны при любых парах точек.

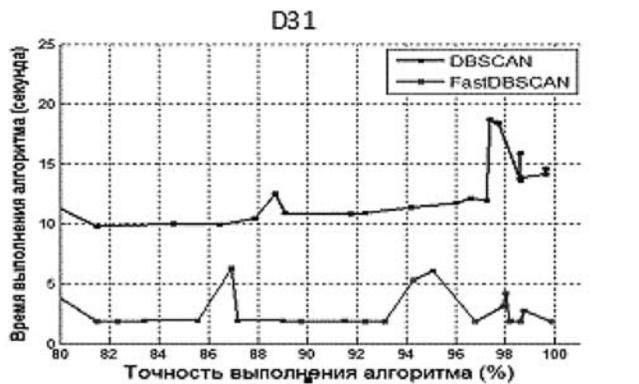
### Заключение

Представлен новый способ ускорения алгоритма кластеризации DBSCAN, основанный на применении алгоритма K-means, названный FastDBSCAN. Такой метод позволяет существенно, до 3–5 раз, ускорить кластеризацию данных при сохранении точности, достигаемой оригинальным DBSCAN (рис. 9, 10). Полученные результаты подтверждены интенсивными экспериментальными исследованиями на ряде тестовых множеств.

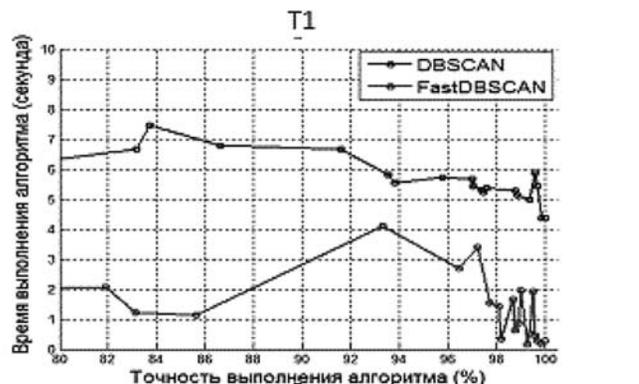
Предложенный способ может быть использован для решения многих классов задач кластеризации, требующих сокращения времени решения, например, таких задач, как распознавание лиц, обнаружение компьютерных атак [5, 6], обработка изображений, документов и др.

Дальнейшее совершенствование метода FastDBSCAN необходимо вести в следующих направлениях:

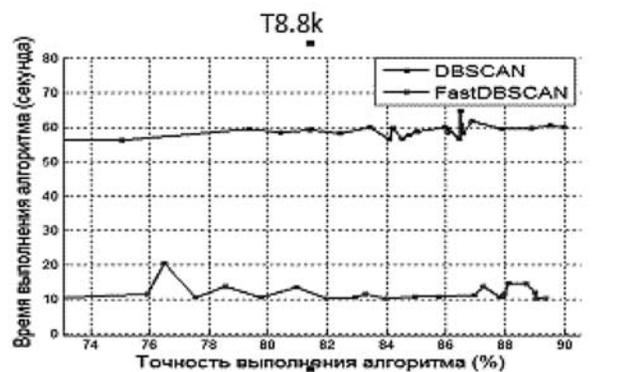
- 1) кластеризация данных, имеющих различную плотность;
- 2) распараллеливание метода на современные параллельные аппаратные средства, такие как суперЭВМ с графическими процессорами и др.;



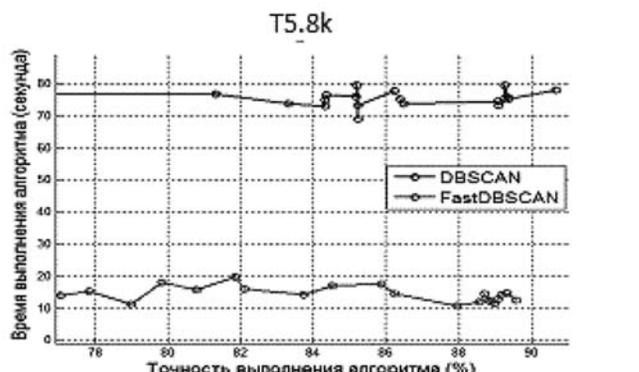
-DBSCAN: Esp = 1, MinPts = 38, время = 14,61 с, точность = 99,67 %  
 -FastDBSCAN: Esp = 1,5, MinPts = 27, время = 1,87 с, точность = 99,89 %



-DBSCAN: Esp = 1, MinPts = 10, время = 4,36 с, точность = 100 %  
 -FastDBSCAN: Esp = 1,8, MinPts = 11, время = 0,27 с, точность = 100 %



-DBSCAN: Esp = 14,4, MinPts = 6, время = 60 с, точность = 89,79 %  
 -FastDBSCAN: Esp = 15,4, MinPts = 2, время = 10,2 с, точность = 89,36 %



-DBSCAN: Esp = 11,6, MinPts = 13, время = 78,01 с, точность = 90,68 %  
 -FastDBSCAN: Esp = 11,4, MinPts = 11, время = 12,47 с, точность = 89,58 %

Рис. 9. Сравнение времени выполнения алгоритмов DBSCAN и FastDBSCAN (в подписях приведены значения параметров для лучших по точности результатов)

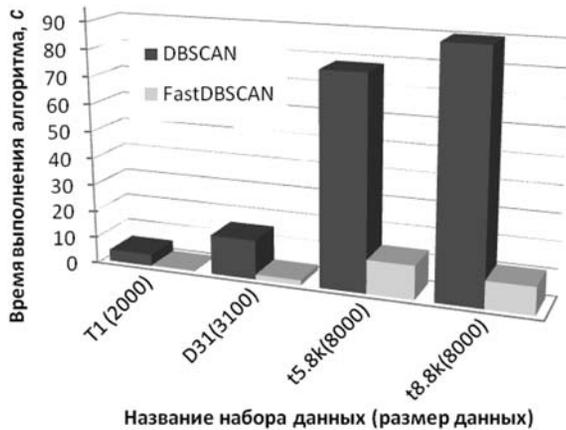


Рис. 10. Зависимость времени выполнения алгоритма от размера набора данных (для лучших по точности результатов)

3) разработка подхода к ускорению гибридного метода обучения с учителем и самообучения (Semi-supervised FastDBSCAN);

4) исследование влияния параметров метода на скорость и точность выполнения;

5) исследование и тестирование алгоритма в многомерном случае.

Vu Viet Thang, Postgraduate Student, thangvuviet84@gmail.com  
Moscow Institute of Physics and Technology (State University)

## Speedup Algorithm Clustering DBSCAN by Using Algorithm K-means

*Clustering is one of the most important tasks of data mining. Although there is a lot to explore ways of clustering such as K-means, Fuzzy C-means et al., But there is a problem of increasing the accuracy and acceleration algorithms for clustering, due to the fact that during the last 10 years the amount of data to be processed has increased substantially. This paper presents a new approach to speed up the clustering algorithm based on density DBSCAN (Density Based Spatial Clustering of Applications with Noise). The practical studies show that the speed of clustering algorithm proposed is higher, while maintaining accuracy.*

**Keywords:** Clustering, DBSCAN, K-means

### References

1. Ester M., Kriegel H. P., Sander J., Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, *Published in Proc. of 2<sup>nd</sup> International Conference on Knowledge Discovery and Data Mining (KDD)*, 1996.
2. Celic M., Dadaser-Celic F., Dokuz A. S. Anomaly Detection in Temperature Data Using DBSCAN Algorithm, *Innovations in Intelligent Systems and Applications (INISTA)*, 2011.
3. Chris Ding, Xiaofeng He. K-means Clustering via Principal Component Analysis, *Proc. of Int'l Conf. Machine Learning (ICML 2004)*, July 2004. P. 225–232.
4. Zhenguo Chen, Yong Fei Li. Anomaly Detection Based on Enhanced DBScan Algorithm, *Proc. Engineering*, 2011, vol. 15.

1. Ester M., Kriegel H. P., Sander J., Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise // *Published in Proc. of 2nd International Conference on Knowledge Discovery and Data Mining (KDD)*, 1996.
2. Celic M., Dadaser-Celic F., Dokuz A. S. Anomaly Detection in Temperature Data Using DBSCAN Algorithm // *Innovations in Intelligent Systems and Applications (INISTA)*, 2011.
3. Chris Ding, Xiaofeng He. K-means Clustering via Principal Component Analysis // *Proc. of Int'l Conf. Machine Learning (ICML 2004)*. July 2004. P. 225–232.
4. Zhenguo Chen, Yong Fei Li. Anomaly Detection Based on Enhanced DBScan Algorithm // *Proc. Engineering*. 2011. Vol. 15.
5. Ranjan R., Sahoo G. A New Clustering Approach for Anomaly Intrusion Detection // *International Journal of Data Mining & Knowledge Management Process (IJDKP)*, March 2014. Vol. 4, n. 2. P. 29–38.
6. Li Xue-yong, Gao Guo-hong, Sun Jia-xia. A New Intrusion Detection Method Based on Improved DBSCAN // *Information Engineering (ICIE)*, 2010, WASE International Conference. 2010. Vol. 2.
7. Karypis G. Chameleon data set available from <http://glaros.dtc.umn.edu/gkhome/cluto/cluto/download>, 2008.
8. Tan P. N., Steinbach M., Kumar V. *Introduction to Data Mining*. Addison-Wesley, 2005.
9. Rand W. M. Objective Criteria for the Evaluation of Clustering Methods // *Journal of the American Statistical Association*. 1971. Vol. 66, Is. 336.

5. Ranian R., Sahoo G. A New Clustering Approach for Anomaly Intrusion Detection, *International Journal of Data Mining & Knowledge Management Process (IJDKP)*, March 2014, vol. 4, pp. 29–38.
6. Li Xue-yong, Gao Guo-Hong, Sun Jia-xia. A New Intrusion Detection Method Based on Improved DBSCAN, *Information Engineering (ICIE)*, 2010, WASE International Conference, 2010, vol. 2.
7. Karypis G. Chameleon data set available from, URL: <http://glaros.dtc.umn.edu/gkhome/cluto/cluto/download>, 2008.
8. Tan P. N., Steinbach M., Kumar V. *Introduction to Data Mining*, Addison-Wesley, 2005.
9. Rand W. M. Objective Criteria for the Evaluation of Clustering Methods, 1971, *Journal of the American Statistickl Association*, vol. 66, Is. 336.

# СТРУКТУРНЫЙ СИНТЕЗ И ПРИНЯТИЕ РЕШЕНИЙ STRUCTURAL SYNTHESIS AND DECISION MAKING

УДК 004.891

А. В. Захаров<sup>1</sup>, канд. физ.-мат. наук, доц., andrewzakhar@mail.ru,  
Р. Р. Рамазанова<sup>2</sup>, аспирант, selen90@mail.ru, И. С. Курилова<sup>2</sup>, аспирант, irina.curilowa@mail.ru

<sup>1</sup> ФГБОУ ВО Уфимский государственный университет экономики и сервиса, г. Уфа

<sup>2</sup> ФГБОУ ВПО Уфимский государственный педагогический университет им. Акмуллы, г. Уфа

## Управление системой, состоящей из набора типовых подсистем, с помощью системы показателей

*Рассматривается методика управления системой, состоящей из набора типовых подсистем, с помощью вычисления их рейтингов. При этом предполагается, что каждая подсистема направлена на увеличение своего рейтинга. Предлагается схема соответствующей экспертной системы поддержки принятия управляющих решений. Проводится теоретическая оценка управления системой, а также результаты практического применения методики.*

**Ключевые слова:** управление системами, балльно-рейтинговая система

### 1. Методология управления системой с помощью системы показателей

#### 1.1. Система показателей подсистем

Системы управления с помощью рейтингов на данный момент достаточно популярны в экономических и социальных системах. Если вопросам формирования совокупности показателей и сбалансированности методики подсчета уделено достаточно внимания (см. например [1]), вопросам управления системами с помощью показателей посвящено ограниченное число, как правило, специализированных работ (см., например, [2]–[4]). Для оценки рейтингов при управлении сложными системами используют следующие типовые модели:

- простая линейная модель взвешенных нормированных экспертных оценок (см., например, [4]);
- модель линейного программирования (см., например, [5]);
- модель метода анализа иерархий на основе экспертных оценок (см., например, [6]);
- различные линейные модели показателей как с использованием стандартных показателей в соответствующей предметной области (например, [1], [7]), так и с обоснованием методики выбора показателей (например [3]).

Недостатками вышеуказанных моделей являются:

- отсутствие математической модели, которая может быть использована для оценки влияния использования рейтингов;
- отсутствие возможностей прогноза поведения систем при сохранении предлагаемой системы рейтингового управления.

В данной работе предпринята попытка с общих позиций управления системами оценить эффективность управления системой (с помощью принципов, закладываемых в экспертную систему) на основе рейтингов ее подсистем. При этом устраняются два указанных выше недостатка существующих моделей рейтингового управления: выполняется оценка результатов рейтингового управления, в том числе асимптотическая оценка, позволяющая осуществить долгосрочный прогноз поведения системы.

Рассмотрим систему, состоящую из набора типовых подсистем. Предположим, что каждая подсистема имеет набор показателей (свойств), которые характеризуют ее развитие. Все показатели распределены по  $n$  логическим группам, в группе  $j$  число показателей обозначим через  $n_j$  ( $j = 1, \dots, n$ ). Каждому показателю соответствует определенное число баллов, соответствующее его значимости.

Обозначим показатели группы  $j$  через  $a_i^j$  ( $j = 1, \dots, n$ ,  $i = 1, \dots, n_j$ ). Число баллов, соответствующее показателю  $a_i^j$ , обозначим через  $b_i^j$ . Если показатель  $a_i^j$  для подсистемы выполнен, подсистеме начисляется соответствующий балл  $b_i^j$ .

Поставим следующие задачи.

1. Разработать систему подсчета общего рейтинга подсистемы, позволяющую повысить ее конкурентоспособность (при условии стремления подсистемы к увеличению своего рейтинга).

2. Разработать методологию управления системой с помощью изменения методики подсчета рейтинга.

Разработанная методология управления системой позволит выбирать те или иные направления развития системы, связанные с конкретными группами показателей.

## 1.2. Методика расчета рейтинга подсистемы

**Оценка подсистемы по группе показателей.** Поскольку значимость показателя каждой группы уже определена соответствующим числом баллов, оценку подсистемы  $b_j$  по группе показателей  $j$  определим как среднее арифметическое по всем показателям группы:

$$b_j = \sum_{i=1}^{n_j} b_i^j / n_j.$$

**Общая оценка подсистемы.** Предположим, что каждая группа показателей имеет свою значимость для конкурентоспособности подсистемы на данный момент времени, выражаемой численно. Обозначим через  $v_j$  значимость группы  $j$  показателей ( $j = 1, \dots, n$ ), и потребуем от набора значимостей выполнения условия

$$\sum_{j=1}^n v_j = 1.$$

Общую оценку подсистемы зададим с помощью линейного выражения:

$$v = \sum_{j=1}^n v_j b_j.$$

## 1.3. Методология управления системой

Мы предполагаем, что каждая подсистема направлена на увеличение своего рейтинга. В этом случае управление системой можно осуществлять следующими методами.

1. Выбор набора показателей подсистем (а также распределение их по группам).

2. Определение значимости каждого показателя для повышения конкурентоспособности подсистемы, а именно значения баллов  $b_i^j$  ( $j = 1, \dots, n$ ,  $i = 1, \dots, n_j$ ).

3. Задание значимостей  $v_j$  каждой группы  $j$  показателей ( $j = 1, \dots, n$ ) на конкретный период.

## 1.4. Пример управления системой

В университете функционирует определенный набор кафедр, от рейтингов которых зависит их финансирование. Все показатели эффективности работы кафедры разбиты на четыре группы.

1. Кадровая и материальная обеспеченность.
2. Учебно-методическая деятельность.
3. Научно-исследовательская и инновационная деятельность.
4. Публикационная активность сотрудников.

Каждый показатель измеряется баллами от 0 до 5. В таком же диапазоне находятся средние арифметические значения по каждой группе показателей (оценка подсистемы по группе показателей). Руководство университета принимает решение усилить публикационную активность сотрудников и устанавливает на следующий год следующие значимости групп показателей:

$$v_1 = v_2 = v_3 = 0,2, v_4 = 0,4.$$

В этом случае кафедре проще увеличить свой рейтинг, усилив публикационную активность сотрудников.

## 2. Система поддержки принятия управляющих решений

### 2.1. Цикл управления системой

Формализуем общий цикл управления системой.

1. Формирование первоначального управляющего воздействия (задание начальной методики расчета рейтингов подсистем).
2. Выполнение рейтингов.
3. Коррекция управляющего воздействия (изменение методики расчета рейтингов подсистем).
4. Переход к п. 2.

### 2.2. Структурная схема процесса управления системой

Для общего описания процесса управления системой приведем его структурную схему (рис. 1).

Как видно из схемы, результат управления в общем случае может привести к изменению не только методики расчета рейтингов подсистем, но и к корректировке самой цели управления. К корректировке и уточнению методики управляющего воздействия может привести не только общий результат, но и информация об изменении объекта управления (подсистемы).

В общем случае оказывается воздействие на информационный уровень объекта управления, т. е. метауправление (коррекция его целей, его взаимоотношений с окружающей средой, мотиваций, способов принятия и реализации решений). В результате управляющего воздействия цели объекта управления сближаются и согласовываются с целями системы управления.

### 2.3. Автоматизация управления системой

Автоматизация управления системой может быть проведена в двух направлениях:

- сбор данных о подсистемах и расчет рейтингов подсистем;
- поддержка принятия решения об управляющем воздействии.



Рис. 1. Структурная схема процесса управления

Автоматизация сбора данных о подсистемах при наличии регламентов соответствующих информационных транзакций, а также расчет рейтингов подсистем представляет собой простую задачу информационного взаимодействия между компонентами системы.

Поддержка принятия решения об управляющем воздействии может осуществляться решающей или советующей экспертной системой. Решающая экспертная система может быть использована в системах управления техническими системами, состоящими из набора типовых систем. Советующая экспертная система необходима к применению в социальных и экономических системах, на предприятиях и в организациях. Один из вариантов системы поддержки принятия решений приведен ниже.

#### 2.4. Модель экспертной системы управления

Опишем информационную модель экспертной системы поддержки принятия решений об изменении методики подсчета рейтингов (рис. 2).

Экспертная система использует базы данных подсистем для подсчета рейтинга и состоит из двух экспертных систем высшего и низшего уровней. Экспертная система низшего уровня используется исключительно для подсчета рейтинга в соответствии с утвержденной методикой на основе баз данных подсистем. Экспертная система высшего уровня состоит из двух модулей: констатирующего и советующего (или решающего) модуля. Констатирующий модуль выводит информацию о текущем положении дел на разных уровнях абстракции, рейтингах, а также информацию, подтверждающую рейтинг той или иной подсистемы. Советующий (или решающий) модуль проводит анализ и отображает информацию о подсистемах, осуществляет прогноз дальнейшего поведения системы, а также генерирует (предполагаемое) решение относительно управляющего воздействия.

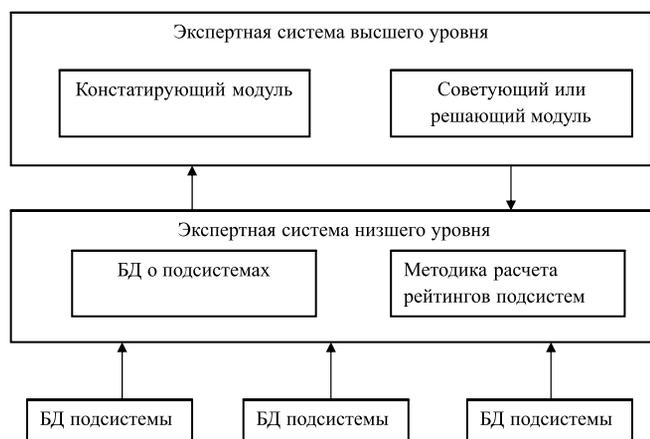


Рис. 2. Информационная модель экспертной системы поддержки принятия решений

### 3. Оценка управления системой, состоящей из набора типовых подсистем

Предположим, что система  $S$  состоит из набора типовых подсистем  $S_1, S_2, \dots, S_N$ , имеющих численные рейтинги соответственно  $R_1(i), R_2(i), \dots, R_N(i)$ , определяемые в дискретные (равноотстоящие) моменты времени  $i = 0, 1, 2, \dots, I$ . Вычислим приведенные рейтинги

$$R_j = R_j(i) / \sum_{j=1}^N R_j(i), j = 1, \dots, N, i = 0, 1, 2, \dots, I.$$

Приведенные рейтинги в сумме дают единицу и, в частности, являются ориентиром для определения будущего роста подсистемы относительно других подсистем.

Предположим, что каждая подсистема  $S_i$  после присваивания того или иного значения рейтинга увеличивает на следующий период времени свою эффективность, что приводит к условному превышению приведенного рейтинга на величину, пропорциональную приведенному рейтингу:

$$\bar{R}_j(i) = R_j(i-1) + e_j R_j^*(i-1), i = 1, 2, \dots, I,$$

где  $e_j$  — коэффициент эффективности использования подсистемой  $j$  своего рейтинга,  $j = 1, \dots, N$ .

**Пример.** В университете функционирует определенный набор кафедр, в конце каждого года подсчитываются рейтинги кафедр, после чего фиксированная каждый год стимулирующая сумма распределяется по кафедрам пропорционально рейтингам (т. е. соответственно приведенному рейтингу). Каждая кафедра имеет возможность улучшить свои показатели за счет стимулирующей суммы в соответствии с эффективностью ее использования.

Так как все рейтинги увеличились, для вычисления реального приведенного рейтинга на следующий период времени необходимо провести нормализацию — разделить на сумму всех рейтингов:

$$R_j^*(i) = \bar{R}_j(i) / \sum_{j=1}^N \bar{R}_j(i).$$

Далее рассмотрим несколько случаев.

**Случай 1. Равные эффективности использования подсистемами своего рейтинга.** Предположим, что коэффициенты эффективности использования подсистемами своего рейтинга равны

$$e_1 = e_2 = \dots = e_N = e.$$

В этом случае приведенные рейтинги систем  $i$  и  $k$  будут равны соответственно:

$$\bar{R}_j(i) = R_j(i-1) + e R_j(i-1) / \sum_{j=1}^N R_j(i), i = 1, 2, \dots, I,$$

$$\bar{R}_k(i) = R_k(i-1) + e R_k(i-1) / \sum_{j=1}^N R_j(i), i = 1, 2, \dots, I.$$

Вычислив их соотношение

$$\frac{\bar{R}_j(i)}{\bar{R}_k(i)} = \frac{R_j(i-1) + e_j R_j(i-1) / \sum_{j=1}^N R_j(i)}{R_k(i-1) + e_k R_k(i-1) / \sum_{j=1}^N R_j(i)} = \frac{\bar{R}_j(i-1)}{\bar{R}_k(i-1)},$$

мы приходим к теореме.

**Теорема 1.** При равных коэффициентах эффективности использования рейтингов приведенные рейтинги подсистем не изменяются со временем.

Теорема означает, что если все подсистемы с одинаковой эффективностью используют ресурсы, получаемые благодаря рейтингам, в рамках рассматриваемой модели их рейтинговое положение в системе не изменяется.

**Случай 2. Разные эффективности использования подсистемами своего рейтинга.** Предположим, что коэффициенты эффективности использования рейтинга, по крайней мере, у двух подсистем не равны, например  $e_j > e_k$ .

В этом случае приведенный рейтинг систем  $i$  и  $k$  будет соответственно

$$\bar{R}_j(i) = R_j(i-1) + e_j R_j(i-1) / \sum_{j=1}^N R_j(i-1), \quad i = 1, 2, \dots, I,$$

$$\bar{R}_k(i) = R_k(i-1) + e_k R_k(i-1) / \sum_{j=1}^N R_j(i-1), \quad i = 1, 2, \dots, I.$$

Оценив их соотношение

$$\frac{\bar{R}_j(i)}{\bar{R}_k(i)} = \frac{R_j(i-1) \left( 1 + e_j / \sum_{j=1}^N R_j(i-1) \right)}{R_k(i-1) \left( 1 + e_k / \sum_{j=1}^N R_j(i-1) \right)} > \frac{\bar{R}_j(i-1)}{\bar{R}_k(i-1)},$$

приходим к теореме.

**Теорема 2.** Если коэффициент эффективности использования рейтингов у одной подсистемы больше, чем у другой, соотношение их рейтингов со временем монотонно возрастает.

**Случай 3. Существование системы с максимальной эффективностью.** Обозначим

$$\gamma_{jk} = \frac{1 + e_j / \sum_{i=1}^N R_j(i)}{1 + e_k / \sum_{j=1}^N R_j(i)}.$$

Переставим индексы всех подсистем, упорядочив их по возрастанию коэффициентов эффективности использования рейтингов:

$$e_1 \leq e_2 \leq \dots \leq e_n.$$

Пусть имеется подсистема с максимальным коэффициентом эффективности использования рейтинга:

$$e_1 \leq e_2 \leq \dots \leq e_{n-1} < e_n.$$

В этом случае

$$\gamma_{nk} = \frac{1 + e_n / \sum_{j=1}^N R_j(i)}{1 + e_k / \sum_{j=1}^N R_j(i)} = \frac{\sum_{j=1}^N R_j(i) + e_j + (e_n - e_j)}{\sum_{j=1}^N R_j(i) + e_j} > 1 + \frac{e_n - e_j}{\sum_{j=1}^N R_j(i) + e_j}.$$

Пусть сумма рейтингов  $\sum_{j=1}^N R_j(i)$  ограничена сверху константой  $S$ . Тогда

$$\gamma_{nk} > 1 + \frac{e_n - e_j}{S + e_j}.$$

Это означает, что величина  $\gamma_{nk} > 1 + \varepsilon$ ,  $\varepsilon > 0$ . В этом случае

$$\frac{\bar{R}_n(i)}{\bar{R}_k(i)} > \frac{\bar{R}_n(i-1)}{\bar{R}_k(i-1)} (1 + \varepsilon), \quad \varepsilon > 0.$$

Отсюда следует, что

$$\lim_{i \rightarrow \infty} \frac{\bar{R}_n(i)}{\bar{R}_k(i)} = \infty,$$

или по-другому

$$\lim_{i \rightarrow \infty} \frac{\bar{R}_k(i)}{\bar{R}_n(i)} = 0.$$

Приходим к следующей теореме.

**Теорема 3.** Если существует подсистема с максимальным коэффициентом эффективности и сумма рейтингов ограничена сверху константой, то при стремлении времени к бесконечности рейтинги всех остальных систем становятся бесконечно малыми величинами.

При сохранении эффективности подсистем и поддержке подсистем с высокой эффективностью система с максимальной эффективностью начинает со временем вытеснять другие системы. Этот вывод согласуется с известными примерами систем: языки отдельных народов со временем вытесняют множество других языков, крупные предприятия и организации со временем разрастаются еще больше и поглощают мелкие предприятия и организации.

## Выводы

Разработана модель экспертной системы управления на основе балльно-рейтинговой оценки, позволяющая численно оценить результаты применения управления подсистемами с помощью рейтингов. Получены теоретические оценки результатов применения предложенной методики управления системой.

Впервые проводится оценка эффективности управления системой (с помощью принципов, закладываемых в экспертную систему) на основе рейтингов ее подсистем. При этом устраняются два указанных выше недостатка существующих моделей рейтингового управления: проводится оценка результатов рейтингового управления, в том числе асимптотическая оценка, позволяющая осуществить долгосрочный прогноз поведения системы. Предложенная модель может служить основой для дальнейших исследований в области применения рейтинговой системы управления для различных предметных областей, а также практического применения для разработки специализированных экспертных систем управления на основе рейтингов.

#### Список литературы

1. **Коровина В. А.** Оценка эффективности деятельности предприятия с использованием системы сбалансированных показате-

телей // Вестник Южно-Уральского государственного университета. Серия: Экономика и менеджмент. 2008. № 20 (120). С. 40—51.

