

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

1(209)  
2014

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ  
Издательство "Новые технологии"

## СОДЕРЖАНИЕ

### ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Баранов М. А. Составные ключевые термины в задаче кластеризации текстовых документов. . . . . 3

### МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

Валеева А. Ф., Гончарова Ю. А., Кошечев И. С. Разработка логистической транспортной системы для решения задачи доставки груза различным клиентам. Часть 2. . . . . 9  
Четырбоцкий А. Н. Численное моделирование вязкости мантии Земли методами радиально-базисных функций в среде MATLAB. . . . . 14  
Булычев Г. Г. Численное моделирование динамики и динамического разрушения арочных и мостовых конструкций . . . . . 19  
Лучкова С. В., Перемитина Т. О., Яценко И. Г. Применение программного комплекса анализа многомерных данных на основе нечеткого и статистического моделирования. . . . . 24

### БЕЗОПАСНОСТЬ ИНФОРМАЦИИ

Тарасюк М. В. Выбор топологии VPN-сети с учетом требований конфиденциальности характеристик телекоммуникационной нагрузки . . . . . 31

### БАЗЫ ДАННЫХ

Набибекова Г. Ч. Об одном методе фаззификации атрибутов хранилища данных в системах поддержки принятия решений в сфере внешней политики. . . . . 36  
Фисун Н. Т., Горбань Г. В. Модели и методы построения системы OLAP для объектно-ориентированных баз данных . . . . . 41

### ОБРАБОТКА ИЗОБРАЖЕНИЙ

Соломеш Н. А., Браиловский И. В. Анализ и оптимизация работы вейвлетного видеокodeка в условиях передачи видеопотока по нестабильному каналу связи. . 46

### Журнал в журнале НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

Савченко А. В. Сегментация речи в задачах автоматического обнаружения и распознавания голосовых команд . . . . . 53  
Аведьян Э. Д., Луганский В. Э. Подход к задаче заполнения числовых пропусков в таблицах и строках, основанный на многослойной нейронной сети и нейронной сети СМАС . . . . . 58  
Сирота Е. А. Прогнозирование нестационарных временных последовательностей на основе иерархических многослойных нейронных сетей. . . . . 67

Приложение. Кухаренко Б. Г. Алгоритмы восстановления и сверхразрешения изображений

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.  
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.  
ГАЛУШКИН А. И., д. т. н., проф.  
ДИМИТРИЕНКО Ю. И., д. ф.-м. н., проф.  
ДОМРАЧЕВ В. Г., д. т. н., проф.  
ЗАГИДУЛЛИН Р. Ш., к. т. н., доц.  
ЗАРУБИН В. С., д. т. н., проф.  
ИСАЕНКО Р. О., к. т. н., с.н.с.  
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.  
РЯБОВ Г. Г., чл.-корр. РАН, д. т. н., проф.  
СОКОЛОВ Б. В., 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

#### INTELLIGENT SYSTEMS AND TECHNOLOGIES

**Baranov M. A.** Composite Key Terms in Task of Clustering Text Documents . . . . . 3

#### MODELING AND OPTIMIZATION

**Valeeva A. F., Goncharova Yu. A., Koshcheev I. S.** The Development of the Logistic Transport System . . . . . 9

**Chetyrbotsky A. N.** Numerical Modeling of the Earth's Mantle Viscosity Methods Radial Basis Functions in MATLAB. . . . . 14

**Bulychev G. G.** Numerical Simulation of Dynamics and Dynamical Destruction of Arch and Bridge Structures . . . . . 19

**Luchkova S. V., Peremitina T. O., Yashchenko I. G.** Application of Software Multidimensional Data Analysis Based on Fuzzy and Statistical Modeling. . . . . 24

#### CRYPTOSAFETY INFORMATION

**Tarasuk M. V.** Choice of the VPN Topology Taking into Account Requirements of the Confidentiality of the Telecommunication Loading Characteristics. . . . . 31

#### DATABASE

**Nabibekova G. Ch.** A Fuzzification Method or Attributes of Data Warehouse of Decision Support System in the Foreign Policy. . . . . 36

**Fisun N. T., Gorban G. V.** Models and Methods for Building OLAP System for Object-Oriented Databases . . . . . 41

#### IMAGE PROCESSING

**Solomeshch N. A., Brailovskiy I. V.** Analysis of Video Quality of Wavelet Video Codec for Unreliable Communicational Channel . . . . . 46

#### Journal-in-journal

#### NEUROTECHNOLOGIES

**Savchenko A. V.** Speech Segmentation in the Problems of Automatic Voice Control Detection and Recognition . . . . . 53

**Aved'yan E. D., Lugansky V. E.** An Approach to the Problem of Filling Gaps in Tables and Lines Based on the Multilayer and CMAC Neural Networks . . . . . 58

**Sirota E. A.** The Unstable Time Sequence Forecasting Based on the Hierarchic Multilayer Neural Network . . . . . 67

**APPLICATION. Kukharevko B. G.** Algorithms of Image Restoration and Superresolution

#### 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.  
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.  
Galushkin A. I., Dr. Sci. (Tech.), Prof.  
Dimitrienko Yu. I., Dr. Sci. (Phys.-Math.), Prof.  
Domrachev V. G., Dr. Sci. (Tech.), Prof.  
Zagidullin R. Sh., Cand. Sci. (Tech.), Ass. Prof.  
Zarubin V. S., Dr. Sci. (Tech.), Prof.  
Isaenko R. O., Cand. Sci. (Tech.)  
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.  
Kukharevko 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.  
Ryabov G. G., Corresp. Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Sokolov B. V., Dr. Sci. (Tech.)  
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"

# ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ INTELLIGENT SYSTEMS AND TECHNOLOGIES

УДК 519.237.8

М. А. Баранов, аспирант,  
Национальный исследовательский университет "Высшая школа экономики", Москва,  
e-mail: thenorthcat@gmail.com

## Составные ключевые термины в задаче кластеризации текстовых документов

*В рамках векторной модели представления документов рассматривается задача кластеризации текстовых документов с использованием словосочетаний, состоящих из более чем двух слов. Показано, что наилучшие результаты достигаются при использовании в процессе кластеризации комбинации одиночных слов со словосочетаниями, состоящими из двух или трех слов.*

**Ключевые слова:** кластеризация текстовых документов, жадный алгоритм, составные ключевые термины

М. А. Baranov

## Composite key Terms in Task of Clustering Text Documents

*In the first time the problem of text document clustering using phrases consisting of more than two words is considered in terms of vector space model. It is shown that best results are achieved when the clustering process combined with a single word phrases consisting of two or three words is used.*

**Keywords:** clustering text documents, greedy algorithm, composite key terms

### Введение

По данным компании Internet Data Corporation (IDC), объем информации, хранящейся во Всемирной Сети, удваивается каждые полтора года [1]. Поэтому задача поиска необходимой информации с каждым годом становится все более актуальной. Одной из наиболее важных задач информационного поиска является задача кластеризации — разбиение исходного множества документов на группы, состоящие из схожих документов. Из факта принадлежности документа кластеру и его релевантности по отношению к некоторому информационному запросу следует, что и остальные документы этого кластера также релевантны этому запросу.

На предварительном шаге перед кластеризацией необходимо создать образ документа. Существует несколько моделей представления документов, например, древовидная [2], векторная [3], но наиболее распространенной для задачи кластеризации является именно векторная модель. Также известно множество способов формирования образов документов в рамках векторной модели, их обобщение выполнено в работе [4].

Один из наиболее важных вопросов заключается в том, каким именно образом отбирать термины для

формирования образов документов. Существуют различные подходы, некоторые из них описаны в работах [5, 6].

В частности, необходимо ответить на вопрос, стоит ли использовать для формирования образа документа составные ключевые термины, т. е. термины, состоящие из нескольких слов. В ряде работ [7—9] было показано, что использование для формирования образа документа одиночных слов совместно со словосочетаниями, состоящими из двух слов, позволяет получить лучшие результаты при кластеризации по сравнению с использованием только лишь одиночных слов.

Цель данной работы заключается в исследовании того, как влияет на результаты кластеризации использование одиночных слов совместно со словосочетаниями различной длины, — будут рассмотрены комбинации одиночных слов со словосочетаниями, состоящими из двух, трех и четырех слов.

### Алгоритм предварительной обработки коллекции

Каждый документ коллекции проходит следующую процедуру обработки.

Из текста документа выделяются последовательности слов между знаками препинания.

**Морфологические шаблоны для словосочетаний, состоящих из двух и трех слов**

Морфологический шаблон	Пример
прил. + сущ.	современная видеокарта
сущ. + сущ.	менеджер процессов
прил. + прил. + сущ.	цветное индикаторное устройство
сущ. + прил. + сущ.	испытания электронного оборудования
прил. + сущ. + сущ.	автоматический поиск информации
сущ. + сущ. + сущ.	автоматизация процессов управления
сущ. + прил. + прил.	медь листовая красная

На следующем шаге все буквы слов переводятся в верхний регистр (это необходимо для корректной работы существующих лемматизаторов, таких как MyStem<sup>1</sup> и PyMorphy<sup>2</sup>). Затем каждое слово приводится к нормальной форме. Для решения этой задачи использовался морфологический анализатор PyMorphy для языка Python.

В случае неоднозначности за нормальную форму принимается первый вариант, возвращаемый морфологическим анализатором. Такая ситуация может возникнуть, например, при нормализации слова "душа": это может быть существительное "душ", существительное "душа" или глагол "душить".

Из полученных последовательностей извлекаются одиночные слова, а также словосочетания, состоящие из двух, трех или четырех слов.

Подсчитывается, сколько раз каждое нормализованное слово или словосочетание встречается в тексте документа. Эти данные будут использованы на следующем шаге для формирования матрицы схожести документов.

### Алгоритм формирования матрицы схожести

Предварительно определяется: 1) словосочетания какой длины будут использованы при формировании образа; 2) их количество; 3) число одиночных слов; 4) используемые морфологические шаблоны.

На первом шаге подсчитывается число появлений каждого одиночного слова и словосочетания заданной длины в текстах коллекции. Эти термины формируют рабочий набор.

Затем одиночные слова и словосочетания заданной длины проходят процедуру фильтрации по стоп-словам (их список взят с сайта разработчика алгоритма Портера<sup>3</sup>): если какое-либо слово из словосочетания присутствует в списке стоп-слов, то словосочетание целиком удаляется из рабочего набора. То же относится и к одиночным словам.

Оставшиеся термины проходят фильтрацию по морфологическим шаблонам: в случае одиночных слов из набора удаляются все слова, не являющиеся существительными или прилагательными; в случае словосочетаниями словосочетание удаляется, если хотя бы одно слово в нем не соответствует морфологическому шаблону.

Здесь опять может возникнуть неоднозначность, как, например, с упомянутым выше словом "душа": невозможно точно установить, является ли данное слово существительным или глаголом. В описываемом в данной статье алгоритме формирования образов документов используется следующее допущение:

<sup>1</sup>Морфологический анализатор MyStem — [company.yandex.ru/technologies/mystem/](http://company.yandex.ru/technologies/mystem/)

<sup>2</sup>Морфологический анализатор PyMorphy — <http://pypi.python.org/pypi/pymorphy>

<sup>3</sup>Список русских стоп-слов — <http://snowball.tartarus.org/algorithms/russian/stop.txt>

ние: если в списке возможных частей речи, возвращаемом морфологическим анализатором для конкретного термина, есть часть речи, описываемая одним из морфологических шаблонов, то данный термин (словосочетание) не будет удален из набора терминов.

В работе использовали морфологические шаблоны для словосочетаний из двух и трех слов [6, 10], приведенные в табл. 1.

Для словосочетаний, состоящих из четырех слов, возникает необходимость учитывать наличие предлогов. Поскольку ручное описание морфологических шаблонов для словосочетаний, состоящих из четырех слов, — достаточно трудоемкая задача, использовалась процедура генерации морфологических шаблонов.

На вход алгоритма генерации шаблонов подается множество  $P$ , состоящее из следующих элементов:

$$P = \{\text{сущ.}, \text{союз}, \text{предл.}, \text{прил.}\},$$

где сущ. — существительное; предл. — предлог; прил. — прилагательное.

Таблица 2

**Наиболее частотные одиночные и двухсловные термины, выделенные из набора ict**

Одиночные слова	Число	Двухсловные термины	Число
длить*	3262	множество решение	372
решение	2038	решение задача	237
уравнение	1691	система уравнение	227
задача	1643	числить моделирование	192
система	1582	турбулентный след	190
метод	1196	свободный поверхность	181
который	963	момент время	160
интервальный	961	вычислительный технология	160
условие	916	границный условие	157
рис	902	дифференциальный уравнение	144
значение	896	результат расчет	143
модель	891	линейный система	138
течение	869	интервальный линейный	120
расчет	819	решение система	118
скорость	817	длить решение	114

\* Подобные конструкции возникают вследствие несовершенства работы морфологических анализаторов

Таблица 3

**Наиболее частотные трехсловные термины,  
выделенные из набора ict**

Термы	Число появлений
интервальный линейный система	119
безымппульсный турбулентный след	78
оценка множество решение	58
институт вычислительный технология	54
объединить множество решение	53
метод дробление параметр	53
полный интервальный арифметик	48
начальный момент время	47
множество решение интервальный	46
российский фонд фундаментальный	45
фонд фундаментальный исследование	45
обыкновенный дифференциальный уравнение	43
оценивание множество решение	42
линейный алгебраический уравнение	39
внешний оценивание множество	39

Таблица 4

**Наиболее частотные четырехсловные термины,  
выделенные из набора ict**

Термы	Число появлений
российский фонд фундаментальный исследование	45
поддержка российский фонд фундаментальный	37
система линейный алгебраический уравнение	35
внешний оценивание множество решение	34
оптимальный внешний оценивание множество	32
дальний безымппульсный турбулентный след	32
решение интервальный линейный система	31
длитель интервальный линейный система	31
финансовый поддержка российский фонд	30
объединить множество решение ислау	22
система обыкновенный дифференциальный уравнение	22
множество решение интервальный система	21
интервальный оценка множество решение	20
оценивание обобщить множество решение	20
дефект продольный компонент скорость	18

Затем мы получаем четвертую Декартову степень данного множества:

$$P^4 = P * P * P * P.$$

Таким образом, получаем множество вида

$$P^4 = \{ \text{сущ.} + \text{сущ.} + \text{сущ.} + \text{сущ.}, \text{сущ.} + \text{сущ.} + \text{сущ.} + \text{союз}, \dots \text{прил.} + \text{прил.} + \text{прил.} + \text{прил.} \}$$

Затем из данного множества удаляются элементы, которые начинаются или заканчиваются предлогом или союзом. В итоге мы получаем 64 морфологических шаблона для словосочетаний, состоящих из четырех слов.

Продемонстрируем качество отбора ключевых слов и ключевых словосочетаний с использованием описанных выше алгоритмов предварительной обработки коллекции и формирования матрицы схожести на одном из тестовых наборов документов — наборе ict (табл. 2, 3, 4).

На заключительном этапе происходит непосредственное формирование образов документов на основе наиболее часто встречающихся одиночных слов  $X$  и наиболее часто встречающихся словосочетаний  $Y$ , состоящих из заданного числа слов (в дальнейшем этот набор будет называться  $N$ -списком).

Образ документа — массив, длина которого равна длине  $N$ -списка. Образ формируется следующим образом: если  $i$ -е слово или словосочетание из  $N$ -списка присутствует в документе, то в  $i$ -й элемент образа документа ставится 1, в противном случае — 0 (т. е. используется булево взвешивание).

На основе полученных образов документов составляется матрица схожести. В качестве меры схожести используется косинусная мера сходства [11]:

$$\cos(X, Y) = \frac{\sum_{i=1}^n X_i Y_i}{\sqrt{\sum_{i=1}^n (X_i)^2} \sqrt{\sum_{i=1}^n (Y_i)^2}},$$

где  $X$  и  $Y$  — векторы (образы) документов.

### Описание алгоритмов кластеризации

При проведении вычислительных экспериментов использовали два алгоритма — первый описан в работах [7, 12] (в дальнейшем будем называть его алгоритмом Greedy), а его модифицированная версия (Greedy2) описана ранее в работе [13].

На вход обоих алгоритмов подается матрица схожести документов размером  $N \times N$ , где  $N$  — число документов в коллекции, а документу с номером  $k$  соответствует  $k$ -й столбец и  $k$ -я строка. В этой матрице в каждой ячейке  $(i, j)$  задается мера схожести между  $i$ -м и  $j$ -м документами. Предлагается подбирать меру схожести таким образом, чтобы ее значение лежало в диапазоне от 0 до 1, где 0 соответствует полному различию документов, а 1 — полному сходству.

Алгоритм Greedy состоит из двух шагов. На шаге 1 в матрице схожести находится строка, сумма элементов которой является максимальной. Документ, соответствующий этой строке, объявляется центром очередного кластера, а сама строка содержит коэффициенты схожести этого документа со всеми остальными документами коллекции.

На шаге 2 в кластер добавляются все документы, коэффициент схожести которых с центром кластера больше или равен некоторому заранее заданному значению (параметр *threshold*). После этого из мат-

рицы удаляются все строки и столбцы, соответствующие попавшим в кластер документам.

Шаги 1 и 2 повторяются до тех пор, пока не останутся документы, не включенных в какой-либо кластер.

Суть модификации в алгоритме Greedy2 состоит в том, что при добавлении нового документа в кластер учитываются коэффициенты схожести этого документа по отношению к другим, уже включенным в кластер документам [13].

### Вычислительные эксперименты

Для проведения вычислительных экспериментов было подготовлено два тестовых набора.

Первый набор (*ferra*) состоит из документов, размещенных на сайтах *ferra.ru*<sup>4</sup> и *ixbt.com*<sup>5</sup>. Эталонное разбиение, с которым в дальнейшем сравнивали результаты работы алгоритмов кластеризации, состоит из пяти кластеров по 40 документов в каждом.

Второй набор (*ict*) составлен из текстов статей, размещенных на сайте журнала "Вычислительные технологии"<sup>6</sup>. Некоторым из размещенных статей приписаны коды классификатора в соответствии с Mathematics Subject Classification 2000 (MSC2000); эти коды использовали при формировании эталонного разбиения. Всего для экспериментов было подготовлено около 100 статей, они распределены по 29 кластерам от двух до семи документов в каждом.

Для оценки результатов кластеризации можно использовать два типа критериев: внутренние и внешние.

Внешние меры предполагают сравнение полученного в результате кластеризации разбиения документов на группы с эталонным разбиением. Внутренние меры предполагают оценку свойств отделимости и компактности полученного разбиения документов [4].