2. **Калинина А. Э., Соколов А. Ф., Елхина И. А.** Информационно-аналитические методы мониторинга эффективности систем управления региональной экономикой // Известия Волгоградского государственного технического университета. 2012. Т. 13, № 7 (94). С. 169—179.

3. **Тарасенко Ф. П.** О применении рейтинговых оценок в управлении вузом // Проблемы управления в социальных системах. 2011. Т. 3. № 5. С. 81—96.

4. **Быстров О. Ф.** Инновационная методика рейтингового оценивания кафедр института в системе управления качеством вуза // Научный вестник МГИИТ. 2010. № 5. С. 46—48.

5. **Болодурин И. П.** Математическая модель управления развитием промышленности регионов // Вестник Оренбургского государственного университета. 2005. № 4. С. 134—138.

6. **Дуканич Л. В., Тимченко А. С.** Рейтинговое управление экономическими системами и процессами: концепция и некоторые результаты применения // Terra economicus. 2005. Т. 3, № 3. С. 83—91.

7. **Ивченко И. Б., Гаврашенко М. О.** Математическая модель рейтингового оценивания та управління банком // Восточно-Европейский журнал передовых технологий. 2010. Т. 2, № 4 (44). С. 48—51.

**A. V. Zakharov**, Associated Professor, e-mail: andrewzakhar@mail.ru,  
Ufa State University of Economics and Service,  
**R. R. Ramazanova**, Graduate Student, selena-9090@mail.ru,  
**I. S. Kurilova**, Graduate Student, irina.curilowa@mail.ru,  
M. Akmullah Bashkir State Pedagogical University

## Management of System Consisting of a Set of Typical Subsystems Using Scorecards

*The control technique of system consisting of a set of typical subsystems by computing their ratings is proposed. It is assumed that each subsystem is aimed at increasing their rating.*

*The control technique includes:*

- Selection of a set of indicators subsystems (as well as their distribution by groups).
- Determination of the significance of each indicator to increase the competitiveness of subsystems.
- Setting the significances of each group for a specific period.

*The control iteration consists of three steps: setting the initial method of calculating subsystems ratings; implementation the calculated ratings; correction of the manipulated variable (changing the parameters, of subsystems ratings calculations).*

*Automation control can be implemented in two directions: data collection and calculation of the subsystems ratings and support for the decision to control the global system.*

*Decision support system are considered and described in terms of automation control. It used database of subsystems and consists of two level. Lower level of decision support system is used for rating computing according with proposed technique. Upper level of decision support system consists of two modules: the module of analytics and the module of forecast. Theoretical estimates of the control system are calculated, among them: estimates of equal and not equal subsystem cases, asymptotical estimate of leader subsystem. The examples of control technique implementation are considered too.*

**Keywords:** system analysis, scorecards, system management, control technique, rating system, system indicator set, decision support system, system forecast, system control estimate, asymptotical estimate of leader subsystem

#### References

1. **Korovina V. A.** Ocenka jeffektivnosti dejatel'nosti predpriyatija s ispol'zovaniem sistemy sbalansirovannyh pokazatelej. *Vestnik Juzhno-Ural'skogo gosudarstvennogo universiteta*. Serija: jekonomika i menedzhment, 2008, no. 20 (120), pp. 40—51.

2. **Kalinina A. Je., Sokolov A. F., Elhina I. A.** Informacionno-analiticheskie metody monitoringa jeffektivnosti sistem upravlenija regional'noj jekonomikoj. *Izvestija Volgogradskogo gosudarstvennogo tehničeskogo universiteta*, 2012, vol. 13, no. 7 (94), pp. 169—179.

3. **Tarasenko F. P.** O primenenii rejtingovyh ocenok v upravlenii vuzom. *Problemy upravlenija v social'nyh sistemah*, 2011, vol. 3, no. 5, pp. 81—96.

4. **Bystrov O. F.** Innovacionnaja metodika rejtingovogo ocenivanija kafedr instituta v sisteme upravlenija kachestvom vuzov. *Nauchnyj vestnik MGIIT*, 2010, no. 5, pp. 46—48.

5. **Bolodurina I. P.** Matematicheskaja model' upravlenija razvitiem promyshlennosti regionov. *Vestnik Orenburgskogo gosudarstvennogo universiteta*, 2005, № 4, pp. 134—138.

6. **Dukanich L. V., Timchenko A. S.** Rejtingovoe upravlenie jekonomicheskimi sistemami i processami: koncepcija i nekotorye rezul'taty primenenija. *Terra economicus*, 2005, vol. 3, no. 3, pp. 83—91.

7. **Ivchenko N. B., Gavrashenko M. O.** Matematichna model' rejtingovogo ocenjuvanija ta upravlinnja bankom. *Vostochno-Evropskij zhurnal peredovyh tehnologij*, 2010, vol. 2, no. 4 (44), pp. 48—51.

УДК 004.7

**В. Н. Тарасов**, д-р техн. наук, проф., зав. каф., e-mail: vt@ist.psati.ru,  
**Н. Ф. Бахарева**, д-р техн. наук, проф., зав. каф., e-mail: bahareva-nf@psuti.ru,  
**Л. В. Липилина**, аспирант, e-mail: mila199113@gmail.com,  
Поволжский государственный университет телекоммуникаций и информатики

## Математическая модель телетрафика на основе системы G/M/1 и результаты вычислительных экспериментов

*Представлены результаты исследования задержки для системы  $H_2/M/1$  типа G/M/1 для широкого диапазона изменения параметров трафика. Известно, что распределенная по гиперэкспоненциальному закону  $H_2$  случайная величина имеет коэффициент вариации больше 1. Также известно, что в системе G/G/1 время ожидания связано с коэффициентами вариаций интервалов поступления и обслуживания квадратичной зависимостью. Следовательно, время ожидания в системе G/M/1 также зависит от коэффициента вариации интервалов поступления. Кроме того, оно зависит и от моментов высших порядков. Учитывая тот факт, что распределение  $H_2$  является трехпараметрическим, приведен механизм аппроксимации произвольных законов распределений гиперэкспоненциальным распределением. Это может быть выполнено как на уровне двух первых моментов, так и на уровне трех первых моментов. Система  $H_2/M/1$  имеет то преимущество перед другими системами с входными распределениями с тяжелым хвостом, что для нее авторами получено точное решение в аналитическом виде.*

**Ключевые слова:** система массового обслуживания  $H_2/M/1$ , среднее время ожидания в очереди, задержка, преобразование Лапласа

### Введение

Как известно из теории массового обслуживания [1], среднее время ожидания требований в очереди является составной частью задержки в сетях пакетной передачи данных. В классической системе массового обслуживания (СМО) M/M/1 оно выражается равенством (здесь и далее используется классическое трехпозиционное обозначение Кендалла)

$$\bar{W} = \frac{\rho/\mu}{1-\rho}, \quad (1)$$

для системы M/G/1 —

$$\bar{W} = \frac{\lambda \bar{X}^2/2}{1-\rho}. \quad (2)$$

Наконец, для системы G/G/1 это время равно

$$\bar{W} = \frac{D_\lambda + D_\mu + (1-\rho)^2/\lambda^2}{2(1-\rho)/\lambda} - \frac{\bar{I}^2}{2\bar{I}}. \quad (3)$$

В этих формулах использованы следующие обозначения:  $\rho$  — коэффициент загрузки системы ( $0 < \rho = \lambda/\mu < 1$ ),  $\lambda$  — интенсивность входного по-

тока,  $\mu$  — интенсивность обслуживания,  $\bar{X}^2$  — второй начальный момент времени обслуживания,  $D_\lambda, D_\mu$  — соответственно дисперсии интервалов поступления и времени обслуживания,  $\bar{I}$  и  $\bar{I}^2$  — соответственно среднее значение и второй начальный момент периода простоя, а черта сверху здесь и далее означает операцию усреднения.

Для системы G/M/1 выражение для времени ожидания неизвестно, поэтому исследование результата для этой системы представляет актуальную задачу. В статье исследуется система G/M/1 для случая ( $c_\lambda > 1, c_\mu = 1$ ), где  $c_\lambda$  и  $c_\mu$  — коэффициенты вариаций интервала между поступлениями и времени обслуживания соответственно.

Учитывая тот факт, что выражение (3) явно содержит дисперсии интервалов входного потока и времени обслуживания, а также возможно включает и моменты высших порядков во втором слагаемом в правой части, при анализе СМО G/M/1 (с произвольным законом поступления в систему), необходимо учитывать не только первые два момента случайного интервала между поступлениями, но и моменты более высокого порядка. Необходимость этого полностью будет подтверждена исследованиями, приведенными ниже.

## Постановка задачи и ее решение

Анализ выражения (3), с другой стороны, показывает, что среднее время ожидания требований в очереди связана с коэффициентами вариаций интервалов поступления и обслуживания квадратичной зависимостью, так как дисперсия случайной величины  $D_\tau$  и коэффициент вариации  $c$  связаны соотношением  $c = \sqrt{D_\tau}/m_\tau$ , где  $m_\tau$  — математическое ожидание величины  $\tau$ . Тогда можем записать соотношения для времени ожидания для различных систем: среднее время ожидания в очереди в системе с входными распределениями, имеющими коэффициенты вариаций интервалов между требованиями входного потока  $c_\lambda < 1$  и времени обслуживания  $c_\mu < 1$ , меньше, чем в системе M/M/1, и меньше, чем в системе G/M/1 при  $c_\lambda > 1$ , и меньше, чем в системе G/G/1 при условии ( $c_\lambda > 1, c_\mu > 1$ ) при одинаковой нагрузке:

$$\begin{aligned} \overline{W} | (c_\lambda < 1, c_\mu < 1) &< \overline{W} | (c_\lambda = 1, c_\mu = 1) < \\ < \overline{W} | (c_\lambda > 1, c_\mu = 1) < \overline{W} | (c_\lambda > 1, c_\mu > 1), \end{aligned} \quad (4)$$

где  $\overline{W}$  — среднее значение величины  $W$ .

Соотношения (4) также отражают эволюцию теории массового обслуживания.

Краткий обзор отечественных и зарубежных источников в области телетрафика позволяет сделать следующие выводы. В работах [2, 3] используются известные классические модели массового обслуживания M/M/1, M/G/1 и верхние границы для времени ожидания требований для системы G/G/1, или же в отсутствие теоретических результатов применяется имитационное моделирование. В зарубежной литературе, например в [4], уже давно известны эвристические формулы для систем G/M/1 и G/G/1. В частном случае, для системы  $H_2/M/1$  в работе [4] приведено выражение для среднего времени ожидания:

$$\overline{W} = \frac{\rho}{\mu(1-\rho)} \left\{ 1 + \frac{1}{\rho} \left[ \sqrt{(1-\theta)^2 + (r-1)} - (1-\theta) \right] \right\}, \quad (5)$$

где промежуточные параметры:  $\theta = \frac{\rho(q-r) - (q-r^2)}{2\rho(r-1)}$ ,

$$q = \frac{\overline{\tau_\lambda^3}}{6(\overline{\tau_\lambda})^3}, \quad r = \frac{\overline{\tau_\lambda^2}}{6(\overline{\tau_\lambda})^2}, \quad \text{а } \overline{\tau_\lambda}, \overline{\tau_\lambda^2}, \overline{\tau_\lambda^3} \text{ — начальные}$$

моменты до 3-го порядка интервалов между поступлениями требований в систему. Выражение (5) получено с помощью преобразования Лапласа—Стилтьеса функции распределения времени ожидания, и это преобразование приведено в работе [1].

Следует отметить, что в работе [5] представлен результат для среднего времени ожидания для частного случая системы G/G/1, а именно  $H_2/H_2/1$ , полученный решением интегрального уравнения Линдли методом спектрального разложения.

В статье на примере системы  $H_2/M/1$  ставится задача исследования зависимости времени ожидания для СМО G/M/1 от моментов высших порядков случайного интервала между поступлениями и построения механизма аппроксимации произвольных законов распределений (G) с тяжелым хвостом с помощью гиперэкспоненциального распределения.

В настоящее время не существует аналитических методов для точного определения характеристик СМО G/M/1 или G/M/m, и, как следствие, это отражается на степени адекватности стохастических сетевых моделей реальным компьютерным и телекоммуникационным сетям и на качестве принимаемых проектных решений. При этом СМО с гиперэкспоненциальными входными распределениями, в отличие от систем с распределениями с тяжелыми хвостами, позволяет получить решение задачи в аналитическом виде. Тот факт, что такие распределения имеют место на практике, подтвержден в работе [6], где приведены результаты анализа интервалов между пакетами входящего трафика на сервер вуза, полученные с помощью программы — дополнения к анализатору Wircshark.

В работе [7] авторами приведены полученные результаты для времени ожидания в системе  $H_2/M/1$  и их практическое применение. Также приведено преобразование Лапласа для функции плотности времени ожидания для системы  $H_2/M/1$ , полученное решением интегрального уравнения Липдли методом спектрального разложения [1]:

$$W^*(s) = \frac{s_1(s+\mu)}{\mu(s+s_1)}, \quad (6)$$

где  $s_1$  — значение отрицательного корня со знаком

$$\begin{aligned} \text{минус } -s_1 = - \left( \sqrt{\frac{c_2^2}{4} + c_1} - \frac{c_2}{2} \right) \text{ квадратного уравнения} \\ s^2 - c_2s - c_1 = 0 \end{aligned} \quad (7)$$

с коэффициентами  $c_1 = \mu[\lambda_1(1-p) + \lambda_2p] - \lambda_1\lambda_2$ ,  $c_2 = \lambda_1 + \lambda_2 - \mu$ . Величины  $p, \lambda_1, \lambda_2$  являются параметрами гиперэкспоненциального закона распределения с функцией плотности

$$a(t) = p\lambda_1 e^{-\lambda_1 t} + (1-p)\lambda_2 e^{-\lambda_2 t} \quad (8)$$

для системы  $H_2/M/1$ .

Среднее время ожидания в очереди равно значению производной от функции преобразования Лапласа (6)

$$\frac{dW^*(s)}{ds} = \frac{s_1\mu(s_1+s) - s_1(s+\mu)\mu}{\mu^2(s+s_1)^2}$$

со знаком минус в точке  $s = 0$ :

$$\bar{W} = -\left. \frac{dW^*(s)}{ds} \right|_{s=0} = \frac{-s_1^2 \mu + \mu^2 s_1}{\mu^2 s_1^2} = \frac{1}{s_1} - \frac{1}{\mu}.$$

Окончательно среднее время ожидания в очереди для СМО  $H_2/M/1$  определяется формулой

$$\bar{W} = 1/s_1 - 1/\mu. \quad (9)$$

Выражение (6) на основе свойств преобразования Лапласа позволяет также определить моменты высших порядков для времени ожидания. Например, начальный момент 2-го порядка времени ожидания равен значению второй производной от преобразования (6) в точке  $s = 0$ , и он будет иметь вид

$$\bar{W}^2 = \left. \frac{d^2 W^*(s)}{ds^2} \right|_{s=0} = \frac{2s_1(\mu - s_1)}{\mu(s + s_1)^3} \Big|_{s=0} = \frac{2(\mu - s_1)}{\mu s_1^2}. \quad (10)$$

В свою очередь, начальный момент 2-го порядка позволяет определить дисперсию времени ожидания:  $D_W = \bar{W}^2 - \bar{W}^2$ . Учитывая определение джиттера в телекоммуникациях как разброс времени ожидания от его среднего значения [8], тем самым получим возможность определения джиттера через дисперсию. Это является важным результатом для анализа трафика, чувствительного к задержкам.

Для практического применения результатов (9) и (10) необходимо определить входящие в них параметры. Определение этих неизвестных параметров рассматривается ниже. Как оказалось, неизвестные параметры закона распределения (8) можно определить путем его аппроксимации с использованием двух первых моментов, а также с использованием трех первых моментов случайного интервала между поступлениями. Таким образом, у нас появится возможность оценки точности обоих подходов.

### Аппроксимация закона распределения на уровне двух первых моментов

Воспользуемся свойством преобразования Лапласа воспроизведения моментов и запишем начальные моменты до 2-го порядка для закона распределения (7):

$$\bar{\tau}_\lambda = \frac{p}{\lambda_1} + \frac{(1-p)}{\lambda_2^2}, \quad (11)$$

$$\bar{\tau}_\lambda^2 = \frac{2p}{\lambda_1^2} + \frac{2(1-p)}{\lambda_2^2}. \quad (12)$$

Рассматривая равенства (11) и (12) как уравнения для метода моментов, найдем неизвестные параметры распределения (8):  $\lambda_1, \lambda_2, p$ . Для этого за-

пишем связующее условие в виде выражения для квадрата коэффициента вариации

$$c_\lambda^2 = \frac{\bar{\tau}_\lambda^2 - (\bar{\tau}_\lambda)^2}{(\bar{\tau}_\lambda)^2}. \quad (13)$$

Заметим, что система уравнений (11) и (12) с тремя неизвестными является недоопределенной. Исходя из вида уравнения (11) положим

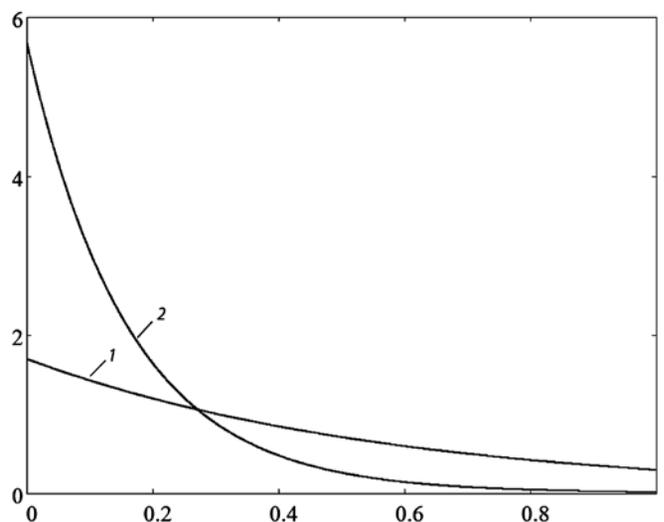
$$\lambda_1 = 2p/\bar{\tau}_\lambda, \quad \lambda_2 = 2(1-p)/\bar{\tau}_\lambda \quad (14)$$

и потребуем выполнения условия (13). Подставив выражения (11), (12) и (14) в (13) и решив квадратное уравнение относительно параметра  $p$ , получим для него два значения:

$$p = \frac{1}{2} \left( 1 \pm \sqrt{\frac{c_\lambda^2 - 1}{c_\lambda^2 + 1}} \right), \quad (15)$$

причем можно воспользоваться любым из них [8]. Таким образом, найдено частное решение недоопределенной системы (11), (12), и гиперэкспоненциальный закон распределения может определяться полностью двумя первыми моментами и перекрывать весь диапазон изменения коэффициента вариации  $c_\lambda$  от 1 до  $\infty$ .

Рассмотрим **пример 1**. Пусть коэффициент загрузки СМО  $\rho = \bar{\tau}_\mu / \bar{\tau}_\lambda = 0,9$ , где  $\bar{\tau}_\lambda$  и  $\bar{\tau}_\mu$  — средние значения интервалов между поступлениями и времени обслуживания. Рассмотрим случай нормированного обслуживания  $\bar{\tau}_\mu = \mu^{-1} = 1$  (1 — единица времени). Тогда средний интервал между поступлениями  $\bar{\tau}_\lambda = 10/9$  (единиц времени). Пусть коэффициент вариации случайной величины — интер-



Графики функции плотности (7):

1 — аппроксимация закона распределения  $H_2$  на уровне двух моментов; 2 — на уровне трех моментов

вала времени между поступлениями  $c_\lambda = 4$ . Аппроксимация на уровне двух первых моментов дает:

$$p \approx 0,9697, \lambda_1 \approx 1,7454, \lambda_2 \approx 0,0546. \quad (16)$$

Таким образом, неизвестные параметры распределения (8) однозначно определены. График функции плотности (8) с этими параметрами приведен на рисунке, кривая 1. Теперь воспользуемся результатом для системы  $H_2/M/1$  (9), приведенным выше. Коэффициенты квадратного уравнения (7) в этом примере равны:  $c_1 \approx 0,0106$ ;  $c_2 = 0,80$ . Найдем отрицательный корень квадратного уравнения (7) с помощью пакета MathCAD:  $-s_1 \approx -0,0130$ . Тогда среднее время ожидания (9) равно  $\bar{W} \approx 75,786$  единиц времени.

### Аппроксимация на уровне трех первых моментов

Учитывая тот факт, что распределение (8) является трехпараметрическим, аппроксимацию можно выполнить и на уровне трех первых моментов, что позволит сравнить полученные результаты. С точки зрения теории вероятностей три момента полнее характеризуют случайную величину, чем два момента, поэтому такая аппроксимация будет точнее. Запишем выражение для момента 3-го порядка, полученное с помощью преобразования Лапласа:

$$\bar{\tau}_\lambda^3 = \frac{6p}{\lambda_1^3} + \frac{6(1-p)}{\lambda_2^3} -$$

для интервалов входного потока.

#### Рассмотрим пример 2.

В рассмотренный выше пример 1 введем в качестве третьего момента коэффициент асимметрии  $A_{S_\lambda}$  и для определенности положим  $A_{S_\lambda} = 7$ . Коэффициент вариации случайной величины — интервала времени между поступлениями  $c_\lambda = 4$  оставляем прежним, чтобы понять, как момент 3-го порядка влияет на конечный результат. Как известно, для пуассоновского потока параметры  $c_\lambda = 1$  и  $A_{S_\lambda} = 2$ .

При тех же значениях входных параметров СМО начальные моменты 2-го и 3-го порядков для распределения (8) соответственно будут:  $\bar{\tau}_\lambda^2 = 17(10/9)^2$ ,  $\bar{\tau}_\lambda^3 = 497(10/9)^3$ . При таких исходных данных для определения неизвестных параметров входного распределения (8)  $\lambda_1, \lambda_2, p$  запишем следующую систему уравнений на основе метода моментов:

$$\begin{cases} \frac{p}{\lambda_1} + \frac{(1-p)}{\lambda_2} = 10/9; \\ \frac{2p}{\lambda_1^2} + \frac{2(1-p)}{\lambda_2^2} = 17(10/9)^2; \\ \frac{6p}{\lambda_1^3} + \frac{3(1-p)}{\lambda_2^3} = 497(10/9)^3, \end{cases} \quad (17)$$

решив которые, найдем искомые параметры. Решение системы (17) в пакете MathCAD после округления дает следующие результаты:

$$p \approx 0,9111, \lambda_1 \approx 6,2291, \lambda_2 \approx 0,0922. \quad (18)$$

График функции плотности (8) с этими параметрами приведен на рисунке (кривая 2).

При этих параметрах коэффициенты квадратного уравнения (7) будут равны:  $c_1 \approx 0,0638$ ;  $c_2 \approx 5,3213$ , и его решение дает следующий отрицательный корень:  $-s_1 \approx -0,012$ . Воспользуемся результатом (9) и определим среднее время ожидания:  $\bar{W} \approx 82,62$  единиц времени. Дисперсия времени ожидания  $D_W = \bar{W}^2 - \bar{W}^2 = 6,991 \cdot 10^3$ . Относительная погрешность среднего времени ожидания в сравнении с результатом аппроксимации на уровне двух моментов составляет 8,27 %. Такая разница в полученных результатах объясняется существенным отличием двух функций плотностей, графики которых приведены на рисунке, так как результаты (16) и (18) для параметров распределения (8), полученные различными способами, сильно различаются.

Таким образом, учет моментов 3-го порядка интервалов поступления показывает существенную зависимость конечного результата (9) от моментов высших порядков. Поэтому результат (9), полученный на уровне трех первых моментов случайной величины, точнее результата, полученного аппроксимацией закона распределения (8) на уровне двух первых моментов. С точки зрения практического применения результатов СМО  $H_2/M/1$ , все же аппроксимация закона распределения на уровне двух первых моментов удобнее из-за простоты вычислений по формулам (7), (14) и (15) без решения системы трех уравнений. Поэтому выбор того или иного способа аппроксимации закона распределения (8) может зависеть от требуемой точности решения поставленной задачи.

*Примечание.* При подстановке исходных данных примера 2 в выражение (5) из работы [4] получим тот же самый результат, что и в примере 2:

$$\bar{W} \approx 82,62,$$

что также подтверждает достоверность полученных результатов.

### Результаты проведенных вычислительных экспериментов и их анализ

С использованием полученных соотношений (9) и (10) проведены вычислительные эксперименты над временем ожидания в системе. Результаты экспериментов сведены в табл. 1. Использован достаточно широкий диапазон изменения параметров трафика, а именно: загрузки системы  $\rho$  от 0,1 до 0,9, а коэффициента вариации интервалов поступле-

ния  $c_\lambda$  от 2 до 10. Это касается входных параметров. Выходными характеристиками являются: среднее время ожидания  $\bar{W}$ , определенное по выражению (9) на уровне двух первых моментов интервалов поступления; дисперсия времени ожидания, определенное через (10). Для сравнения полученных результатов в табл. 1 приведены также результаты классической системы M/M/1.

Анализ данных табл. 1 подтверждает квадратичную зависимость времени ожидания от коэффици-

ента вариации интервалов поступления. Кроме того, время ожидания резко возрастает с ростом коэффициента загрузки  $\rho$ .

Следует заметить, что классическая система M/M/1, использующая только средние значения временных интервалов, как следует из табл. 1, дает слишком оптимистичные результаты по задержкам.

Это говорит об адекватности полученного результата (9). Значения дисперсий времени ожидания  $D_W$  позволяют находить разброс времени ожидания от его среднего значения (джиттер), например, с использованием правила  $3\sigma$ .

Перейдем теперь к исследованию зависимости результатов (9) от моментов высших порядков. С учетом результатов предыдущего раздела о зависимости конечного результата времени ожидания для системы G/M/1 от моментов высших порядков, проведем расчеты по установлению степени такой зависимости.

Для этого рассмотрим следующие характерные случаи: малой нагрузки  $\rho = 0,1$ ; средней нагрузки  $\rho = 0,5$  и высокой нагрузки  $\rho = 0,9$ .

Расчеты, проведенные по методике предыдущего раздела, для этих случаев приведены в табл. 2. Данные табл. 2 необходимо воспринимать следующим образом: например, при малой нагрузке  $\rho = 0,1$ , при коэффициенте вариации интервала между поступлениями  $c_\lambda = 2$  и увеличении коэффициента асимметрии  $A_{S_\lambda}$  с 4 до 18 время ожидания убывает с 0,34 до 0,13 единиц времени. При  $c_\lambda = 4$  и увеличении  $A_{S_\lambda}$  с 7 до 18 время ожидания убывает с 1,26 до 0,17 единиц времени и т. д.

Анализ данных табл. 2 показывает, что с ростом коэффициентов асимметрий (3-го момента) при одной и той же нагрузке время ожидания уменьшается.

## Заключение

Результаты проведенных экспериментов по системе  $H_2/M/1$  подтверждают зависимость основной характеристики — времени ожидания в системе G/M/1 — от моментов высших порядков случайной величины — интервала между поступлениями. При этом при фиксированных значениях коэффициентов загрузки  $\rho$  и вариации интервала между поступлениями  $c_\lambda$  увеличение коэффициента асимметрии (момента 3-го порядка) приводит к уменьшению времени ожидания (см. табл. 2). Таким образом, влияние моментов высшего порядка на время ожидания в системе G/M/1 нельзя считать несущественным и им нельзя пренебрегать. В отличие от двойственной системы M/G/1 (2), характеристики которой определяются двумя первыми моментами времени обслуживания, система G/M/1 зависит от моментов высших порядков интервала между поступлениями.

Таблица 1

### Результаты экспериментов с использованием двух моментов

Входные параметры		Выходные характеристики		
$\rho$	$c_\lambda$	$\bar{W}$ $H_2/M/1$	$D_W$	$\bar{W}$ M/M/1
0,1	2	0,187	0,41	0,11
	4	0,230	0,51	
	6	0,241	0,54	
	8	0,245	0,55	
	10	0,247	0,55	
0,3	2	0,811	2,28	0,43
	4	1,201	3,84	
	6	1,344	4,49	
	8	1,407	4,9	
	10	1,438	4,94	
0,5	2	2,162	9,0	1,0
	4	4,831	33,0	
	6	7,602	73,0	
	8	10,402	129,0	
	10	13,213	201,0	
0,7	2	5,573	42,21	2,33
	4	17,391	337,24	
	6	36,582	1411	
	8	63,311	4135	
	10	97,629	9727	
0,9	2	22,409	547	9,0
	4	75,786	5895	
	6	164,685	27451	
	8	289,134	84177	
	10	449,135	202621	

Таблица 2

### Результаты экспериментов с использованием трех моментов

Входные параметры		Диапазоны изменения коэффициента асимметрии и времени ожидания	
$\rho$	$c_\lambda$	$A_{S_\lambda}$	$\bar{W}$
0,1	2	(4;18)	(0,34;0,13)
	4	(7;18)	(1,26;0,17)
	6	(10;18)	(3,79;0,24)
0,5	2	(4;18)	(3,14;1,23)
	4	(7;18)	(13,6;2,34)
	6	(10;18)	(32,1;7,6)
0,9	2	(4;18)	(23,5;18,3)
	4	(7;18)	(82,6;67,8)
	6	(10;18)	(182;165)

Также установлено, что классическая система  $M/M/1$ , использующая только средние значения временных интервалов, как следует из табл. 1, дает слишком оптимистичные результаты по задержкам. Поэтому полученные результаты могут быть с успехом использованы в теории современного телетрафика, чувствительного к задержкам пакетов.

*Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 15-47-02686.*

#### Список литературы

1. Клейнрок Л. Теория массового обслуживания: Пер. с англ. // Под редакцией В. И. Неймана. М.: Машиностроение, 1979. 432 с.
2. Крылов В. В., Самохвалова С. С. Теория телетрафика и ее приложения. Санкт Петербург: БХВ, 2005. 288 с.

3. Ложковский А. Г. Теория массового обслуживания в телекоммуникациях: учебник. Одесса: ОНАС им. А. С. Попова, 2012. 112 с.

4. Myskja A. An improved heuristic approximation for the  $GI/GI/1$  queue with bursty arrivals // *Teletraffic and datatraffic in a Period of Change*, ИТС-13. Elsevier Science Publishers, 1991. P. 683–688.

5. Тарасов В. Н., Карташевский И. В. Определение среднего времени ожидания требований в управляемой системе массового обслуживания  $H_2/H_2/1$  // *Системы управления и информационные технологии*. 2014. № 3 (57). С. 92–96.

6. Тарасов В. Н., Горелов Г. А., Ушаков Ю. А. Восстановление моментных характеристик распределения интервалов между пакетами входящего трафика // *Инфокоммуникационные технологии*. 2014. № 2. С. 40–44.

7. Тарасов В. Н., Бахарева Н. Ф., Горелов Г. А., Малахов С. В. Анализ входящего трафика на уровне трех моментов распределений временных интервалов // *Информационные технологии*. 2014. № 9. С. 54–59.

8. Вишнеvский В. М. Теоретические основы проектирования компьютерных сетей. М.: Техносфера, 2003. 512 с.

V. N. Tarasov, Doctor, Professor, N. F. Bahareva, Doctor, Professor,  
L. V. Lipilina, Assistant, e-mail: mila199113@gmail.com  
Volga State University of Telecommunications and Informatics

## Mathematical Model of Teletraffic on the Based $G/M/1$ System and Results of Computational Experiment

*The article presents the results of a study delay  $QS H_2/M/1$  type  $G/M/1$  for a wide range of parameters of traffic. It is known that the distribution in the law Hyperexponential  $H_2$  random variable has a coefficient of variation is greater than 1. It is also known that in the  $QS G/G/1$  waiting time due to the coefficient of variation intervals and service revenues quadratic dependence. Therefore, the waiting time in the  $QS G/M/1$  is also dependent on the coefficient of variation, of arrival intervals. Moreover, it also depends on the higher-order moments. Given the fact that the distribution of  $H_2$  is a three-parameter, the article is an approximation mechanism for arbitrary distribution laws Hyperexponential distribution. This can be done at the level of the first two moments, and at the level of the first three moments. The  $H_2/M/1$  system has that advantage before other systems with input distributions with a heavy tail that for it authors received the exact closed form solution.*

**Keywords:**  $QS H_2/M/1$ , the average waiting time in the queue, delay, Laplace transform

#### References

1. Kleinrock L. *Queueing Systems*. Russ. ed.: D. Neiman V. I. Moscow: Mashinostroenie, 1979, vol. 1: Teory, 432 p.
2. Krylov V. V., Samokhvalova S. S. *Teoriya teletrafika i ee prilozheniya* [Teletraffic Theory and its applications]. Saint Petersburg: BHV Publ., 2005, 288 p.
3. Logkovskii A. G. *Teoriya massovogo obslugivaniya v telekommunikatsiyah: ushebnyk* [Queueing theory in telecommunications. Textbook]. Odessa: Odessa National Academy of Telecommunications Publ., 2012, 112 p.
4. Myskja A. An improved heuristic approximation for the  $GI/GI/1$  queue with bursty arrivals. *Teletraffic and datatraffic in a Period of Change*, ИТС-13. Elsevier Science Publishers, 1991, pp. 683–688.
5. Tarasov V. N., Kartashevskii I. V. *Opredelenie srednego vremeni ogidaniya trebovaniy v upravlyemoy sisteme massovogo obslugivaniya  $H_2/H_2/1$*  [Determination of the average waiting time requirements in

a controlled queuing system]. *Control Systems and Information Technology*, 2014, no. 3 (57), pp. 92–96.

6. Tarasov V. N., Gorelov G. A., Ushakov Yu. A. *Vosstanovlenie momentnykh harakteristik raspredeleniya intervalov mezdu paketami vkhodyashego trafika* [Restoring torque characteristics of the distribution of intervals between packets of incoming traffic]. *Infocommunicationve Tehnologii*, 2014, no. 2, pp. 40–44.

7. Tarasov V. N., Bakhareva N. F., Gorelov G. A., Malakhov S. V. *Analiz vkhodyashego trafika na urovne trech momentov raspredelenii vremennvch intervalov* [Analyzing the incoming traffic at the three moments distribution of time intervals]. *Information Technology*, 2014, no. 9, pp. 54–59.

8. Vishnevskv V. M. *Teoretisheskie osnovy proektirovaniya komputernykh setei* [Theoretical bases of designing computer networks]. Moscow: Tehnosfera, 2003. 512 p.

# СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ CAD-SYSTEMS

УДК 004.421

**А. В. Амирханов**<sup>1</sup>, канд. физ.-мат. наук, руководитель группы,  
**А. А. Гладких**<sup>2</sup>, канд. техн. наук, доц., e-mail: gladkikhalexci@gmail.ru,  
**А. А. Глушко**<sup>2</sup>, канд. техн. наук, доц., **В. В. Макачук**<sup>2</sup>, канд. техн. наук, доц.,  
**В. А. Шахнов**<sup>\*\*</sup>, чл.-корр. РАН, зав. каф.

<sup>1</sup>Федеральное государственное бюджетное учреждение науки Научно-исследовательский институт системных исследований Российской академии наук (НИИСИ РАН),

<sup>2</sup>Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования "Московский государственный технический университет имени Н. Э. Баумана" (МГТУ им. Н. Э. Баумана)

## Автоматизированная система формирования топологической информации для изготовления комплекта фотошаблонов сверхбольших интегральных схем

*Рассмотрен подход к автоматизации процесса формирования топологической информации для производства фотошаблонов субмикронных сверхбольших интегральных схем. Разработано программное обеспечение, интегрированное с системой автоматизированного проектирования "Calibre" фирмы "Mentor Graphics".*

**Ключевые слова:** САПР, фотошаблон, литография, программное обеспечение, Python, Tcl, СБИС

### Введение

При производстве современных СБИС для создания маскирующих областей применяется процесс проекционной литографии. От его качества зависят значения электрофизических параметров структур СБИС, а значит, и ее функционирование [1].

Формирование топологического рисунка осуществляется с помощью фотошаблона (ФШ), представляющего собой изделие из оптически прозрачного материала, на поверхности которого сформирован рисунок требуемой конфигурации, состоящий из материала, не пропускающего экспонирующее излучение от системы освещения литографического оборудования. ФШ позволяет избирательно "защитить" от экспонирующего излучения слой светочувствительного материала (резиста), нанесенного на полупроводниковую пластину в целях формирования защитной маски, в свою очередь предназначенной для выполнения избирательных процессов травления, осаждения или ионного легирования.

Следует отметить, что любая ошибка, совершенная разработчиком при проектировании ФШ, может привести к неработоспособности всей изготовленной СБИС. Поэтому разработка ФШ — чрезвычайно сложная и ответственная задача. Она состоит из нескольких этапов, среди которых проектирование топологии ядра кристалла (центральная часть)

и его периферийных технологических областей, а также объединение ядра и периферии с последующим их размещением в рабочей области ФШ. Интерактивное и неавтоматизированное формирование окончательного рисунка ФШ крайне нежелательно, поскольку такая работа связана с возможными субъективными ошибками, появление которых должно быть исключено. Решение подобного рода задач является важной составляющей общей парадигмы автоматизированного проектирования СБИС [2]. Существующие в настоящее время системы автоматизированного проектирования (САПР) охватывают весьма общие средства решения задач проектирования ФШ, в то время как необходимы специализированные инструменты, позволяющие максимально автоматизировать завершающий этап разработки рисунка ФШ перед его изготовлением. Таким образом, формирование подходов и разработка эффективных средств автоматизации формирования топологической информации для изготовления ФШ является актуальной и важной задачей любого производства СБИС [3].

### Анализ топологических принципов проектирования ФШ СБИС

Обычно при проектировании ФШ его разделяют на несколько областей: па рабочее поле и на пе-

риферийные области, на которых располагается вспомогательная служебная информация, необходимая для работы литографического оборудования с этим шаблоном. В рабочем поле ФШ создается одно или несколько изображений топологического рисунка слоев СБИС, а по периферии — маркировка, включая штрих-код, служащий для его идентификации и выполняемый по установленным в литографическом оборудовании требованиям, а также знаки предварительной ориентации ФШ и метки совмещения, необходимые для его точного позиционирования.

При малых объемах выпуска СБИС в целях экономии на одном ФШ формируют сразу несколько изображений масочных слоев. В общем случае компоновка (структура рисунка) такого многослойного ФШ, содержащего несколько изображений топологических слоев СБИС, представлена на рис. 1.

Исходной информацией для формирования ФШ служат: название комплекта ФШ, вертикальный и

горизонтальный размер спроектированного кристалла СБИС (шаг мультипликации), информация о технологическом процессе, в котором он должен быть изготовлен. Помимо этого в качестве исходной информации используются полный перечень масочных слоев, их описание в виде номеров в формате GDSII [4] и номер слоя в штрих-коде ФШ, тип используемых дополнительных меток совмещения SPM (Scribe-line Primary Marks) и координаты центра каждого из слоев на ФШ. Все эти данные подготавливаются при разработке компоновки ФШ, пример которой приведен на рис. 2. В нем ФШ состоит из изображений четырех масочных слоев ACTIVE, NWELL, INACTIV, PWELL, центры которых имеют координаты  $(-5,7; 5,7)$ ,  $(5,7; 5,7)$ ,  $(-5,7; -5,7)$ ,  $(5,7; -5,7)$ .

При разработке компоновки ФШ следует учитывать ограничения литографического оборудования. Например, максимальные размеры поля экспонирования не могут превышать размеры  $X_{\max}$ ,  $Y_{\max}$ , а расстояние между изображениями не должно быть менее  $D$  (см. рис. 1). Следует также учитывать коэффициент уменьшения проекционного объектива литографического оборудования.

Пусть шаги мультипликации по осям абсцисс и ординат составляют соответственно  $X$  и  $Y$ . Тогда максимальное число изображений топологических слоев кристалла СБИС  $n_X$  и  $n_Y$  на одном ФШ по осям  $Ox$  и  $Oy$  будет определяться из ограничений

$$\begin{aligned} n_X X + (n_X - 1)D &\leq X_{\max}; \\ n_Y Y + (n_Y - 1)D &\leq Y_{\max}. \end{aligned} \quad (1)$$

Отсюда следует, что число изображений  $n_X$  и  $n_Y$  на одном ФШ должно выбираться исходя из следующих условий:

$$n_X \leq \frac{X_{\max} + D}{X + D}, \quad n_Y \leq \frac{Y_{\max} + D}{Y + D}.$$

Для определения области допустимых значений координат центров изображений введем по осям  $Ox$  и  $Oy$  шаги изменения их изображений  $dX_C$  и  $dY_C$ , а через  $X_C$  и  $Y_C$  обозначим минимальные положительные координаты центров. Будем считать, что они располагаются симметрично относительно центральной точки ФШ (хотя, в общем случае, это и не обязательно).

Условия-ограничения (1) и ограничения на расстояние между изображениями будут записываться следующим образом:

$$\begin{aligned} (n_X - 1)dX_C + X &\leq X_{\max}; \\ (n_Y - 1)dY_C + Y &\leq Y_{\max}; \\ dX_C - X &\geq D; \quad dY_C - Y &\geq D. \end{aligned}$$

При таком подходе к распределению площади ФШ важно для всех слоев выбрать координаты их центров на ФШ. Следует отметить, что сдвиг топологического слоя кристалла СБИС, даже около 1 мкм (при общем размере рабочей области ФШ порядка

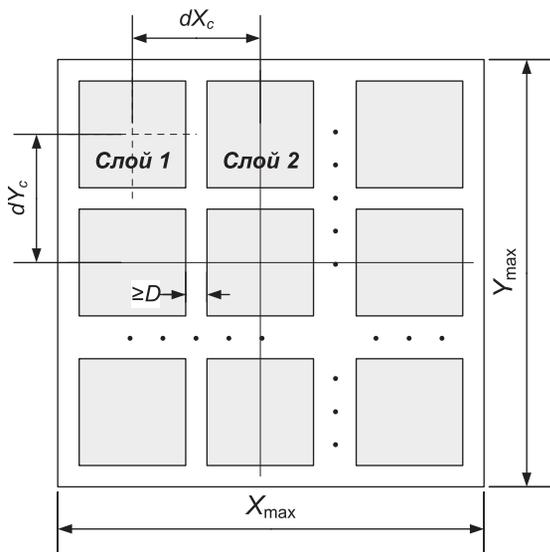


Рис. 1. Компоновка многослойного ФШ

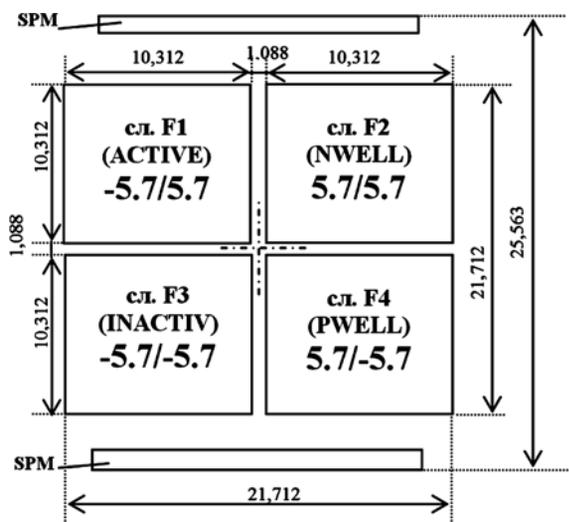


Рис. 2. Пример компоновки многослойного ФШ

20 мм), в результате использования интерактивного и неавтоматизированного формирования рабочей области ФШ приведет к рассовмещению при изготовлении СБИС и, следовательно, к ее полной неработоспособности. Визуально отследить такую ошибку практически невозможно, поскольку само определение координаты центра изображения трудоемко и сопряжено с возможными субъективными погрешностями.

Поэтому в связи с изложенным выше процесс размещения изображений на ФШ желательно полностью автоматизировать, исключив тем самым человеческий фактор. Для этого необходимо в используемую САПР, например, такую как "Calibre" фирмы "Mentor Graphics" [5], разработать и встроить программные модули формирования топологической информации комплекта ФШ из отдельных слоев кристалла СБИС в соответствии с заданной компоновкой. Помимо сказанного компоновка ФШ должна быть формализована в виде реляционной базы данных (БД) в целях удобства обработки информации и ее использования разработанными программными модулями. Кроме того, разработка специализированных программных модулей позволит в дальнейшем провести автоматизацию документооборота в части генерации чертежей так, как показано на рис. 2, и иной конструкторской документации, связанной с разработкой ФШ.

#### **Анализ и выбор средств автоматизации формирования топологической информации для изготовления ФШ СБИС**

В работе [6] был проведен анализ возможности использования объектно-ориентированного языка программирования совместно с САПР "Calibre" фирмы "Mentor Graphics" [5]. Кроме этого, был обоснован выбор языка программирования Python [7] и САПР "Mentor Graphics Calibre" для решения задачи автоматизированной подготовки топологической информации. Кратко рассмотрим основные аспекты предложенного в работе [6] подхода. "Calibre WORKbench" (один из основных продуктов фирмы "Mentor Graphics" в области обработки топологической информации) имеет специальный режим командной строки (называемый batch-mode), который позволяет выполнять все доступные из графического интерфейса пользователя команды редактирования топологии.

Изначально вызов команд обработки топологической информации осуществляется с помощью встроенного в Calibre WORKbench языка программирования Tcl (Tool command language). Благодаря встроенному в САПР "Mentor Graphics Calibre WORKbench" интерпретатору Tcl возможна разработка различных расширений, позволяющих в значительной степени автоматизировать как сам процесс проектирования СБИС, так и подготовку к ее производству.

Однако программы такого типа должны обладать высокой степенью надежности и быть легко поддерживаемы, поскольку решаемые в данном случае задачи требуют большой ответственности, и все ошибки при их реализации ведут к неприемлемым финансовым и временным затратам.

Многие высокоуровневые языки программирования построены по схожей архитектуре и представляют собой виртуальные машины, которые выполняют так называемый байт-код, транслированный из исходного кода программы. Очевидно, что его выполнение внутри виртуальной машины имеет более низкое быстродействие, нежели исполнение машинного кода. Поэтому в целях увеличения быстродействия таких языков высокого уровня в них предусматривается возможность расширения с помощью низкоуровневых языков. Для интеграции языка программирования Python в САПР "Calibre WORKbench" была использована именно эта особенность интерпретатора Tcl. В итоге для разработчика в интерпретаторе языка программирования Python становится доступен модуль calibre.py, с помощью которого возможно обращение к командам, классам и методам, предоставляемым САПР "Calibre WORKbench". В настоящий момент в этом модуле реализованы обработчики только тех классов и методов, которые необходимы для создания программного обеспечения автоматизированной подготовки топологической информации при производстве ФШ.

#### **Разработка системы автоматизации формирования топологической информации для изготовления ФШ СБИС**

Поскольку сборка топологической информации в соответствии с компоновкой является, как уже упоминалось, трудоемкой задачей, в которой не допускаются какие-либо ошибки, актуальной является проблема автоматизированной сборки топологической информации для изготовления комплекта ФШ. В этом случае его компоновка должна быть формализована. Наиболее удобным способом решения этой задачи является представление информации о расположении изображений топологических слоев (компоновки) в виде БД. Поскольку от этой БД не требуется обеспечения хранения большого объема информации и высокой скорости ее выборки, то предпочтительно использовать относительно "простую" в применении систему управления базами данных (СУБД), например SQLite [8]. Использование такой СУБД не требует разворачивания отдельной программной службы — сервера управления базами. В этом случае вся СУБД SQLite напрямую встраивается в виде, например, динамически загружаемой библиотеки в основное приложение. Все это позволяет использовать формализованное хранение важной информации. Структура конфигурационной базы данных (КБД) для хранения ин-



Рис. 3. Структура БД автоматизированной подготовки топологической информации для изготовления комплекта ФШ

Таблица 1

Описание структур таблицы БД

Ключ	Имя поля	Тип	Описание поля
<b>Структура таблицы Projects</b>			
PK*	name	text	Название комплекта ФШ
	size_x	real	Размер кристалла по оси X
	size_y	real	Размер кристалла по оси Y
<b>Структура таблицы Masks</b>			
PK	number	integer	Порядковый номер ФШ в комплекте
	gds	integer	Номер результирующего масочного GDSII-слоя
FK*	spm_name	text	Имя SPM, которая должна быть размещена на ФШ
FK	project_name	text	Название комплекта, к которому относится данный ФШ
<b>Структура таблицы Layers</b>			
PK	id	integer	Идентификатор записи
	name	text	Текстовое название слоя
	number	integer	Порядковый номер слоя
	gds	integer	GDSII номер слоя
	center_x	real	Центр слоя по оси X
	center_y	real	Центр слоя по оси Y
FK	project_name	text	Название комплекта ФШ, к которому относится слой
FK	mask_number	integer	Порядковый номер ФШ в комплекте, к которому относится топологический слой
<b>Структура таблицы SPM</b>			
PK	name	text	Название (тип) структуры SPM
	desc	text	Текстовое описание структуры SPM
	gds	integer	Номер слоя GDSII
	md5	text	Контрольная сумма топологической структуры SPM
	data	blob	Данные в формате GDSII топологической структуры SPM
<b>Структура таблицы Log</b>			
PK	id	integer	Идентификатор записи
	user	text	Пользователь, выполнивший команды для обработки БД
	date	text	Дата выполнения команды
	command	text	Строка выполненной команды
<b>Структура таблицы Info</b>			
PK	id	integer	Идентификатор записи
	version	integer	Номер версии формата представленной БД
*PK — первичный ключ, FK — внешний ключ			

формации о компоновке ФШ кристаллов СБИС приведена на рис. 3.

КБД содержит шесть таблиц, где четыре таблицы (Projects, Masks, Layers и SPM) связаны между собой, а две таблицы (Log и Info) — вспомогательные. Их описание представлено в табл. 1.

**Разработка алгоритма автоматизации формирования топологической информации ФШ СБИС.** Блок-схема алгоритма точки входа программного обеспечения автоматизированного формирования топологической информации ФШ СБИС приведена на рис. 4. Из нее видно, что первым шагом алгоритма является считывание КБД в оперативную память в структурированном виде. Для загрузки данных используется библиотека storm [9] ORM (Object-Relational Mapping) [10]. После выполнения этой функции на ее выходе будет получен специальный объект, в полях которого сохранены основные конфигурационные параметры будущих ФШ. Более того, этот объект обеспечивает связь с внутренней структурой КБД, поэтому необходимость разработки специальных SQL-запросов отпадает (так как эта функциональность ложится полностью на систему ORM).

Алгоритм допускает выполнение по двум различным веткам в зависимости от параметра options.export\_database, передаваемого через внешние опциональные настройки. Этот параметр имеет тип булевой переменной. При его "истинном" значении программное обеспечение будет выполнять преобразование КБД в текстовый формат, а в случае "ложного" — будет вызывать функцию CalibreCreateMasks подготовки топологической информации о комплекте ФШ согласно компоновке (по КБД).

Рассмотрим подробнее основную функцию CalibreCreateMasks разрабатываемого программного обеспечения. Блок-схема алгоритма ее работы приведена на рис. 5.

Как видно из рисунка, большая часть алгоритма работы этой функции является линейной. Первоначально она создает объект топологии в памяти САПР Calibre WORKbench, в котором будут храниться все данные на протяжении ее работы. Далее

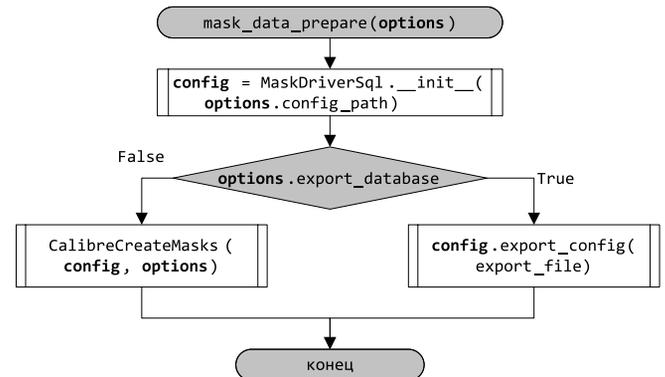


Рис. 4. Алгоритм программы автоматизированного формирования топологической информации ФШ СБИС

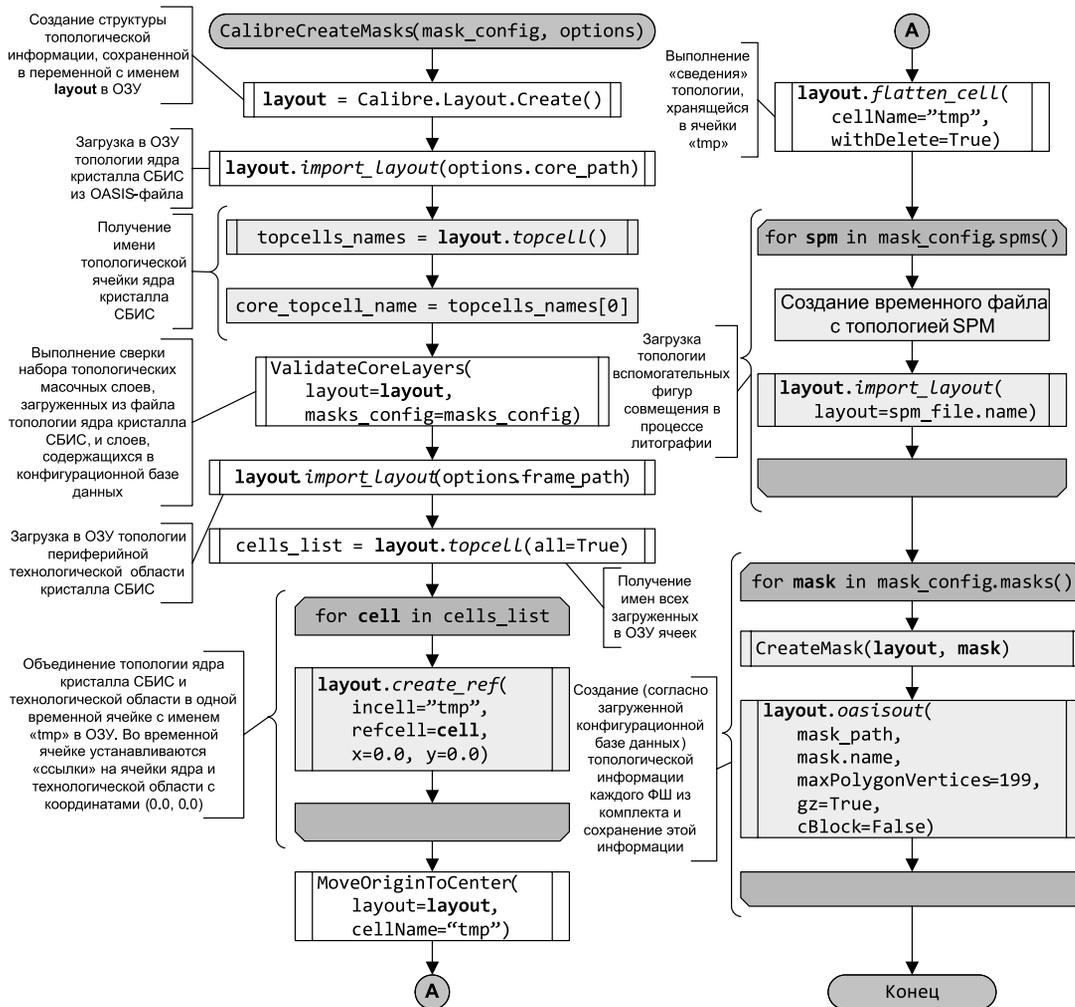


Рис. 5. Блок-схема алгоритма формирования топологической информации по всем фотошаблонам СБИС согласно КБД

выполняется загрузка топологии ядра кристалла СБИС и периферийной технологической области. Следующий шаг заключается в проверке соответствия информации о топологических слоях из КБД и загруженной информации из файла топологии ядра кристалла СБИС. После успешного прохождения проверки слоев программное обеспечение объединяет ячейки топологической информации ядра кристалла и технологической области в одну ячейку и выполняет ее "сведение".