В данной работе использованы четыре внешних меры кластеризации: точность (*Precision*), полнота (*Recall*), ошибка классификации (*Error*) и  $F_1$ -мера, которые вычисляются по следующим формулам [5, 11]:

$$Precision = \frac{TP}{TP + FP}; Recall = \frac{TP}{TP + FN};$$

$$Error = \frac{FP + FN}{TP + FP + FN + TN}; F_1 = \frac{2 * Precision * Recall}{Precision + Recall},$$

где  $TP$  (*True Positives*) — число пар документов, которые как в эталонном, так и в получившемся в результате кластеризации разбиения принадлежат к одному классу;

$TN$  (*True Negatives*) — число пар документов, которые как и в эталонном, так и в получившемся разбиениях принадлежат к разным классам;

<sup>4</sup>URL: <http://www.ferra.ru/>

<sup>5</sup>URL: <http://www.ixbt.com/>

<sup>6</sup>Сайт журнала "Вычислительные технологии" — URL: <http://ict.nsc.ru/jct/>

$FP$  (*False Positives*) — число пар документов, которые в эталонном разбиении принадлежат к разным классам, однако в результате кластеризации были отнесены к одному классу (ошибки второго рода);

$FN$  (*False Negatives*) — число пар документов, которые в эталонном разбиении принадлежат к одному классу, однако в результате кластеризации были отнесены к разным классам (ошибки первого рода).

В табл. 5 и 6 представлены результаты экспериментов с исходным и модифицированным алгоритмами на двух описанных ранее наборах документов. Для оценки качества результатов кластеризации использовалось значение ошибки кластеризации *Error* (лучшим считался результат с наименьшим значением *Error*).

При проведении экспериментов использовали следующие данные. Число ключевых слов (и словосочетаний), с использованием которых формировались образы документов, лежало в диапазоне от 20 до 2000 слов с шагом 20. Таким образом, минимальная длина  $N$ -списка составляла 40 термов, максимальная — 4000 термов.

В работе [7] для параметра *threshold* использовался диапазон значений от 0,4 до 0,6 с шагом 0,1. Вследствие низких результатов при кластеризации значение 0,4 было исключено из рассмотрения, за-

Таблица 5

Результаты кластеризации текстовых коллекций алгоритмом *Greedy*

Мера	Коллекция <i>ferra</i>			
	Комбинация слов и словосочетаний			
	1	1 + 2	1 + 3	1 + 4
<i>Precision</i>	0,8708	<b>0,9326</b>	0,8918	0,8708
<i>Recall</i>	0,8364	<b>0,8085</b>	0,8451	0,8364
<i>Error</i>	0,0564	<b>0,049</b>	0,0505	0,0564
$F_1$	0,8533	<b>0,8661</b>	0,8678	0,8533
Дополнительные параметры				
Значение параметра <i>threshold</i>	0,7	<b>0,7</b>	0,7	0,7
Состав $N$ -списка	80	<b>80 + 400</b>	60+1000	80+20
Мера	Коллекция <i>ict</i>			
	Комбинация слов и словосочетаний			
	1	1 + 2	1 + 3	1 + 4
<i>Precision</i>	0,7755	0,75	<b>0,7568</b>	0,6974
<i>Recall</i>	0,2857	0,3835	<b>0,4211</b>	0,3985
<i>Error</i>	0,0259	0,0242	<b>0,0232</b>	0,0252
$F_1$	0,4176	0,5075	<b>0,5411</b>	0,5072
Дополнительные параметры				
Значение параметра <i>threshold</i>	0,7	0,7	<b>0,7</b>	0,7
Состав $N$ -списка	600	140+360	<b>220+180</b>	180+1780

Таблица 6

Результаты кластеризации текстовых коллекций алгоритмом *Greedy2*

Мера	Коллекция <i>ferra</i>			
	Комбинация слов и словосочетаний			
	1	1 + 2	1 + 3	1 + 4
<i>Precision</i>	0,9344	0,9361	<b>0,9351</b>	<b>0,9351</b>
<i>Recall</i>	0,8615	0,8531	<b>0,8613</b>	<b>0,8613</b>
<i>Error</i>	0,039	0,0402	<b>0,0389</b>	<b>0,0389</b>
$F_1$	0,8965	0,8927	<b>0,8967</b>	<b>0,8967</b>
Дополнительные параметры				
Значение параметров <i>threshold_center</i> , <i>threshold_other</i> и <i>admission</i>	t_c = 0,5 t_o = 0,7 adm = 25	t_c = 0,6 t_o = 0,7 adm = 25	<b>t_c = 0,6</b> <b>t_o = 0,7</b> <b>adm = 25</b>	<b>t_c = 0,6</b> <b>t_o = 0,7</b> <b>adm = 25</b>
Состав <i>N</i> -списка	80	60 + 220	<b>80 + 80</b>	<b>80 + 380</b>
Мера	Коллекция <i>ict</i>			
	Комбинация слов и словосочетаний			
	1	1 + 2	1 + 3	1 + 4
<i>Precision</i>	0,7333	<b>0,7654</b>	0,7333	0,7333
<i>Recall</i>	0,4962	<b>0,4662</b>	0,4962	0,4962
<i>Error</i>	0,0222	<b>0,022</b>	0,0222	0,0222
$F_1$	0,5919	<b>0,5794</b>	0,5919	0,5919
Дополнительные параметры				
Значение параметров <i>threshold_center</i> , <i>threshold_other</i> и <i>admission</i>	t_c = 0,5 t_o = 0,6 admission = 1	<b>t_c = 0,5</b> <b>t_o = 0,6</b> <b>admission = 2</b>	t_c = 0,5 t_o = 0,6 admission = 1	t_c = 0,5 t_o = 0,6 admission = 1
Состав <i>N</i> -списка	760	<b>1680 + 1720</b>	760 + 20	740 + 820

тем диапазон был расширен двумя новыми значениями: 0,7 и 0,8. Таким образом, значения параметров *threshold*, *threshold\_center* и *threshold\_other* лежат в диапазоне от 0,5 до 0,8 с шагом 0,1. Для параметра *admission* задавали следующие значения:

- 1, 2, 5, 10, 15, 20, 25 — при кластеризации коллекции *ferra*;
- 1, 2, 3, 4, 5 — при кластеризации коллекции *ict*.

Всего было проведено около 6 млн экспериментов. Наилучшие результаты для каждой серии выделены в табл. 5, 6 жирным шрифтом. Значения в столбцах под строкой "Комбинация слов и словосочетаний" показывают, каким образом формировался *N*-список: единица соответствует одиночным словам, двойка — словосочетаниям, состоящим из двух слов, и т. д. Например, запись "1+ 3" означает, что для формирования *N*-списка использовались одиночные слова и словосочетания, состоящие из трех слов. Записи в последней строчке таблицы ("Состав *N*-списка") расшифровываются следующим образом: первая цифра — число наиболее час-

то встречающихся одиночных слов, вторая (после знака "+") — число наиболее часто встречающихся словосочетаний, состоящих из заданного числа слов. Для оценки качества результатов кластеризации использовалось значение ошибки кластеризации *Error* (лучшим считался результат с наименьшим значением *Error*). В таблицах используются следующие сокращения: t\_c — *threshold\_center*, t\_o — *threshold\_other*, adm — *admission*.

Можно заметить, что наилучшие результаты кластеризации получаются при использовании для формирования *N*-списка комбинации одиночных слов со словосочетаниями, состоящими из двух или трех слов.

Проведенные эксперименты также показали, что наилучшие результаты кластеризации текстовых коллекций с использованием алгоритма *Greedy* получаются при значении параметра *threshold*, равном 0,7. Это еще одно подтверждение гипотезы, высказанной в работе [13], о том, что такое значение является оптимальным для данного алгоритма.

Для алгоритма *Greedy2* использование комбинаций одиночных слов и словосочетаний не дает значительного улучшения результатов кластеризации, в отличие от алгоритма *Greedy*, для которого данное улучшение является заметным.

Также в работе [13] было высказано предположение о том, что наилучшие результаты для алгоритма *Greedy* получаются при использовании меньшего числа термов для формирования матрицы схожести, чем при использовании алгоритма *Greedy2*. Возможно, это предположение верно, если применять для формирования *N*-списка только одиночные слова, но оно не выполняется при использовании комбинации одиночных слов и словосочетаний: например, наилучшие результаты для алгоритма *Greedy2* были получены при использовании 80 одиночных слов и 80 словосочетаний, состоящих из трех слов, а лучшие результаты для алгоритма *Greedy* — при использовании 80 одиночных слов и 400 словосочетаний, состоящих из двух слов.

## Заключение

Как уже упоминалось выше, для оценки результатов кластеризации использовали значение ошибки кластеризации. Наилучшие результаты были получены при использовании одиночных слов совместно со словосочетаниями, состоящими из двух или трех слов.

В ближайшее время планируется создать параллельные версии алгоритмов *Greedy* и *Greedy2*, что позволит сократить время выполнения данных алгоритмов. Отметим, что ранее для ряда алгоритмов кластеризации уже были созданы их параллельные версии — см., например, работы [14, 15].

## Список литературы

1. **Gantz J. F.** The Expanding Digital Universe: A Forecast of Worldwide Information Growth Through 2010. URL: <http://www.emc.com/collateral/analyst-reports/expanding-digital-white-paper.pdf> (дата обращения: 11.03.2013).
2. **Dawei G. C., Dawei Song, Bruza P.** Suffix tree clustering on post-retrieval documents. URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.6.429&rep=rep1&type=pdf> (дата обращения: 14.03.2013).
3. **Salton G., Wong A., Yang C. S.** A Vector Space Model for Automatic Indexing // Communications of the ACM. 1975. Vol. 18, Is. 11. P. 224—227.
4. **Пескова О. В.** Автоматическое формирование рубрикатора полнотекстовых документов // Электронные библиотеки: перспективные методы и технологии, электронные коллекции: Труды Десятой всероссийской научной конференции "RCDL'2008" (Дубна, 7—11 октября 2008 г.). Дубна: Изд. ОИЯИ, 2008. С. 139—148.
5. **Большакова Е. И., Клышинский Э. С., Ландэ Д. В., Носков А. А., Пескова О. В., Ягунова Е. В.** Автоматическая обработка текстов на естественном языке и компьютерная лингвистика: учеб. пособие. М.: Изд. МИЭМ, 2011.
6. **Браславский П. И., Соколов Е. А.** Сравнение четырех методов автоматического извлечения двухсловных терминов из текста // Компьютерная лингвистика и интеллектуальные технологии. Труды Международной конференции "Диалог 2006". М.: Изд-во РГГУ, 2006. С. 88—94.
7. **Баракхин В. Б., Ткачев Д. А.** Кластеризация текстовых документов на основе составных ключевых термов // Вестник Новосибирского государственного университета. Серия: Информационные технологии. 2010. Т. 8. № 2. С. 5—14.
8. **Bekkerman R. R., Allan J.** Using Bigrams in Text Categorization. URL: <http://people.cs.umass.edu/~ronb/papers/bigrams.pdf> (дата обращения: 30.03.2013).
9. **Chade-Meng Tan, Yuan-Fang Wang, Chan-Do Lee.** The use of bigrams to enhance text categorization // Information Processing & Management. July 2002. Vol. 38, Is. 4. P. 529—546.
10. **Белоголов Г. Г., Кузнецов Б. А.** Языковые средства автоматизированных информационных систем. М.: Наука. Главная редакция физико-математической литературы, 1983.
11. **Маннинг К. Д., Рагхаван П., Шюгце Х.** Введение в информационный поиск.: пер. с англ. М.: Вильямс, 2011. 528 с.
12. **Баракхин В. Б., Нехаева В. А., Федотов А. М.** О задании меры сходства для кластеризации текстовых документов // Вестник Новосибирского государственного университета. Серия: Информационные технологии. 2008. Т. 6, № 1. С. 3—9.
13. **Баранов М. А.** Модификация жадного алгоритма кластеризации // Прикладная информатика. 2013. № 2. С. 78—88.
14. **Островский А. А.** Реализация параллельного выполнения алгоритма FCM-кластеризации // Прикладная информатика. 2009. № 2. С. 101—106.
15. **Chang D., Kantardzic M. and Ouyang M.** Hierarchical clustering with CUDA/GPU // Proc. of ISCA PDCCS. 2009. P. 130—135.

## ИНФОРМАЦИЯ



# VIII Международная научная конференция "Параллельные вычислительные технологии" (ПаВТ'2014)

31 марта—4 апреля 2014 г.,  
Южный федеральный университет (г. Ростов-на-Дону)

**ПаВТ** — серия международных научных конференций, представляющих собой авторитетный и престижный форум в области применения параллельных вычислительных технологий в различных областях науки и техники.

Учредителями конференции являются Российская академия наук и Суперкомпьютерный консорциум университетов России.

В первый день работы конференции будет объявлена **20-я редакция списка Top50** самых мощных компьютеров СНГ.

**Тематика конференции** охватывает следующие основные направления (но не ограничивается ими):

- Технологии параллельных и распределенных вычислений
- Грид и облачные вычисления
- Перспективные многопроцессорные архитектуры
- Параллельные и распределенные системы баз данных
- Администрирование, мониторинг и тестирование многопроцессорных систем
- Вычислительная математика
- Вычислительная физика
- Вычислительная химия
- Гидро-газодинамика и теплообмен
- Высоконелинейные и быстротекущие процессы в задачах механики
- Биоинформатика
- Нанотехнологии
- Климат и экология
- Криптография
- Визуализация
- Компьютерная алгебра
- Суперкомпьютерные научно-образовательные центры.

В рамках конференции будет работать **индустриальная сессия**. На индустриальную сессию принимаются высококачественные презентации по инновационному коммерческому аппаратному и программному обеспечению, ориентированному на применение суперкомпьютерных и параллельных вычислительных технологий в различных областях науки и техники.

**Официальный сайт конференции:** <http://ПаВТ.РФ>

# МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 519.7

А. Ф. Валеева, д-р техн. наук, проф., e-mail: aida\_val2004@mail.ru  
Ю. А. Гончарова, аспирант, И. С. Кошчев, аспирант,  
Уфимский государственный авиационный технический университет

## Разработка логистической транспортной системы для решения задачи доставки груза различным клиентам. Часть 2

*Рассматривается прототип логистической транспортной системы для решения задачи доставки груза различным клиентам (3L-EVRP, Three-Dimensional Extended Vehicle Routing Problem). Представлены алгоритмы оптимизационного ядра системы, а именно: алгоритм муравьиной колонии, основанный на популяции, для поиска рациональных маршрутов доставки груза различным клиентам и эволюционный алгоритм (1+1)-EA3D для поиска рационального размещения емкостей параллелепipedной и цилиндрической форм с учетом требуемых ограничений (учет центра масс, размещения емкостей на поддонах и т. д.).*

**Ключевые слова:** маршрутизация, размещение грузов в автомобильные транспортные средства, транспортная логистика, логистическая транспортная система, алгоритм муравьиной колонии, основанный на популяции, эволюционный алгоритм

Valeeva A. F., Goncharova Yu. A., Koshcheev I. S.

## The Development of the Logistic Transport System

*In this article the prototype of the logistic transport system for solving the problem of goods delivery to customers (3L-EVRP, Three-Dimensional Extended Vehicle Routing Problem) is considered. Mathematical model for 3L-EVRP problem and the algorithms of the optimization kernel of system such as a population based ant colony optimization algorithm for looking for rational routes and evolutionary algorithm (1+1)-EA3D for looking for rational packing of contents with cuboid and cylindrical forms, taking into account required restriction (the consideration of the center of gravity, palletization etc) are presented. Some numerical experiments on random generated data, taking into account such restrictions as vehicle capacity, time windows, split delivery, delivery period, multi depot and so on, and tested examples, taken from International OR-library tests, are resulted.*

**Keywords:** routing, goods displacement in transportation means, transportation logistics, logistic transport system, population based ant colony optimization algorithm, evolution algorithm

### Введение

Расходы на транспортную логистику занимают существенную часть расходных статей многих компаний. В этих условиях необходимо иметь возможность принимать рациональные решения "точно в срок". В связи с этим возникает необходимость в создании логистической транспортной системы, позволяющей компании за приемлемое время получить рациональные маршруты доставки производимой ею продукции, а также карту размещения емкостей с реагентом в автомобильные транспортные средства (ТС).

Поскольку большинство практических задач транспортной логистики имеют большую размерность, то ввиду больших затрат вычислительного

времени для их решения применяют эвристические методы. Среди них важное место занимают алгоритмы, основанные на аналогиях с живой природой и физическими процессами. К таким алгоритмам можно отнести генетические алгоритмы, алгоритмы имитации отжига, нейронные сети, эволюционные алгоритмы, а также алгоритмы муравьиной колонии. Данная статья продолжает исследования в этом направлении и описывает разработанный алгоритм муравьиной колонии, основанный на популяции, для поиска рациональных маршрутов доставки груза различным клиентам, эволюционный алгоритм (1+1)-EA3D (данный алгоритм состоит из алгоритма (1+1)EA и декодера, основанного на работе [10]) для поиска рационального размещения

емкостей параллелепипедной и цилиндрической форм с учетом требуемых ограничений (учет центра масс, размещения емкостей на поддонах и т. д.).

Одним из методов приближенного решения задач комбинаторной оптимизации является оптимизация методом муравьиной колонии. Этот метод показывает хорошую производительность в сравнении с другими приближенными методами решения задач комбинаторной оптимизации [1]. Алгоритмы оптимизации методом муравьиной колонии являются частью научного направления роевого интеллекта, которое занимается изучением алгоритмов, основанных на наблюдениях за поведением децентрализованных самоорганизующихся систем.

Оптимизация методом муравьиной колонии была сформулирована как метаэвристика в работе [2] и нашла широкое применение для решения многих задач комбинаторной оптимизации, таких как задача коммивояжера [3], квадратичная задача о назначениях [4], задача маршрутизации [5].

Муравьиные алгоритмы серьезно исследуются европейскими учеными с середины 1990-х годов. На сегодня уже получены хорошие результаты муравьиной оптимизации таких сложных комбинаторных задач, как задачи коммивояжера, задачи оптимизации маршрутов грузовиков, задачи раскраски графа, квадратичной задачи о назначениях, оптимизации сетевых графиков, задачи календарного планирования и других.

В данной работе предлагается структура логистической транспортной системы, которая нацелена на решение следующих задач: прогнозирование спроса поставляемого клиентам груза, поиск оптимальных маршрутов доставки груза с одновременным получением карты его наилучшего размещения в автомобильных ТС. Последняя задача (3L-EVRP, Three-Dimensional Extended Vehicle Routing Problem) объединяет задачи маршрутизации и упаковки.