Для понимания операции "сведения" необходимо рассмотреть варианты хранения топологической информации. Она может быть представлена в двух вариантах, назовем их иерархическим и геометрическим. В случае геометрического представления топология строится из обычных двумерных геометрических объектов — многоугольников, называемых полигонами. При иерархическом представлении топология, построенная из полигонов, разбивается на отдельные топологические ячейки, которые могут быть вставлены одна в другую, вследствие чего образуется иерархия ячеек в топологии. Это позволяет сократить объем обрабатываемых данных за счет

того, что дублирующиеся полигоны заменяются ссылкой на отдельную ячейку.

Операция "сведения" выполняет замещение всех установленных топологических ячеек содержащихся в них геометрическим объектами. Ее применение в данной функции необходимо для упрощения работы с топологией на последующих этапах. Кроме того, это является требованием к топологической информации рабочей области ФШ при его изготовлении. Завершающими шагами алгоритма являются следующие операции:

- чтение топологической информации структур SPM из КБД;
- окончательное формирование для ФШ топологической информации с последующим сохранением ее в выходной файл в сжатом GZIP-формате GDSII или OASIS [11], что позволяет сократить объем памяти, необходимый для ее хранения.

Как это видно из блок-схемы алгоритма, функция, которая отвечает за формирование топологической информации отдельного ФШ согласно КБД, называется CreateMask, а ее блок-схема приведена на рис. 6.

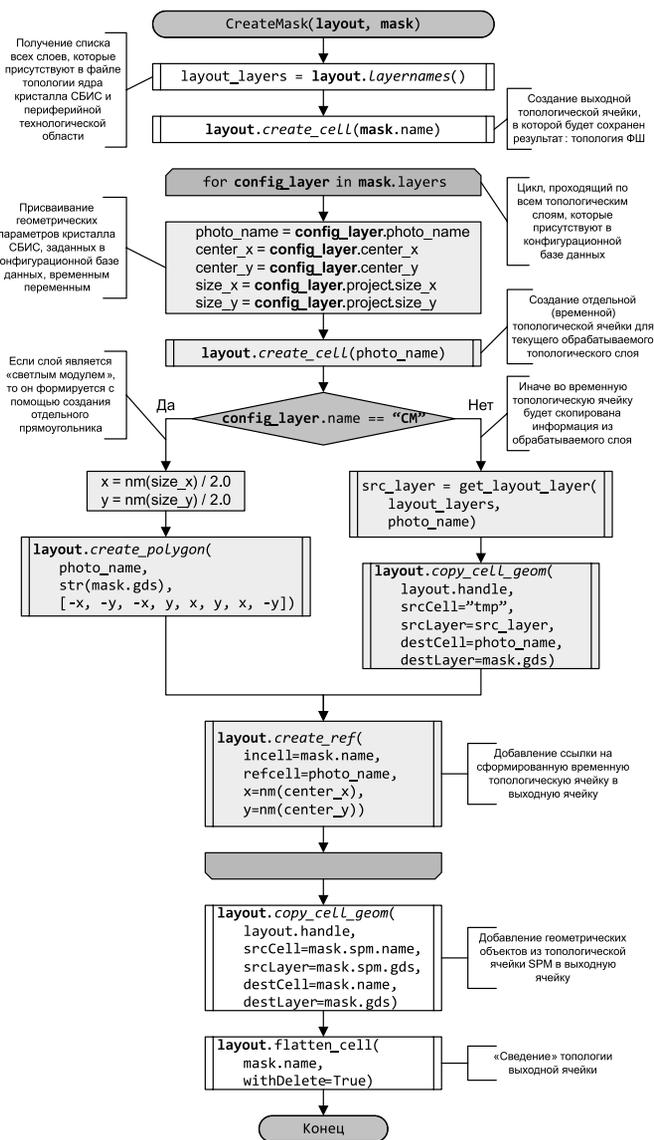


Рис. 6. Алгоритм формирования топологической информации отдельного ФШ согласно БЧД

По сути, она выполняет задачу поочередного копирования информации о каждом слое в отдельную топологическую ячейку и помещение ее в выходную топологическую ячейку ФШ. С этой целью первоначально строится список всех слоев, присутствующих в топологии ядра кристалла СБИС, а затем создается отдельная ячейка в выходном файле, которая представляет собой непосредственно топологию обрабатываемого ФШ. В созданной ячейке происходит добавление ячеек, относящихся к конкретным топологическим слоям. Это реализуется с помощью последовательной обработки "конфигурационных слоев" из БД, относящихся к текущему ФШ.

Основная часть обработки заключается в том, чтобы выполнить проверку, является ли добавляемый слой обычным топологическим слоем из ядра кристалла СБИС или это вспомогательный технологический слой, называемый "светлый модуль" (в алгоритме представлен как CM). В этом случае на данное место добавляется прямоугольник с заданным размером, равным по размеру кристаллу СБИС. В противном случае в созданную ранее ячейку топологического слоя выполняется копирование всех геометрических объектов из топологии кристалла СБИС. На заключительном этапе, после того как были сформированы ячейки для всех топологических слоев на обрабатываемом ФШ, проводится так называемое "сведение" всех объектов в одну топологическую ячейку, а созданные ранее ячейки топологических слоев после "сведения" удаляются.

### Тестирование и оценка разработанного подхода

Разработанный алгоритм и программное обеспечение, интегрированное с САПР "Calibre WORKbench" с помощью динамически подключаемых модулей, были протестированы для различных комплектов ФШ СБИС. Проведенное сравнение выходных данных разработанного программного обеспечения (набор файлов в формате OASIS) с ранее разработанными комплектами ФШ показало отсутствие каких-либо ошибок. Сравнение проводилось путем выполнения логической операции XOR (исключающее "ИЛИ") между топологической информацией, полученной с помощью разработанного автоматизированного подхода и с помощью интерактивного и неавтоматизированного подхода.

Кроме того, была проведена оценка временных затрат на формирование комплектов ФШ для кристаллов СБИС с использованием предложенного автоматизированного подхода и разработанного программного обеспечения. Полученные результаты приведены в табл. 2 (тестирование проводилось на рабочей станции со следующей конфигурацией: AMD Opteron Processor 6234 2.4 ГГц, ОЗУ 128 Гб).

Как видно из приведенных данных, полученных при использовании рабочей станции с указанной конфигурацией, время обработки топологии для

Таблица 2

#### Оценка времени формирования топологической информации

Число полигонов/прямоугольников	Размер ядра oas.gz, Мбайт	Число масок/слоев на маске	Общее число слоев	Время обработки, мин	Минимальная проектная норма, мкм
47/154 млн	1272	14/2	27	30	0,25 ОРС*
33/113 млн	1075	8/4	27	23	0,25 ОРС*
5,5/34 млн	203	10/2	19	4,5	0,35
1/10 млн	37	2/9	18	1,1	0,50
1/8 млн	22	3/9	19	1,0	0,50

\*ОРС — ядро кристалла СБИС было обработано по методу Model-based Optical Proximity Correction (Коррекция эффекта оптической близости на основе моделирования).

формирования информации, необходимой для изготовления комплекта ФШ СБИС, лежит в пределах 30 мин.

### Заключение

Разработанный подход и программное обеспечение, интегрированное с САПР "Mentor Graphics Calibre", для автоматизации формирования топологической информации комплекта ФШ СБИС позволяют практически исключить возможность возникновения субъективных ошибок во время обработки топологической информации на заключительных этапах проектирования, а также снизить время, затрачиваемое на подготовку топологии кристалла СБИС к производству.

Работа выполнена при частичной финансовой поддержке программы Президента РФ по государственной поддержке ведущих научных школ (грант НШ-2903.2014.9) и Российского фонда фундаментальных исследований (грант № 15-07-03116\_a).

### Список литературы

1. **Киреев В. Ю.** Введение в технологии микроэлектроники и нанотехнологии. М.: ФГУП "ЦНИИХМ", 2008. 428 с.
2. **Амирханов А. В., Гладких А. А., Глушко А. А., Макаручук В. В., Новоселов А. С., Родионов И. А., Шахнов В. А.** Разработка парадигмы проектирования СБИС с учетом результатов конструкторско-технологического моделирования // Датчики и Системы. 2013. № 9. С. 38–51.
3. **Глушко А. А., Гладких А. А., Зотов С. К.** Алгоритм оптимизации сложной топологии элементов СБИС // Тр. НИИСИ РАН. Математическое и компьютерное моделирование систем: прикладные аспекты. М.: НИИСИ РАН, 2011. Т. 1. № 1. С. 4–5.
4. **GDSII™** Stream Format. Calma Company. 1987. 47 p.
5. **URL:** <http://www.mentor.com/products/ic-manufacturing/>
6. **Гладких А. А.** Использование языка программирования Python для расширения системы автоматизированного проектирования // Вестник Екатеринбургского института. 2014. № 3 (27). С. 7–16.
7. **URL:** <https://www.python.org/>
8. **Kreibich Jay A.** Using SQLite. O'Reilly Media, Inc., 2009, 528 p.
9. **URL:** <https://storm.canonical.com/>
10. **Ireland C., Bowers D., Newton M., Waugh K.** Understanding Object-Relational Mapping: A Framework Based Approach. The Open University, Milton Keynes, UK // International Journal on Advances in Software. 2009. Vol. 2, N. 2 & 3.
11. **SEMI P39-0308 — OASIS™** — Open Artwork System Interchange Standard. 2008.

**A. V. Amirkhanov**, Head of Department of Scientific,

**A. A. Gladkikh**, Associate Professor, e-mail: [gladkikhalexei@mail.com](mailto:gladkikhalexei@mail.com), **A. A. Glushko**, Associate Professor,

**V. V. Makarchuk**, Associate Professor, **V. A. Shakhnov**, Head of Chair,

Scientific Research Institute of System Analysis (SRISA RAS)

## Computer Aided Design System for Photomask Workspace Layout Generation of Very Large Scale Integrated Circuit

*An approach of computer aided design (CAD) photo mask workspace layout generation process for very large scale integrated circuit (VLSI) is considered in this paper. The problems of automated CAD approaches lack that occurred during photomask layout design showed. Analysis of photomask construction and workspace designing was considered. Types and photolayers arrangement within photomask workspace is determined. CAD Software architecture and structure development is considered. The main developed software algorithms block-schemes are exposed. Configuration database format and structure for mask layers store of VLSI project arrangement within mask workspace are showed. The developed software integration with Mentor Graphics Calibre CAD system considered. Complex testing of developed CAD software for different types of VLSI layouts (microprocessor and static memory) was done. Analyses of obtained testing results were done. Also timings required for workspace layout generation using developed CAD software were estimated.*

**Keywords:** CAD, Photomask, lithography, software, Tcl, Python, VLSI

### References

1. **Kireev V. Ju.** Vvedenie v tehnologii mikrojelektroniki i nanotehnologii. Moscow: FGUP "ЦНИИХМ", 2008. 428 p.
2. **Amirhanov A. V., Gladkikh A. A., Glushko A. A., Makarchuk V. V., Novoselov A. S., Rodionov I. A., Shakhnov V. A.** Razrabotka paradigmy proektirovaniya SBIS s uchtom rezul'tatov konstruktorsko-tehnologicheskogo modelirovaniya. *Datchiki i Sistemy*, 2013, no. 9, pp. 38–51.
3. **Glushko A. A., Gladkikh A. A., Zotov S. K.** Algoritim optimizatsii slozhnoj topologii jelementov SBIS. *Tr. NIISI RAN. Matematicheskoe i komp'yuternoe modelirovanie sistem: prikladnye aspekty*. Moscow: Izdatel'stvo "NIISI RAN", 2011, vol. 1, no. 1, pp. 4–5.
4. **GDSII™** Stream Format. Calma Company. 1987. 47 p.
5. **URL:** <http://www.mentor.com/products/ic-manufacturing/>
6. **Gladkikh A. A.** Ispol'zovanie jazyka programirovaniya Python dlja rasshirenija sistemy avtomatizirovannogo proektirovaniya. *Vestnik Ekaterininskogo instituta*, 2014, no. 3 (27), pp. 7–16.
7. **https://www.python.org/**
8. **Kreibich Jay A.** Using SQLite. O'Reilly Media, Inc., 2009. 528 p.
9. **URL:** <https://storm.canonical.com/>
10. **Ireland C., Bowers D., Newton M., Waugh K.** Understanding Object-Relational Mapping: A Framework Based Approach. The Open University, Milton Keynes, UK. *International Journal on Advances in Software*, 2009, vol. 2, no. 2 & 3.
11. **SEMI P39-0308 — OASIS™** — Open Artwork System Interchange Standard.

**В. Н. Гридин**<sup>1</sup>, д-р техн. наук, проф., директор, e-mail: info2@dict.ras.ru,  
**В. И. Анисимов**<sup>1</sup>, д-р техн. наук, проф., e-mail: ailaristov@inbox.ru,  
**Г. Д. Дмитриевич**<sup>2</sup>, д-р техн. наук, проф., **А. И. Ларистов**<sup>2</sup>, канд. техн. наук, доц.,  
**Я. М. Аль-Шамери**<sup>2</sup>, аспирант

<sup>1</sup>Центр информационных технологий в проектировании, г. Одинцово, МО

<sup>2</sup>Санкт-Петербургский государственный электротехнический университет (ЛЭТИ)

## Организация информационного обеспечения web-ориентированных схемотехнических САПР

*Рассматриваются вопросы организации информационного обеспечения web-ориентированных схемотехнических САПР на основе технологий баз данных. Анализируется состав проектных данных схемотехнических САПР. Предлагается сформировать распределенную структуру информационного обеспечения web-ориентированных САПР, в основе которой лежит использование единого проектного пространства на Web-сервере Интернет-ресурсов САПР. Взаимодействие пользователей с Web-сервером Интернет-ресурсов САПР возможно с помощью встроенного браузера и серверных web-приложений доступа к базам данных. Рассматривается состав баз данных, входящих в единое информационное пространство Web-сервера, и предлагаются методы для реализации его составляющих.*

**Ключевые слова:** web-ориентированные схемотехнические САПР, информационное обеспечение САПР, технологии баз данных, Web-сервер Интернет-ресурсов САПР

### Введение

В настоящее время технология проектирования электронных схем полностью ориентирована на сквозное применение САПР на всех стадиях разработки, начиная от этапа получения и анализа принципиальной схемы устройства и заканчивая этапом топологического проектирования. При этом современные интегрированные САПР в области проектирования РЭА (EDA — Electronic Design Automation) широко представлены на рынке [1]. Среди наиболее широко используемых в настоящее время EDA следует отметить зарубежные системы компаний *Cadence*, *Mentor Graphics*, *Zuken*, *Atium* и *Synopsys*, предлагающие средства проектирования СБИС, в том числе и систем на кристалле. На российском рынке большой популярностью пользуется система *Altium Designer*, наследующая технологии системы P-CAD. Успешно развиваются системы *ToroR* и *SimOne* отечественных разработчиков компании "Эремекс" [2].

Следует отметить, что стоимость лицензий на использование этих систем достаточно высока и составляет от нескольких десятков до сотен тысяч долларов и, по мере усложнения этих систем, она будет только возрастать. При этом алгоритмическое и программное обеспечение систем является одной из основных составляющих САПР, определяющих их стоимость в настоящее время. Кроме того, для обеспечения проектирования цифровых электронных схем большой степени сложности (более нескольких сот компонентов при нескольких тысячах контактов и/или трасс) при использовании таких систем, как *Specetra*, *Expedition* или *Synopsys*, требуются высокопроизводительные серверные платформы, стоимость которых сопоставима

со стоимостью программного обеспечения САПР. В этой связи приобретение современных САПР, их официальное использование доступно только для крупных корпоративных пользователей, широко практикующих разработку современной радиоэлектронной и электронно-вычислительной аппаратуры. Для пользователей, имеющих относительно небольшой объем разработки мелкосерийной электронной аппаратуры, покупка лицензий таких САПР оказывается экономически невыгодной.

Одним из возможных решений данной проблемы является использование web-ориентированных версий интегрированных САПР с облегченным режимом лицензирования. Перемещение производственных систем (CAD, CAE, PLM, ...), используемых в процессе проектирования, разработки технологий и производства радиоэлектронной аппаратуры, в Интернет-среду может развиваться в следующих направлениях [3, 4, 10]:

- создание нового класса распределенных прикладных систем на основе web-технологий;
- перенос в Интернет существующих приложений и средств хранения данных с использованием облачных технологий.

Применительно к схемотехническим САПР каждое из направлений имеет значительные различия и особенности. Однако общим для решения данной проблемы является процесс перестройки информационного фонда САПР на платформу баз данных и разработка web-ориентированных версий специализированных СУБД. При этом в рамках каждого направления накладываются определенные ограничения и требования к построению информационного обеспечения САПР.

## Анализ составляющих информационного фонда схмотехнических САПР

Процесс проектирования объекта можно представить как связанную совокупность этапов преобразования одних данных в другие. При этом данные, полученные в результате одного процесса преобразования, могут быть исходными для другого процесса (промежуточные данные). Это приводит к различным трактовкам термина "данные" в САПР. Так, для проектирующих подсистем к данным относится совокупность исходных и результирующих чисел, необходимых для выполнения конкретной проектной процедуры; пользователю САПР в качестве данных требуется иметь в своем распоряжении исходную проектную документацию, справочные данные, типовые проектные решения и т. д.

Основными задачами при построении информационного обеспечения Web-ориентированных схмотехнических САПР являются: определение состава информационного фонда с учетом функционального назначения используемой информации; распределение компонентов информационного фонда между рабочей станцией и web-сервером Интернет ресурсов САПР; разработка инфологических моделей данных на семантическом уровне для определения структуры компонентов информационного фонда; переход от инфологических моделей к даталогическим моделям и формирование структур хранения данных; выбор инструментальных средств для организации хранения компонентов информационного фонда на основе технологий баз данных; определение оптимальных методов доступа к данным, учитывая распределенный характер их размещения в сети Интернет; разработка систем управления базами данных схмотехнических САПР.

Определим состав информационного фонда. В зависимости от назначения информации можно выделить следующие компоненты информационного фонда схмотехнических САПР:

Компонент *А*. Информация о структуре проектируемой схемы и ее внешних выводах.

Компонент *В*. Информация о структуре моделей схемных компонентов, входящих в состав схемы, и значениях параметров их моделей.

Компонент *С*. Исходные данные, определяющие задание на моделирование схемы и условия выполнения проектных операций и процедур.

Компонент *Д*. Выходные данные, являющиеся результатом выполнения проектной процедуры или операции (результаты расчетов). Эти данные обновляются для каждой проектируемой схемы.

Компонент *Е*. Текущая и итоговая проектная документация отражает состояние и ход проектирования схемы. Эта группа данных слабо структурирована и существует в форме текстовых документов и чертежей.

Компонент *Ф*. Нормативно-справочная проектная документация (НСПД). В состав этой группы данных входят справочные данные о материалах, элементах схем, унифицированных узлах и конструкциях.

Компонент *Г*. Государственные и отраслевые стандарты, регламентирующие документы, руководящие материалы и указания.

Компонент *Н*. Готовые проекты и типовые проектные решения, полученные с помощью САПР.

Следует отметить, что для непосредственного моделирования электронной схемы необходимы только те данные, которые считываются и выводятся программными модулями системы. К ним относятся компоненты данных *А*, *В*, *С* и *Д*. Компоненты данных *Ф*, *Г* и *Н* представляют собой комплекты документов, используемые инженером-схмотехником в процессе синтеза принципиальной схемы. Готовые проектные решения соответствуют компоненту данных *Е*. Эти данные представляют собой полный комплект проектных документов на разрабатываемую схему в соответствии с Единой системой конструкторской документации (ЕСКД). Источником формирования компонента данных *Е* могут служить наряду со схмотехническими САПР также конструкторские и технологические САПР. Часть документов этого вида может быть получена с помощью офисных приложений (MS Word, MS Excel, Adobe Acrobat и др.). Совокупность всех данных компонентов *А*, *В*, *С*, *Д* и *Е*, относящихся к одному проекту, является единицей хранения для компонента данных *Н*. Этот компонент содержит описание проектируемой схемы, файлы результатов моделирования и полный комплект проектной документации на разрабатываемую схему.

## Базы проектных данных схмотехнических САПР

Проблему организации и ведения информационного фонда схмотехнических САПР можно рассматривать в содержательном и организационном аспектах. Содержательный аспект информации, используемой при проектировании, полностью определяется принятой методикой проектирования, структурой программного обеспечения, разработанными алгоритмами решения проектных задач [5]. Организационный аспект определяет технологии работы с данными, способы хранения и ведения информационного фонда. Наиболее универсальным является использование специализированных и универсальных СУБД для организации информационного обеспечения САПР [6]. Данный способ позволяет: выполнить структурирование данных в виде, удобном для проектировщика; выделить коллективно используемые данные и сформировать централизованный информационный фонд САПР; обеспечить поиск нормативно-справочной и проектной документации. Кроме того, при таком способе хранения данных обеспечивается централизованное копирование и восстановление данных, разграничение доступа к данным, возможность доступа к данным в сети Интернет. Универсальность данного подхода позволяет использовать СУБД для хранения практически всех компонентов информационного фонда САПР.

Следует отметить, что в настоящее время в САПР наиболее часто применяют специализированные СУБД [7], создаваемые одновременно с разработкой программного обеспечения САПР. Такие СУБД ориентированы, как правило, на обеспечение информацией проектирующих систем и не содержат нормативно-справочных и архивных проектных документов. Применение универсальных СУБД в САПР позволяет наиболее полно реализовать все функции по ведению информационного фонда САПР.

В соответствии с функциональным назначением компонентов информационного фонда можно выделить следующий перечень баз данных.

*Базы данных, непосредственно используемые в процессе функционирования САПР.*

- База данных сеанса проектирования (БДСП) — содержит компоненты данных *A, C, D, E* (исходное описание схемы; входные и результирующие данные, необходимые при выполнении программных модулей). Следует отметить, что содержимое БДСП соответствует электронной схеме, моделируемой в рамках текущего сеанса проектирования, и полностью обновляется при переходе к моделированию другой схемы. Исходное описание схемы и результаты моделирования могут быть сохранены в файлах для последующего использования в рамках нового сеанса проектирования или переданы на долговременное хранение в архив проектных решений.
- База данных моделей компонентов (БДМК) — информация о структуре моделей схемных компонентов, входящих в состав проектируемой схемы, и значениях параметров их моделей (компонент данных *B*). Данная информация является общей для всех проектов, выполняемых в САПР, и изменяется достаточно редко (только при добавлении новых схемных компонентов и их моделей).

Перечисленные базы данных (библиотеки данных) в той или иной форме присутствуют практически во всех современных схемотехнических САПР [8]. В дополнение к ним в состав САПР необходимо включить *базы данных, ориентированные на хранение проектных документов и окончательных результатов проектирования*. Эти базы данных непосредственно не связаны с проектирующими подсистемами САПР и могут использоваться автономно в рамках подсистемы хранения проектных данных. Они предназначены для хранения компонентов данных *E, F, G, H, K*.

- База данных нормативно-справочной проектной документации (БДНСПД) — справочные данные о материалах, элементах схем, унифицированных узлах и конструкциях; государственные и отраслевые стандарты; руководящие материалы и указания. Хранящиеся в БДНСПД сведения используют во всех проектах, выполняющихся в САПР, и имеют малую частоту обновления.
- База данных рабочих проектов (БДРП) — входные данные, результаты моделирования и текущая

проектная документация, отражающая состояние и ход выполнения проекта. Хранящиеся данные должны быть привязаны к конкретным проектам и обновляться в процессе выполнения очередного сеанса проектирования. Процессом пополнения БДРП управляет инженер-схемотехник, используя в качестве источников данные из БДСП и файлы документов, полученные с помощью внешних приложений (Word, Excel и др.).

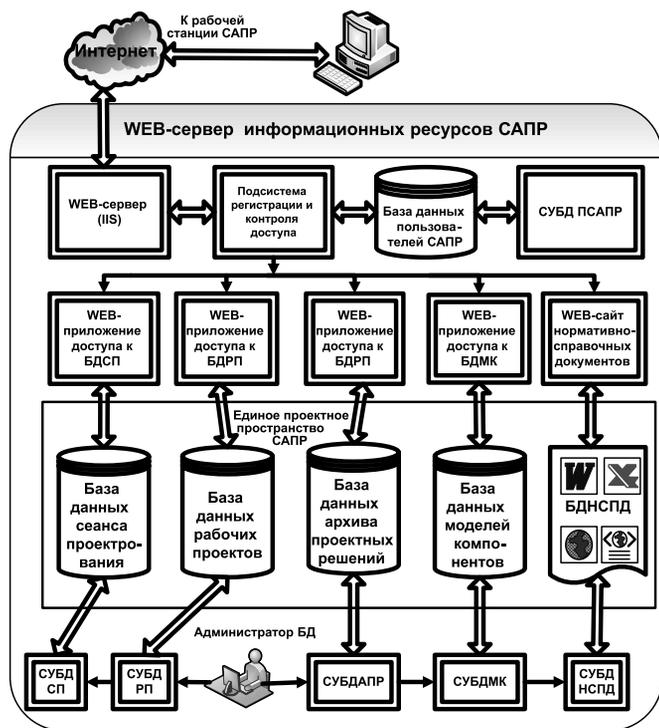
- База данных архива проектных решений (БДАПР) — законченные и типовые проектные решения, выполненные в САПР. Информация из БДПР, подлежащая долговременному хранению, передается в БДАПР. При этом должна быть реализована возможность поиска и извлечения данных, относящихся к конкретному проекту.

Основой для переноса процесса автоматизированного проектирования РЭА в Интернет может служить создание единого проектного пространства САПР (ЕПП САПР) и размещение его в глобальной сети на Web-сервере(-ах) информационных ресурсов САПР (СИР САПР). При этом основными требованиями к распределенной структуре единого проектного пространства САПР являются:

- обеспечение коллективной работы пользователей САПР в сети Интернет над общим проектом, выполняемым в САПР;
- наличие оперативного доступа из среды САПР к централизованным базам данных схемных компонентов;
- возможность сохранения на Web-сервере информационных ресурсов САПР текущих проектных данных, полученных на рабочей станции, и последующее их использование в рамках следующего сеанса проектирования;
- использование в проекте единых нормативных и справочных баз для всех пользователей САПР, работающих над проектом;
- обеспечение формирования единого архива проектных решений, доступного для всех пользователей САПР;
- протоколирование сеансов доступа пользователей САПР к базам данных и изменений, вносимых в проектные данные.

На основе анализа перечисленных требований и состава баз данных предлагается сформировать распределенную структуру информационного обеспечения Web-ориентированной САПР, представленную на рисунке. Особенностью предложенной архитектуры является использование единого проектного пространства на Web-сервере Интернет-ресурсов САПР. Взаимодействие пользователей с Web-сервером Интернет-ресурсов САПР возможно с помощью встроенного браузера и серверных web-приложений доступа к базам данных. При этом пользователям необходимо выполнить процедуру авторизации на сервере.

Следует отметить, что базы данных, ориентированные на хранение документов, практически отсутствуют в составе современных схемотехниче-



Базы данных схемотехнических САПР

ских САПР. Не решены полностью проблемы формирования архивов проектных решений и систем поддержки коллективной работы над проектами. Встречаются отдельные реализации фирменных электронных справочников по схемным компонентам в сети Интернет, однако их содержимое относится, как правило, к изделиям одного производителя. Таким образом, в процессе автоматизирован-

ного проектирования пользователь САПР вынужден обращаться к разнородным источникам информации и использовать традиционные справочники на бумажном носителе для поиска и отбора технической информации и, кроме того, самостоятельно обеспечивать формирование и хранение проектных данных, полученных в результате взаимодействия с САПР. Для решения перечисленных проблем предлагается включить в состав единого информационного пространства САПР (ЕИП САПР) централизованную базу данных моделей компонентов и подсистему хранения проектных данных, размещаемых на Web-сервере Интернет-ресурсов САПР, и обеспечить к ним доступ из рабочих станций САПР.

### Выбор методов реализации баз данных Web-сервера Интернет-ресурсов САПР

Рассмотрим способы организации ЕИП схемотехнических САПР, размещаемого на Web-сервере. Во многом структуры данных определяются назначением компонентов информационного фонда. На основе анализа содержания баз данных предлагается использовать следующие подходы к реализации информационного обеспечения рассматриваемых САПР, которые нашли отражение в таблице.

Структура файлов БДСП полностью определяется форматами данных используемой САПР. Для сохранения этих файлов на рабочей станции пользователь может воспользоваться стандартными файловыми менеджерами. В случае передачи этих файлов на сервер Интернет-ресурсов САПР в базу данных рабочих проектов необходимо в проводнике выполнить соединение со службой FTP Web-сервера, пройти процедуру регистрации и после отправ-

№ пп	Наименование БД	Назначение БД	Способ хранения информации	Способ ведения	Способ доступа
1	БДСП	Файлы промежуточных и конечных результатов проектирования и файлы с описанием схемы	Двоичные и текстовые файлы операционной системы	Файлы создаются, модифицируются и удаляются программными модулями САПР	Открытие файлов и считывание данных программными модулями САПР
2	БДМК	Информация о структуре моделей схемных компонентов и значениях их параметров	Таблицы реляционной БД	Система управления БД	Web-приложение доступа к БД, Интернет-браузер
3	БДРП	Файлы промежуточных и конечных результатов проектирования и файлы с описанием схемы. Текущая проектная документация, отражающая состояние и ход выполнения проекта	Картотека файлов в виде таблиц реляционной БД, двоичные и текстовые файлы операционной системы, файлы документов в формате внешних приложений	Система управления БД. Обмен файлами через Интернет по протоколу FTP	Web-приложение доступа к БД, Интернет-браузер
4	БДАПР	Законченные и типовые проектные решения, выполненные в САПР	Таблицы реляционной БД. Файлы исходных и результирующих данных САПР, файлы документов в формате внешних приложений	Система управления БД. Обмен файлами через Интернет по протоколу FTP	Web-приложение доступа к БД, Интернет-браузер
5	БДНСПД	Справочные данные о материалах, элементах схем, унифицированных узлах и конструкциях; государственные и отраслевые стандарты, руководящие материалы	HTML-файлы, файлы документов в формате внешних приложений	Обмен файлами через Интернет по протоколу FTP	Web-сайт нормативно-справочных документов, Интернет-браузер

ки файлов занести соответствующую информацию в таблицы БДСП. Реализацию БДМК целесообразно проводить на основе реляционной модели данных, так как информация о моделях схемных компонентов хорошо структурирована и может быть представлена в виде таблиц.

Для построения БДРП предлагается использовать комбинированный подход хранения данных, который предполагает:

- непосредственное хранение файлов проектных документов в формате внешних приложений и файлов, полученных в результате проектирования объекта в САПР, на устройстве внешней памяти;
- хранение реквизитов файлов, характеризующих их принадлежность определенному проекту, иерархическую вложенность, историю изменений и т. д., в таблицах реляционной базы данных.

Данный подход широко применяется в системах электронного документооборота [9]. Каждый документ в системе электронного документооборота имеет набор однозначно характеризующих его реквизитов (часто этот набор называют карточкой документа). Карточка может включать тип документа (например, принципиальная схема), имя автора, даты формирования и изменения документа, отдел, в котором должен храниться документ, и многое другое. Хранение и поиск проектных документов осуществляется также в соответствии с реквизитами, указанными в карточке. Для хранения файлов данных САПР необходимо добавить в базу данных соответствующие карточки.

Организация БДАПР также может быть выполнена на основе комбинированного подхода, но в этом случае необходимо дополнительное сжатие проектных файлов для минимизации занимаемого дискового пространства и наличие процедуры поиска архивированных данных. Содержание БДНСПД представляет собой совокупность электронных версий справочников, нормативных документов, государственных стандартов и других документов в формате внешних приложений MS Word, Adobe Acrobat и др. Часто для формирования электронных документов применяют также HTML-страницы. Все перечисленные документы носят статический характер и редко подвержены изменениям, кроме того они не привязаны к конкретным проектам. Учитывая перечисленные обстоятельства, реализацию БДНСД целесообразно выполнить в виде Web-сайта со статической структурой.

### Заключение

Наиболее часто для организации информационного фонда схмотехнических САПР используют текстовые и двоичные файлы операционной системы; библиотечные текстовые файлы; специализированные СУБД. Такие способы хранения данных ориентированы, как правило, на обеспечение ин-

формацией проектирующих систем и не содержат нормативно-справочных и архивных проектных документов. Применение универсальных СУБД в САПР является наиболее комплексным подходом к хранению данных, который позволяет: провести структурирование данных в виде, удобном для проектировщика; выделить коллективно используемые данные и сформировать централизованный информационный фонд САПР; обеспечить поиск нормативно-справочной и проектной документации.

В случае использования технологий баз данных задача организации информационного обеспечения Web-ориентированных САПР сводится к построению специализированных баз данных различного назначения. При этом основными требованиями к распределенной структуре информационного фонда являются: обеспечение коллективной работы пользователей САПР в сети Интернет над общим проектом, выполняемым в САПР; наличие оперативного доступа из среды САПР к централизованным базам данных схемных компонентов. Для решения перечисленных проблем авторы предлагают разработать централизованное хранилище проектных данных на Web-сервере Интернет-ресурсов САПР. Взаимодействие пользователей с этим Web-сервером осуществляется посредством встроенного браузера и серверных web-приложений доступа к базам данных. При этом пользователям необходимо выполнить процедуру авторизации на сервере.

*Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта 15-07-0119\_а.*

### Список литературы

1. **Воейков Д.** Тенденции на рынке ИТ глазами САПР-компании // PC Week. 2011. RE № 16 (766).
2. **Лузин С. Ю., Лячек Ю. Т., Петросян Г. С., Полубасов О. Б.** Модели и алгоритмы автоматизированного проектирования радиоэлектронной аппаратуры. СПб: БХВ-Петербург, 2010. 224 с.
3. **Анисимов В. И., Гридин В. Н.** Методы построения систем автоматизированного проектирования на основе Internet-технологий и компактной обработки разреженных матриц // Информационные технологии в проектировании и производстве. 2009. № 1.
4. **Точилкин С.** Перевод графических приложений в облака // Открытые системы. СУБД. № 3. 2012. С. 21–27.
5. **Altium Designer Schematic Capture and PCB Editing training**, 2006. Altium Limited. 248 с.
6. **Date C. J.** SQL and Relational Theory: How to Write Accurate SQL Code. 2<sup>nd</sup> ed. Published by O'Reilly Media, Inc., 2012. 446 с.
7. **Ларистов А. И., Лячек Ю. Т., Абу Сара М. Р.** Интегрированные базы данных в программных системах проектирования электронных схем // Информационно-управляющие системы. 2009. № 3 (40). С. 69–71.
8. **Heineman R.** PSPICE Einführung in die Elektroniksimulation. Carl Hanser Verlag München. 2011, 400 Seiten.
9. **Гридин В. Н., Анисимов В. И., Каменьков Д. В.** Разработка типового встроенного модуля документооборота в системах управления данными об изделии // Информационные технологии в проектировании и производстве. 2011. № 2. С. 18–21.
10. **Гридин В. Н., Дмитриевич Г. Д., Анисимов Д. А.** Построение веб-сервисов систем автоматизации схмотехнического проектирования на основе методов сжатия данных // Информационные технологии и вычислительные системы. 2012. № 4. С. 79–84.

V. N. Gridin, Professor, Director, ingo@ditc.ras.ru,  
Center for Information Technology in the Design of the RAS,  
V. I. Anisimov, Professor, vianisimov@inbox.ru, G. D. Dmitrevich, Professor,  
A. I. Laristov, Ph. D., Associate Professor, ailaristov@inbox.ru,  
Ya. M. A. Al-Shameri, Graduate Student, saprfkti@mail.ru,  
St. Petersburg Electrotechnical University

## Organization of Information Support of Web-Based Circuit Design CAD

*The questions of the organization of information support of web-based circuit design CAD on the basis of database technology. Analyzes the composition of the circuit design data CAD. It is proposed to form a distributed structure of information support of web-based CAD systems, which is based on the use of a single design space on a web server of Internet resources CAD. User interaction with a web server of Internet resources CAD systems is possible via the built-in browser and server-side web applications to access databases. Discusses the composition of the databases included in a single information space of the web server, and methods for implementation of its components.*

**Keywords:** Web-based circuit design CAD, information CAD software, database technology, Web-server Internet resources CAD

### References

1. Voikov D. Tendentsii na rynke IT glazami S APR-kompanii, *PC Week RE*, 2011, no. 16 (766). (in Russian).
2. Luzin S. Yu., Lyachek Yu. T., Petrosyan G. S., Polubasov O. B. *Modeli i algoritmy avtomatizirovannogo proektirovaniya radioelektronnoi apparatury*. Saint Petersburg, BKhV-Petersburg, 2010, 224 p. (in Russian).
3. Anisimov V. I., Gridin V. N. Metody postroeniya sistem avtomatizirovannogo proektirovaniya na osnove Internet-tehnologii i kompaktnoi obrabotki razrezhennykh matrits, *Informatsionnye tekhnologii v proektirovanii i proizvodstve*, 2009. no 1. (in Russian).
4. Tochilkin S. Perevod graficheskikh prilozhenii v oblaka, *Otkrytye sistemy*, SUBD, 2012, no. 3, pp. 21–27 (in Russian).
5. Altium Designer Schematic Capture and PCB Editing training, 2006, Altium Limited, 248 p.
6. Date C. J. SQL and Relational Theory: How to Write Accurate SQL Code (2nd edition), Published by O'Reilly Media, Inc., 2012, 446 p.
7. Laristov A. I., Lyachek Yu. T., Abu Sara M. R. Integrirovannyye bazy dannykh v programmnykh sistemakh proektirovaniya elektronnykh skhem, *Informatsionno-upravlyayushchie sistemy*, 2009, no. 3 (40), pp. 69–71. (in Russian).
8. Heineman R. *PSPICE Einfuhrung in die Elektroniksimulation*. Carl Hanser Verlag, München, 2011, 400 p.
9. Gridin V. N., Anisimov V. I., Kamen'kov D. V. Razrabotka tipovogo vstroennogo modulya dokumentooborota v sistemakh upravleniya dannyimi ob izdelii. *Informatsionnye tekhnologii v proektirovanii i proizvodstve*, 2011, no. 2, pp. 18–21. (in Russian).
10. Gridin V. N., Dmitrevich G. D., Anisimov D. A. Postroenie veb-servisov sistem avtomatizatsii skhemotekhnicheskogo proektirovaniya na osnove metodov szhatiya dannykh. *Informatsionnye tekhnologii i vychislitel'nye sistemy*, 2012, no. 4, pp. 79–84. (in Russian).

УДК 004.3

A. V. Surkov, ст. науч. сотрудник, e-mail: surkov@cs.niisi.ras.ru  
Федеральное государственное учреждение науки  
Научно-исследовательский институт системных исследований РАН, г. Москва

## Использование статического временного анализа для проектирования самосинхронных схем

*Рассматривается вопрос адаптации существующего маршрута проектирования синхронных схем для работы с самосинхронными схемами. Современные САПР используют статический временной анализ в качестве промежуточного этапа для большинства операций со схемой. Основным требованием для работы статического временного анализа является отсутствие в схеме обратных связей и асинхронных элементов, что делает невозможным анализ самосинхронных схем. Предложенная методика изменения Liberty-моделей элементов позволяет устранить обратные связи в самосинхронных схемах с волновой синхронизацией, и проводить статический временной анализ. Изменения касаются только описания функций элементов с сохранением значений задержек, полученных при характеристизации. В результате открывается возможность автоматизированного проектирования самосинхронных конвейерных схем с использованием современных САПР.*

**Ключевые слова:** асинхронные, самосинхронные, временной анализ, САПР

## Введение

Существует ряд задач, где применение синхронных схем не всегда удобно или доступно. К примеру, в смешанных аналого-цифровых схемах синхронная логика является источником помех, распространяющихся по линиям питания и земли в аналоговую часть схемы. Существуют применения в условиях нестабильного питания и экстремальных температур, при которых использование синхронных схем ограничено ввиду накладываемых ограничений на время распространения сигнала внутри схемы. Известны также и применения, работающие при напряжении питания около порога срабатывания транзисторов, где синхронная логика не используется ввиду большого разброса задержек переключения. Все перечисленные свойства выгодно отличают *самосинхронный* подход к проектированию: низкие шумы при работе; работа при напряжениях питания около порога срабатывания транзисторов, а также инвариантность к задержкам при любых вариациях питания и температуры, при которых сохраняются переключательные свойства транзисторов. При этом асинхронность может являться и недостатком: самосинхронные схемы работают в логических отсчетах времени, которые не являются постоянными в физическом течении времени [1]. Неспособность работать в физических отсчетах времени в ряде случаев затрудняет взаимодействие самосинхронных схем с элементами синхронной среды. Тем не менее, самосинхронные схемы востребованы, и основным препятствием к их использованию является отсутствие специализированных средств автоматизации проектирования, в результате чего до недавнего времени самосинхронные схемы проектировали вручную.

Исследования В. И. Варшавского [2] показали возможность синтеза самосинхронной схемы путем преобразования синхронной схемы-прототипа. Использование уже разработанных синхронных блоков позволяет значительно сокращать время разработки их самосинхронных аналогов, поскольку экономится время на тестировании и выявлении ошибок. Также, в целях сокращения времени проектирования, в качестве синхронного прототипа удобно брать не исходную *RTL-модель*, а уже синтезированную, с использованием элементов библиотеки, схему. В связи с этим представляется наиболее удобным адаптировать современную методику разработки микросхем для работы с самосинхронной логикой, поскольку это существенно сократит время разработки.

### 1. Обзор маршрутов проектирования микросхем

Все этапы проектирования заказных микросхем (ASIC) условно можно разбить на две части: маршрут аналогового проектирования и маршрут цифрового проектирования. Оба маршрута полностью автоматизированы за счет использования САПР, при этом

для аналогового проектирования используют одни САПР, а для цифрового проектирования — другие.

Процесс автоматизации проектирования включает определенную формализацию входных данных для уменьшения сложности используемых алгоритмов в САПР. В частности, программы для аналогового проектирования работают со *Spice-моделями* транзисторов, емкостей, сопротивлений и индуктивностей, которые предоставляет фабрика. Замена физической топологии моделями позволяет существенно упростить, и как следствие, ускорить расчеты. А программы цифрового маршрута проектирования используют уже модели целых элементов, не опускаясь до уровня их транзисторной схемы. В этом случае, как правило, используют *Liberty-модели* элементов, которые описывают характер и значения задержек между входами и выходами элемента, а также упрощенно описывают функцию выходов. *Liberty-модель* элемента создают с помощью процедуры, называемой *характеризацией* на основе вариативного *Spice-моделирования* транзисторной схемы элемента, с учетом экстрагированных из топологии параметров *spice-моделей* сопротивлений проводов, паразитных емкостей и индуктивностей.

Вариативность моделирования заключается в дискретном переборе таких параметров как питающее напряжение, температура, крутизна переходных характеристик сигналов на входах элемента, и подключенной емкостной нагрузки к его выходам. В результате полученные значения заносятся в таблицы и сохраняются в виде библиотеки в формате *Liberty*. Полученную библиотеку вследствие табличного задания значений также называют точечной (*dot-lib*). В дальнейшем САПР всегда может восстановить поведение элемента и переходную характеристику его выходов, взяв соответствующие заданным параметрам значения из таблиц *Liberty-модели* элемента и проведя требуемую интерполяцию по ближайшим полученным из таблицы точкам.

Существует и более точная модель элементов (*CCS — Composite Current Source*), учитывающая эффект Миллера при переключении входных каскадов элементов, а также более близко описывающая поведение выхода элемента при большом сопротивлении нагрузки. Для расчета задержек с использованием этой модели выход элемента представляется как источник тока, который зависит от крутизны переходной характеристики сигнала на входе, емкости выходной нагрузки, времени, и задается в виде четырех таблиц — при переключении из логического 0 в логическую 1 и обратно, до и после порога переключения.

Емкость входа элемента также задается в виде четырех таблиц и зависит от крутизны подаваемого сигнала, но при расчете задержек может учитывать и емкость нагрузки на выходе элемента. Использование источников тока при описании выхода в модели элемента удобно при анализе шумов и на-

водов (*crosstalk*), а также при анализе просадки питания (*IR-drop*).

Описанная модель с использованием источников тока является значительно более сложной с точки зрения характеристики и использования, и поэтому для работы с самосинхронными элементами на данный момент не рассматривается.

## 2. Синтез самосинхронных схем

Маршрут проектирования цифровой части микросхем условно можно разделить на два этапа: создание RTL-модели схемы и ее синтез с применением библиотек элементов, а затем проектирование топологии схемы с использованием результатов синтеза. RTL-модель синхронной схемы создается путем описания функциональных узлов на языках описания аппаратуры, таких как Verilog или VHDL.

Для функциональной верификации модели используют специальные программы для моделирования, позволяющие провести верификацию с проверкой покрытия тестами. В НИИСИ для моделирования используют САПР Cadence Incisive. Синтез RTL-модели проводится в два этапа: вначале выполняется логический синтез схемы во внутренний формат САПР, а затем с использованием библиотек элементов в формате *Liberty* проводится трансформация синтезированной модели в схему, состоящую из элементов библиотеки и их соединений, и называемую *нетлистом*. В НИИСИ для логического синтеза используют программу *Synopsys Design Compiler*. Полученный нетлист в дальнейшем можно моделировать с использованием подключаемых *Verilog*-моделей библиотечных элементов, которые получают на этапе характеристики. При этом существует возможность рассчитать реальные задержки элементов, исходя из нагруженности проводов, путем использования инструментов САПР. Рассчитанные задержки записывают в отдельный файл (формат SDF), который используется при моделировании нетлиста. Следует помнить, что на этапе синтеза САПР не может учитывать все паразитные задержки в схеме, поскольку информация о них появится только после проектирования топологии.

Разрабатываемый в НИИСИ маршрут проектирования самосинхронных схем не предусматривает создания модели самосинхронных схем с использованием языков HDL. Вместо этого используется методика [3] автоматического преобразования нетлиста синхронной схемы, являющейся прототипом. Преобразование проводится с помощью специального скрипта, написанного на языке Perl. Полученный таким образом нетлист самосинхронной схемы также можно моделировать с экстрагированными из библиотеки задержками. Для экстракции задержек, полученных в результате трансформации, нетлист необходимо загрузить в САПР вместе с *Liberty*-библиотеками элементов (включая и самосинхронные элементы), и выполнить экс-

тракцию в формат SDF. Таким образом, САПР синтеза используют для создания самосинхронной схемы лишь косвенным образом: на первом этапе для синтеза прототипа, и на завершающем этапе — для экстракции задержек, используемых при верификации функции самосинхронной схемы с помощью моделирования.

## 3. Статический временной анализ

Программа автоматического синтеза выполняет две основные задачи: цифровой синтез с использованием булевых преобразований, трансформация синтезированной модели в схему из стандартных библиотечных элементов. Эти и другие операции учитывают взятые из *Liberty*-моделей внутренние задержки и емкости выводов элементов, что позволяет проводить оптимизацию схемы по таким параметрам, как площадь, потребление и производительность. Все работающие в этих операциях алгоритмы используют особую методику расчета задержек, называемую статическим временным анализом (*STA — Static Timing Analysis*). Аналогичная методика применяется и в САПР по проектированию топологии, поскольку позволяет оценивать правильность размещения элементов, расположения соединений с использованием металлов и т. д. В первую очередь STA является методикой оценки временных характеристик схемы, а поскольку все используемые в САПР алгоритмы обязаны вести контроль над результатами проводимых операций, то без STA полноценная работа САПР невозможна.

В статическом временном анализе используется графовая модель соединений между элементами: элементы являются вершинами графа, а связи — соединениями. При этом используется такой термин, как путь (*path*) сигнала, представляющий собой траекторию распространения сигнала, проходящую по соединениям (ребрам графа) через элементы логики (вершины графа). Пути сигналов можно разделить на две группы — синхронные и асинхронные.

Синхронные пути сигналов всегда конечны: они начинаются на тактовом входе синхронного триггера-источника сигнала, проходят внутри модели триггера на выход, затем распространяются по проводам через модели элементов логики и заканчиваются на входе данных синхронного триггера-приемника сигнала. Элементы логики, включенные между триггерами, являются частью пути сигнала вместе с соединяющими их проводами. Пример пути сигнала в синхронной схеме показан пунктиром на рис. 1.

Для работы САПР используют в основном синхронные пути сигналов, поскольку они наиболее удобны для расчетов ввиду своей конечности. Однако в некоторых случаях пути сигналов могут замыкаться в петлю, что нарушает работу САПР. Например, асинхронные входы синхронных элементов, такие как асинхронный сброс у триггера, также мо-

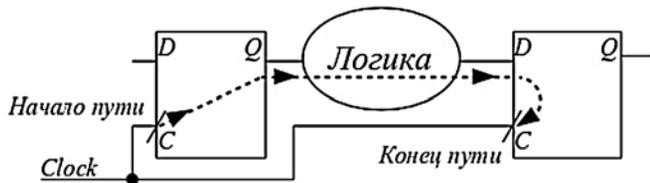


Рис. 1. Путь распространения сигнала в синхронной схеме

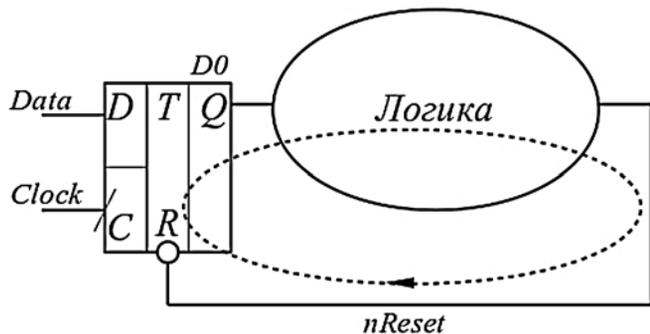


Рис. 2. Пример петли обратной связи в синхронной схеме

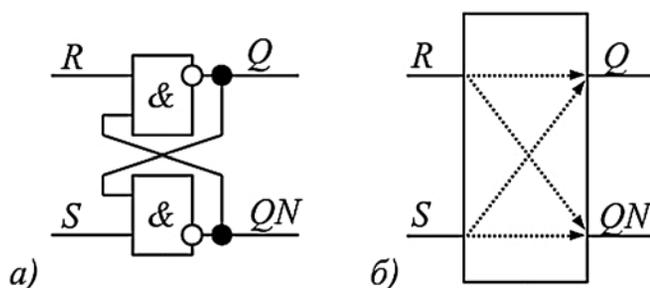


Рис. 3. Принципиальная схема RS-зашелки (а), и асинхронные связи между входами и выходами внутри Liberty-модели RS-зашелки (б)

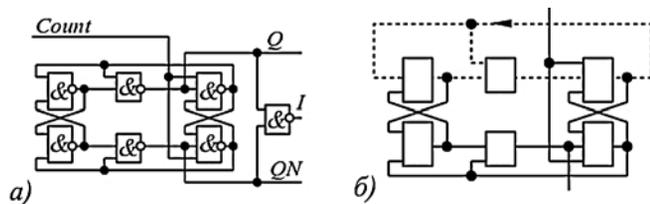


Рис. 4. Схема самосинхронного счетного триггера (а), и пример обратной связи внутри самосинхронной схемы (б)

гут стать частью пути сигнала и быть использованы при расчете. В результате, траектория сигнала замыкается через обратную связь, образовав петлю. Схема с приведенной петлей обратной связи показана на рис. 2.

Схема соединений на рис. 2, считается непригодной к использованию. При обнаружении замкнутого пути сигнала, обозначенного на рис. 2 пунктиром, САПР попытается искусственно разорвать эту связь, удалив информацию о соединении в некоторой точке петли. Что, в свою очередь, приводит к ошибке при расчете задержки прохождения сигнала по указанному пути.

Тем не менее, существуют и асинхронные пути сигналов, для активации расчета которых в САПР

необходимо использовать специальные ключи. Асинхронные пути сигналов проходят только через входы полностью асинхронных элементов, таких как RS-зашелка, показанная на рис. 3.

Асинхронные пути сигналов также могут быть использованы в расчетах, но только в том случае, если они не замыкаются через обратную связь. Все известные САПР цифрового маршрута проектирования работают исключительно с конечными путями сигналов — как синхронными, так и асинхронными. Но поскольку в подавляющем большинстве проектируемых схем асинхронных элементов нет, то по умолчанию при расчетах используют только синхронные пути сигналов.

Невозможность работать с замкнутыми путями сигналов является основной преградой к использованию современного цифрового маршрута для проектирования самосинхронных схем. В качестве примера рассмотрим схему самосинхронного счетного триггера (рис. 4) и путей сигналов внутри.

На рис. 4, а показана принципиальная схема самосинхронного счетного триггера со счетным входом (сигнал *Count*), парафазным выходом (сигналы *Q* и *QN*), и выходом индикации окончания переходных процессов (сигнал *I*). На рис. 4, б в качестве примера замкнутой обратной связи пунктиром обозначена траектория сигнала через три элемента И-НЕ; функции элементов скрыты для наглядности рисунка. Ввиду множества обратных связей в самосинхронных схемах алгоритмы STA не работают, а значит, и не работают такие функции САПР, как размещение элементов, трассировка, различные оптимизации скорости, площади схемы, потребления питания и т. д. Поэтому для использования САПР при проектировании самосинхронных схем, необходимо избавиться от обратных связей, препятствующих работе статического временного анализа.

В ходе исследований было найдено единственное на данный момент решение задачи о ликвидации обратных связей в схеме, заключающееся в изменении описания Liberty-моделей. Асинхронные элементы с помощью моделей удобно представлять как элементы логики, либо как синхронные триггеры, в результате чего асинхронные пути сигналов становятся конечными и могут быть описаны как синхронные. В результате изменения моделей элементов САПР может работать с самосинхронной схемой так, как если бы она являлась синхронной. С помощью САПР становится возможным автоматически размещать элементы, проводить трассировку и оптимизации.

#### 4. Проектирование и характеристика самосинхронных элементов

На данный момент для проектирования самосинхронных конвейеров потребовалось создать всего два типа самосинхронных элементов, отсутствующих в стандартной библиотеке (TSMC 65 нм). Первый тип элементов — С-элемент Маллера спро-

ектирован с использованием статической схемы [4] для двух входов (рис. 5) и с использованием мостиковой схемы [4] для трех входов (рис. 6).

Также для схемы синхростратума для волновой обработки данных [1] потребовалось спроектировать двухвходовой *C*-элемент со сбросом, который используется для начальной установки схемы. Второй тип элементов — это RS-зашелка с дополнительным сбросом, используемая для хранения данных. RS-зашелку можно построить и из двух библиотечных элементов И-НЕ (см. рис. 3, а), однако в этом случае мы получаем две замкнутые обратные связи. Поэтому оказалось целесообразным спроектировать RS-зашелку в виде отдельного элемента библиотеки: в этом случае обратные связи оказываются скрытыми внутри *Liberty*-модели.

Топологическое проектирование выполняла САПР Virtuoso; DRC и LVS проведены с помощью САПР Calibre. Проектирование топологии самосинхронных элементов ничем не отличается от проектирования топологии других устройств.

Для характеристики элемента в САПР SiliconSmart требуется *Spice-нетлист*, который представляет собой совокупность *spice*-моделей транзисторов, паразитных емкостей, индуктивностей и сопротивления проводов, а также всех межсоединений. *Spice-нетлист* получается на завершающем этапе разработки топологии элемента с использованием САПР Calibre. Также для характеристики требуется описать функцию устройства, чтобы САПР мог составить список тестов для измерения электрических и временных характеристик элемента, которые впоследствии будут сохранены в табличном виде в описании *Liberty*-модели элемента. Поскольку *C*-элемент Маллера не является распространенным устройством, то его функциональное описание удобно представить для характеристики как RS-зашелку (см. рис. 3). В качестве примера рассмотрим двухвходовой *C*-элемент на рис. 4, поведение выходов которого описывается следующими формулами:

$$R = !A \& !B; \quad (1)$$

$$S = A \& B, \quad (2)$$

где *A* и *B* — входы *C*-элемента, а выход *C*-элемента берется с прямого выхода (*Q*) RS-зашелки.