### 1. Структура логистической транспортной системы для решения задачи 3L-EVRP

Схема взаимодействия модулей представлена на рис. 1. Порядок вызовов отдельных компонентов следует сверху вниз. Для решения задач прогнозирования и получения матрицы расстояний/ визуализации маршрутов логично воспользоваться готовыми решениями, например пакетами Statistica и картами google/yandex maps api. Связи 1—2 показывают взаимодействие при составлении прогноза объемов спроса реагента. Связь 3 показывает взаимодействие задачи прогнозирования объемов спроса реагента и задачи составления рациональных маршрутов доставки емкостей с реагентом автомобильными ТС различным клиентам. Связи 4—5 показывают взаимодействие при заполнении таблиц расстояний (может использоваться *static api*). Связь 6 — взаимодействие задачи составления рациональных маршрутов доставки емкостей с реагентом автомо-

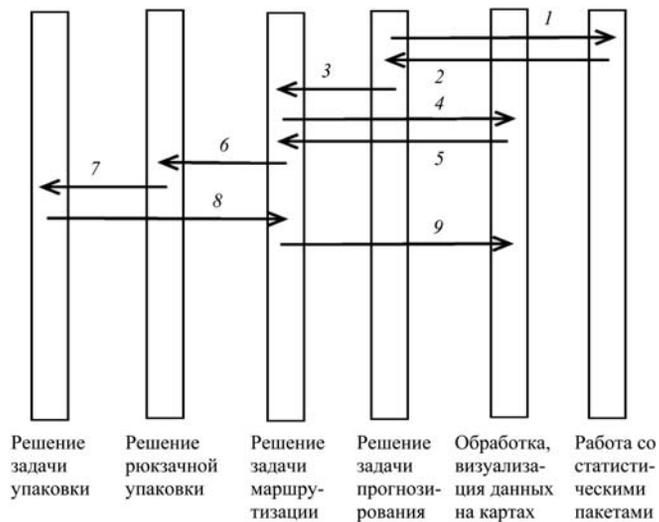


Рис. 1. Схема взаимодействия модулей

бильными ТС различным клиентам и задачи выбора типов емкостей для перевоза в них реагента в ТС. Связь 7 — взаимодействие задачи выбора типов емкостей для перевоза в них реагента в ТС и задачи размещения емкостей в ТС с выдачей плана размещения емкостей с реагентом в ТС. Связь 8 — взаимодействие задачи размещения емкостей в ТС с выдачей плана размещения емкостей с реагентом в ТС и задачи составления рациональных маршрутов доставки емкостей с реагентом автомобильными ТС различным клиентам. Связь 9 — отображение карт пользователю.

Для решения поставленной задачи 3L-EVRP предложен алгоритм P-ACO-3L-EVRP, представленный ниже.

#### Алгоритм P-ACO-3L-EVRP

**Входные параметры:** объем спроса клиентов в реагенте  $q_{iini}$ ; число городов и клиентов  $n$ ; расстояния между пунктами  $i$  и  $j$  —  $c_{ij}$  для всех пунктов  $i$  и  $j$ ; время в пути между пунктами  $i$  и  $j$  —  $e_{ij}$  для всех пунктов  $i$  и  $j$ ; параметры алгоритма  $\alpha$ ,  $\beta$ ; феромон  $\tau_{ij}$ ; период времени, за который должны быть обслужены все клиенты  $M$ ; время обслуживания в городе  $s_i$ ; граничные значения временных окон  $a_i$ ,  $b_i$ ; число агентов  $l$ ; штраф за начало обслуживания клиента после окончания временного окна, связанного с этим клиентом,  $penalty\_time_i$ ; параллелепипеды  $itp_i$  с известной длиной  $lp_i$ , шириной  $wp_i$ , высотой  $hp_i$ , массой  $mp_i$ ; цилиндры  $itc_k$  с известным радиусом  $rc_k$ , высотой  $hc_k$ , массой  $mc_k$ ; поддоны с известной длиной  $lpa_j$ , шириной  $wpa_j$ , высотой  $hpa_j$ , массой  $mpa_j$ , грузоподъемностью  $qpa_j$ , максимальной допустимой высотой предметов на поддоне  $hpa_j$ ; ТС  $v$  с известной длиной  $L^v$ , шириной  $W^v$ , высотой  $H^v$ , грузоподъемностью  $Q^v$ , стоимость емкостей  $h_z$ , число емкостей цилиндрической формы  $Item_{cyl}$ , число

емкостей параллелепипедной формы  $Item_{par}$  объем емкостей  $vol_z$ .

**Выходные параметры:** множество маршрутов, карта размещения емкостей с реагентом в ТС.

### Алгоритм *opt-3L-EVRP*

#### Шаг 1. ПРОГНОЗИРОВАНИЕ

(вход:  $q_{iinit}$ ;

выход: прогнозные значения объема спроса клиентов в реагенте  $q_{it}$ );

#### Шаг 2. P-ACO-3L-EVRP

(вход:  $n, c_{ij}, e_{ij}, \alpha, \beta, \tau_{ij}, q_{it}, M, s_i, a_i, b_i, l, penalty\_time_i, ip_i, lp_i, wp_i, hp_i, mp_i, itc_k, rc_k, hc_k, mc_k, lpal_j, wpal_j, hpal_j, mpal_j, qpal_j, hpal_j, L^v, W^v, H^v, Q^v$

выход: матрица популяции  $P$ , состоящая из  $R$  рациональных маршрутов  $Tour^r$ .

#### Шаг 3. ВЫБОР ЕМКостей

(вход:  $h_z, Item_{cyl}, Item_{par}, vol_z, q_{it}$

выход: выбранные емкости

$num = (num_1, \dots, num_{|Item_{cyl}| + |Item_{par}|})$ ).

#### Шаг 4. (1+1)-EA3D

(вход:  $num = (num_1, \dots, num_{|Item_{cyl}| + |Item_{par}|})$ ,  $ip_i$ ,

$lp_i, wp_i, hp_i, mp_i, itc_k, rc_k, hc_k, mc_k, lpal_j, wpal_j, hpal_j, mpal_j, qpal_j, hpal_j, L^v, W^v, H^v, Q^v$

выход: карта размещения емкостей с реагентом в ТС  $v$ , значение  $fpack^v$ ).

Опишем подробнее процедуры алгоритма *opt-3L-EVRP*.

#### Шаг 1. ПРОГНОЗИРОВАНИЕ.

Для получения прогнозных значений использовался пакет *Statistica*.

#### Шаг 2. P-ACO-3L-EVRP.

При решении рассматриваемой задачи 3L-EVRP агенты имитируют ТС, маршруты строятся пошагово выбором следующего клиента до тех пор, пока не будут обслужены все клиенты. Изначально к каждому депо прикрепляются ближайшие к нему клиенты, и алгоритм применяется несколько раз для каждого депо, т. е. задача преобразуется в задачу с единственным депо. Агент стартует из депо, и список клиентов, включенных в его маршрут, пуст. Агент выбирает следующего клиента из списка доступных клиентов и обновляет свою текущую загрузку перед выбором следующего. Агент возвращается в депо, когда достигнута его максимальная грузоподъемность либо когда все клиенты уже обслужены. Суммарный путь  $L^l$  рассчитывается как значение целевой функции полного маршрута  $l$ -го искусственного агента. Алгоритм P-ACO-3L-EVRP строит полный маршрут для первого агента перед тем, как второй начнет свое движение. Это продолжается до тех пор, пока заранее определенное число агентов  $k$  не построят полный маршрут.

### Начало алгоритма P-ACO-3L-EVRP

#### МЕТОД КИНИ\_РАЙФА

(вход:  $c_{ij}$

выход:  $c_{ij}$ )

$P := \emptyset$

#### Повторять

Для каждого агента  $l \in \{1, \dots, k\}$  {построение решения}

$S := \{1, \dots, n\}$  {множество заданных городов и клиентов}

$Tour^l := \emptyset$

$Time^l := \emptyset$

$i := 0$  (депо)

$Tour^l := Tour^l + \{i\}$

Для  $num := 1$  до **выполнить**

**Выбрать** город или клиента  $j$  с вероятностью  $p_{ij}$  по формуле (1) (см. ниже)

**Если**  $j$  — это клиент, **то**

**Вызвать** алгоритм

#### ВЫБОР ЕМКостей

(вход:  $h_z, Item_{cyl}, Item_{par}, vol_z, q_{it}$ )

выход:  $num = (num_1, \dots, num_{|Item_{cyl}| + |Item_{par}|})$ )

**Вызвать** алгоритм

(1 + 1)-EA3D

(вход:  $num = (num_1, \dots, num_{|Item_{cyl}| + |Item_{par}|})$ ,  $ip_i$ ,

$lp_i, wp_i, hp_i, mp_i, itc_k, rc_k, hc_k, mc_k, lpal_j, wpal_j, hpal_j, mpal_j, qpal_j, hpal_j, L^v, W^v, H^v, Q^v$

выход:  $fpack^v$ )

**Если**  $(Q - q_j \leq 0)$  и  $(fpack^v \leq 1)$ ,  $Q = Q^v$ , **то**  $penalty\_pack^v := 0$  по формуле (14) из части 1 статьи;

**Если** раздельная доставка, **то**

обслуживать клиента  $j$  продолжает другое ТС  $v \in \{1, \dots, m\}$  с грузоподъемностью  $Q := Q^v \geq q_j$ , происходит удовлетворение спроса в реагенте  $j$ -го клиента  $Q := Q - q_j$

**В противном случае**

ТС возвращается в депо, дозагружается и обслуживает клиента  $j$

**Конец Если**

**Конец Если**

**Если**  $(fpack^v > 1)$ , **то**

$penalty\_pack^v := const$  по формуле (14) из части 1 статьи;

$L^l := L^l + penalty\_pack^v$

**Конец Если**

**Если**  $(Q - q_j > 0)$

Спрос  $j$ -го клиента продолжает удовлетворять предыдущее ТС

**Конец Если**

Клиент  $j$  удаляется из множества  $S$ , если его спрос удовлетворен

$S := S - \{j\}$

Если  $w_j^{yu} > b_i$ , то

$Time^l := Time^l + penalty\_time_j$

Конец Если

$Time^l := Time^l + e_{ij} + s_j$

$i := j$

Конец Если

Если  $j$  — это город, то

город  $j$  удаляется из множества  $S$

$S := S + \{j\}$

$i := j$

Конец Если

Строится тур  $Tour^l := Tour^l + \{j\}$

Если в множестве  $S$  не осталось клиентов

Конец Для

Конец Если

Конец Для

Вычисляется длина  $L^l$  тура  $Tour^l$

Если  $Time^l \geq M$ , то к длине  $L^l$  тура  $Tour^l$  добавляется штраф в виде константы  $const$ :  $L^l := L^l + const$  Конец Если

Конец Для

Вызвать алгоритм

ВЫЧИСЛЕНИЕ\_МАТРИЦЫ\_ПОПУЛЯЦИИ

(вход:  $l$ ,  $P$  выход:  $P'$ )

До тех пор, пока не выполнен критерий останова

Выбрать лучшее решение из матрицы популяции  $P'$

Конец алгоритма P-ACO-3L-EVRP.

МЕТОД\_КИНИ\_РАЙФА

(вход:  $c_{ij}$

выход: обновленное расстояние  $c_{ij}$  между пунктами  $i$  и  $j$  для всех пунктов  $i$  и  $j$ )

Для решения задачи 3L-EVRP вначале строится исходный граф маршрутов  $G = (V, A)$ , из которого формируется матрица расстояний  $c_{ij}$  между пунктами  $i$  и  $j$ , к графу далее применяется метод Кини—Райфа [6], позволяющий выбрать дуги графа, удовлетворяющие взаимонезависимым по предпочтениям условиям  $Y^1, \dots, Y^k$  (например, качество дорог, уровень преступности на дорогах и т. д.); в преобразованном графе  $G' = (V', A')$ , из которого формируется матрица расстояний  $c'_{ij}$  между пунктами  $i$  и  $j$ , ищется кратчайший путь по алгоритму муравьиной колонии, основанному на популяции (Population Based Ant Colony Optimization, P-ACO) [7].

ВЫЧИСЛЕНИЕ\_МАТРИЦЫ\_ПОПУЛЯЦИИ

(вход:  $l$ ;  $P$

выход: обновленная матрица популяции  $P'$ ).

Если  $|P| = l$ , то удалить самое "худшее" решение  $\tilde{\pi}$  из популяции:

$$P' := P - \tilde{\pi}.$$

Определить самое лучшее решение итерации  $\pi^*$  и добавить его в популяцию:

$$P' = P + \pi^*.$$

Вычислить новую матрицу феромона  $\tau_{ij}$  из  $P$  [8].

Используется следующий критерий останова: если  $STOP = const$ , где  $STOP$  — число шагов, то считается, что критерий останова выполнен.

Выбор очередного города или клиента проводится случайным образом на основе вероятностной формулы

$$p_{ij}^l = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{z \in N^l} [\tau_{iz}(t)]^\alpha [\eta_{iz}]^\beta}, & \text{если } j \in N^l, \\ 0, & \text{в противном случае} \end{cases} \quad (1)$$

где  $N^l$  — множество вершин, которые агент  $i$  еще

не посетил;  $\eta_{ij} = \frac{1}{c_{ij}}$  — эвристическая информация;

величина  $\tau_{ij}(t)$ , где  $t$  — номер итерации алгоритма P-ACO-3L-EVRP, является уровнем феромона, который показывает "желательность" перехода в вершину  $j$  из вершины  $i$ , другими словами, желательность появления ребра  $(i, j)$  в решении; параметры  $\alpha$  и  $\beta$  — параметры алгоритма, показывающие относительное влияние феромона и эвристической информации. При  $\alpha = 0$  алгоритм вырождается в жадный, когда выбирается ближайший город без учета количества феромона.

Шаг 3. ВЫБОР\_ЕМКОСТЕЙ

Из емкостей с наибольшей потребительской оценкой найти емкости наименьшей стоимости и максимальной заполненности реагентом с помощью метода динамического программирования [9]:

1. Если  $z < vol_0 = vol_1$ , то  $f(z) = 0$  для всех таких  $z$ , иначе

2. Если  $z \in [vol_0, q_{ji}]$ , то  $f(z) = \min_{i \in I_z} \{-h_i - f(z - vol_i)\}$ ,

$I_z = \{i | vol_i \leq z\}$ .

3. Получение требуемого набора емкостей  $num = (num_1, \dots, num_{|Item_{cyl}| + |Item_{par}|})$ .

Шаг 4. (1+1)-EA3D

Сформировать приоритетный список  $\pi = (1, 2, \dots, |Item_{par}| + |Item_{cyl}|)$

Разместить предметы в контейнере согласно процедуре  $ABLP^{3D}(\pi)$ , вычислить значение целевой функции  $f(ABLP^{3D}(\pi))$ .

Повторять

Сгенерировать новый приоритетный список  $\pi^*$  такой, что  $\pi^*(j) = \pi(i)$ ,  $\pi^*(i) = \pi(j)$ ,  $\pi^*(m) = \pi(n)$ ,  $\pi^*(n) = \pi(m)$  для предметов  $it_i, it_j, it_m, it_n$ , где  $i, j, m, n \in [0, |Item_{cyl}| + |Item_{par}|]$ ,  $i \neq j, n \neq m$ ; для остальных элементов  $\pi^*(k) = \pi(k)$ ,  $k \in \{0, \dots, |Item_{cyl}| + |Item_{par}|\} \setminus \{m, n, i, j\}$

Определить значение целевой функции  $f(ABLP^{3D}(\pi^*))$

Если  $f(ABLP^{3D}(\pi^*)) < f(ABLP^{3D}(\pi))$ , то  $\pi = \pi^*$

До тех пор, пока критерий остановки не выполнен. Подать на выход упаковочный план согласно решению  $\pi$  и значение  $f_{pack}^v$ .

Приведем общую схему процедуры  $ABLP^{3D}$  для рационального размещения заданных предметов. В дальнейшем будем считать, что:

— допустимые позиции — позиции, которые обеспечивают допустимое размещение предметов, а именно удовлетворяют ограничениям (17)–(29) из части 1 статьи;

— лучшая позиция из списка — позиция, размещение предмета в которую обеспечивает меньшее значение целевой функции.

$ABLP^{3D}$  (вход: приоритетный список  $\pi$   
выход: значение целевой функции  $f(\pi)$ )

1. Поместить первый предмет в угол ТС  $v$ , совпадающий с началом координат
2. Для всех неразмещенных предметов выполнить:
  - 2.1. Для всех размещенных в ТС  $v$  предметов
    - 2.1.1. Получить допустимые позиции  $P_1$  относительно уже размещенного предмета
    - 2.2. Для всех пар размещенных предметов
      - 2.2.1. Получить допустимые позиции  $P_2$  относительно пар уже размещенных предметов
3. Получить список допустимых позиций  $P^*$ , удалив позиции, которые не обеспечивают допустимое размещение

- 3.1. Удалить недопустимые позиции из списка  $P_1 \cup P_2$
- 3.2. Поместить предмет в лучшую позицию из списка  $P^*$

Более подробно данный алгоритм описан в работе [9].

## 2. Архитектура прототипа логистической транспортной системы

Архитектура прототипа логистической транспортной системы, предложенной для решения задачи 3L-EVRP, приведена на рис. 2. Она состоит из БД, хранящей информацию о поддонах, бочках, коробках, заказчиках, заказах, расположениях клиентов, ТС, оптимизационного ядра, включающего модули, реализующие решения следующих подзадач: прогнозирование объемов спроса реагента; составление рациональных маршрутов доставки емкостей с реагентом автомобильными ТС различным клиентам на базе алгоритма муравьиной колонии, основанного на популяции (P-ACO-3L-EVRP, Population Based Ant Colony Optimization for 3L-EVRP); выбор типов емкостей для перевоза в них реагента в ТС; размещение емкостей в ТС с выдачей плана размещения емкостей с реагентом в ТС на базе алгоритма (1+1)-EA3D. Для визуализации найденных маршрутов используют карты google/yandex maps api.

Для прототипа логистической транспортной системы для решения задачи 3L-EVRP была составлена статистическая модель данных (рис. 3, см. третью сторону обложки), на основе которой в дальнейшем будет разработан программный комплекс. Как уже было сказано, в базе данных предполагается хранить информацию о поддонах, бочках, коробках, заказчиках, заказах, расположениях клиентов, ТС. Данная модель — модель физического уровня. Каждый прямоугольник представляет таблицу в БД, внутри описаны соответствующие столбцы и их типы. Линиями представлены связи между таблицами: так как эта модель физического уровня, связи "многие-ко-многим" разорваны с помощью дополнительных таблиц и связей "один-ко-многим".