Полученная в результате характеристики *Liberty*-модель *C*-элемента подходит для экстракции задержек в САПР Synopsys DC, а полученная Verilog-модель может быть использована вместе с экстрагированными задержками в моделировании нетлиста. Однако для статического временного анализа полученная *Liberty*-модель не годится, поскольку содержит только асинхронные пути сигнала, изображенные на рис. 3, б. Чтобы САПР мог считать задержку прохождения сигнала через *C*-элемент, необходимо вручную исправить *Liberty*-модель, изменив функцию устройства с зашелки на элемент логики, такой как И или ИЛИ. В этом случае пути сигнала внутри модели можно считать синхронны-

ми и использовать при расчете задержек в САПР. Изменение функции элемента можно считать корректным, поскольку применяемые сейчас в большинстве САПР алгоритмы STA не используют при расчетах функцию элемента, беря вместо этого наихудшую из задержек сигнала (вход-выход), содержащихся в *Liberty*-модели элемента. Измененную *Liberty*-модель можно использовать только в САПР синтеза и топологии. Для функционального моделирования необходимо использовать оригинальные модели элементов, полученные на этапе характеризации.

Более сложной задачей является изменение *Liberty*-модели RS-зашелки, применяемой для хранения данных на выходах стадий самосинхронного конвейера. Стадия самосинхронного конвейера условно показана на рис. 7.

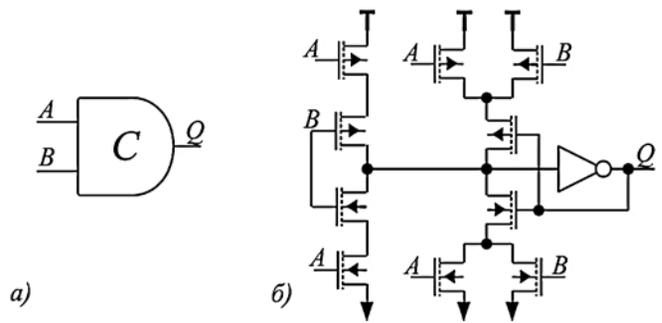


Рис. 5. Статическая схема двухвходового *C*-элемента Маллера

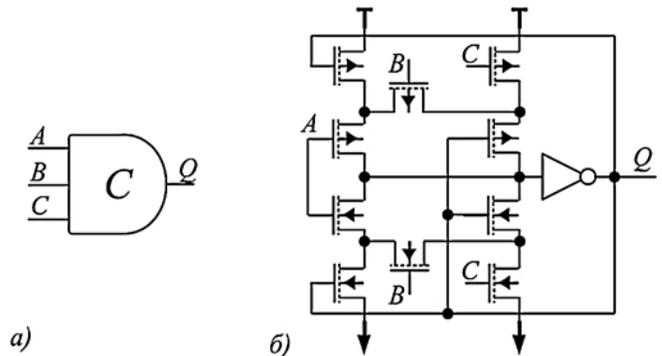


Рис. 6. Мостиковая схема трехвходового *C*-элемента Маллера

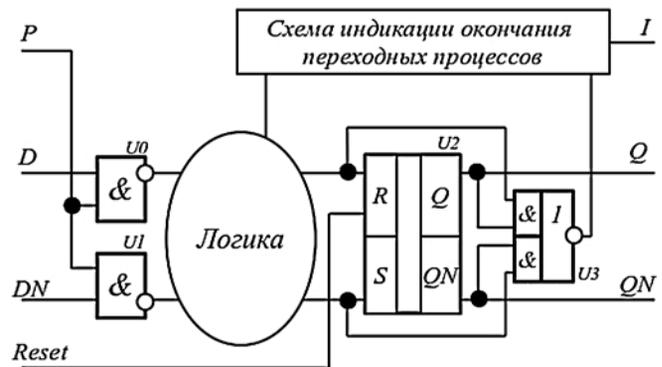


Рис. 7. Стадия самосинхронного конвейера

В целях упрощения рисунка у стадии конвейера изображены только один входной ( $D$ ,  $DN$ ) и один выходной ( $Q$ ,  $QN$ ) парафазные разряды данных, но подразумевается, что схема содержит множество входных и выходных разрядов данных. Входной сигнал  $P$  является фазовым для всей стадии, и управляет элементами блокировки входных сигналов  $U0$  и  $U1$ . Элемент  $U3$  осуществляет индикацию окончания переходных процессов в защелке  $U2$ , формирующей выходы одного разряда данных. Сигнал  $I$  является выходом схемы окончания переходных процессов всей стадии конвейера, включая логику и защелки хранения. Элементы блокировки входных сигналов являются логическими элементами и могут быть использованы при расчете путей сигнала. В то же время защелка  $U2$  является непрозрачной (содержит только асинхронные пути сигнала) для СТА, и ее *Liberty*-модель должна быть изменена. Поскольку в синхронных схемах начало и конец пути сигнала могут находиться только внутри модели  $D$ -триггера, то целесообразно изменить и *Liberty*-модель защелки наподобие синхронного триггера. Пути сигналов в *Liberty*-модели защелки после характеристики и после изменения показаны на рис. 8. Для проектирования самосинхронных конвейеров удобно использовать защелки с двумя сбросовыми входами.

На рис. 8, а, в отличие от рис. 3, б, показаны все существующие пути сигналов внутри модели, включая сквозные пути сигнала, и связь между входами. Сквозные пути учитываются при расчете задержки распространения сигнала, а пути между входами используются лишь для контроля взаимовлияния (*Setup* и *Hold*) входных сигналов во время их установки и снятия.

Модель защелки необходимо изменить таким образом, чтобы асинхронные пути стали синхронными. На рис. 8, б показано, как внутри *Liberty*-модели помещаются два виртуальных синхронных  $D$ -триггера, которые тактируются с входа  $R1$ , на который согласно схеме стадии конвейера (см. рис. 7) заведен сигнал сброса. Сигнал сброса является глобальным для всего конвейера, поэтому с точки зрения САПР может быть объявлен как тактирующий. В результате проведенного изменения *Liberty*-модели схему самосинхронного конвейера на рис. 7 можно рассматривать как синхронную, по-

скольку при расчете задержек сквозные пути со входов ( $D$ ,  $DN$ ) до выходов ( $Q$ ,  $QN$ ) отсутствуют, а в схеме появляются виртуальные синхронные триггеры и схема тактирования.

Прделанные изменения *Liberty*-модели не являются полностью корректными с точки зрения расчета временных параметров схемы, поскольку лишь незначительная часть асинхронных путей сигналов, получившихся на этапе характеристики, наследуется в измененной модели. В частности, задержка между входом  $R0$  и  $Q$  значительно меньше, чем между  $S$  и  $Q$ , что следует из принципиальной схемы защелки (см. рис. 3, а). В связи с этим задержку до выхода  $Q$  в измененной модели целесообразно взять равной асинхронной задержке между  $S$  и  $Q$ , поскольку она заведомо больше задержки между  $R0$  и  $Q$ . В результате, алгоритмы статического временного анализа схемы с измененными элементами будут оперировать реальными значениями емкостей выводов элементов, но внутренняя задержка этих элементов будет взята с запасом, что существенно ограничивает вариативность анализа.

Представление *Liberty*-моделей *RS*-защелок в качестве сдвоенного синхронного  $D$ -триггера, у которого к тактовому входу подключен сигнал сброса, позволяет представить схему самосинхронного конвейера (см. рис. 7) в качестве синхронной схемы, состоящей их элементов логики,  $D$ -триггеров, и у которой линия сброса может рассматриваться как псевдотактирующий сигнал. Если после указанной замены в схеме остались обратные связи, то достаточно выбрать удобный элемент для разрыва обратной связи, и изменить его модель на синхронный  $D$ -триггер. Для расчета скорости работы стадии самосинхронного конвейера с использованием СТА, необходимо выполнить расчет наихудшего пути сигнала, проходящего через фазовый вход и через выход окончания переходных процессов рассматриваемой стадии конвейера.

Таким образом, изменение *Liberty*-моделей элементов решает основную задачу использования САПР для проектирования самосинхронных схем, но расчет задержек возможен только для наихудшего случая, поскольку при изменении *Liberty*-модели асинхронного элемента часть задержек не учитывается.

## 5. Практические результаты

Маршрут проектирования самосинхронных схем использован при разработке самосинхронного блока умножения с накоплением (FMA) двойной точности с применением технологии TSMC 65 нм. Спроектирована библиотека асинхронных элементов, содержащая *RS*-защелку и *C*-элементы Маллера. В качестве прототипа взята синхронная RTL модель блока FMA, использующаяся в процессоре КОМДИВ-64 [5]. На первом этапе разработки осуществлен синтез синхронного прототипа FMA в базис стандартных библиотечных элементов. Далее

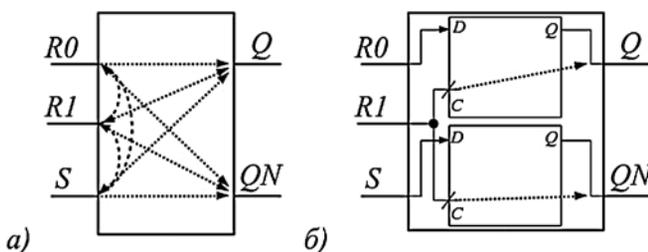


Рис. 8. Пути сигналов в *RS*-защелке после характеристики (а), и после изменения *Liberty*-модели (б)

выполнено перекрестное преобразование с помощью скрипта в прямой и инверсный каналы передачи данных с восстановлением монотонности полученных в результате преобразования функций. И в заключение, синхронные триггеры заменены на RS-защелки, построена схема индикации завершения переходных процессов в стадиях конвейера, добавлены элементы синхростратума для волновой обработки данных и выписан результирующий нетлист. Для дальнейшей работы потребовалось изменить библиотеки асинхронных элементов в соответствии с приведенной методикой. В результате получены два набора библиотек: библиотека с использованием оригинальных *Liberty*-моделей элементов, используемая для выписывания файла экстрагированных задержек элементов (SDF), и библиотека для использования алгоритмов STA. Проведено моделирование полученной самосинхронной схемы с экстрагированными задержками в САПР Incisive, а также сделан статический временной анализ наихудших путей распространения сигнала в САПР Design Compiler. С учетом внесенных изменений в модели элементов, ухудшающих точность расчетов, результаты моделирования и временного анализа совпали.

### Заключение

Использование алгоритмов STA является ключевым условием для работы современных САПР. При этом STA работает только с конечными путями распространения сигналов в схеме и на практике используется в основном с синхронными моделями элементов. Самосинхронные же схемы состоят из элементов логики и самосинхронных элементов

памяти, поэтому большая часть путей сигналов внутри схемы замкнута. Замкнутые пути сигналов, в свою очередь, препятствуют автоматизации проектирования с использованием САПР.

В НИИСИ разработана методика изменения асинхронных моделей элементов таким образом, чтобы самосинхронные конвейерные схемы становились неотличимы от синхронных схем с точки зрения STA и проектирования в САПР. Новый маршрут также включает в себя использование уже разработанных ранее синхронных блоков. Таким образом, новая методика позволяет не только сократить время проектирования самосинхронных конвейеров за счет использования САПР, но и экономит время на разработку логической модели и тестирование за счет применения имеющихся синхронных разработок.

*Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 14-29-09238 офи\_м.*

### Список литературы

1. **Мараховский В. Б., Мелехин В. Ф.** Проектирование средств синхронизации блоков глобально асинхронных систем с произвольной локальной синхронизацией // Информационно-управляющие системы. 2010. № 1. С. 29–38.
2. **Varshavsky V. I., Marakhovsky V. B., Chu T. A.** Asynchronous Timing of Arrays with Synchronous Prototype // Proc. of the Second Intern. Conf. on Massively Parallel Computing Systems (MPCS'96). 1996. P. 47–54.
3. **Сурков А. В.** Использование Synopsys Design Compiler для синтеза самосинхронных схем // Программные продукты и системы. 2014. № 4. С. 24–30.
4. **Moreira M., Oliveira B., Moraes F., Calazans N.** Impact of C-Elements in Asynchronous Circuits // 13<sup>th</sup> Int'l Symposium on Quality Electronic Design. 2012. P. 437–452.
5. **Бобков С. Г.** Архитектура микропроцессоров суперЭВМ экасафлпсного диапазона // Информационные технологии. 2012. № 12. С. 2–9.

**A. V. Surkov**, Senior Researcher, surkov@cs.niisi.ras.ru

Scientific Research Institute of System Analysis, Russian Academy of Sciences, Moscow, Russia

## STA Algorithms for Developing Self-Timed Schemes

*The article covers the problem of adopting the synchronous workflow for developing self-timed schemes. The modern EDA tools use the static timing analysis algorithms (STA) as interim step for the most of operations. The major requirement for the STA to work is the absence of feedback interconnections in the analyzed circuit. Suggested approach eliminates the feedbacks in the self-timed pipelines by the modifying the Liberty-model of used elements. The modification affects the functional description of modified elements, so it does not change the arc delays. This allows STA to work, so it becomes possible to use the modern EDA tools to automate design of the self-timed pipelines.*

**Keywords:** asynchronous, self-timed, STA, EDA tools

### References

1. **Marakhovsky V. B., Melekhin V. F.** Proektirovanie sredstv synchronizatsii blockov globalno asynchronnih sistem s proizvolnoy localnoy synchronizatsiyey, *Information and Control Systems*, 2010, no. 1, pp. 29–38 (in Russian).
2. **Varshavsky V. I., Marakhovsky V. B., Chu T. A.** Asynchronous Timing of Arrays with Synchronous Prototype, *Proc. of the Second Intern. Conf. on Massively Parallel Computing Systems (MPCS'96), Ischia, Italy*, 1996, pp. 47–54.
3. **Surkov A. V.** Synthesis of burst-mode asynchronous schemes using Synopsys Design Compiler, *Programmnye produkty i sistemy*, 2014, no. 4, pp. 24–30 (in Russian).
4. **Moreira M., Oliveira B., Moraes F., Calazans N.** Impact of C-Elements in Asynchronous Circuits, *13<sup>th</sup> Int'l Symposium on Quality Electronic Design*, 2012, pp. 437–452.
5. **Bobkov S. G.** Arhitektura mikroprocessorov super-EVM ekasaflopsnogo diapazona, *Informacionnye Tekhnologii*, Moscow, 2012, no. 12, pp. 2–9 (in Russian).

УДК 004.89

**А. М. Бершадский**<sup>1</sup>, д-р техн. наук, проф., зав. каф., e-mail: bam@pnzgu.ru,

**А. С. Бождай**<sup>1</sup>, д-р техн. наук, проф. каф., e-mail: bozhday@yandex.ru,

**В. С. Мкртчян**<sup>2</sup>, д-р техн. наук, проф., e-mail: hhhuniversity@gmail.com

<sup>1</sup> ФГБОУ ВПО "Пензенский государственный университет", г. Пенза, Россия

<sup>2</sup> Triple H Namalsaran of HNH TECHNOLOGY Inc. "HNH University", г. Сидней, Австралия

## Принципы построения общедоступной самоадаптирующейся системы дистанционного обучения на основе модели изменчивости и сервис-ориентированной архитектуры\*

*Обсуждается подход к построению единой общедоступной среды для оказания электронных образовательных услуг. Рассматривается сервис-ориентированная схема взаимодействия интеллектуальных агентов обучаемого и преподавателя. Предлагается метод самоадаптации агента обучаемого на основе гиперграфовой модели изменчивости, позволяющий организовать непрерывный цикл электронного обучения и существенно увеличить жизненный цикл всей системы дистанционного обучения.*

**Ключевые слова:** система дистанционного образования, E/U Learning, модель изменчивости, сервис-ориентированная архитектура, гиперграф, технология IDSPL

### Введение

Стремительное развитие информационно-компьютерных технологий (ИКТ) и средств телекоммуникации определило основные тенденции интеллектуализации средств обучения. Отправной точкой этих тенденций является возникновение технологии E-Learning [1, 2], определяющей основные аспекты электронного обучения с использованием средств ИКТ, мультимедиа и Интернета. К настоящему моменту технология E-Learning претерпела существенную эволюцию. В начальной версии E-Learning 1.0 были заложены следующие базовые принципы: создание и доставка электронных обучающих материалов; самостоятельная работа с обучающим материалом; получение консультаций и сдача экзаменов удаленным способом; создание образовательных web-ресурсов и порталов; получение современных знаний без территориальной привязанности и др. В последующих трактовках технологии E-Learning 1.3 и E-Learning 2.0 произошло значительное расширение функций электронного обучения: видеосеминары и видеолекции; форумы; создание пакетов специфических заданий (кей-

сов); использование средств компьютерной графики и анимации.

Сегодня научное сообщество подошло к осознанию новой фазы электронного образования (E-Learning 3.0), в которой важную роль будут играть распределенные компьютерные системы, облачные технологии, мобильные персональные устройства, системы искусственного интеллекта и средства виртуальной реальности. Особенно актуальными являются разработки в направлении интеграции процессов интернационализации и интеллектуализации образования. Это приводит к возникновению фундаментальной научной проблемы создания единой межгосударственной интеллектуальной среды для оказания электронных образовательных услуг, в которую будут включены гетерогенные формы представления знаний, межгосударственные стандарты и формы обучения, международный преподавательский и студенческий состав. Таким образом, можно говорить о принципиально новой стадии доступности электронного образования — всепроникающем образовании (All-pervading E-Learning или E/U Learning).

Другой важной проблемой ближайшего будущего является многоуровневая самоорганизация (адаптируемость) компонентов электронной образовательной среды без лишних процессов перекомпи-

\* Работа выполнена при поддержке РФФИ (научный проект № 15-07-01553).

ляции, что приведет к их универсализации и продлению жизненного цикла. В частности, должны быть решены проблемы адаптируемости компонентов к изменениям объема и качества знаний в предметных областях, изменениям требований общества к знаниям и навыкам специалистов, изменениям к формам и методам обучения, изменениям технических средств вычислений и коммуникаций.

В итоге, можно кратко сформулировать основные специфические требования к формам и методам дистанционного образования ближайшего будущего:

1) мобильность и оперативность образовательных технологий;

2) адаптивность образовательных курсов и систем как с точки зрения содержания, так и с точки зрения форм передачи знаний;

3) сервис-ориентированность, позволяющая гибким образом подстраивать функционал систем дистанционного образования (СДО) под требования каждого конкретного человека, а также эффективным образом использовать распределенные образовательные ресурсы.

Таким образом, создание СДО, обладающих свойствами мобильности, адаптивности и сервис-ориентированности является в настоящее время актуальной научно-практической проблемой и перспективным направлением развития электронного образования в целом.

В широком смысле под мобильностью дистанционных технологий понимается возможность получения образовательных услуг и различной справочной информации с использованием персональных мобильных устройств (смартфонов, планшетных компьютеров, нетбуков и др.) и беспроводных (*wireless*) сетей доступа к данным (WiFi, WiMAX, 3G/4G, WAP) [3]. Пользователю предоставляется возможность постоянного присутствия в образовательном облаке без привязки к конкретным географическим территориям, жизненным обстоятельствам или программно-техническим платформам.

Адаптивность образовательных курсов и систем необходимо рассматривать с трех основных точек зрения:

— адаптивность к постоянно изменяющемуся уровню знаний обучаемого;

— адаптивность к изменяющемуся материалу в рамках отдельных учебных курсов;

— адаптивность к изменяющимся требованиям со стороны работодателей и рынков труда.

Поскольку принцип мобильности и оперативности подразумевает непрерывное проведение дистанционного обучения и оказание справочно-образовательных услуг в изначально неопределенных условиях для заранее неопределенных групп пользователей, то вынужденным образом меняется и концепция жизненного цикла электронной образовательной системы. Принцип адаптивности уже нецелесообразно реализовывать путем выпуска це-

почек обновленных версий. Разработчик просто не имеет нужного времени на перекомпиляцию, отладку и верификацию каждой новой версии системы или учебного курса — дистанционные образовательные услуги необходимо оказывать непрерывно и с одинаково хорошим уровнем качества. От СДО требуется самонастройка, прозрачная для пользователя, без перекомпиляции исходного программного кода. Следовательно, архитектура такой системы изначально должна базироваться на модели изменчивости (VM — *Variability Models*), управляемой контурами обратной связи с обучаемым, преподавателем и работодателем (то есть учитывающая все три перечисленных точки зрения на адаптивность).

Для реализации подобного принципа адаптации целесообразно применять методы, используемые в настоящее время в сфере инженерии линейек программных продуктов (SPLE — *Software Product Line Engineering*). В частности, парадигма повторного использования программных продуктов за счет динамической реконфигурации их программного кода разработана в рамках технологий DSPL (*Dynamic Software Product Lines* — динамические линейки программных продуктов) [4, 5].

Кроме того, структурный состав системы должен иметь распределенный модульный вид с четко выделенными автономными сервисами и гибкими алгоритмами их взаимодействия. На сегодняшний день одним из лучших подходов к решению этих задач считается сервис-ориентированная технология, в рамках которой архитектура системы выстраивается на основе автономных, совместно функционирующих web-сервисов. Такую архитектуру принято называть сервис-ориентированной (SOA — *Service-Oriented Architecture*) [6].

Совокупности различных web-сервисов СДО позволяют построить интеллектуальные агенты (аватары), способные адаптироваться к текущему уровню знаний обучаемого, к требованиям окружающей среды (например, рынкам труда), к конкретным методикам преподавания и формам обучения, к различным национальным образовательным системам и стандартам [7].

## 1. Основные технологические составляющие общедоступной (всепроникающей) СДО

Идеология E-Learning базируется на принципах многократного использования (*reusability*) и свободного распространения (*shareability*) авторских курсов. Поэтому разработчики учебных курсов должны придерживаться общепринятых стандартов. К настоящему моменту наибольшее распространение получили следующие модели учебных курсов.

1. Модель IEEE LOM (*Learning Object Model*), разработанная организацией LTSC (*Learning Technology Standard Committee*) в 2002 г. Вся совокупность обучающих объектов разделена на девять составных иерархий (категорий): общие (*General*),

жизненный цикл (*Life Cycle*), метаданные (*Metadata*), технические (*Technical*), образовательные (*Education*), правовые (*Rights*), связи (*Relation*), аннотации и классификации (*Annotation and Classification*).

2. Система спецификаций консорциума IMS (таких, например, как *Content Packaging Specification*, *Metadata Specification*, *Digital Repositories Interoperability*, *Digital Repositories*).

3. Спецификации комитета AICC (*Aviation Industry Computer-Based Training Committee*), предназначенные изначально для развития компьютерных обучающих систем и технологий в области авиационной промышленности.

4. Спецификация SCORM (*Shareable Course Object Reference Model*), разработанная в рамках программы ADL (*Advanced Distributed Learning*), выполняемой по инициативе Министерства обороны США. Это промышленный стандарт для обмена учебными материалами на базе адаптированных спецификаций ADL, IEEE, IMS, AICC. Основой модели SCORM является модульное построение учебного материала за счет выделения отдельных автономных учебных единиц (SCO — *Shareable Content Objects*) и представления их в специальных Web-репозиториях. Модули SCO могут объединяться друг с другом в различных сочетаниях и компилироваться в электронные учебные пособия с помощью LMS-системы.

Таким образом, если в первых E-Learning системах преподавателю приходилось самостоятельно собирать обучающие курсы, хранить у себя на персональном компьютере и далее вручную организовывать в единый образовательный контент, то с появлением таких спецификаций как SCORM эта работа автоматизируется с возможностью использования технологий Web 2.0 и сервис-ориентированного подхода.

Другой актуальной проблемой современных электронных образовательных систем является проблема создания модели обучаемого на основе отслеживания личной информации, связанной с обучением, траекториями прохождения по различным обучающим модулям или web-сервисам, пройденным курсам, сданным тестам. Для этих целей также существует ряд спецификаций, наиболее известными из которых являются:

- IEEE PAPI (*Personal and Private Information*);
- IMS LIP (*Learner Information Package*).

При этом используется язык XML (*eXtension Markup Language*) для записи в профиль пользователя его биографических данных, истории обучения, владения языками, предпочтения в использовании компьютерных платформ, пароли доступа к средствам обучения и т. п. Эти сведения в дальнейшем используют для учета индивидуальных особенностей обучаемого при определении оптимальных средств и методик обучения.

Для квалификационной аттестации обучения (*learning competency assessment*) также используют унифицированные спецификации. Например, IMS QTI (*Question and Test Interoperability*), где описывается иерархическая структура тестирующей информации, формы представления заданий и списка ответов, возможные сценарии тестирования и методы обработки полученных результатов.

Осенью 2013 года Европейская комиссия инициировала новый проект — *Opening up Education*, нацеленный на создание и развитие единого общедоступного, открытого, интероперабельного образовательного пространства для электронного обучения. Проект опирается на новейшие достижения в области E-Learning. В частности, на интеграцию web-ресурсов MOOC (*Massive Open Online Courses* — массовые открытые онлайн-курсы), образовательных порталов OER (*Open Education Resources* — открытые образовательные ресурсы), открытых стандартов качества обучающих материалов OQS (*Open Quality Standards*). При этом большой упор планируется сделать на мобильность учащихся, беспроводные коммуникации и передовые образовательные технологии (облачные технологии, компьютерные игры, средства виртуальной и расширенной реальности).

Тем не менее, стоит отметить несколько проблем, актуальных на сегодняшний день практически для всех перспективных проектов E-Learning.

Несмотря на столь успешную работу по стандартизации электронных обучающих систем, сохраняются существенные трудности преобразования обучающего контента, созданного в рамках различных стандартов, для их совместного использования конечным пользователем. Предлагаемая в следующих разделах аватар-технология позволяет в некоторой степени разрешить эту проблему. В частности, аватар, являясь интеллектуальным агентом конечного пользователя, выступает в качестве дополнительного средства унификации. Собирая модули курсов, выполненных по разным стандартам, аватар делает прозрачными для конечного пользователя их структурные и технические различия. Обучаемый настраивает под себя интерфейсную часть аватара, которая и будет определять тот внешний вид, в котором будет визуализирован материал учебного курса. Аналогичным образом аватар преподавателя позволяет абстрагироваться от технических подробностей отдельных стандартов и спецификаций. Преподаватель настраивает интерфейсную часть аватара, в рамках которой ему удобно вносить материал обучающего курса. Функциональная часть аватара, скрытая от пользователя, позволит подстроить введенный материал под требования определенного стандарта и спецификации.

Кроме того, не существует общепринятого понимания концепции жизненного цикла электронных модулей, курсов и образовательных систем. До сих пор остаются открытыми следующие вопросы.

- На какой научно-технологической базе должна происходить самоадаптация компонентов СДО, чтобы одновременно учесть противоречивые требования со стороны работодателей, образовательных учреждений и самих обучаемых?
- Какие модели изменчивости при этом использовать?

Таким образом, можно схематично рассмотреть основные научные и технологические аспекты, из которых должна складываться всепроникающая СДО настоящего и ближайшего будущего (рис. 1).

## 2. Многоуровневая модель дистанционного обучения на основе моделей изменчивости и сервис-ориентированной архитектуры

Ключевым технологическим аспектом создания и использования всепроникающих СДО является их самоадаптируемость к целому ряду внешних и внутренних условий. Свойство самоадаптируемости является определяющим при решении задач расширения жизненного цикла программных компонентов СДО. Поскольку всепроникающие технологии обучения подразумевают непрерывный цикл оказания дистанционных образовательных услуг, то самоадаптация должна быть прозрачной для конечного пользователя и проходить в реальном режиме времени без остановки функционирования на время перекомпиляции и отладки новых версий.

К настоящему времени задача проектирования линеек программных продуктов без выделения этапа перекомпиляции решается в русле технологии DSPL. При этом какой-либо программный продукт рассматривается не в контексте одной лишь текущей версии, жизненный цикл которой заканчивается с появлением новой версии, а в контексте совокупности всех возможных модификаций и изменений продукта в зависимости от условий окружающей среды и требований пользователя. Неотъемлемой частью такого продукта должна быть модель изменчивости предметной области, заранее предусматривающая и формализующая различные допустимые варианты ее изменения. Далее, на основе возникновения новых вариантов предметной области генерируется новая конфигурация программной части продукта.

В данной работе мы предлагаем концептуальную структуру общедоступной (всепроникающей) СДО, включающую как традиционные компоненты электронного образования, так и компоненты, обеспечивающие управление изменчивостью и самоадаптацией (рис. 2, см. четвертую сторону обложки).



Рис. 1. Основные составляющие общедоступной СДО

В предлагаемой структуре можно выделить пять основных уровней.

1. Уровень взаимодействия (*Level of Interaction*). Включает взаимодействующих аватаров основных участников образовательного процесса: студента и преподавателя. Аватары являются интерфейсной, т. е. видимой для участника образовательного процесса, частью СДО. Аватар студента это интеллектуальная программная оболочка, устанавливающая формат и выборку образовательного контента, определяющая интерфейс для его визуализации, выполняющая адаптацию контента для различных видов программно-аппаратных платформ и каналов передачи данных. Источником контента для аватара студента является аватар преподавателя — интеллектуальная программная оболочка, предоставляющая преподавателю доступ к функциям системы управления обучением (LMS) и тестирования (QTS). Аватары являются самоадаптирующейся частью СДО. Их конфигурация собирается из набора отдельных web-сервисов под управлением соответствующих моделей изменчивости (VM).

2. Уровень управления изменчивостью (*Level of Variability*). Включает модели изменчивости, управляющие текущими конфигурациями аватаров.

3. Уровень сервисов (*Level of Services*). Включает базовый набор web-сервисов, из которых выстраиваются конфигурации аватаров и предоставляется доступ к системам функционального уровня.

4. Функциональный уровень (*Level of Systems*). Включает основные системы управления образовательным процессом, в том числе: систему управления обучением (LMS), систему тестирования и контроля знаний (QTS), систему контроля личной информации (PIMS). LMS отвечает за управление контент-

ной составляющей аватаров в рамках отдельных учебных курсов. QTS обеспечивает процессы промежуточного тестирования и передачу их результатов сначала в LMS и PIMS, а затем оттуда осуществляется связь с уровнем управления изменчивостью для изменения конфигурации аватара студента.

5. Уровень данных (*Level of Data*). Включает базы данных для информационного обеспечения систем функционального уровня. База модулей курсов и база образовательных стандартов обеспечивает LMS. База тестов обеспечивает QTS. База студентов с личной информацией связана траекториями обучения и текущей успеваемостью с PIMS.

Рассмотрим модель изменчивости аватара студента при изучении отдельно взятого курса. Для примера выбран курс "Функциональное программирование" (рис. 3).

Данная модель включает в себя описание трех изменяемых составляющих.

1. *Контент курса*. Определяется структурой изучаемого курса, весь материал которого условно разделен на начальную, базовую и продвинутую части. По мере изучения модулей курса начальной части они должны заменяться на модули базовой, а затем на модули продвинутой частей. Тем самым, по мере успешного освоения модулей курса обеспечивается постепенное изменение контента, транслируемого студенту через его аватар.

2. *Интерфейс аватара*. Определяет внешние формы, в рамках которых студенту транслируется образовательный контент, а также предоставляются элементы управления просмотром. Изменяемость этой составляющей более медленная, поскольку она зависит частично от навыков пользования компьютером студентом, а частично от его психофизических особенностей восприятия информации с экрана.

3. *Программно-аппаратная платформа*. Складывается, во-первых, из типа вычислительного устройства и пропускной способности телекоммуникационного канала, через который обучаемый получает доступ к контенту; во-вторых, из типа базового ПО (операционной системы) и прикладного ПО. Например, при изучении курса "Функциональное программирование" в качестве прикладного ПО могут выступать компиляторы и интерпретаторы для различных видов изучаемых функциональных языков, текстовые редакторы для набора и редактирования программного кода, браузеры для просмотра теоретического материала и мультимедийных приложений, а также программные системы дистанционного компьютерного тестирования.

Таким образом, в ходе изучения курса аватар студента будет постоянно самоадаптироваться, подстраиваясь под уровень текущей успеваемости студента, доступных ему в данный момент аппаратно-вычислительных и программных ресурсов, персо-



Рис. 3. Модель изменчивости аватара студента на примере изучения курса "Функциональное программирование"

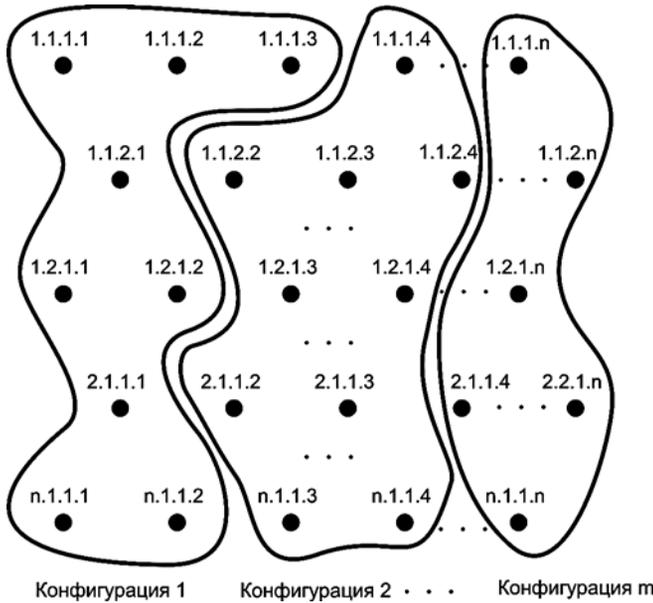


Рис. 4. Гиперграф, в котором веса вершин соответствуют индексам характеристик в модели изменчивости (см. рис. 3), а гиперребра соответствуют возможным конфигурациям аватара

нальных психофизических особенностей. В основе всех этих изменений лежит модель изменчивости аватара, которую можно математически описать в терминах теории гиперграфов. Все пространство возможных конфигураций аватара представляется гиперграфом с динамически изменяемой структурой гиперребра. При этом в качестве вершин гиперграфа выступают все возможные характеристики изменчивости, а динамически изменяемые гиперребра описывают возможные конфигурации аватара, построенные на различных множествах вершин (характеристик изменчивости) (рис. 4).

При этом стоит отметить, что одни характеристики изменчивости модели могут влиять на выбор других характеристик. Например, текущая пропускная способность телекоммуникационных каналов может повлечь изменение интерфейсной составляющей, а составляющая контента может накладывать требования на аппаратное обеспечение и состав требуемого ПО.

Зададим в качестве модели изменчивости гиперграф  $VMG$ , состоящий из двух множеств и предиката:

$$VMG = (V, U, P). \quad (1)$$

Множество  $V$  описывает структуру гиперграфа на уровне вершин:

$$V = \{v_i, (weight)\}, i = 1, 2, \dots, N, \quad (2)$$

где  $N$  — общее число вершин, соответствующее общему числу характеристик модели изменчивости;  $weight$  — вес вершины вида п. 1.1.1 (рис. 3), представляющий собой индекс соответствующей характеристики в иерархической структуре модели изменчивости.

Множество  $U$  имеет мощность, соответствующую числу возможных конфигураций аватара:

$$U = \{u_j\}; j = 1, 2, \dots, K, \quad (3)$$

где  $K$  — число гиперребра.

Очевидно, что в зависимости от объема и структуры каждого учебного курса, от технических и телекоммуникационных возможностей, доступных студенту в определенный момент и от некоторых других особенностей мощность множества  $U$  может существенным образом меняться.

Предикат  $P$  — определяет инцидентность вершин и гиперребра каждого слоя.  $P$  определен на множестве всех пар  $(v \in V, u \in U)$ . Областью истинности предиката  $P$  является множество  $R$  переменной мощности  $B_r \neq \text{const}$ :

$$F(P) = \{(v, u) | P(v, u)_r\}, \quad (4)$$

где  $v \in V, u \in U, r \in R = \{1, 2, \dots, B_r\}$ .

Переменность мощности множества  $R$  обусловлена теми же причинами, что и переменность множества  $U$  в уравнении (3).

Рассмотренное теоретико-множественное представление модели изменчивости позволяет определить матричное представление этой модели, удобное для создания программно-алгоритмического обеспечения СДО. Матричное представление (матрица инцидентности размера  $N \times K$ ) гиперграфа будет иметь следующий вид:

$$M_f = \|m_{ij}\|_{N \times K}, \quad (5)$$

где

$$m_{ij} = \begin{cases} 1, & \text{если } (v_i, u_j) \in F(P), v \in V, u \in U, \\ 0, & \text{если } (v_i, u_j) \notin F(P), v \in V, u \in U. \end{cases}$$

В некоторых случаях удобнее использовать матрицу связности вершин гиперграфа, которая имеет размер  $N \times N$  и отражает попарные отношения связности вершин через инцидентные гиперребра:

$$M_c = \|m_{ij}\|_{N \times N}, \quad (6)$$

где

$$m_{ij} = \begin{cases} 1, & \text{если для } (v_i, v_j) \exists u_k, (v_i, u_k) \in F(P), \\ & (v_j, u_k) \in F(P), v \in V, u \in U, \\ 0, & \text{если для } (v_i, v_j) \neg (\exists u_k), (v_i, u_k) \in F(P), \\ & (v_j, u_k) \in F(P), v \in V, u \in U. \end{cases}$$

На рис. 5 показан процесс взаимодействия модели изменчивости и конфигураций аватара студента в зависимости от успешности прохождения учебных курсов (по результатам тестирования и индивидуальным ответам на вопросы преподавателя), доступных технических и телекоммуникационных возможностей, психофизических особенностей студента. На основе анализа текущего профиля студента выбирается оптимальная конфигурация

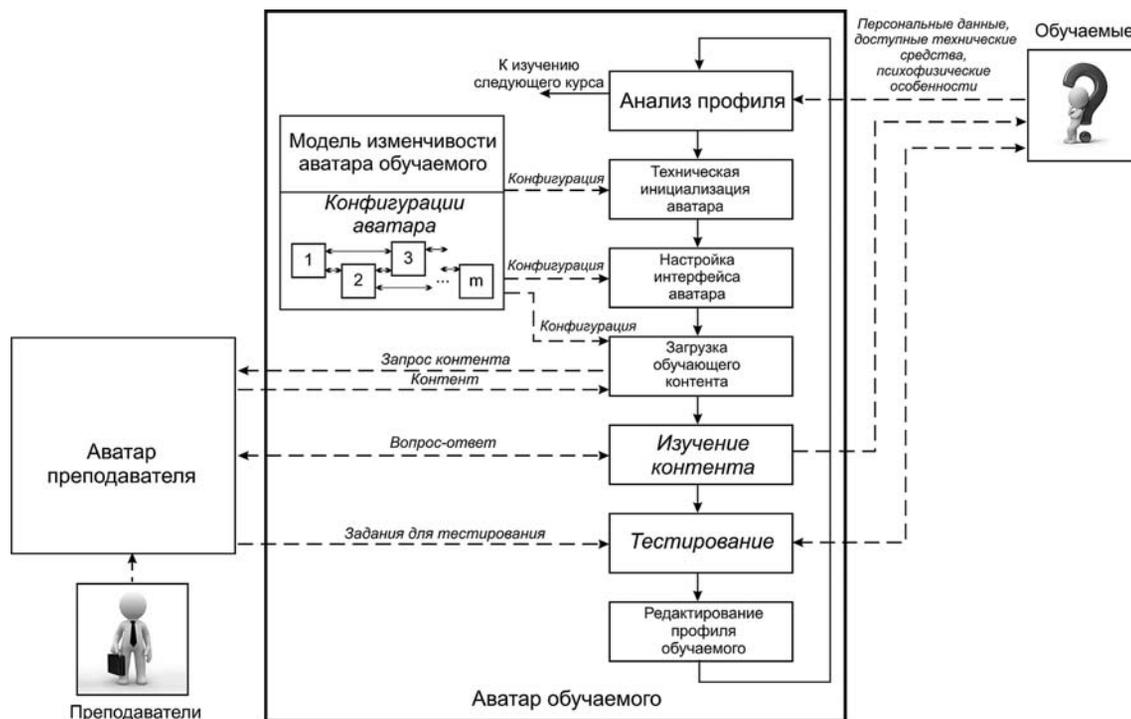


Рис. 5. Процесс взаимодействия модели изменчивости и конфигураций аватара студента

аватара (гиперребро графа VMG), определяемая технической, интерфейсной и контентной составляющей (вершинами, инцидентными данному гиперребру).

После загрузки и изучения контента курса, прохождения тестирования и удаленного собеседования преподавателя со студентом (через аватар преподавателя) в профиль студента вносятся соответствующие изменения. После этого конфигурация аватара студента меняется, происходит подгрузка следующих разделов контента и интерфейсная настройка его визуализации. Если курс пройден полностью, то возможен переход к изучению следующего курса с загрузкой в аватар студента новой модели изменчивости.

Удобство предлагаемого подхода заключается в том, что модульная структура учебных курсов имеет типовую иерархическую структуру и допускает вставку специальных тегов (администратором системы или преподавателем), разграничивающих курс по тематическим разделам и уровням сложности. Наличие таких тегов позволяет далее строить гиперграфовые модели изменчивости для каждого курса в автоматическом режиме уже без участия преподавателя.

### Заключение

Таким образом, создание систем дистанционного обучения, обладающих свойствами мобильности, адаптивности и сервис-ориентированности, является в настоящее время актуальной научно-практической проблемой и перспективным направле-

нием развития электронного образования в целом. В статье предложен один из возможных подходов к построению общедоступной (всепроницающей) единой интеллектуальной среды для оказания электронных образовательных услуг.

Предлагаемый подход основан на интеграции технологий сервис-ориентированного проектирования, инженерии динамических линеек программных продуктов, беспроводной телекоммуникации и взаимодействия интеллектуальных агентов электронной образовательной среды (hhh-технология). Структурной основой предлагаемой системы дистанционного образования является взаимосвязанная пара интеллектуальных программных агентов (аватаров): аватара студента и аватара преподавателя, реализованных с использованием сервис-ориентированной архитектуры. Аватары способны самоадаптироваться к текущему уровню знаний обучаемого, доступному в текущий момент программно-техническому и телекоммуникационному обеспечению, требованиям окружающей среды (например, рынкам труда) и различным образовательным стандартам. В качестве основы самоадаптации аватаров предложена модель изменчивости, включающая три базовых иерархии характеристик: образовательный контент; интерфейс; программно-техническое обеспечение. При этом математическое описание модели изменчивости реализовано с использованием теории гиперграфов.

Достигнутые результаты позволяют подойти к решению фундаментальной проблемы создания единой межгосударственной интеллектуальной среды для оказания электронных образовательных услуг,

в которую будут включены гетерогенные формы представления знаний, межгосударственные стандарты и формы обучения, международный преподавательский и студенческий состав. Построение такой среды открывает новую фазу развития электронного образования (E-Learning 3.0), в которой ведущую роль будут играть распределенные компьютерные системы, облачные технологии, мобильные персональные устройства, системы искусственно-го интеллекта и средства виртуальной реальности.

#### Список литературы

1. Bjork E., Ottosson S., Thorsteinsdottir S. E-Learning for All. Chapter 3. In: E-Learning: 21<sup>st</sup> Century Issues and Challenges. Nova Science Publishers Inc., 2008. P. 49–69.
2. Bershady A. M., Krevsky I. G. The Organization of Distance Education in Region // Proceedings of the Second Interna-

tional Conference on Distance Education in Russia: Open and Distance Learning as a Development Strategy, Moscow, 1996. P. 174–176.

3. Бершадский А. М., Финогеев А. Г., Бождай А. С. Разработка и моделирование гетерогенных инфраструктур для беспроводного информационного обеспечения процессов мониторинга // Известия высших учебных заведений. Поволжский регион. Технические науки. 2010. № 1. С. 36–46.

4. Czarnecki K., Grunbacher P., Rabiser R., Schmid K., Wadowski A. Cool Features and Tough Decisions: A Comparison of Variability Modeling Approaches, VaMoS'12, January 25–27, 2012, Leipzig, Germany.

5. Czarnecki K., Helsen S. Feature-based survey of model transformation approaches // IBM Systems Journal. 2006. N. 45 (3). P. 621–646.

6. Papazoglou M. P., Heuvel W.-J. Service oriented architectures: approaches, technologies and research issues // The VLDB Journal. 2007. Vol. 16, N. 3. P. 389–415.

7. Mkrttchian V. Modeling using of Triple H-Avatar Technology in online Multi-Cloud Platform Lab / Mehdi Khosrow-Pour (Ed). Encyclopedia of Information Science and Technology (3<sup>rd</sup> Ed.). Hershey, PA: IGI Global, 2014. P. 116–141.

A. M. Bershady, D. Sc., Professor, Chief of CAD dept., bam@pnzgu.ru,

A. S. Bozhday, D. Sc., Professor of CAD dept., bozhday@yandex.ru,

Penza State University, Penza, Russia,

V. S. Mkrttchian, D. Sc., Professor of CAD dept., hhhuniversity@gmail.com

Triple H Hamalsaran of HHH TECHNOLOGY Inc. "HHH University", Sydney, Australia

## Construction Principles of Pervading Self-Adapting Distance Learning System Based on Variability Model and Service-Oriented Architecture

*The paper is devoted to the development of a new phase of e-learning (E-Learning 3.0) in which will play an important role: distributed computer systems, cloud computing, mobile personal devices and wireless networks, artificial intelligence and virtual reality tools. Today, there is a fundamental scientific problem of creating a unified interstate intellectual environment for e-education services. This environment should include heterogeneous forms of knowledge representation, interstate standards and forms of learning, international faculty and student teams.*

*The paper suggests a possible approach to the construction of an all-pervading intellectual environment for e-learning services. The proposed approach is based on the integration of multiple technologies, such as: service-oriented design, engineering of DSPL (Dynamic Software Product Lines), wireless telecommunications, interaction of intelligent e-learning agents. Structural basis of this approach is interconnected pair of intelligent software agents ("avatars"): student's avatar and teacher's avatar. These avatars are able to adapt itself: to the current level of student's knowledge; to the available software, hardware and network equipment; to the current demands of the environment (e.g. labor markets); to the modern educational standards. As a basis of avatars self-adaptation is a variability model, including the three basic features hierarchy: educational content, interface, software and technical support. The mathematical description of this variability model is implemented using the theory of hypergraphs. Application of such variability model does not require recompilation of the source code (to make changes in the avatars properties) and allows to organize a continuous process of e-learning and significantly increase the life cycle of the entire system of distance learning.*

**Keywords:** system of distance education, E/U Learning 3.0, variability model, service-oriented architecture, hypergraph, DSPL, intelligent e-learning agents, knowledge representation

#### References

1. Bjork E., Ottosson S., Thorsteinsdottir S. E-Learning for All. Chapter 3, *E-Learning: 21<sup>st</sup> Century Issues and Challenges*, Nova Science Publishers Inc., 2008, pp. 49–69.
2. Bershady A. M., Krevsky I. G. The Organization of Distance Education in Region, *Proceedings of the Second International Conference on Distance Education in Russia: Open and Distance Learning as a Development Strategy*, Moscow, 1996, pp. 174–176.
3. Bershady A. M., Finogeev A. G., Bozhday A. S. Development and modeling of heterogeneous infrastructures for wireless information support of monitoring process, *Proceedings of Higher Education. Volga region. Technical sciences*. 2010, no. 1, pp. 36–46.

4. Czarnecki K., Grunbacher P., Rabiser R., Schmid K., Wadowski A. Cool Features and Tough Decisions: A Comparison of Variability Modeling Approaches, VaMoS'12, January 25–27, 2012, Leipzig, Germany.

5. Czarnecki K., Helsen S. Feature-based survey of model transformation approaches, *IBM Systems Journal*, 2006, no. 45 (3), pp. 621–646.

6. Papazoglou M. P., Heuvel W.-J. Service oriented architectures: approaches, technologies and research issues, *The VLDB Journal*, 2007, vol. 16, no. 3, pp. 389–415.

7. Mkrttchian V. Modeling using of Triple H-Avatar Technology in online Multi-Cloud Platform Lab. In Mehdi Khosrow-Pour (Ed). *Encyclopedia of Information Science and Technology (3<sup>rd</sup> Ed.)*. Hershey, PA: IGI Global, 2014.

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИМЕДИЦИНСКИХ СИСТЕМАХ INFORMATION TECHNOLOGIES IN BIOMEDICAL SYSTEMS

УДК 60::004.3

**А. А. Кухтичев**, аспирант, e-mail: a.kukhtichev@mail.ru,  
**Е. А. Клёнов**, аспирант, e-mail: eaklenov@gmail.com,  
**С. В. Скородумов**, доцент, канд. техн. наук, e-mail: skorodum@gmail.com,  
Московский авиационный институт (национальный исследовательский университет),  
Россия, г. Москва

## Разработка архитектуры информационной системы "ЦифроМед" цифровой медицины в авиации и космонавтике

*Представлен системный подход к созданию web-сервисов цифровой медицины в авиации и космонавтике на основе методов и технологий биологической обратной связи (БОС), а также рассмотрены исторические предпосылки БОС-технологий. Основное внимание в работе направлено на описание архитектуры информационной системы "ЦифроМед", которая после ее реализации позволит в режиме реального времени контролировать состояние здоровья летных экипажей и авиадиспетчеров (т. е. именно тех пользователей системы, от которых зависит безопасность полетов).*

**Ключевые слова:** цифровая медицина (ЦМ), информационная система (ИС), информационный портал (ИП), биологическая обратная связь (БОС), носимые устройства микроэлектроники (НУМ)

### Введение

В основе цифровой медицины (ЦМ) как нового направления развития здравоохранения в XXI веке лежат методы биологической обратной связи (БОС) с человеком как биологическим объектом управления, на основе измерения его параметров жизнедеятельности (частоты сердечных сокращений, артериального давления, мышечной силы и др.). Научной основой для создания методов БОС стали фундаментальные исследования механизмов регуляции физиологических процессов у человека и животных [1–8].

Благодаря развитию инфокоммуникационных технологий (ИКТ) появилась возможность предоставлять пользователям обратную связь не только в виде графиков и звуковых сигналов, но и создавать полноценные пользовательские интерфейсы (GUI), в том числе игровые сюжеты, управление которыми также основано на технологиях БОС.

По программе научных исследований Московский авиационный институт выполняет комплексную тему НИОКР "Цифровая медицина в авиации и космонавтике" на основе взаимодействия творческих проектных команд из числа студентов, аспирантов и преподавателей МАИ факультетов "Аэрокосмического", "Системы управления, информати-

ка и электроэнергетика", "Прикладная математика и физика" в целях создания информационного портала цифровой медицины "ЦифроМед" в авиации и космонавтике. Основное внимание исследований направлено на создание инструментария для прототипирования и разработки специальных носимых устройств микроэлектроники (НУМ) в виде датчиков первичной информации, данные которых передаются в облако и накапливаются в базах данных (БД) в центре обработки данных (ЦОД), чтобы как медицинские специалисты, так и конечные пользователи системы постоянно были на связи, в результате чего самочувствие пользователей находилось бы под контролем 24 ч в сутки. Система "ЦифроМед" при ее внедрении позволит космонавтам, пилотам, штурманам, диспетчерам и другим авиаспециалистам контролировать параметры жизнедеятельности организма (показатели здоровья) и оперативно пересылать результаты измерений на виртуальную машину в центре обработки данных (ЦОД), эксперту, специалисту в области ЦМ, а также личному врачу.

Целью программы научных исследований МАИ является создание информационного портала "ЦифроМед" — цифровой медицины, который бы обеспечивал пользователей оперативной информа-

цией и предоставлял доступ к медицинским релевантным данным.

Одно из главных направлений развития технологий БОС в рамках цифровой медицины связано с биосенсорами, датчиками первичной информации и носимыми устройствами микроэлектроники (НУМ), которые реально дают возможность пользователям БОС-системы контролировать состояние своего организма. В свою очередь, система "ЦифроМед" может быть интегрирована в центр подготовки космонавтов для контроля показателей жизнедеятельности космонавтов как во время подготовки, так и на всех этапах полета, а также во время пребывания экипажа на орбите.

### Разработка архитектуры информационной системы "ЦифроМед"

Применение технологий БОС за последние 10 лет уже дало существенные результаты, когда десятки тысяч людей в мире получили возможность реально контролировать состояние своего здоровья.

В рамках системы "ЦифроМед" предусмотрена возможность использования 3D-модели организма основного пользователя системы (космонавта, пилота, штурмана, авиадиспетчера и др.). Для построения такой модели планируется использовать

результаты применения метода компьютерной томографии. Такие разработки ведутся, в частности, в рамках специальных программ Нижегородского государственного университета [9].

Концептуальная модель в виде интеллект-карты портала "ЦифроМед" представлена на рис. 1.

Основными пользователями системы "ЦифроМед" являются нилоты, космонавты, штурманы и другие авиаспециалисты, но в перспективе применение системы "ЦифроМед" возможно и в гражданском секторе (для всех жителей Российской Федерации). Для контроля показателей их жизнедеятельности требуются новые решения в области использования НУМ для мониторинга состояния жизненных показателей пользователя. И таким решением может стать концептуальная одежда пилота. Одежда состоит из нескольких модулей:

1. *Умный костюм.* Включает в себя датчики БОС, которые собирают информацию об основных показателях (ЭКГ, ЧСС, ДАС и др.).

2. *Умный браслет.* Отслеживает каждое движение пользователя, напоминает о необходимости движения, когда пользователь слишком долго не проявляет физической активности, осуществляет автоматический контроль часов сна, когда фиксируется соотношение легкого и глубокого сна и времени пробуждения.