### Заключение

В работе представлен прототип логистической транспортной системы для решения задачи доставки груза различным клиентам (3L-EVRP, Three-Dimensional Extended Vehicle Routing Problem), включающий следующие модули: прогнозирование спроса, поиск рациональных маршрутов доставки

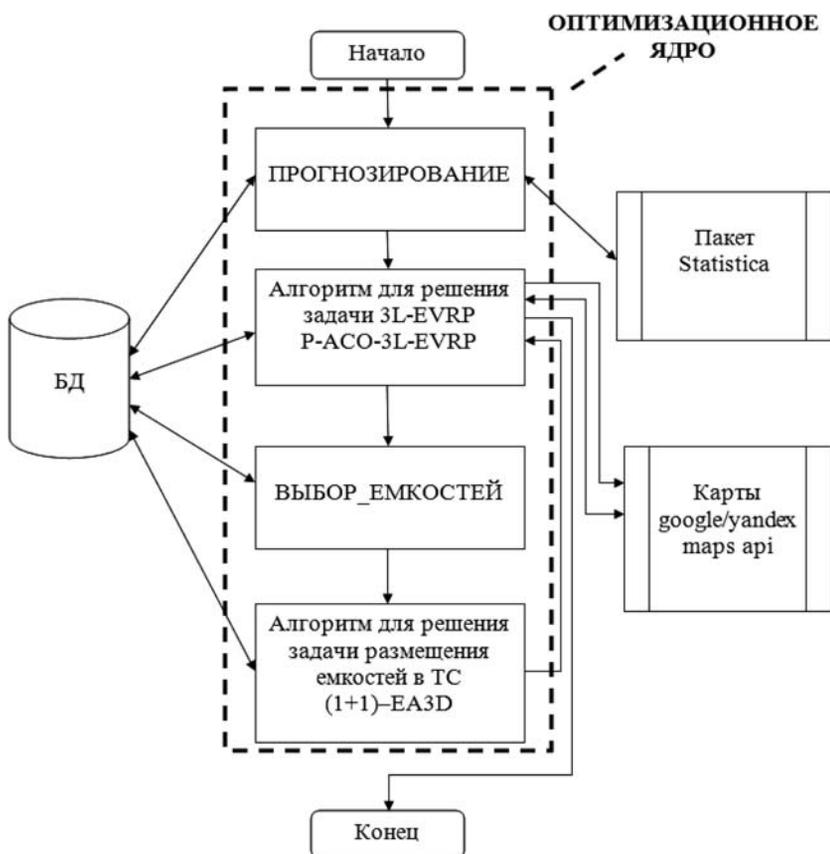


Рис. 2. Архитектура прототипа логистической транспортной системы для решения задачи 3L-EVRP

груза различным клиентам на базе алгоритма муравьиной колонии, основанного на популяции, а также размещение емкостей с реагентом в автомобильные ТС на базе эволюционного алгоритма (1+1)-EA3D. При этом модули поиска рациональных маршрутов и размещения емкостей с реагентом в автомобильные ТС объединены в единую систему, являющуюся частью разрабатываемой логистической транспортной системы.

Цель дальнейшей работы — создание логистической транспортной системы для решения задачи 3L-EVRP, объединяющей модули, решающие все вышеперечисленные задачи, в единый программный комплекс, как показано на рис. 1.

*Работа поддержана грантом РФФИ № 13-07-00579.*

#### Список литературы

1. **Dorigo M., Maniezzo V., Colomi A.** Ant system: optimization by a colony of cooperating // IEEE Transactions on Systems Man and Cybernetics. Part B. 1996. Vol. 26. N 1. P. 29—41.
2. **Dorigo M., Di Caro G., Gambardella L. M.** Ant Algorithms for Discrete Optimization // Artificial Life. 1999. Vol. 5, N 2. P. 137—172.

3. **Dorigo M., Gambardella L. M.** Ant colonies for the travelling salesman problem // BioSystems. 1997. Vol. 43, N 2. P. 73—81.
4. **Gambardella L. M., Taillard E. D., Dorigo M.** Ant Colonies for the Quadratic Assignment Problem // The Journal of the Operational Research Society. 1999. Vol. 50, N 2. P. 167—176.
5. **Bullnheimer B., Hartl R. F., Strauss C.** Applying the ant system to the vehicle routing problem // Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization, 1999. P. 109—120.
6. **Кини Р. Л., Райф Х.** Принятие решений при многих критериях: предпочтения и замещения. М.: Радио и связь, 1981. 560 с.
7. **Angus D.** Population-Based Ant Colony Optimization for Multi-objective Function Optimization // Proceedings ACAL. 2007. P. 232—244.
8. **Scheuermann B., So K., Guntsch M., Middendorf M., Dessel O., ElGindy H., Schmeck H.** FPGA implementation of population-based ant colony optimization // Applied Soft Computing. 2004. P. 303—322.
9. **Юсупова Н. И., Валеева А. Ф., Рассадникова Е. Ю., Кошечев И. С., Латыпов И. М.** Многокритериальная задача доставки грузов различным потребителям // Логистика и управление цепями поставок. 2011. № 46. С. 60—81.
10. **Hifi M., M'Hallah R.** Approximate algorithms for constrained circular cutting problems // Computers and Operation Research. 2004. Vol. 31, N 5. P. 675—694.

УДК 519.6

**А. Н. Четырбоцкий**, д-р физ.-мат. наук, вед. науч. сотр., e-mail: Chetyrbotsky@yandex.ru,  
Дальневосточный геологический институт ДВО РАН, Дальневосточный федеральный университет

## Численное моделирование вязкости мантии Земли методами радиально-базисных функций в среде MATLAB

*Рассматриваются формы аналитического представления характеристики внутреннего трения (вязкости) в мантии Земли. Обсуждаются вопросы согласования статистических выборок, которые используются для проведения в среде MATLAB параметрической идентификации модели мантийной вязкости. Для построения модели используется аппарат радиально-базисных функций. Приводятся способы графического представления результатов вычислительных экспериментов.*

**Ключевые слова:** мантия Земли, вязкость мантии, статистическая выборка, параметрическая идентификация

**A. N. Chetyrbotsky**

## Numerical Modeling of the Earth's Mantle Viscosity Methods Radial Basis Functions in MATLAB

*We consider the form of the analytic representation of characteristics of internal friction (viscosity) in the Earth's mantle. The issues of harmonization of statistical samples that are used to carry out in MATLAB parametric identification of the model of mantle viscosity. To construct the model of the machine is used radial basis functions. The methods of the graphical representation of the results of computational experiments.*

**Keywords:** Earth's mantle, the viscosity of the mantle, statistical sampling, parametric identification

## Введение

Вязкость среды является важной характеристикой контактного взаимодействия одних частей рассматриваемой системы относительно ее других частей. Ее численное представление обычно выполняется с помощью единой аналитической формулы, в которой она является функцией температуры и давления. Имеется также ряд моделей, где присутствует только одна из этих величин. Анализ выборочных радиальных распределений вязкости мантии Земли (широкий сферический слой, который простирается от глубин 35...45 км от поверхности до глубин порядка 2900 км [7]) показывает неприемлемость подобного представления.

Между тем вязкость присутствует в соответствующих уравнениях в частных производных, определяющих движение мантийного вещества. Для изучения динамики этого вещества требуется аналитическое представление вязкости, в рамках которого ее функция имеет непрерывные частные производные по пространственным переменным. В связи с этим актуальной является проблема построения такой функции вязкости. Для выполнения ее параметрической идентификации актуальным также является согласование неравных по объемам, доступных для исследований статистических выборок.

В статье рассматриваются способы решения поставленных задач. Сначала для вязкости приводятся основные формы представления вязкости. Далее выполнен анализ привлеченных для исследований статистических выборок. В рамках радиально-базисных функций (RBF) обсуждаются вопросы представления вязкости. Приводится описание разработанной в среде MATLAB процедуры параметрической идентификации вязкости, изучаются свойства полученных решений. Для разработки программного обеспечения используется MATLAB, что обусловлено его возможностями и постоянным совершенствованием заложенных алгоритмов [1].

### Форма представления вязкости мантии Земли

Построение моделей вязкости выполняется на основании положений, которые не являются физическими законами. В связи с этим они представляют собой эмпирические и полуэмпирические приближения, описывающие кривые течения в определенном интервале скоростей его сдвига [6]. Для выражения вязкости обычно используют различные модификации активационной модели арениусовского типа

$$\eta(T, P) = \eta_0 \exp\left[\frac{E_{act}}{kT}\right], \quad (1)$$

где полагается линейная зависимость активационной энергии  $E_{act} = E_0 + PV_{act}$  от давления  $P$ ;  $V_{act}$  — соответствующий активационный объем;  $k$  — посто-

янная Больцмана;  $T$  — абсолютная температура;  $\eta_0$  — значения вязкости при нулевом показателе экспоненты (когда числитель существенно меньше знаменателя). Например, в работе [4]  $\eta_0 = 8 \cdot 10^{11}$  Па · с,  $E_{act} = 12,1 + 45H/(1 + 0,73H)$ ,  $kT = T + T^* + 0,4$ ;  $H$  — толщина мантии;  $T$  — температура, выраженная в единицах ее перепада в мантии  $-2500$  К;  $T^*$  — безразмерная адиабатическая температура (определяется эмпирическим соотношением  $T^* = 0,36H$ ).

Выражение (1) отражает факт роста/убывания вязкости при росте/убывании давления и убывании/росте температуры. Простая запись такого процесса изменения  $\eta$  определяется выражением

$$\eta(T, P) = \eta_0 \exp(-aT + bP), \quad (2)$$

где неотрицательные коэффициенты  $a$  и  $b$  характеризуют соответствующие "вклады"  $T$  и  $P$  в изменение вязкости. В работе [13] полагается гидростатичность давления

$$P = \bar{\rho}g(H - z),$$

где  $\bar{\rho}$  — средняя плотность мантийного вещества;  $g$  — ускорение свободного падения. Тогда  $a = 4,6$  и  $b = 0,92/\bar{\rho}gH$ .

Для моделирования динамики мантийного вещества в работе [5] использовали подобное (2) выражение безразмерной вязкости, где  $\eta_0 = 5$ ,  $a = -6,9$  и  $b = 4,6/\bar{\rho}gH$ . При этих значениях вязкость мантии (от поверхности до ядра) за счет роста только температуры уменьшается на три порядка, а при росте только давления она увеличивается на два порядка.

Анализ положений, на основании которых разрабатываются современные модели мантийной вязкости, показывает стремление разработки единого аналитического ее представления. Однако подобные представления в некоторой мере являются адекватными реальной вязкости только в узких зонах мантии Земли.

### Выборочные глубинные распределения мантийной вязкости

Необходимость выполнения шага статистических исследований в различных предметных областях обусловлена тем, что на основании результатов его проведения можно выявить функциональную структуру изучаемого объекта [9]. В определенных ситуациях статистическое исследование является одним из основных способов количественного изучения. Кроме того, изучение выборочного материала и выявление основных свойств функционирования системы является одним из предварительных этапов построения численной модели.

Представляется уместным напомнить особенность вычисления среднего значения для случая не-

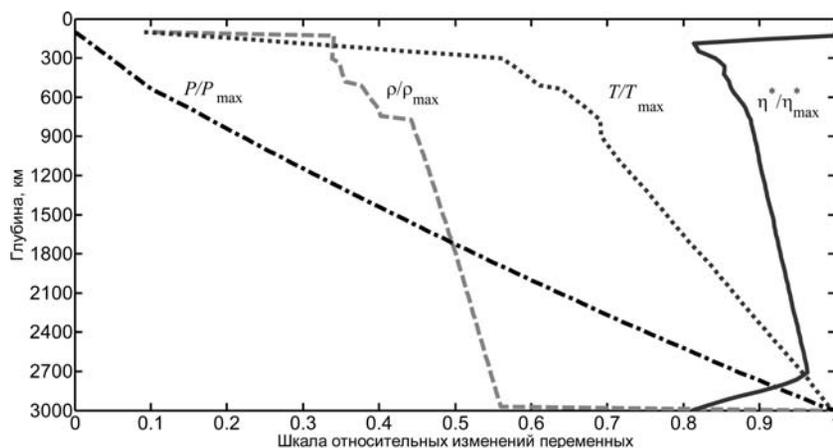


Рис. 1. Относительные изменения плотности (штриховая линия), температуры (пунктирная линия), давления (штрих-пунктирная линия) и десятичного логарифма вязкости  $\eta^*$  (сплошная линия)

равномерного распределения узлов (здесь — глубин). Если среднее вычисляется простым суммированием и делением на число слагаемых, то тем самым неявно в узлах полагается скачкообразный характер изменения переменной. Так, вычисленное таким способом среднее значение плотности для выборки из работы [3] оказалось равным  $5,061 \text{ г/см}^3$ , тогда как средняя плотность самой Земли равна  $5,515 \text{ г/см}^3$  [11]. Для устранения этой особенности следует посредством формирования весовых множителей (длины интервала между узлами, деленной на весь интервал изменения) учитывать длины интервалов между узлами. Легко заметить, что при равноотстоящих узлах эти веса формируются автоматически.

Вычисленные по выборке элементарные статистики принимают значения:  $\rho_{\max} = 9,961 \text{ г/см}^3$ ,  $\bar{\rho} = 4,692 \text{ г/см}^3$ ,  $T_{\max} = 3151 \text{ К}$ ,  $\bar{T} = 2408 \text{ К}$ ,  $P_{\max} = 636,5 \text{ кбар}$ ,  $\bar{P} = 1397,7 \text{ кбар}$ ,  $\eta_{\max}^* = 24,207$ ,  $\bar{\eta}^* = 22,116$  (черта сверху указывает на среднее значение переменной; здесь и далее  $\eta^* = \ln \eta$ ).

Изучение глубинного распределения мантийной вязкости здесь выполнялось на основании трех неравных по числу наблюдений статистических выборок. Так, выборка распределения плотности  $\rho$  насчитывает 90 наблюдений [11], выборка распределения температуры  $T$  и давления  $P$  — 18 наблюдений [3] и выборка распределения десятичного логарифма вязкости  $\{\eta^*\}$  — 50 наблюдений [12]. Особенность этих выборок состоит в том, что наблюдения каждой выборки приурочены к разным наборам неравномерно распределенных глубин. Для согласования этих выборок по одному и тому же набору равноотстоящих глубин целесообразным видится решение трех задач 1D-интерполяции (для каждой выборки отдельная задача) выборочных значений в равноотстоящий набор глубин.

В среде MATLAB для решения задачи интерполяции предназначена процедура `interp1` [1]. Форма ее вызова:

$$YI = \text{interp1}(h, Y, H, \text{Meth}),$$

где  $h$  — набор глубин измерения  $Y$ ;  $H$  — единый для всех выборок набор равноотстоящих глубин;  $YI$  — искомые значения интерполируемой переменной в узлах  $H$ ;  $\text{Meth}$  специфицирует способ проведения интерполяции.

Вычислительные эксперименты показывают незначительные отличия полученных при разных значениях  $\text{Meth}$  результатов интерполяции. Здесь использовалась линейная интерполяция  $\text{Meth} = \text{'linear'}$ . По результатам решения задач интерполяции была сформирована выборка наблюдений, каждая набором  $\{\rho_i, T_i, P_i, \eta_i^*\}$ , где  $i$  — номер наблюдения (здесь объем выборки был выбран равным 100, что принципиально не меняет ход дальнейших рассуждений).

На рис. 1 представлены графики относительных значений переменных. Конфигурации кривых указывают на совпадение глубин скачков значений  $\rho$ ,  $T$ ,  $\eta^*$  и границ сейсмических разделов верхней мантии (глубины не превышают 670 км). Сам факт наличия этих границ как раз и объясняется тем, что они (границы) являются зонами кристаллических перестроек мантийных минералов [2]. В связи с этим происходит скачкообразное изменение физических свойств параметров состояния рассматриваемой системы. Анализ кривых на рис. 1 отражает факт радиального роста значений  $\rho$ ,  $T$  и  $P$ . Близкое к 1 значение коэффициента корреляции (здесь он равен 0,998) явным образом указывает на гидростатичность распределения давления.

### RBF-представление мантийной вязкости

Нелинейность механизма внутреннего трения (вязкости) следует из анализа изменения выборочных распределений. Поскольку вязкость входит в соответствующие уравнения динамики мантийного вещества вместе со своими частными производными [8], то, чтобы не выполнять ресурсозатратное численное дифференцирование функции вязкости, надо для нее сформировать единое аналитическое представление. В дальнейшем это способствует существенному снижению вычислительных затрат.

RBF-представление следует решению многоэкстремальной задачи

$$\begin{aligned} \min_P \min_a \Phi(a, B) = \\ = \sum_{i=1}^{100} \left[ \eta_i^* - a_0 - \sum_{j=1}^J a_j \varphi(\|X_i - C_j\|, \varepsilon) \right]^2, \end{aligned} \quad (3)$$

где  $\{a_j, j = 0, \dots, J\}$  — искомые веса;  $J$  — число центров для RBF или число скрытых нейронов (определяется при вычислительных экспериментах);  $\varphi(\|X_i - C_j\|, \varepsilon)$  — заданная RBF с центром в точке  $C_j$ ;  $C \equiv \{C_j \in D \subset R^n, j = 1, \dots, J\}$  — центры RBF;  $\varepsilon \equiv \{\varepsilon_g, g = 1 \dots G\}$  — параметры RBF;  $G$  — их общее число;  $\|\cdot\|$  — введенная на  $R^n$  некоторая метрика (в большинстве случаев — евклидова метрика);  $B \equiv \{\{C\}, \{\varepsilon\}, \{x\}\}$  определяет параметры задачи (3). В рассматриваемом здесь случае  $X_i = (\rho_i, T_i, P_i)$ , где  $i = 1, \dots, 100$ ;  $\varepsilon = (\varepsilon_\rho, \varepsilon_T, \varepsilon_P)$  и  $C = \{\{\tilde{\rho}_j, \tilde{T}_j, \tilde{P}_j\}, j = 1, \dots, J\}$ .

Для решения задачи (3) здесь использовалась модификация так называемой гауссовской формы RBF

$$\varphi(X_i, C_j, \varepsilon) = \exp\left[-\sum_k^J \varepsilon_k (X_{ik} - C_{jk})^2\right]$$

или [10]

$$\begin{aligned} & \{\varphi(X_i, C, \varepsilon)\}_{ij} = \\ & = \exp[-\varepsilon_\rho(\rho_i - \tilde{\rho}_j)^2 - \varepsilon_T(T_i - \tilde{T}_j)^2 - \varepsilon_P(P_i - \tilde{P}_j)^2]. \end{aligned}$$

Сущность этой модификации состоит в следующем. При использовании традиционной гауссовской формы полагается тождественность коэффициентов [14], т. е.  $\varepsilon_k \equiv \text{const} \forall k = 1, \dots, J$ . Таким образом, в исходной постановке принимается одинаковость масштабов изменения переменных и единство их размерностей. Понятно, что в общем случае это противоречит реальности.

Решение типовых задач (3) приводится в [10]. Стоит только упомянуть, что итерационная процедура решения состоит из *этапа вычисления коэффициентов в показателе экспоненты* и *этапа расчета линейных весов*. Представляется полезным обсуждение вопросов выбора начального приближения и процедуры расчета линейных весов. Для реализации первого этапа следует указать границы допустимой многомерной области. Необходимо заметить, что поскольку показатель экспоненты безразмерное число, то размерность отдельного коэффициента обратно пропорциональна его сомножителю. А поскольку эти коэффициенты являются масштабирующими множителями, то их характерные значения определяются из сопоставления порядков отдельных членов. Далее сами границы многомерной области можно уже определить посредством проведения вычислительных экспериментов исходя из характерных значений.

Для нахождения решения задачи (3) необходимо указать число и координаты центров RBF. Если указано только число центров, то уже сами координаты становятся аргументами функционала задачи (3). Далее их расчет составляет расширенную процедуру первого этапа.

При выполнении вычислительных экспериментов было замечено существенное снижение вычислительных затрат, когда координатами центров выступали локальные экстремумы выборочных распределений. Здесь координаты центров как раз и выбирались согласно этому эмпирическому факту.

Решение второго этапа реализуется методом наименьших квадратов

$$Aa = b, \quad (4)$$

где элементы матрицы  $A$  и вектора  $b$  в рамках принятых здесь положений и обозначений представлены выражениями

$$\begin{aligned} A_{ij} &= \begin{cases} \varphi(\|X_i - C_j\|, \varepsilon) & \text{при } i \leq I, j \leq J; \\ 1 & \text{при } i \leq I, j = J + 1; \end{cases} \\ b &= \{\eta_i^*, i \leq I\}. \end{aligned}$$

Решение системы (4) можно выполнить несколькими способами. Один из них состоит в прямом нахождении обратной  $A$  матрицы  $A^{-1}$ . Для реализации в среде MATLAB этой процедуры следует вызвать функцию  $[U, S, V] = \text{svd}(A)$ , где  $U$  и  $V$  — унитарные матрицы, а  $S$  — диагональная матрица неотрицательных элементов. Размерности всех матриц совпадают и  $U \cdot S \cdot V = A$ . При большом числе центров RBF этот способ решения (4) не является экономичным, поскольку обращение матрицы большого размера требует значительных вычислительных затрат.

Другой способ состоит во введении спектрально близкой к  $A$  легко обрабатываемой матрицы  $B$ . Существенное снижение числа обусловленности  $A$  достигается умножением обеих частей (4) на матрицу  $B^{-1}$ . Для ее нахождения можно использовать стандартное  $LU$ -разложение матрицы, которое в среде MATLAB реализуется вызовом функции  $[L, U] = \text{lu}(A)$ . В результате формируются нижняя треугольная матрица  $L$  и верхняя треугольная матрица  $U$ , произведение которых есть  $A$ . При наличии этих матриц решение (4) распадается на два этапа. На первом из них решается уравнение  $Lc = b$ , а на втором — уравнение  $Ua = c$ . При большом числе центров этот метод более экономичный по сравнению с первым методом.

Численное решение задачи (3) определяется выражением

$$\begin{aligned} \eta^*(\rho, T, P) &= a_0 + \sum_{j=1}^4 a_j \exp[-\beta_\rho(\rho - \tilde{\rho}_j)^2 - \\ & - \beta_T(T - \tilde{T}_j)^2 - \beta_P(P - \tilde{P}_j)^2], \end{aligned}$$

где  $a_0 = 22,419$ ,  $a_1 = -3,074$ ,  $a_2 = 1,331$ ,  $a_3 = -3,724$ ,  $a_4 = 3,461$ ,  $\beta_\rho = 1,573 \cdot 10^{-2}$ ,  $\beta_T = 1,457 \cdot 10^{-6}$ ,  $\beta_P = 8,997 \cdot 10^{-6}$ ,  $\tilde{\rho} = (0,914, 5,388, 3,369, 1,020)$ ,  $\tilde{T} = (3,151, 2,985, 1,140, 2,880) \cdot 10^3$ ,  $\tilde{P} = (1,398, 1,187, 3,767, 0) \cdot 10^3$ .

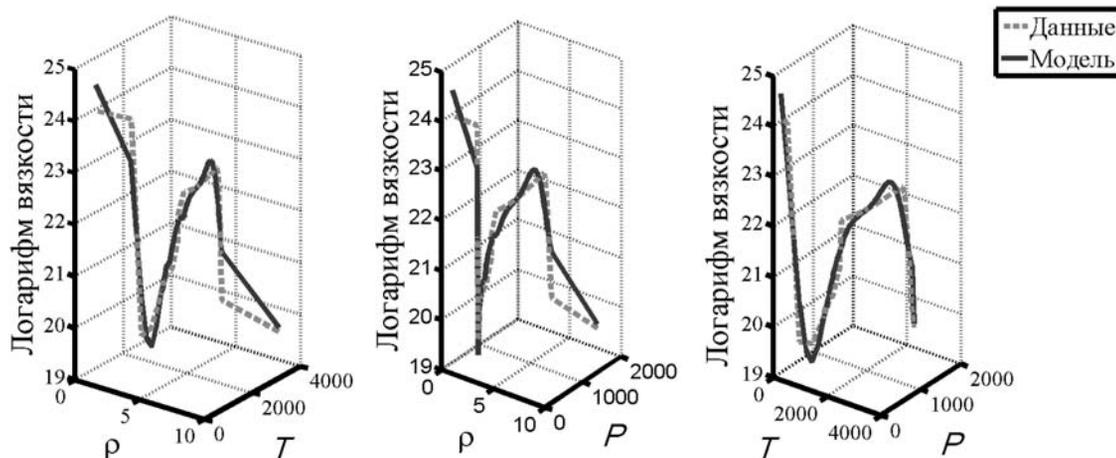


Рис. 2. Выборочные и модельные совместные распределения

Взаимное расположение кривых выборочных и модельных совместных распределений на рис. 2 отражает высокую степень адекватности выборочным данным.

Мерой адекватности выборочного и модельного распределений здесь выступает коэффициент корреляции между ними. Его высокое значение  $r = 0,939$  указывает на достоверность этого факта.

Значения коэффициентов  $\beta_\rho$ ,  $\beta_T$  и  $\beta_P$  являются мерами зависимости мантийной вязкости от плотности, температуры и давления. Их анализ показывает, что значения отдельных членов в показателе экспоненты (5) примерно равны. Иными словами, внутреннее трение в равной мере определяется плотностью, температурой и давлением.

### Заключение

В статье сформулированы основные положения, которые используются для построения моделей мантийной вязкости. Выполнен анализ выборочных глубинных распределений плотности, температуры, давления и логарифма вязкости. Выполнено построение RBF-приближения последней величины. Показана адекватность ее выборочных и модельных распределений. Выполнен анализ полученных результатов.

### Список литературы

1. Ануфриев И. Самоучитель MATLAB 5.3/6.x. СПб.: БХВ-Петербург, 2002. 712 с.

2. Пушаровский Ю. М., Пушаровский Д. Ю. Геология мантии Земли. М.: Геос, 2010. 140 с.

3. Сорохтин О. Г. Жизнь Земли. М.-Ижевск: НИЦ "Регулярная и хаотическая динамика"; Институт компьютерных исследований, 2007. 452 с.

4. Трубицын В. П. Основы тектоники плавающих континентов // Физика Земли. 2000. № 9. С. 4–40.

5. Трубицын В. П., Рыков В. В., Трубицын А. П. Роль конвективных процессов при образовании высоковязкой континентальной литосферы // Вестник ОГГГН. 2001. № 4(9). 25 с.

6. Фройштетер Г. Б., Трильский К. К., Ишук Ю. А., Ступак П. М. Реологические и теплофизические свойства пластичных смазок / Под ред. Г. В. Виноградова. М.: Наука, 1980.

7. Уайли П. Земная мантия // УФН. 1977. Т. 121. Вып. 1. С. 139–156.

8. Хашпель Дж., Бреннер Г. Гидродинамика при малых числах Рейнольдса. М.: Мир, 1976. 631 с.

9. Четырбоцкий А. Н. Крупномасштабное математическое моделирование пространственно-временной динамики морского ледяного покрова (на примере Японского моря). Владивосток: Дальнаука, 2009. 192 с.

10. Четырбоцкий А. Н. Параметрическая идентификация радиальных базисных функций нейронных сетей методами глобальной оптимизации // Информационные технологии. 2011. № 11. С. 54–58.

11. Anderson D. L. Theory of Earth—Blackwell Scientific Publications, 1989. 367 p.

12. Čížková H., Van den Berg A. P., Spakman W., Matyska C. The viscosity of Earth's lower mantle inferred from sinking speed of subducted lithosphere // Phys. of the Earth and Plan. Int. 2012. N 1. P. 1–14.

13. Cristensen U. R. Heat transport by variable viscosity convection for the Earth's thermal evolution // Phys. Earth and Planet Interiors. 1984. Vol. 35. P. 262–282.

14. Kansa E. J. Multiquadrics — a scattered data approximation scheme with applications of computational fluid dynamics. Part II: Solutions to parabolic hyperbolic and elliptic differential equations // Comput. Math. Appl. 1990. Vol. 19. P. 147–161.

Г. Г. Булычев, д-р физ.-мат. наук, проф., e-mail: geo-bulychev@mail.ru,  
 Московский государственный технический университет радиоэлектроники и автоматики  
 (МГТУ МИРЭА)

## Численное моделирование динамики и динамического разрушения арочных и мостовых конструкций

*Рассматриваются организация проекта, математическая модель, методика и результаты численного моделирования динамики и динамического разрушения арочных и мостовых конструкций. Для моделирования используются численный метод пространственных характеристик, развитый и численно реализованный автором применительно к данному классу задач, и авторская машинная графика. Определены характер разрушений конструкций и предельные разрушающие нагрузки, проведена оптимизация конструкции по динамической прочности.*

**Ключевые слова:** численное моделирование, механика деформированного твердого тела, динамика, разрушение, рамные конструкции

G. G. Bulychev

## Numerical Simulation of Dynamics and Dynamical Destruction of Arch and Bridge Structures

*In this article the project's architecture, mathematical model, method and results of the numerical simulation of dynamics and dynamic destruction of arches and bridges is considered. For simulation the numerical spatial characteristics method was developed and numerically realised by the author as applied to the given class of problems, and an author's machine graphics was used for it. The nature of destruction of structures and magnitudes of its limiting breaking loads was determined, structure's optimisation on a dynamic strength was conducted.*

**Keywords:** a numerical simulation, a mechanics of the deformed solid body, dynamics, fracture, frame structures

### Введение

Современные персональные ЭВМ (ПЭВМ), оснащенные многоядерными процессорами, позволяют осуществить кластерное решение сложных вычислительных задач на одном процессоре. Кроме того, внутрипроцессорное использование директив OpenMP (Multi Processing and Open means) и примитивов OpenGL (Open Graphics Library), свойственное многим современным процессорам и операционным системам, позволяет не только распараллеливать вычисления, но и ускорять графическую обработку их результатов с помощью встроенной графики любого вида.

Настоящий проект выполнен на языке ФОРТРАН-90. Его программы выполнены таким образом, чтобы допустить распараллеливание вычислений. Этого удалось добиться, объединяя группы скалярных переменных одного типа и назначения в векторы и матрицы, указывая способ их обработки и включая опции распараллеливания. При этом на четырехядерном процессоре скорость вычислений удалось увеличить более чем в 3 раза по сравнению с одноядерным. Для создания и компиляции файлов источников, а также отладки, сборки и тести-

рования построенных объектных файлов использовались оболочка Microsoft Visual Studio 2008 и встроенный в нее компилятор (Intel Composer XE fortran compiler v12).

### Этапы проекта и их краткое содержание

Схема проекта, представленная на рис. 1, состоит из шести последовательных этапов.

*На первом этапе* проводится моделирование динамики рассматриваемых конструкций.

Результаты моделирования, полученные на этом этапе, описывают динамику конструкции и представляют собой массивы напряжений, возникающих в ней под действием нагрузки в последовательные моменты времени.

*На втором этапе* полученные массивы обрабатываются с помощью программ машинной графики в целях определения максимальных напряжений и их эволюции. При этом используются примитивы OpenGL и, как надстройка над ними, подпрограммы построения контуров и изолиний, разработанные автором на базе известного пакета ГРАФОР [2] применительно к данной задаче и современным ПЭВМ с операционной системой WINDOWS-7.



Рис. 1

Анализ этих напряжений позволяет правильно выбрать и построить двойную сетку, необходимую для моделирования динамического разрушения конструкции.

На третьем этапе выбираются критерии начала разрушения и условия на трещинах. В качестве критериев начала разрушения используются условия накопления неупругих напряжений и обусловленных ими пластических деформаций до заранее заданных значений, обычно определяемых из опыта, и затем, в зависимости от напряженно-деформированного состояния (НДС) на берегах трещины, использования тех или иных условий на этих берегах.

На четвертом этапе проводится моделирование динамического разрушения конструкции при различных нагружающих напряжениях  $P$ , равномерно распределенных по поверхности и не меняющихся во времени. В качестве критерия полного разрушения конструкции принимается полное разрушение какого-либо ее сечения. Анализ разрушения проводится автоматически, а выполнение указанного критерия используется для окончания счета. Предполагается также, что разрушение происходит за время, не превышающее 50 времен пробега продольной волны по максимальному размеру конструкции. В выводимых массивах разрушающих напряжений задаются коды, позволяющие вручную проанализировать значения напряжений и области разрушений в конструкции.

На пятом этапе массивы, полученные в момент полного разрушения конструкции, подвергаются графической обработке в целях определения областей катастрофического разрушения.

И наконец, на шестом, последнем этапе решается задача определения минимальных нагрузок, приводящих к полному разрушению конструкции. Также на этом этапе для мостовой конструкции решается задача оптимизации конструкции по динамической прочности. При этом предполагается, что

суммарный объем опор и размеры несущей части мостовой конструкции заданы и не меняются в процессе моделирования, а меняются число опор и размеры каждой опоры (все опоры одинаковые). Предполагается также, что опоры и несущая часть конструкции сделаны из одного и того же материала.

### Математическая модель динамики

Исходная математическая модель, используемая при моделировании динамики и записанная в декартовой координатной системе  $\{x_j\}$ , состоит из следующих

условий и уравнений: уравнений движения элементов объема тела, условий аддитивности деформаций, закона Гука, закона пластического течения и, наконец, соотношений Коши, записанных для скоростей деформаций:

$$\partial \sigma_{ij} / \partial x_j = \rho \partial V_i / \partial t, \quad \varepsilon_{ij} = \varepsilon_{ij}^e + \varepsilon_{ij}^{vp}, \quad (1)$$

$$\varepsilon_{ij}^e = a_{ijkl} \sigma_{ij}, \quad \partial \varepsilon_{ij}^p / \partial t = \tau_0^{-1} S^{-1} \Phi(S - k_S),$$

$$\partial \varepsilon_{ij} / \partial t = 0,5(\partial V_i / \partial x_j + \partial V_j / \partial x_i), \quad a_{ijkl} = c_{ijkl}^{-1}, \quad C = \|c_{ijkl}\|, \quad (2)$$

где  $\sigma_{ij}$  — напряжения в теле;  $\rho$  — плотность тела;  $V_i$  — скорости частиц тела;  $\varepsilon_{ij}$  — полные деформации в теле;  $\varepsilon_{ij}^e$  — упругие деформации в теле;  $\varepsilon_{ij}^{vp}$  — вязкопластические деформации в теле;  $a_{ijkl}$  — тензор податливостей;  $c_{ijkl}$  — тензор упругости;  $C$  — матрица жесткостей рассматриваемого тела:

$$C = \begin{pmatrix} 2\mu + \lambda & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & 2\mu + \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & 2\mu + \lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & 2\mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 2\mu & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\mu \end{pmatrix}, \quad (3)$$

$\Phi(S - k_S)$  — функция пластичности;  $\tau_0$  — время задержки текучести;  $S = \sqrt{s_{ij}s_{ij}/2}$  — интенсивность напряжений в теле;  $s_{ij} = \sigma_{ij} - \delta_{ij}p$  — девиатор напряжений;  $p = \sigma_{ii}/3$  — гидростатическое давление;  $k_S$  — статический предел текучести материала тела при сдвиге;  $\mu$  и  $\lambda$  — параметры Ламе;  $\delta_{ij}$  — единичный тензор;  $t$  — время;  $i, j, k, l = 1, 2, 3$ , по повторяющимся латинским индексам здесь и далее проводится суммирование.

Представим систему уравнений (1), (2) в характеристической форме так, как это делалось ранее в работе [1].

Уравнения для движущихся разрывов, распространяющихся вдоль оси  $x_\alpha$ , совпадающей с одной из осей выбранной координатной системы  $\{x_1, x_2, x_3\}$ , имеют следующий вид:

- для продольной волны, распространяющейся в обе стороны вдоль оси  $x_\alpha$ ,

$$(\partial/\partial\tau \pm \partial/\partial x_\alpha)(\sigma_{\alpha\alpha} \mp V_\alpha) = \partial/\partial x_\beta(v_1 V_\beta \pm \sigma_{\alpha\beta}) + \partial/\partial x_\gamma(v_1 V_\gamma \pm \sigma_{\alpha\gamma}) - F_{\alpha\alpha} - v_1(F_{\beta\beta} + F_{\gamma\gamma}); \quad (4)$$

- для поперечных волн, распространяющихся вдоль той же оси,

$$(\partial/\partial\tau \pm \xi\partial/\partial x_\alpha)(\sigma_{\alpha\beta} \mp \xi V_\beta) = \partial/\partial x_\beta(v_2 V_\alpha/2 \pm \xi\sigma_{\beta\beta}) \pm \partial/\partial x_\gamma(\xi\sigma_{\beta\gamma}) - F_{\alpha\beta}, \quad (5)$$

$$(\partial/\partial\tau \pm \xi\partial/\partial x_\alpha)(\sigma_{\alpha\gamma} \mp \xi V_\gamma) = \partial/\partial x_\gamma(v_2 V_\alpha/2 \pm \xi\sigma_{\gamma\gamma}) \pm \partial/\partial x_\beta(\xi\sigma_{\beta\gamma}) - F_{\alpha\gamma}; \quad (6)$$

- для неподвижных разрывов с нормалью  $x_\alpha$  характеристические уравнения имеют вид

$$\partial/\partial\tau(\sigma_{\beta\beta} - v_1\sigma_{\alpha\alpha}) = v_2(1-v)^{-1}(\partial V_\beta/\partial x_\beta + v\partial V_\gamma/\partial x_\gamma - F_{\beta\beta} - vF_{\gamma\gamma}), \quad (7)$$

$$\partial/\partial\tau(\sigma_{\gamma\gamma} - v_1\sigma_{\alpha\alpha}) = v_2(1-v)^{-1}(\partial V_\gamma/\partial x_\gamma + v\partial V_\beta/\partial x_\beta - F_{\gamma\gamma} - vF_{\beta\beta}), \quad (8)$$

$$\partial/\partial\tau\sigma_{\beta\gamma} = v_2[(\partial V_\beta/\partial x_\gamma + v\partial V_\gamma/\partial x_\beta)/2 - F_{\beta\gamma}], \quad (9)$$

где все переменные приведены к безразмерному виду с помощью соотношений

$$\tilde{x}_i = x_i/x_0, \quad \tau = c_0 t/x_0, \quad \tilde{\sigma}_{ij} = \sigma_{ij}/k_S, \quad \tilde{S} = S/k_S, \quad \tilde{V}_i = V_i/V_S, \quad (10)$$

$$k_S = \rho c_0 V_S, \quad c_0 = \sqrt{(2\mu + \lambda)/\rho}, \quad \delta = x_0/(V_S \tau_0);$$

$x_0$  — нормировочная константа, за которую принят максимальный размер арки;  $c_0$  — скорость продольной волны в теле;  $V_S$  — скорость частиц тела, при которой начинается пластическое течение;  $v_1 = v/(1-v)$ ,  $v_2 = 1-v_1$ ;  $\alpha \neq \beta$ ,  $\beta \neq \gamma$ ,  $\gamma \neq \alpha$ ,  $\alpha, \beta, \gamma = 1, 2, 3$ ;  $\xi = c_\perp/c_0$  — отношение скорости поперечной волны  $c_\perp = \sqrt{\mu/\rho}$  к скорости продольной волны  $c_0$ , по повторяющимся латинским индексам проводится суммирование, а по повторяющимся греческим индексам суммирование не проводится.