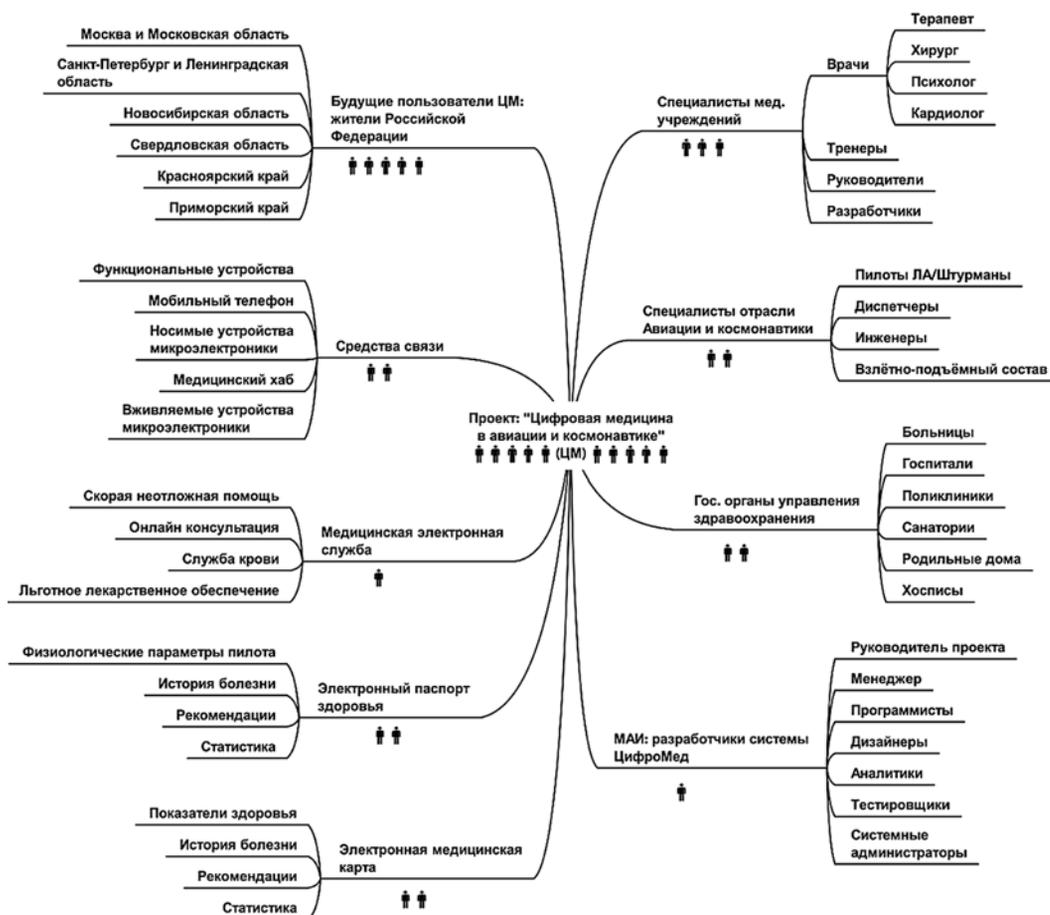


Рис. 1. Концептуальная модель проекта "ЦифроМед" развития методов цифровой медицины

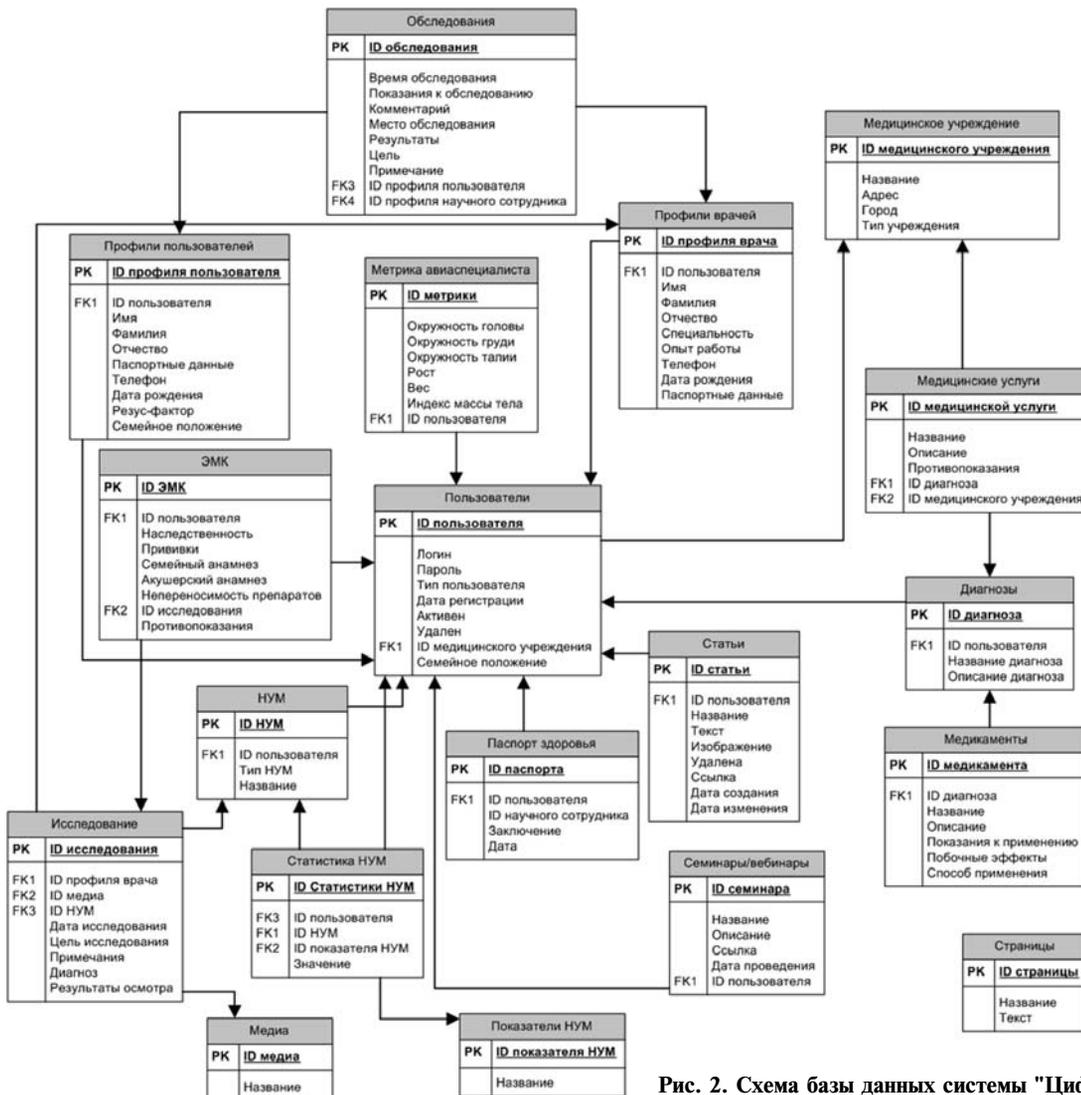


Рис. 2. Схема базы данных системы "ЦифроМед"

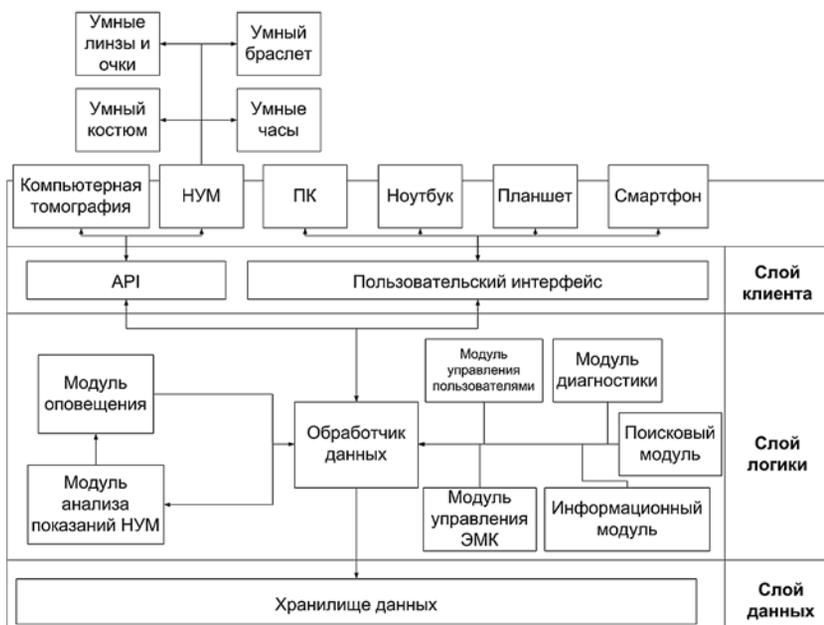


Рис. 3. Архитектура информационной системы "ЦифроМед"

3. *Умные часы*. Отображают показатели самочувствия пользователя в режиме реального времени, предупреждают об отклонениях.

4. *Умные линзы и очки*. Помогают следить за маршрутом полета с помощью расширенной реальности, записывают и транслируют в сеть аудио/видео данные окружающей обстановки.

Схема базы данных системы "ЦифроМед" показана на рис. 2.

Здесь представлены основные классы-модели БД, которые реализованы на портале "ЦифроМед". В качестве основных приняты те разделы электронной медицинской карты (ЭМК), которые предложены Министерством здравоохранения Российской Федерации в документе "Основные разделы Электронной медицинской карты" от 11 ноября 2013 г. [10]. Все разделы и

поля, перечисленные в этом документе не могут быть отображены на рис. 2, так как схема базы данных получилась бы большой и малопонятной. В разрабатываемом прототипе системы "ЦифроМед" используются динамические таблицы, позволяющие добавлять необходимые поля.

Для оптимизации базы данных и уменьшения числа операций JOIN, схема БД выстроена вокруг центрального элемента — пользователя.

Таблица пользователей предусматривает определение типа пользователя, в зависимости от которого подключается либо профиль пользователя (пациента), либо профиль лечащего врача. Если тип пользователя — "пациент", то к таблице также подключается "Метрика пациента", описывающая его основные физические характеристики.

С таблицей пользователя также связана таблица "Медицинское учреждение" (в зависимости от типа пользователя — либо это место работы, либо место обследований/лечения). Медицинское учреждение предполагает наличие "Медицинских услуг", которые в свою очередь относятся к определенному "Диагнозу". Для каждого диагноза, помимо медицинских услуг, существует набор медикаментов из соответствующей таблицы.

К пользователю с типом профиля "пациент" привязана таблица "ЭМК", а также таблицы "НУМ", "Показатели НУМ" и "Статистика НУМ", работающие с данными, полученными от носимых устройств микроэлектроники.

Пользователи с типом профиля "лечащий врач" могут писать статьи и организовывать семинары и/или вебинары. В системе также предусмотрена таблица для создания статической отчетности по страницам портала.

К таблице "ЭМК" привязана таблица "Исследование", которая связана в свою очередь с таблицами "Профиль врача", "НУМ" и "Медиа". В таблицу "Медиа" добавляются пути к видео/аудио/графическим файлам в файловой системе, поступающие в результате проведения исследования (МРТ, КТ, УЗИ и др.). Благодаря столбцу "Дата исследования" и "НУМ" можно получить данные от носимых устройств микроэлектроники, которые были задействованы в период исследования.

В основе системы "ЦифроМед" лежит трехуровневая архитектура (рис. 3) — архитектурная модель, предполагающая наличие трех компонентов: клиента (слой клиента — пользовательский интерфейс); сервера приложений (слой логики — программные модули и обработчик данных) и сервера базы данных (слой данных — хранилище большого объема данных).

Пользовательский интерфейс предназначен для обмена данными с основным пользователем системы и его лечащим врачом, а также для получения и передачи данных посредством специального API от НУМ и результатов проведения компьютерной томографии. Слой логики обеспечивает все вычислительные процессы, подготовку и обработку данных. Хранение больших объемов данных обеспечивается с помощью хранилища данных. Взаимодействие между модулями и архитектурными слоями обеспечивает обработчик данных.

Рассмотрим более детально слой логики:

- модуль анализа показателей НУС анализирует показатели НУС, если они превышают норму, передает информацию модулю оповещения;
- модуль оповещения передает сигнал на НУМ в виде вибрации или звонка, а также данные о результатах анализа показателей пользовательскому интерфейсу;
- модуль управления пользователями обеспечивает логику регистрации, входа, разграничивает области доступа для разных типов пользователей, а также организует информацию для личного кабинета врача;
- модуль управления электронной медицинской картой организует логическую последовательность действий при работе с ЭМК;
- модуль диагностики управляет статистикой показаний, статистикой данных с НУМ в составе "паспорта здоровья";
- информационный модуль систематизирует информацию, разбивая ее по отдельным рубрикам, а также отвечает за механизм публикаций,

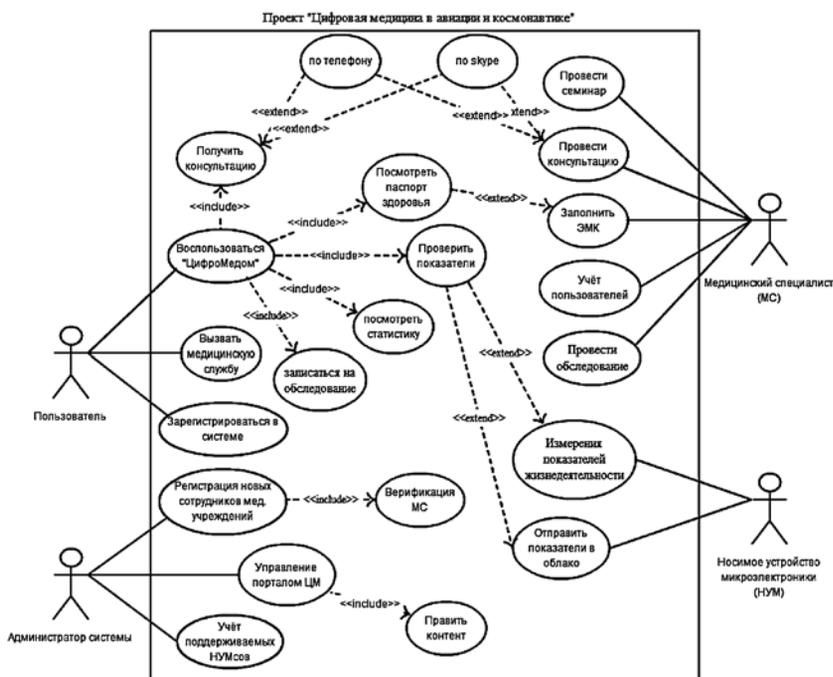


Рис. 4. Диаграмма использования UseCase проекта "ЦифроМед"

редактирования и удаления различных элементов контента портала;

- поисковый модуль обеспечивает корректную выдачу результатов поиска.

Информационная архитектура (ИА) системы относится к слою клиента и проектируется до создания GUI — пользовательского интерфейса, поскольку сам интерфейс лишь обеспечивает взаимодействие между пользователем и БОС-системой (человеком и информацией). На основе полученной информации пользователь портала "ЦифроМед" сможет принимать решения.

В данной работе под информационной архитектурой будем понимать систему организации информации по технологиям БОС на портале "ЦифроМед".

Информационная система "ЦифроМед" должна помогать специалистам решать следующие задачи.

1. Измерение показателей жизнедеятельности т. е. оцифровывание или "цифроризация" конечного пользователя.

2. Привязка пользователя к медицинским учреждениям и специалистам (врачу или группе врачей), которые должны контролировать показатели жизнедеятельности данного пользователя.

3. Выявление отклонений в здоровье пользователя на ранних этапах и выдача рекомендаций по лечению заболевания.

4. Проведение консультаций и семинаров специалистов.

Система БОС (рис. 4) в процессе работы должна использоваться как инструмент и взаимодействовать с другими системами. В качестве экторов здесь представлены следующие сущности.

- *Пользователь* — пилот, штурман, диспетчер и другие профессионалы, для которых при выполнении их профессиональных обязанностей необходим контроль состояния здоровья со стороны медицинских специалистов.
- *Сотрудник-эксперт* — профессиональный врач, помогающий пользователю пройти обследование, отвечает на все интересующие его вопросы, выполняет свою главную функцию — контролирует состояние конечного пользователя, а также может изучать новые методы или делиться своими знаниями по медицинским темам.
- *Администратор портала*. Регистрирует новых пользователей, добавляет по необходимости новый контент, следит за актуальностью и обновлением данных, а также обслуживает заказы пользователей.
- *Носимые устройства микроэлектроники (НУМ)*. Контролируют показатели здоровья (параметры пользователя) и отправляют их в хранилище данных информационной системы на информационный портал.

При проектировании ИА главное внимание уделяется двум активным экторам: 1) основной пользователь, в интересах которого и создается информационная БОС-система "ЦифроМед"; 2) врач как сотрудник медицинского учреждения, контролирующего состояние здоровья пользователя. Для каждого из указанных экторов разрабатывается собственный пользовательский интерфейс и особый доступ к нужной информации.

Конечный пользователь системы и лечащий врач в роли медицинского специалиста (МС) будут вовлечены в процессы и процедуры с использованием механизмов информационного портала "ЦифроМед", которые приведены в табл. 1.

Важность данных процессов и процедур ранжируется экспертами.

Таблица 1

Перечень процессов и процедур в портале "ЦифроМед"

Пользователь	Лечащий врач
Авторизация в системе	Авторизация в системе
Проверка показателей с НУМ	Просмотр паспорта здоровья пользователя
Просмотр паспорта здоровья	Анализ статистики показаний с НУМ
Просмотр ЭМК	Проведение консультации
Анализ статистики показаний	Проведение обследования
Консультация со специалистами	Заполнение ЭМК
Вызов медицинской службы	Проведение online-семинара/вебинара
Запись на обследование	Учет пользователей
Посещение online-семинаров/вебинаров	
Поиск информации к установленному врачом диагнозу	

Таблица 2

Основные сущности портала "ЦифроМед" для экторов

Пользователь	Лечащий врач
Кнопка экстренного оповещения МС и служб медицинской помощи	Профили пользователей — все его данные, а также оповещения о новых показателях и экстренных вызовах
Профиль лечащего врача пользователя (содержит контактные данные и пр.), а также форма записи на обследование	Личный кабинет — информация о специалисте, статьи, организация семинаров, график записи пользователей, история правок по ЭМК, связан с механизмом авторизации
Личный кабинет (профиль пользователя с диагнозом) — показатели НУМ, статистика показаний, паспорт здоровья, ЭМК, план лечения, связан с механизмом авторизации	Поиск по сайту
Механизм консультации	Информационная часть портала
Поиск по сайту	
Информационная часть портала — персонализируется с учетом диагноза показателей состояния здоровья пользователя	

На основе указанных процессов и процедур формируются основные сущности информационной системы (табл. 2).

Основываясь на указанных здесь сущностях портала "ЦифроМед" для пользователя и врача, проранжированных в порядке важности экспертами, а также на базе методологии из опубликованной в работах [11–13], разработана информационная архитектура портала цифровой медицины, представленная на рис. 5 (для пользователя) и рис. 6 (для лечащего врача).

На рис. 7 (см. четвертую сторону обложки) представлен разработанный прототип портала "ЦифроМед". Интерфейс системы построен на основе информационной архитектуры. "ЦифроМед" представляет собой набор взаимосвязанных функциональных модулей, взаимодействующих с пользователем посредством веб-интерфейсов множества страниц сайта. Доступ к отдельным страницам сайта регулируется ролевой политикой безопасности.

### Заключение

Результаты данного исследования легли в основу проекта создания информационного портала "ЦифроМед" для online коммуникаций участников научного сообщества по проблеме "Цифровая медицина в авиации и космонавтике".

В ходе исследования определена аудитория проекта "ЦифроМед", разработана концептуальная модель системы, а также ее основные модули, подготовлены учебные материалы и документация для взаимодействия всех участников проекта, а также разработчиков с пользователями системы. На этапе реализации разработаны схема базы данных системы, информационная архитектура, а также макеты графического интерфейса портала. На основе результатов данной работы реализован прототип системы цифровой медицины "ЦифроМед".

Внедрение системы "ЦифроМед" в аэрокосмическую отрасль позволит свести к минимуму вероятность возникновения аварий и авиакатастроф, вызванных человеческим фактором, исключить разнообразные риски, а также улучшить качество подготовки будущих авиаспециалистов.

### Список литературы

1. Сеченов И. М. Рефлексы головного мозга. М.: Изд-во Академии наук СССР, 1961. 100 с.
2. Павлов И. П. Условный рефлекс. Л.: Лениздат, 2014. 224 с.
3. Быков К. М., Курцин И. Т. Кортико-висцеральная теория патогенеза язвенной болезни. М.: Изд-во Академии наук СССР, 1952. 271 с.

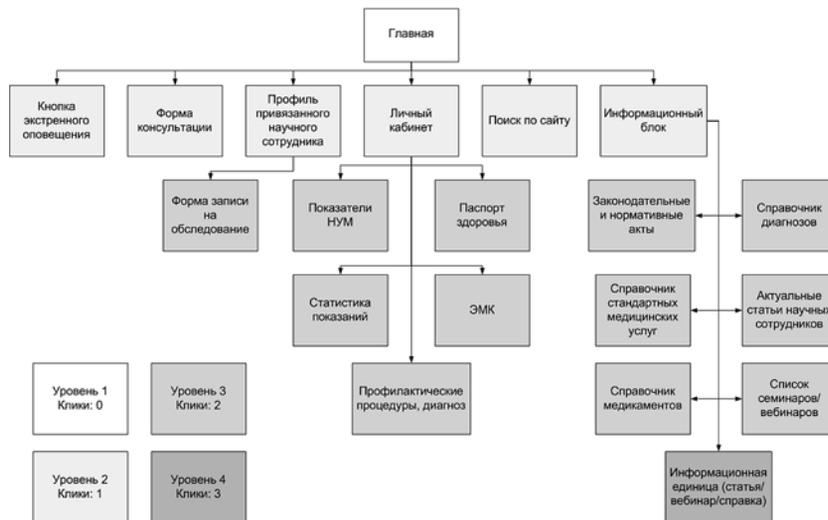


Рис. 5. Информационная архитектура портала ЦМ для конечного пользователя

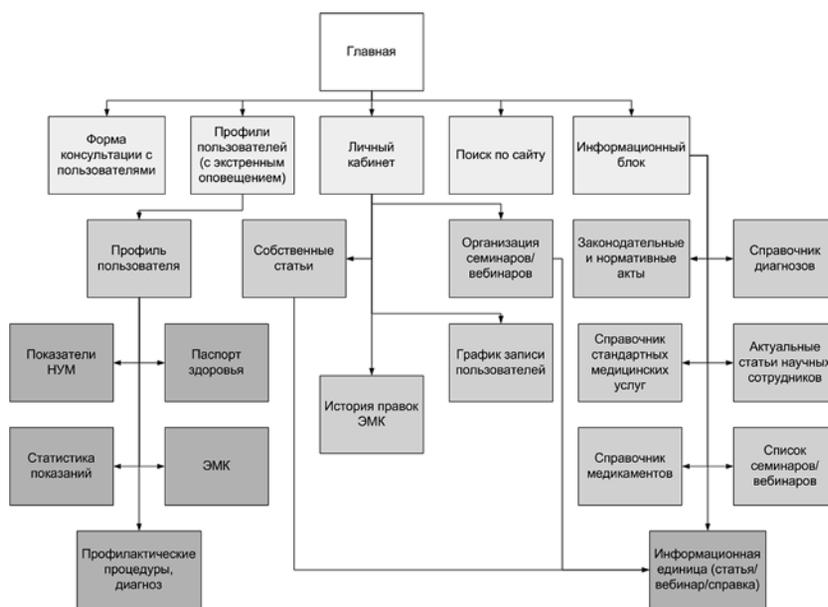


Рис. 6. Информационная архитектура портала ЦМ для лечащего врача

4. Анохин П. К. Избранные труды. Философские аспекты теории функциональной системы. М.: Наука, 1978. 400 с.
5. Бехтерева Н. П. Нейрофизиологические аспекты психической деятельности человека. Л.: Медицина, 1974. 151 с.
6. Peniston E. G., Kulkosky P. J. Alcoholic personality and alpha-theta brainwave training // Journal of Medical Psychotherapy. 1991. N. 2. P. 37–55.
7. Сметанкин А. А. Дыхание по Сметанкину. СПб.: Биосвязь, 2007. 160 с.
8. Базанова О. М. Современная интерпретация альфа-активности электроэнцефалограммы // Успехи физиологических наук. 2009. Т. 40. № 3. С. 32–53.
9. 3D-диагностика: нижегородские ученые нашли способ облегчить труд врачей. URL: <http://www.ntv.ru/novosti/140-1882/?f> (дата обращения: 07.05.2015).
10. Минздравом России утверждена структура электронной медицинской карты / Министерство здравоохранения Российской Федерации. URL: <http://www.rosminzdrav.ru/news/2013/11/20/1314-minzdravom-rossii-utverzhdena-struktura-elektronnoy-meditsinskoy-karty> (дата обращения: 12.12.2014).
11. Wurman R. S. Information Architects. Zurich: Graphis Press, 1997. 235 p.
12. Rosenfeld L., Morville P. Information architecture for the World Wide Web. O'Reilly & Associates, 2006. 528 p.
13. Garrett J. J. A visual vocabulary for describing information architecture and interaction design. URL: <http://www.jjg.net/ia/vis-vocab/> (дата обращения: 07.05.2015).

A. A. Kukhtichev, Postgraduate Student, a.kukhtichev@mail.ru,  
E. A. Klenov, Postgraduate Student, eaklenov@gmail.com,  
S. V. Skorodumov, Ph. D., Associate Professor, skorodum@gmail.com  
Moscow Aviation Institute (National Research University), MAI

## Development of Architecture of Digital Health Informational System "CifroMed" in Aviation and Aerospace

Digital health (DH) as a new direction of the healthcare development in the XXI century is based on Biofeedback (BFB). In its turn, fundamental studies of regulation mechanisms of physiological processes in humans and animals provided the basis for the development of biofeedback techniques.

One of the main directions of development of technologies of the digital health is connected with the medical biosensors, sensors of the primary information and wearable devices of microelectronics (WDM), which really enable users of biofeedback system to monitor the status of the organism. The using of biofeedback technologies in the past decade has yielded significant results, when ten thousands humans in the world had the opportunity to take control of their health.

The focus of this work is aimed at the architecture of information system "TsifroMed" which after its implementation will enable the real-time monitor the health of flight crews and air traffic controllers (ie, it is the users of the system, on which depends the safety of flight).

The aim of the science research program of Moscow aviation institute (MAI) is the creation of information portal "TsifroMed" — digital health, which would provide to users of operational information and provide an access to relevant medical data.

The system "TsifroMed" has a three-tier architecture — an architectural model that assumes the presence of three components: the client (the client layer — the user interface), the application server (the logic layer — software modules and data handler) and database server (data layer — the repository of a large volume data).

The results of this research formed the basis of the project to creation of an information portal "TsifroMed" for on-line communication of participants of the scientific community on the issue of "digital health in aviation and aerospace."

The use case diagram illustrates what actions to perform each member of the project on the portal "TsifroMed." Formed general requirements for information system architecture "Tsifromed."

A conceptual model of the system for further details and documentation has been prepared to communicate with users of the system.

Picked audience of the project and the main modules of the system "TsifroMed", developed the information architecture of the portal. At the stage of implementation developed the database scheme of the system "TsifroMed."

**Keywords:** Digital health (DH), informational architecture (IA), biofeedback (BFB), wearable device of microelectronics (WDM)

### References

1. Sechenov I. M. *Refleksy golovnogo mozga* (Reflexes of the Brain), Moscow, Izdatel'stvo Akademii nauk SSSR, 1961, 100 p.
2. Pavlov I. P. *Uslovnyy refleks* (Conditioned Reflexes), Saint Petersburg, Lenizdat, 2014, 224 p.
3. Bykov K. M., Kurtsin I. T. *Kortiko-vistseral'naya teoriya patogeneza yazvennoy bolezni* (The Corticovisceral Theory of the Pathogenesis of Peptic Ulcer), Moscow, Izdatel'stvo Akademii nauk SSSR, 1952, 271 p.
4. Anokhin P. K. *Izbrannye trudy. Filosofskie aspekty teorii funktsional'noy sistemy* (Theory of functional systems in the scientific school), Moscow, Nauka, 1978, 400 p.
5. Bekhtereva N. P. *Neyrofiziologicheskie aspekty psikhicheskoy deyatelnosti cheloveka* (The Neurophysiological Aspects of Human Mental Activity), Leningrad, Meditsina, 1974, 151 p.
6. Peniston E. G., Kulkosky P. J. Alcoholic personality and alpha-theta brainwave training, *Journal of Medical Psychotherapy*, 1991, vol. 2, pp. 37–55.
7. Smetankin A. A. *Dykhaniye po Smetankiny* (Breathing with Smetankin), Saint Petersburg, Biosvyaz, 2007, 160 p.
8. Bazanova O. M. *Uspehi fiziologicheskikh nauk*, 2009, vol. 40, no. 3, pp. 32–53.
9. *3D-diagnostika: nizhegorodskie uchenye nashli sposob oblegchit' trud vrachey*, available at <http://www.ntv.ru/novosti/1401882/?f> (accessed 07.05.2015).
10. *Minzdravom Rossii utverzhdena struktura elektronnoy medicinskoj karty*, Ministerstvo zdravoohraneniya Rossiyskoy Federacii, available at <http://www.rosminzdrav.ru/news/2013/11/20/1314-minzdravom-rossii-utverzhdena-struktura-elektronnoy-meditsinskoj-karty> (accessed 12.12.2014).
11. Wurman R. S. *Information Architects*, Zurich, Graphis Press, 1997, 235 p.
12. Rosenfeld L., Morville P. *Information architecture for the World Wide Web*. O'Reilly & Associates, 2006, 528 p.
13. Garrett J. J. *A visual vocabulary for describing information architecture and interaction design*, available at <http://www.jjg.net/ia/visvocab/> (accessed 07.05.2015).

### Адрес редакции:

107076, Москва, Стромынский пер., 4

Телефон редакции журнала (499) 269-5510

E-mail: [it@novtex.ru](mailto:it@novtex.ru)

Технический редактор *Е. В. Конова*.

Корректор *Т. В. Пчелкина*.

Сдано в набор 04.12.2015. Подписано в печать 27.01.2016. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ ИТ216. Цена договорная.

Журнал зарегистрирован в Министерстве Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-15565 от 02 июня 2003 г.

Оригинал-макет ООО "Авансед солюшнз". Отпечатано в ООО "Авансед солюшнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.