Затем значок "тильда" над безразмерными переменными убирается. Для упруговязкопластического тела функции в правых частях уравнений (4)—(9) имеют вид

$$F_{\eta\zeta} = \delta \frac{\bar{\Phi}(S-1)}{S} s_{\eta\zeta}, \quad \eta, \zeta = \alpha, \beta, \gamma.$$

Для упрощения вычислений выбираем

$$\bar{\Phi}(S-1) = \sqrt{S-1} \quad \text{при } S > 1 \text{ и}$$

$$\bar{\Phi}(S-1) = 0 \quad \text{при } S \leq 1.$$

### Метод моделирования

Для моделирования используется численный метод пространственных характеристик, построенный на характеристической форме представления исходных уравнений, т. е. на уравнениях (4)—(9), которые записываются в конечно-разностной форме. Такая форма записи эквивалентна интегрированию указанных уравнений по соответствующим шаблонам с помощью формул Гаусса—Остроградского [2, стр. 72—77].

По сравнению с классическими сеточными методами этот метод обладает следующими достоинствами:

- существенно сближена область зависимости дифференциальных уравнений и их конечно-разностных аналогов и тем самым резко повышена точность вычислений;
- на единицу уменьшено число независимых переменных, причем определяемые переменные на характеристиках связаны попарно. Это позволяет резко сократить время счета и объем требуемой оперативной памяти;
- дифференциальные части уравнений, составляющих модель, имеют дивергентную форму, т. е. удовлетворяют законам сохранения. Это позволяет точно описывать скачки решений и отслеживать изменение условий на трещинах;
- при первом порядке аппроксимации остаточные члены имеют параболический вид, что дает "естественный" характер затухания процесса. Поэтому используемые алгоритмы могут быть применены для описания не только быстрой динамики, но и переходных (от быстрой динамики к статике) процессов в рассматриваемых телах.

**Параметры моделирования.** Моделирование динамики вышеуказанных конструкций проводилось на ПК с помощью численного интегрирования формул (4)—(9) с учетом особенностей точек, в которых проводилось интегрирование (например, внутренняя точка, граничная точка, внешняя угловая точка, внутренняя угловая точка, точка ребра и т. д.). При разработке алгоритмов использовались нормировки (10), поэтому задавались только значение внешней нагрузки  $P_0$  ( $P_0 = 0,01...0,5$ ), коэффициент Пуассона  $\nu$  ( $\nu = 0,1, 0,2, 0,3$ ), безразмерная величина  $\delta$  ( $\delta = 0,05$ ) и геометрические параметры.

Для арок *граничные условия* соответствуют схеме нагружения, показанной на рис. 2. На верхней поверхности арки (поверхность 1) действует нормальная нагрузка  $P$ , на поверхностях 2—4 заданы условия жесткой заделки, поверхности 5—12 свободны от напряжений.

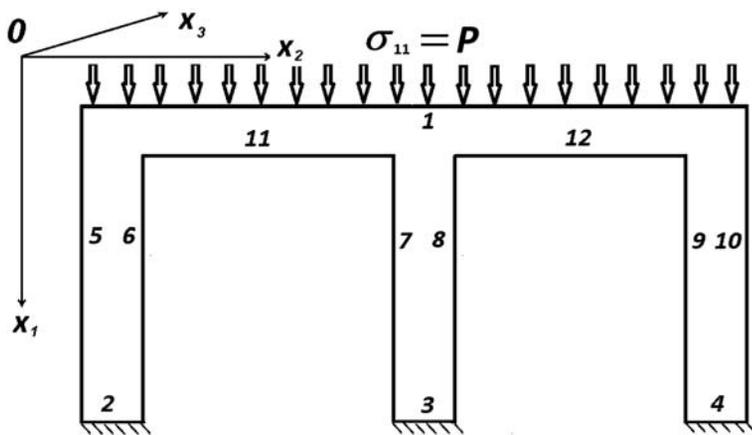


Рис. 2

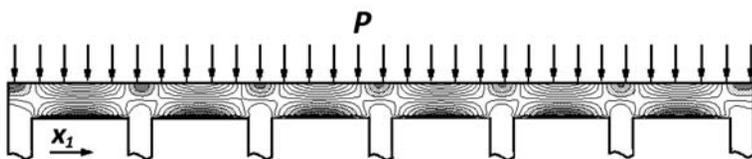


Рис. 3

Для моста граничные условия несколько более сложные: на границе 1 также действует сжимающая нагрузка, на границах  $x_2 = \min$  и  $x_2 = \max$  заданы условия продолжения конструкции (для волн, движущихся, соответственно, налево от  $x_2 = \min$  и направо от  $x_2 = \max$  [3]), на части опор, прилегающей к их нижним границам, заданы условия жесткой заделки, а остальные границы свободны от напряжений.

### Результаты моделирования динамики конструкций

Как оказалось, при указанной выше нагрузке наибольшими напряжениями являются  $\sigma_{11}$ ,  $\sigma_{22}$ ,  $\sigma_{12}$ , которые образуют области повышенных напряжений посередине между опорами —  $\sigma_{22}$ , на нагружаемой поверхности и вблизи оснований опор —  $\sigma_{11}$  и вблизи внутренних угловых точек, образуемых опорами, —  $\sigma_{12}$ .

На рис. 3 показаны схема нагружения и изолинии напряжений  $\sigma_{22}$  шестипролетного моста; видно, что эти напряжения концентрируются, в основном, посередине между пролетами, а также над опорами.

Проведенный анализ позволил выявить напряжения, способные привести конструкцию к ее разрушению, и определить направление двойной сетки, необходимой для исследования разрушения конструкции под действием данной нагрузки.

### Критерии образования трещины и условия на ней

Предположим, что трещины возникают из-за накопления пластических деформаций, которые, в свою очередь, определяются напряжениями, вы-

ходящими за пределы упругой области. Тогда критерии образования трещины в рассматриваемой конструкции можно принять в виде  $J_n = A$  или  $J_\tau = B$ , где

$$J_n = \int_{\tau_0}^{\tau} \langle \sigma_{11}(\tau) / \sqrt{3} - 1 \rangle d\tau,$$

$$J_\tau = \int_{\tau_1}^{\tau} \langle \sigma_{12}(\tau) - 1 \rangle d\tau,$$

а константы  $A$  и  $B$  определяются экспериментально.

Здесь  $\langle f \rangle = f$  при  $f > 0$  и  $\langle f \rangle = 0$  при  $f \leq 0$ ;  $A = B = 1$ ;  $\tau_0$  и  $\tau_1$  — моменты времени, в которые соответствующие напряжения впервые выходят в пластическую область.

До выполнения условий разрушения во всех точках плоскостей  $x_1 = \text{const}$  задаются условия непрерывности скоростей частиц и нормальных и касательных напряжений, т. е. условия

$$\begin{aligned} \sigma_{11}^+ &= \sigma_{11}^-, \sigma_{12}^+ = \sigma_{12}^-, \sigma_{13}^+ = \sigma_{13}^-, \\ V_1^+ &= V_1^-, V_2^+ = V_2^-, V_3^+ = V_3^-, \end{aligned} \quad (11)$$

в которых знаками плюс и минус обозначены верхний и нижний берега рассматриваемой плоскости.

После выполнения соответствующего условия разрушения на образовавшейся трещине задается одна из двух последовательностей условий.

#### 1. Трещина отрыва, выполнен первый критерий.

В момент  $\tau_0$  образования трещины на соответствующем участке задаются условия свободных (от напряжений) берегов трещины

$$\sigma_{11}^+ = \sigma_{12}^+ = \sigma_{13}^+ = \sigma_{11}^- = \sigma_{12}^- = \sigma_{13}^- = 0 \quad (12)$$

и начинает вычисляться расстояние между берегами трещины

$$\Delta u_1 = \int_{\tau_0}^{\tau} [V_1^+(x_1, \tau) - V_1^-(x_1, \tau)] d\tau. \quad (13)$$

До тех пор, пока  $\Delta u_1 > 0$ , условие (12) сохраняется, при  $\Delta u_1 < 0$  (что достигается лишь при  $\sigma_{11} < 0$ ) трещина закрывается и на ее берегах задаются условия

$$\begin{aligned} V_1^+ &= V_1^-, \sigma_{11}^+ = \sigma_{11}^-, \\ \sigma_{12}^+ &= -\alpha |\sigma_{11}| \text{sgn}(V_2^+ - V_2^-), \\ \sigma_{13}^+ &= -\alpha |\sigma_{11}| \text{sgn}(V_3^+ - V_3^-), \\ \sigma_{12}^- &= +\alpha |\sigma_{11}| \text{sgn}(V_2^+ - V_2^-), \\ \sigma_{13}^- &= +\alpha |\sigma_{11}| \text{sgn}(V_3^+ - V_3^-), \Delta u_1 = 0. \end{aligned} \quad (14)$$

Условия (14) сохраняются до тех пор, пока  $\sigma_{11} < 0$ ; при  $\sigma_{11} \geq 0$  снова задаются условия (13) и начинается вычисляться интеграл для  $\Delta u_1$ .

2. **Трещина сдвига или комбинированная трещина, выполнен второй критерий.** В момент  $\tau_1$  образования трещины определяется  $\sigma_{11}$  и в зависимости от ее знака выбираются либо условия (12), (13), либо условие (14). Тело предполагается разрушенным, если для всех точек какого-нибудь его сечения выполняются условия разрушения.



Рис. 4

### Динамическое разрушение конструкции

**Начальные условия моделирования.** В качестве начальных условий задаются условия отсутствия напряжений в конструкции. В момент  $t = 0$  к поверхности  $x_2 = 0$  прикладываются постоянные растягивающие напряжения  $P_f$ , выводящие в процессе нагружения материал тела в пластическую область. Значение этих напряжений подбирается минимально необходимым для того, чтобы обеспечить разрушение моста за время, равное 20 временам пробега продольной волны по его длине.

**Параметры моделирования.** Моделирование динамики мостовых конструкций выполнялось с помощью численного интегрирования формул (1)–(6) с учетом особенностей точек, в которых проводилось интегрирование (например, внутренняя точка, граничная точка, внешняя угловая точка, внутренняя угловая точка, точка ребра и т. д.). При разработке алгоритмов использовались вышеуказанные нормировки, поэтому задавались только значение внешней нагрузки  $P_0$  ( $P_0 = 0,1 \dots 0,9$ ), коэффициент Пуассона  $\nu$  ( $\nu = 0,1, 0,2, 0,3$ ), безразмерная величина  $\delta$  ( $\delta = 0,05$ ) и геометрические параметры моста. После нормировок длина моста  $l$  равнялась  $320 \pm 20$  ячеек и подбирались более точно из того условия, что размеры всех опор должны быть одинаковы, высота моста равнялась 80 ячейкам, а его толщина — 20 ячейкам.

**Граничные условия** задавались в следующем виде: на верхней границе действует сжимающая нагрузка  $P_0$ , значение которой подбирается в целях определения нижнего предела разрушения несущей части моста  $P_f$ ; на границах  $x_1 = \min$  и  $x_1 = \max$  заданы условия продолжения конструкции [3] (для волн, движущихся, соответственно, влево от  $x_1 = \min$  и направо от  $x_1 = \max$ ), на нижней части опор заданы условия жесткой заделки, а остальные границы свободны от напряжений.

На рис. 4 показано срединное сечение мостовой конструкции в момент полного разрушения одного из ее сечений (между второй и третьей опорами); черным цветом закрашены области разрушения. Разрушение произошло за счет напряжений  $\sigma_{22}$ . Рисунок получен с помощью примитивов OpenGL и библиотеки ГРАФОР.

### Оптимизация динамической прочности моста

С помощью последовательности реализаций программ моделирования динамики разрушения мостовых конструкций решалась задача оптимизации их формы. Предполагалось, что несущая часть и опоры моста изготовлены из одного и того же материала. Длина, ширина и высота несущей части моста заданы и не меняются в процессе моделирования. Также постоянна суммарная масса опор, их высота и толщина. Все опоры одинаковые, их число  $N$  меняется за счет изменения их ширины. К конструкции приложены нагрузки  $P$  так, как это указано на рис. 3. Посредством моделирования определялись минимальные разрушающие (несущую часть) напряжения  $P_f$  как функции  $N$ . Возможность разрушения опор не рассматривалась.

Результаты моделирования представлены серией графиков на рис. 5; по оси абсцисс отложено число опор, по оси ординат — минимальная разрушающая нагрузка, нормированная на предел текучести материала на сдвиг. График, помеченный треугольным маркером, соответствует коэффициенту Пуассона  $\nu = 0,1$ , график с круглым маркером —  $\nu = 0,2$ , с квадратным маркером —  $\nu = 0,3$ .

Анализ графиков показывает, что с увеличением числа опор прочность конструкции возрастает. При этом, однако, следует учесть, что в заданной постановке значительное увеличение числа опор приведет к соответствующему уменьшению их сечений, а значит, первоначальные разрушения будут происходить уже не в несущей части, а в опорах (что не учитывается), а затем, после выхода из строя опор, — в несущей части.

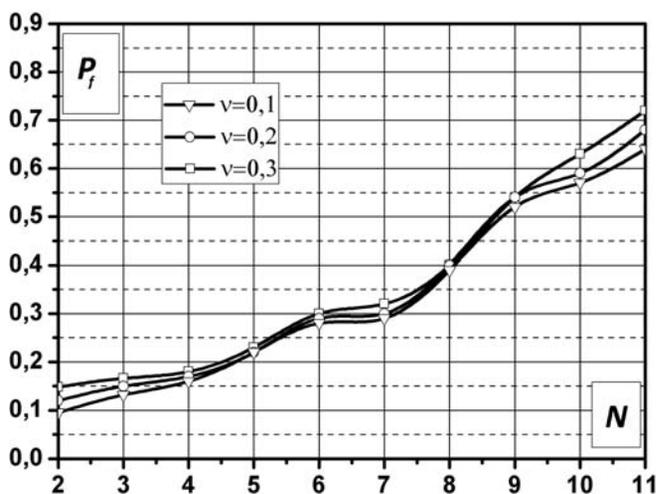


Рис. 5

## Заключение

Рассмотренный проект позволяет численно исследовать динамику и динамическое разрушение арочных и мостовых конструкций.

Представление дифференциальных уравнений, составляющих математическую модель, в характеристической форме приводит к сокращению числа независимых переменных с четырех до трех, при этом в каждом из этих уравнений вдоль характеристик (или вырожденных характеристик) изменяются только одно напряжение и одна скорость движения частиц материала тела. При алгоритмировании таких уравнений удастся создавать простые и быстрые в реализации вычислительные схемы. Эти схемы относятся к классу прямых, явных сеточно-характеристических схем первого порядка точности [4], вследствие чего возможно использование матричной записи определяемых переменных и независимый расчет напряженно-деформированного состояния в различных точках тела. Это, в свою очередь, позволяет распараллеливать вычисления

при использовании многоядерных процессоров, что также ускоряет работу программы.

Программа состоит из набора отдельных подпрограмм и функций; такой принцип построения позволяет легко изменять ее при изменении реологии материала, критериев разрушения, условий на трещинах, а также при изменении формы рассматриваемых конструкций.

Применение машинной графики позволяет наглядно представлять многомерные массивы данных, получаемых в процессе счета.

## Список литературы

1. Бульчев Г. Г. Построение матричной характеристической формы уравнений динамики анизотропных упруговязкопластических сред // Известия РАН. Механика твердого тела. 1995. № 1. С. 91–95.
2. Баяковский Ю. М., Галактионов В. А., Михайлова Т. Н. Графор. Графическое расширение фортрана. М.: Наука, 1985. 288 с.
3. Бульчев Г. Г. Моделирование динамики и динамического разрушения в ограниченных областях протяженных тел // Строительная механика и расчет сооружений. 2010. № 5. С. 39–43.

УДК 004.94

С. В. Лучкова, аспирант, Т. О. Перемитина, канд. техн. наук, науч. сотр.,  
И. Г. Ященко, канд. геол.-минер. наук, науч. сотр.,  
Институт химии нефти СО РАН, г. Томск, e-mail: sonetta@gmail.com

## Применение программного комплекса анализа многомерных данных на основе нечеткого и статистического моделирования

*Рассмотрен комплексный подход к анализу данных с пропущенными значениями, включающий восстановление данных с помощью нечеткой системы на основе эволюционной стратегии и выявление зависимостей свойств объектов с помощью факторного анализа с вращением. Описывается модель восстановления данных и алгоритм факторного анализа. Приведены результаты экспериментальных исследований для определения влияния состава и условий залегания нефтей на численность, распространение и активность пластовой микрофлоры.*

**Ключевые слова:** нечеткая система, восстановление данных, факторный анализ, физико-химические свойства нефти, микробиологические свойства пластовой воды

S. V. Luchkova, T. O. Peremitina, I. G. Vashchenko

## Application of Software Multidimensional Data Analysis Based on Fuzzy and Statistical Modeling

*In this paper we consider an integrated approach to the analysis of multidimensional data with missing values. This approach includes data recovered by fuzzy system based on evolutionary strategy and identifying dependencies of object properties by factor analysis with "varimax" rotation. The model of data recovery and algorithm of factor analysis are described. Experimental researches results are presented.*

**Keywords:** fuzzy system, data recovered, factor analysis, integrated approach, physic-chemical properties of oil, microbiological properties of produced water

## Введение

Природные объекты чаще всего являются очень сложными для выявления значимых зависимостей в их структуре и построения математических моделей, поэтому для более детального исследования такие объекты описывают большим числом признаков (параметров). Такие свойства исследуемых данных, как многомерность данных, отсутствующие значения, разнородность, приводят к ряду проблем при выявлении структурных зависимостей и требуют последовательного применения нескольких методов для получения результатов анализа, формируя комплексный анализ данных.

Одним из подходов к решению проблемы о пропущенных значениях является применение методов нечеткого моделирования, которые позволяют наиболее точно восстановить недостающие данные, снимая требования к исходным данным о нормальном распределении и однородности [1].

При решении проблемы многомерности объектов рассматривается вопрос о сокращении числа свойств объекта с помощью отбрасывания части из них или замены меньшим числом, при котором сохранится вся необходимая информации об объекте. Одним из подходов к решению данной задачи является использование метода факторного анализа, который позволяет сжать информацию и сделать объекты более наглядными для анализа. Также в исследовании сложных объектов факторный анализ позволяет выявить скрытые взаимосвязи объекта и их структурные особенности, что является важным аспектом, например, при анализе свойств такого природного объекта, как нефть.

Цель данной работы — описать комплексный подход к анализу многомерных данных с пропущенными значениями и показать применение программного комплекса на основе данного подхода на примере определения влияния состава и условий залегания нефти на численность, распространение и активность пластовой микрофлоры в задачах повышения нефтеотдачи.

### Модель восстановления пропущенных значений

Основой модели восстановления является нечеткая система типа сингтон [2], для которой выбрана треугольная функция принадлежности. Параметры функции принадлежности входных переменных нечеткой системы можно представить в виде вектора параметров [1]. Тогда параметрическая идентификация нечеткой системы рассматривается как процесс оптимизации параметров построенной модели, такой чтобы ошибка вывода была минимальной [2].

Для настройки вектора параметров нечеткой системы в нашей модели используется метод эволюционной стратегии. Стратегия основана на механизмах естественного отбора и наследования.

Преимущества этой стратегии перед другими методами оптимизации заключаются в параллельной обработке множества альтернативных решений [3, 4].

Модель восстановления пропущенных значений на основе нечеткой системы и метода эволюционной стратегии работает по следующему алгоритму [5]:

*Вход:* таблица наблюдений с пропусками в записях.

*Шаг 1.* Загружаем входные данные (таблицу наблюдения).

*Шаг 2.* Задаем параметры нечеткой системы.

*Шаг 3.* Выбираем параметры алгоритма эволюционной стратегии.

*Шаг 4.* Строим нечеткую систему.

*Шаг 5.* Настраиваем нечеткую систему, применяя эволюционную стратегию.

*Шаг 6.* Отбираем лучшую хромосому.

*Шаг 7.* Восстанавливаем пропущенные значения на основе сформированной базы правил и лучшей хромосомы. Если достигнуто условие выхода, переходим к *Шагу 8*, иначе *Шаг 5*.

*Шаг 8.* Выводим результат.

*Выход:* таблица наблюдений с восстановленными значениями.

### Алгоритм факторного анализа с вращением

Факторный анализ является одним из разделов многомерного статистического анализа. Он основан на многомерном нормальном распределении, т. е. каждый из используемых признаков изучаемого объекта должен иметь нормальный закон распределения. Факторный анализ исследует внутреннюю структуру ковариационной и корреляционной матриц системы признаков изучаемого объекта [1].

Пусть в изучаемом объекте отобрано  $N$  записей. В каждой из них измерены значения  $K$  параметров и получены значения случайных многомерных нормально распределенных величин. Эти значения случайных многомерных величин обусловлены различными причинами, которые называются факторами. Предполагается, что число этих факторов всегда меньше, чем число  $K$  измеряемых параметров изучаемого объекта. Эти факторы являются скрытыми, их нельзя непосредственно измерить, и поэтому они представляются гипотетическими. Однако имеются методы их выявления, которые и составляют сущность факторного анализа.

В факторном анализе решаются следующие задачи [6—8]:

- определить число действующих факторов и указать их относительную интенсивность;
- выявить признаковую структуру факторов, т. е. показать, какими признаками объекта обусловлено действие того или иного фактора и в какой относительной мере;
- выявить факторную структуру изучаемых параметров объекта, т. е. показать долю влияния ка-

ждого из факторов на значение того или иного параметра этого объекта;

- воссоздать в факторном координатном пространстве облик изучаемого объекта, используя вычисляемые значения факторов для каждого наблюдения исходной выборочной совокупности.

Алгоритм факторного анализа с вращением:

*Вход:* таблица наблюдений без пропусков (с восстановленными данными).

*Шаг 0.* Загружаем данные и выбираем анализируемые параметры.

*Шаг 1.* Нормируем данные.

*Шаг 2.* Рассчитываем матрицы ковариации и корреляции.

*Шаг 3.* Вычисляем собственные числа и собственные векторы (применяя разложение Холецкого, *LU*-разложение [9]).

*Шаг 4.* Рассчитываем факторы и вычисляем долю влияния каждого из факторов на значения параметров.

*Шаг 5.* Выявляем наиболее значимые факторы.

*Шаг 6.* Воссоздаем в факторном координатном пространстве изучаемый объект (отображаем на пространственном графике).

*Шаг 7.* Применяем ортогональное вращение методом "Варимакс" [10] для увеличения критерия качества каждого фактора.

*Шаг 8.* Классифицируем данные, применяя метод *K*-средней кластеризации [11] для разделения данных на классы. Если достигнуто условие завершения анализа, переходим к *Шагу 9*, иначе *Шаг 0*.

*Шаг 9.* Выводим результаты.

*Выход:* таблицы с рассчитанными данными и графические отображения.

## Программная реализация

В среде визуального программирования Microsoft Visual Studio 2012 был реализован программный комплекс для комплексного анализа многомерных данных. Реализованное приложение имеет простую систему классов для выполнения основных функций комплексного анализа на основе нечеткого и статистического моделирования.

Программный комплекс включает в себя следующие основные блоки (рис. 1):

- блок "Моделирование системы", который позволяет провести идентификацию параметров нечеткой системы;
- блок "Восстановление данных", который позволяет с помощью подобранных параметров восстанавливать пропущенные данные в массиве данных. Также модуль включает в себя подраздел "Имитации восстановления", в котором на основе метода скользящего экзамена [12] можно рассчитать адекватность подобранной системы;
- блок "Анализа данных", который включает в себя факторный анализ с вращением и классификацию данных. Также можно посмотреть результаты основных расчетов в виде таблиц ковариации, корреляции, сводных таблиц для собственных чисел и векторов, графическое отображение данных в пространстве факторов, гистограммы факторных нагрузок и отображение данных с учетом классификации.

Согласно представленной структуре программного комплекса наш подход комплексного анализа будет состоять из последовательного выполнения нескольких шагов:

- моделирования системы;
- восстановления данных;
- проведения факторного анализа и классификации.

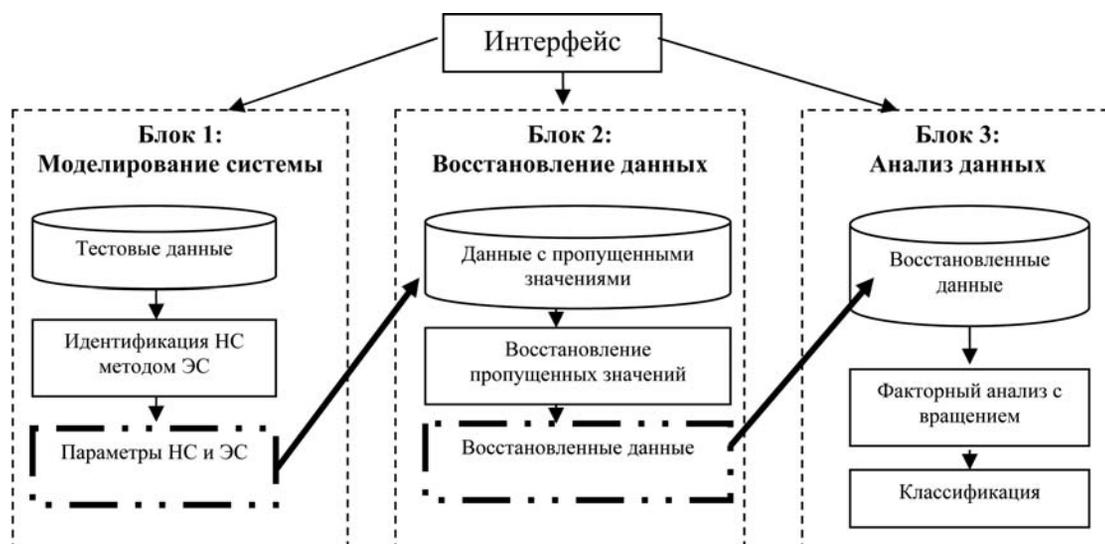


Рис. 1. Структура программного комплекса

## Применения программного комплекса

Разработанный программный комплекс был применен на сформированном многомерном массиве описаний физико-химических свойств нефти и микробиологических данных соответствующих пластовых вод (табл. 1).

### Подготовка данных

Первоначальный анализ многомерного массива с данными о свойствах нефти и пластовой воды (включающий 18 образцов нефти и воды по 13 свойствам, табл. 1) показал, что из 234 значений пропущено 19 (8 %).

Таблица 1

#### Перечень исследуемых свойств нефти и пластовой воды

Характеристики	Обозначение	Исследуемые свойства
1. Физико-химические	1.1	Плотность, г/см <sup>3</sup>
	1.2	Вязкость при 20 °С, мм <sup>2</sup> /с
	1.3	Содержание серы, %
	1.4	Содержание смол, %
	1.5	Содержание твердого парафина, %
	1.6	Содержание асфальтенов, %
2. Микробиологические	2.1	Степень кислотности или щелочности воды <i>pH</i>
	2.2	Окислительно-восстановительный потенциал <i>Eh</i> , мВ
	2.3	Сухой остаток (минерализация), г/л
	2.4	Гетеротрофы, тысяч клеток/мл
	2.5	Сульфовосстанавливающие бактерии (СВБ), тысяч клеток/мл
	2.6	Денитрифицирующие бактерии (ДНБ), тысяч клеток/мл
	2.7	Углеводородокисляющие бактерии (УОБ), тысяч клеток/мл

### Моделирование системы

Согласно предложенной методике алгоритм действий, которого необходимо придерживаться для достижения наилучшего результата моделирования, выглядит следующим образом.

**Шаг 1. "Тестовая выборка".** Этот этап заключается в подготовке тестового массива без пропущенных значений из начального массива данных, т. е. мы формируем тестовую выборку, удаляя все образцы нефти, у которых есть пропущенные свойства.

**Шаг 2. "Идентификация параметров".** На тестовой выборке осуществляется подбор оптимальных параметров для нечеткой и эволюционной систем.

### Восстановление данных

**Шаг 1. "Проверка подобранных параметров".** На этом шаге в тестовом массиве вводятся искусственные пропуски и проводится восстановление с подобранными параметрами в блоке "Моделирование системы". Это необходимо, чтобы удостове-

риться, что подобранные параметры действительно оптимальны. Далее, после получения восстановленного тестового массива рассчитывается критерий адекватности выбранной модели восстановления, и если результат удовлетворяет, то можно переходить к следующему шагу, иначе возвращаемся в блок "Моделирование системы".

**Шаг 2. "Восстановление параметров".** Проводим процедуру восстановления на начальном массиве с подобранными параметрами.

### Факторный анализ и классификация

Переходим в блок "Анализ данных". Загружаем массив с восстановленными данными и переходим к анализу полученных результатов.

Рассмотрим полученные результаты для наших данных. Корреляционные матрицы нормированных признаков представлены в табл. 2, 3. Нумерация признаков соответствует списку в табл. 1.

Таблица 2

#### Корреляционная матрица физико-химических свойств

Признаки	Коэффициенты корреляции					
	1.1	1.2	1.3	1.4	1.5	1.6
1.1	1	<b>0,82</b>	<b>0,52</b>	<b>0,57</b>	0,11	0,27
1.2	<b>0,82</b>	1	0,26	0,20	0,18	0,08
1.3	<b>0,52</b>	0,26	1	<b>0,88</b>	-0,31	0,01
1.4	<b>0,57</b>	0,20	<b>0,88</b>	1	-0,11	0,30
1.5	0,11	0,18	-0,31	-0,11	1	-0,03
1.6	0,27	0,08	0,01	0,30	-0,03	1
2.1	-0,25	-0,21	0,04	-0,27	-0,24	-0,32
2.2	-0,45	-0,71	-0,26	0,01	0,32	0,01
2.3	<b>0,48</b>	<b>0,62</b>	0,18	0,20	-0,25	0,11
2.4	0,21	0,39	0,11	-0,002	-0,16	-0,06
2.5	<b>0,67</b>	<b>0,86</b>	0,36	0,18	-0,16	0,13
2.6	<b>0,47</b>	<b>0,67</b>	0,39	0,13	-0,23	0,01
2.7	<b>0,68</b>	<b>0,87</b>	0,34	0,15	-0,12	0,08

Таблица 3

#### Корреляционная матрица микробиологических свойств

Признаки	Коэффициенты корреляции						
	2.1	2.2	2.3	2.4	2.5	2.6	2.7
1.1	-0,25	-0,45	<b>0,48</b>	0,21	<b>0,67</b>	<b>0,47</b>	<b>0,68</b>
1.2	-0,21	-0,71	<b>0,62</b>	0,39	<b>0,86</b>	<b>0,67</b>	<b>0,87</b>
1.3	0,04	-0,26	0,18	0,11	0,36	0,39	0,34
1.4	-0,27	0,01	0,20	-0,002	0,18	0,13	0,15
1.5	-0,24	0,32	-0,25	-0,16	-0,16	-0,23	-0,12
1.6	-0,32	0,01	0,11	-0,06	0,13	0,01	0,08
2.1	1	-0,12	-0,45	0,003	0,03	0,18	0,04
2.2	-0,12	1	<b>-0,57</b>	-0,43	<b>-0,90</b>	<b>-0,69</b>	<b>-0,86</b>
2.3	-0,45	<b>-0,57</b>	1	<b>0,55</b>	<b>0,59</b>	0,30	<b>0,55</b>
2.4	0,003	-0,43	<b>0,55</b>	1	0,45	0,32	<b>0,46</b>
2.5	0,03	<b>-0,90</b>	<b>0,59</b>	0,45	1	<b>0,82</b>	<b>0,98</b>
2.6	0,18	<b>-0,69</b>	0,30	0,32	<b>0,82</b>	1	<b>0,82</b>
2.7	0,04	<b>-0,86</b>	<b>0,55</b>	<b>0,46</b>	<b>0,98</b>	<b>0,82</b>	1

**Примечание:** коэффициенты корреляции являются значимыми ( $\beta = 0,05$ ) при их абсолютном значении не менее 0,46

Анализ коэффициентов корреляции показывает, что признак 1.1 (плотность нефти) связан одновременно с вязкостью нефти (0,82), содержанием смол (0,57), серы (0,52), микробиологическими данными УОБ (0,68), СВБ (0,67), сухим остатком (0,48) и ДНБ (0,47), но не связан и биполярен со следующими признаками: содержание парафина, асфальтенов,  $pH$ , гетеротрофов и  $Eh$ . При этом гетеротрофы связаны с УОБ (0,46).

Вязкость нефти также связана с микробиологическими признаками — СВБ (0,86), УОБ (0,87), ДНБ (0,67) и сухим остатком (0,62), но с остальными признаками биполярна и не связана.

Содержания серы и смол связаны между собой (0,88), а с остальными признаками не связаны или биполярны.

Анализ корреляционной матрицы признаков позволяет выявить структуру взаимосвязей признаков, которая графически может быть представлена в виде иерархической дендрограммы (рис. 2).

Далее нам необходимо выделить факторы и их нагрузки. Перейдем к рассмотрению собственных чисел и собственных векторов корреляционной матрицы. Данные представлены в табл. 4.

Значения собственных чисел  $\lambda_j$  (факторные нагрузки  $a_{ij}$ ) и векторов (веса факторов, %) факторной нагрузки признаков в табл. 4 показывают, что на численность, распространение и активность пластовой микрофлоры оказывают влияние на 44,23 % фактор  $F_1$  на 15,96 % фактор  $F_2$  и на 12,75 % фактор  $F_3$ .

Анализ признаковой структуры фактора  $F_1$  показывает, что нагрузка этого фактора значимо определяется концентрациями СВБ (0,959), УОБ (0,946), вязкостью нефти (0,895), ДНБ (0,790), плотностью нефти (0,781), сухого остатка (0,692), гетеротрофов (0,516), содержанием серы (0,503) и также имеет значимую отрицательную связь с  $Eh$  (-0,840).



Рис. 2. Диаграмма связи признаков

Самое значимое влияние оказывают СВБ, концентрация которых влияет на процессы и качество нефтедобычи [13]. Так, адаптация СВБ в нефтяном пласте вызывает интенсивные процессы образования сероводорода. А сероводород усиливает коррозию нефтяного оборудования и ухудшает товарное качество добываемой нефти и к тому же создает дополнительные технологические сложности при очистке и переработке добытой нефти. Также эти бактерии могут закупоривать пластовые поры за счет образования слизи и осадков сульфата железа, который образуется в результате взаимодействия с выделяющимися сероводородом и ионами железа. Все это снижает объем нефтеотдачи.

Значимость же УОБ объясняется тем, что продукты жизнедеятельности этих бактерий служат источниками питания для СВБ и достаточного количества сульфатов. В комплексном развитии с дру-

Таблица 4

Факторные нагрузки, собственные значения и векторы

Признаки	Факторные нагрузки $a_{ij}$												
	$F_1$	$F_2$	$F_3$	$F_4$	$F_5$	$F_6$	$F_7$	$F_8$	$F_9$	$F_{10}$	$F_{11}$	$F_{12}$	$F_{13}$
1.1	<b>0,781</b>	-0,422	0,097	0,254	0,039	0,035	0,230	0,242	-0,137	-0,030	0,058	0,005	0,0002
1.2	<b>0,895</b>	-0,005	0,319	0,247	0,085	-0,042	0,052	0,080	0,035	-0,056	-0,115	-0,014	0,008
1.3	<b>0,503</b>	<b>-0,483</b>	<b>-0,660</b>	0,054	0,232	0,011	-0,050	-0,116	0,017	-0,027	0,013	-0,056	0,035
1.4	0,360	<b>-0,819</b>	-0,404	0,004	0,137	0,079	-0,005	-0,065	0,043	0,027	-0,030	0,050	-0,038
1.5	-0,190	-0,176	<b>0,625</b>	<b>0,638</b>	0,231	0,205	-0,034	-0,166	0,095	-0,018	0,038	-0,006	-0,003
1.6	0,142	<b>-0,499</b>	0,192	-0,161	<b>-0,762</b>	0,288	0,013	-0,042	0,041	-0,025	-0,003	-0,021	0,006
2.1	-0,098	<b>0,596</b>	<b>-0,602</b>	0,295	-0,089	0,287	0,268	0,054	0,143	-0,010	-0,0003	0,001	-0,006
2.2	<b>-0,840</b>	-0,374	0,037	0,072	0,110	0,160	-0,138	0,262	0,097	0,108	-0,015	0,004	0,021
2.3	<b>0,692</b>	-0,075	0,309	<b>-0,542</b>	0,177	-0,153	0,122	0,048	0,233	-0,007	0,036	-0,010	-0,002
2.4	<b>0,516</b>	0,271	0,127	-0,440	0,327	<b>0,573</b>	-0,092	0,001	-0,082	-0,013	-0,004	0,003	-0,0002
2.5	<b>0,959</b>	0,177	0,025	0,092	-0,131	-0,016	0,003	-0,095	0,016	0,074	0,009	0,086	0,034
2.6	<b>0,790</b>	0,258	-0,208	0,212	-0,155	-0,037	-0,409	0,135	0,069	-0,068	0,030	0,005	-0,013
2.7	<b>0,946</b>	0,203	0,044	0,139	-0,084	-0,009	-0,015	-0,029	-0,030	0,172	0,001	-0,061	-0,018
Собственные значения $\lambda_j$	5,7493	2,0743	1,658	1,2137	0,9218	0,595	0,3411	0,2141	0,13	0,0583	0,0217	0,0177	0,0049
Вес факторов, %	44,23	15,96	12,75	9,34	7,09	4,58	2,62	1,65	1	0,45	0,17	0,14	0,04

гими группами аэробных и анаэробных микроорганизмов СВБ участвуют в разложении простых и сложных субстратов, в том числе белков, жиров, углеводов, клетчатки. В сообществе с УОБ они принимают участие в окислении углеводородов [14]. Это, в свою очередь, объясняет большую связь с параметром сухого остатка (минерализации) [15, 16], выделение сульфатов показывает, что исследуемые месторождения обладают солоноватыми пластовыми водами.

Вязкость нефти является важным фактором в технологическом плане, так как именно значение вязкости учитывается для оценки скорости фильтрации в пласте и выборе вытесняющего агента, так как вязкость характеризует силу трения между смежными слоями внутри жидкости при их взаимном перемещении.

Значимость ДНБ объясняется тем, что они в большей степени способны к использованию жидких легкокипящих *n*-алканов и ароматических углеводородов. Другой причиной их преимущественного распространения в нефтяных пластах является принадлежность к факультативным анаэробам, т. е. способности существовать в анаэробных и микроаэрофильных условиях.

Высокая значимость плотности нефти объясняется наличием выше описанных бактерий, к тому же среднее значение плотности по всем образцам нефти равно 0,865, что позволяет отнести данные образцы нефти к классу тяжелой нефти, которая обладает повышенной концентрацией смолисто-асфальтеновых компонентов.

Отрицательную связь *Eh* можно объяснить тем, что окислительно-восстановительные процессы зависят от концентрации окисленной и восстановленной форм элемента, температуры и особенно сильно — от степени кислотности (*pH*) раствора. У окислителей, молекулы которых содержат кислород, *Eh* возрастает с увеличением концентрации ионов водорода, а  $F_1$  практически не обладает связью с концентрацией *pH*.

Анализ признаков структуры фактора  $F_2$ , наоборот, показывает именно значимость степени кислотности *pH*. Влияние содержания твердого парафина выявляется факторами  $F_3$  и  $F_4$ .

Определим минимальное число факторов, адекватно воспроизводящих наблюдаемые корреляции. Наиболее часто число выделяемых факторов определяется числом собственных чисел, больших единицы. В нашем случае это факторы с 1-го по 9-й. Но как видно из табл. 4, факторы с 5-го по 9-й не обладают значимыми коэффициентами исследуемых свойств. Поэтому лучше обратиться к другому критерию определения значимых факторов — через графическое изображение собственных чисел. Выделение заканчивается на том факторе, после которого исследуемая зависимость близка к горизонтальной линии. Как видно на рис. 3, после 4-го фактора

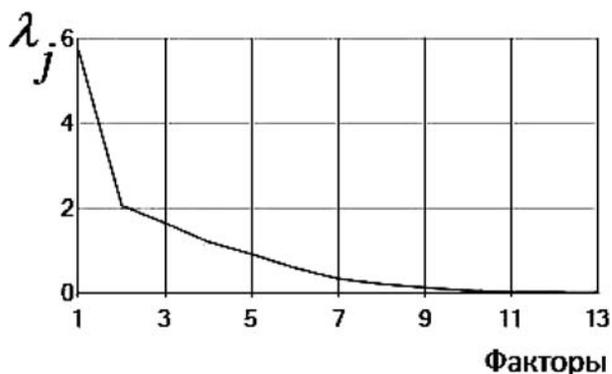


Рис. 3. Зависимость собственных значений и факторов

график стремится к более горизонтальному расположению, но окончательное решение должно базироваться на приемлемости с точки зрения научных представлений в данной области.

Следующим шагом с помощью процедуры вращения выявляются наиболее легко интерпретируемые факторы, т. е. целью вращения является получение наиболее простой факторной структуры или достижение простоты интерпретации признаков и факторов. В нашем программном комплексе используется метод "Варимакс" в предположении, что он дает лучшее разделение факторов, так как в нем для каждого признака добиваются, чтобы дисперсия квадратов нагрузок фактора была максимальна.

Как видно из табл. 5, вращение не привело к улучшению, а поворот более чем на  $4^\circ$  приводит, наоборот, к снижению весов факторов, поэтому оставим данные без вращения.

Таблица 5  
Факторные нагрузки, собственные значения после вращения

Признаки	Факторные нагрузки $a_{ij}$			
	$F_1$	$F_2$	$F_3$	$F_4$
1.1	<b>0,7166</b>	-0,4208	0,0967	0,2544
1.2	<b>0,9471</b>	-0,0343	0,3188	0,2466
1.3	<b>0,5031</b>	<b>-0,4827</b>	<b>-0,6375</b>	0,0537
1.4	0,3600	<b>-0,8186</b>	-0,4379	0,0067
1.5	-0,1900	-0,1756	<b>0,6246</b>	<b>0,6384</b>
1.6	0,1417	<b>-0,4984</b>	0,1917	-0,1611
2.1	-0,0974	<b>0,5962</b>	<b>-0,6018</b>	0,2953
2.2	<b>-0,8398</b>	-0,3742	0,0365	0,0718
2.3	<b>0,6912</b>	-0,0747	0,3084	<b>-0,5416</b>
2.4	<b>0,5159</b>	0,2707	0,1273	-0,4399
2.5	<b>0,9580</b>	0,1769	0,0251	0,0921
2.6	<b>0,7893</b>	0,2579	-0,2080	0,2121
2.7	<b>0,9456</b>	0,2026	0,0437	0,1385
Собственные значения $\lambda_j$	5,7427	2,0739	1,6566	1,2137
Вес факторов, %	44,17	15,95	12,74	9,34
Критерий "Варимакс" до вращения	0,2466	0,1341	0,1112	0,0846
Критерий "Варимакс" после вращения	0,2467	0,1341	0,1113	0,0846

Получаем, что из графического представления и численных вычислений, три-четыре фактора наиболее полно характеризуют признаковую структуру данных месторождений, на которой четко выделяются три класса месторождений.

"Класс 1" включает в себя три месторождения: это Северное, Усинское и Урманское, которые очень сильно выделяются по параметру концентрации гетеротроф.

В "Класс 2" вошло 11 месторождений: Вахское, Самотлорское, Советское (проба 2004 г.), Вахское (проба 2004 г.), Зуунбаян, Тамсагулаг, Русское, Ульяновское, Фуларти, Шингинское и Арчинское.

"Класс 3" содержит четыре месторождения: Советское, Крапивинское, Белый Тигр и Цыган-Элс.

Так как в литературе встречаются описания полномасштабных изучений месторождений "Класс 1", мы их рассматривать не будем. Для решения же вопроса о "существовании или отсутствии значимого различия между свойствами образцов "Класс 2" и "Класс 3" проанализируем данные этих классов отдельно.

Месторождения "Класс 2" отличаются максимальной концентрацией смол и асфальтенов. Образцы "Класс 3" обладают максимальными концентрациями УОБ, ДНБ, гетеротрофов, наибольшим содержанием парафинов и наибольшей вязкостью, при этом данные характеристики в 2...55 раз выше, чем у образцов класса 2, что и является ответом на вопрос о значимых различиях классов.

Оба класса месторождений являются схожими по следующим признакам: концентрации  $pH$ ,  $Eh$ , сухого остатка, СВБ, содержания серы и плотности нефти. Значимыми же физико-химическими характеристиками исследуемых образцов нефти являются вязкость нефти при 20 °С и содержание парафина. Образцы класса 2 также отличаются содержанием смол. Значимыми микробиологическими свойствами пластовой воды являются гетеротрофы и окислительно-восстановительный потенциал. А для класса 3 выделяются значимость ДНБ и УОБ.

Таким образом, сопоставление результатов комплексного анализа позволяет выделить взаимосвязи свойств микрофлоры и физико-химических характеристик нефти: высоковязкие, парафинистые нефти обладают пластовыми водами с наибольшей концентрацией гетеротроф, ДНБ и УОБ. При этом для менее вязких, смолистых нефтей характерно почти одинаковые  $Eh$ , минерализация и концентрация ДНБ, СВБ и УОБ.

### Заключение

В работе рассмотрено применение программного комплекса для анализа многомерных данных с пропущенными значениями, включающего в себя нечеткую систему на основе эволюционной стратегии

для восстановления пропущенных значений, факторный анализ с вращением для проведения анализа. Анализ включает в себя выявление взаимосвязей различных признаков природных объектов, главных действующих факторов, их признаковой структуры и факторной структуры. Более того, метод позволяет воссоздать в факторном координатном пространстве облик изучаемого природного объекта и указать его характерные признаки и отличительные особенности. Применение методов нечеткого моделирования и факторного анализа для исследования многомерных данных о микробиологических свойствах пластовых вод и физико-химических характеристиках нефти позволило выявить взаимосвязи между данными исследуемых месторождений, что имеет важное научное значение и практическую ценность при совершенствовании технологий повышения нефтеотдачи нефтяных запасов.

### Список литературы

1. **Перемитина Т. О., Лучкова С. В.** Применение программного комплекса "Нечеткая система на основе эволюционной стратегии" для задачи импутирования // Информационные технологии. 2013. № 2. С. 47–50.
2. **Ходашинский И. А., Гнездилова В. Ю., Дудин П. А., Лавыгина А. В.** Основанные на производных и метаэвристические методы идентификации параметров нечетких моделей // Труды VIII международной конференции "Идентификация систем и задачи управления" SICPRO'08 (Москва). 2009. С. 501–528.
3. **Рутковская Д., Пилиньский М., Рутковский Л.** Нейронные сети, генетические алгоритмы и нечеткие системы. М.: Горячая линия — Телеком, 2006. 383 с.
4. **Noche S. and Wrobel S.** A Comparative Evaluation of Feature Set Evolution Strategies for Multirelational Boosting // Proc. 13th Int. Conf. on ILP. 2003.
5. **Лучкова С. О.** Идентификация нечеткой системы методом эволюционной стратегии // Матер. сб. трудов Всероссийского конкурса научно-исследовательских работ студентов и аспирантов в области информатики и информационных технологий в рамках Всероссийского фестиваля науки (Белгород). 2011. С. 92–101.
6. **Лоули Д., Максвелл А.** Факторный анализ как статистический метод. М.: Мир, 1967. 144 с.
7. **Белонин М. Д., Голубева В. А., Скублов Г. Т.** Факторный анализ в геологии. М.: Недра, 1982. 269 с.
8. **Ким Дж., Мюллер Ч. У.** Факторный, дискриминантный и кластерный анализ. М.: Финансы и статистика, 1989. 215 с.
9. **Lloyd N., Bau D.** Numerical linear algebra // SIAM: Society for Industrial and Applied Mathematics. 1997. 263 p.
10. **Харман Г.** Современный факторный анализ. М.: Статистика, 1972. 483 с.
11. **Vance F.** Clustering and the Continuous k-Means Algorithm // Los Alamos Science. 1994. N 22. P. 138–144.
12. **Загоруйко Н. Г.** Методы распознавания и их применение. М.: Сов. радио, 1972. 216 с.
13. **Сухотин А. М.** Коррозия под действием теплоносителей, хладагентов и рабочих тел. Л.: Химия, 1988. 360 с.
14. **Зайнуллин Р. С.** Ресурс трубопроводных систем. Уфа: БЭСТС, 2005. 836 с.
15. **Лукнер Р., Шестаков В. М.** Моделирование миграции подземных вод. М.: Недра, 1986. 208 с.
16. **Яценко И. Г., Сваровская Л. И., Перемитина Т. О.** Исследование влияния химического состава и условий залегания нефтей на численность и активность пластовой микрофлоры в задачах повышения нефтеотдачи на основе статистического анализа // Экспозиция Нефть Газ. 2013. № 2. С. 35–38.

УДК 004.322.067

**М. В. Тарасюк**, канд. техн. наук, доц., системный аналитик,  
ЗАО "Институт сетевых технологий", г. Санкт-Петербург  
e-mail: miket@int.spb.ru

## Выбор топологии VPN-сети с учетом требований конфиденциальности характеристик телекоммуникационной нагрузки

*Предложена модель оценки информативности межпакетных интервалов в канале связи, с учетом которой сформулированы принципы выбора топологии виртуальных каналов в сетях специальной связи, построенных на основе VPN (Virtual Private Network)-технологий, учитывающие требования по обеспечению конфиденциальности характеристик телекоммуникационной нагрузки.*

**Ключевые слова:** скрытый канал, виртуальный канал, VPN-сеть, маскировка трафика, скрытность связи

M. V. Tarasuk

## Choice of the VPN Topology Taking into Account Requirements of the Confidentiality of the Telecommunication Loading Characteristics

*In this paper the model of the amount information estimation in the packets time delays, transfer throw untrusted telecommunication canal, is proposed. The principles of the construction virtual channels topology for the special crypto security network on the base of the VPN technology are define for considering security requirements of the telecommunication loading characteristics such network.*

**Keywords:** covert channel, virtual channel, virtual private network, crypto-router, traffic masquerading

### 1. Обеспечение скрытности объектов сети шифрованной связи

Помимо требований стойкости шифрования (определяются категорией конфиденциальности передаваемой информации), сети шифрованной связи (СШС), создаваемые в интересах государственного и военного управления, зачастую должны удовлетворять дополнительным требованиям по обеспечению конфиденциальности (скрытности) объектов, обслуживаемых СШС.

Для СШС, построенных на принципах коммутации каналов и линейных шифраторов, данные требования имеют отношение лишь к физическим полям и сигналам, возникающим при функционировании каналообразующей аппаратуры СШС. При вскрытии информационного направления анализ структурных характеристик шифрованного трафика объекта (узла) СШС оказывается невозможным, поскольку на выходе шифратора имеет место однородный поток канальных символов, относительно которого не могут быть выделены события, отно-

сящиеся к конкретным пакетам, абонентам, приложениям или комплексам технических средств (за исключением синхроросылок и сигнализации установления соединений).

В СШС с пакетной коммутацией используется технология VPN и сетевое шифрование, а средствами построения СШС являются криптомаршрутизаторы. Общеизвестно, что сети с пакетной коммутацией обеспечивают более эффективное использование ресурсов связи, а также являются более надежными и отказоустойчивыми при выходе из строя отдельных узлов и линий коммуникаций, что достигается возможностью автоматического выбора альтернативного маршрута доведения данных при сбоях и отказах. Однако сетевое шифрование VPN-сети не обеспечивает конфиденциальность структурных характеристик телекоммуникационной нагрузки в канале связи, в результате чего данная технология более уязвима и в ряде случаев требует принятия ряда дополнительных мер защиты.

При использовании сетевого шифрования в канале связи передаются пакеты, инкапсулирующие

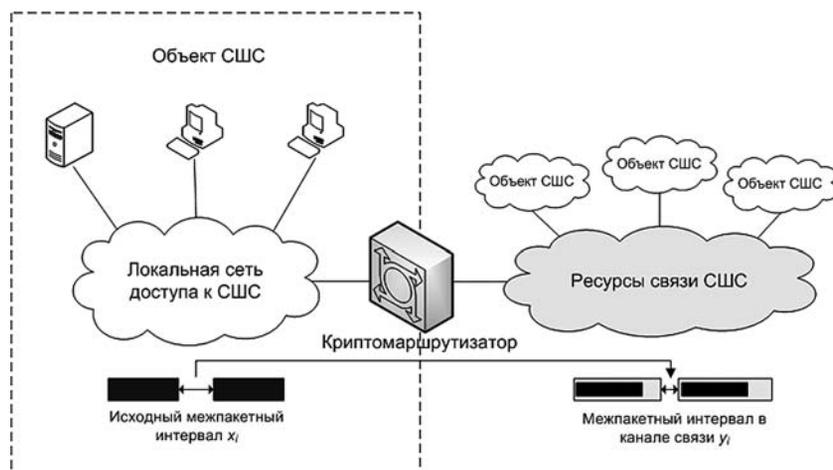


Рис. 1. Схема прохождения пакетов в СШС

(в зашифрованном на соответствующем ключе виде) исходные сетевые пакеты от локальных абонентов объекта СШС. Выделяются внешние (передаваемые по открытому каналу) и внутренние (передаваемые от локальных абонентов СШС в криптомаршрутизатор) сетевые пакеты, связь между которыми определяется посредством задания так называемых виртуальных каналов. Виртуальный канал определяет набор правил построения VPN-сети, применяемых к сетевым пакетам с определенными параметрами (например, адресами, типами сетевых соединений и пр.). В частности, виртуальный канал определяет ключ и алгоритм шифрования пакета, адрес сетевого устройства (криptomаршрутизатора), в котором должна быть выполнена его расшифровка. Совокупность виртуальных каналов, "наложенных" на физическую сеть с заданной на них схемой маршрутизации, обычно и называют виртуальной сетью (VPN).

При вскрытии (на физическом уровне) информационного направления оказывается возможным анализ параметров сетевого трафика, целями которого могут быть раскрытие назначения и функций объекта СШС [1], либо перехват информации, закодированной в параметрах трафика, при условии, что в объект СШС внедрены "программные закладки".

Для защиты от указанных угроз используют меры, направленные на обеспечение безопасности программной среды абонентов СШС, либо маскировку (передача в канал связи служебного трафика, не несущего полезной информации и рассматриваемого как "шум" при демодуляции данных в межпакетных интервалах) параметров сетевого трафика (в открытом канале). Первый способ не оказывает влияния на пропускную способность и вероятностно-временные характеристики (ВВХ) сети, однако возможность его применения ограничена объектами СШС, обслуживаемыми абонентские рабочие места с доверенной программно-аппаратной средой (модель нарушителя по 1-му уровню [2]).

Второй способ не имеет подобного ограничения, но при этом приводит к значительным накладным расходам на передачу служебного (маскировочного) трафика и ухудшению ВВХ обмена данными, особенно при передаче асинхронной (нерегулярной во времени) нагрузки с высоким коэффициентом пульсаций. Для снижения накладных расходов можно использовать адаптивные методы, обеспечивающие "подстройку" интенсивности маскировочного трафика под текущую (фоновую) нагрузку сети с учетом требований по уровню обеспечиваемой при этом безопасности, например, как предложено в работах [3] и [4]. Интенсивность маскировочного трафика в каждый момент времени можно определять исходя из нормы безопас-

ности (пропускная способность канала утечки), выбираемой с учетом категории объекта СШС (уровня секретности передаваемой информации).

Для оценки информативности межпакетных интервалов в условиях передачи фоновой нагрузки удобно воспользоваться методами теории массового обслуживания. Будем далее предполагать, что в составе объекта СШС имеются источники нагрузки, трафик которых кодирует информацию межпакетными интервалами, а схема прохождения пакетов в канал связи соответствует рис. 1. Для простоты предполагается, что все пакеты в СШС имеют одинаковый размер (выровнены на фиксированную границу) и экспоненциальный закон распределения для межпакетных интервалов и времени, затрачиваемого на коммутацию и передачу в канал связи при условии отсутствия потерь на сетевых интерфейсах криптомаршрутизатора.

## 2. Модель оценки информативности межпакетных интервалов

Плотность распределения промежутков времени между пакетами, поступающими от источников нагрузки, обозначим  $f(x) = ae^{-ax}$  ( $a$  — среднее время между моментами поступления пакетов), а плотность распределения для времени обработки пакета в криптомаршрутизаторе, соответственно,  $g(x) = be^{-bx}$  ( $b$  — среднее время передачи пакета по информационному направлению СШС).

В процессе передачи по каналу связи исходные межпакетные интервалы ( $x_i$ ) искажаются вследствие очередей на интерфейсах криптомаршрутизатора объекта СШС. Общее время нахождения пакета в криптомаршрутизаторе  $w$  с учетом буферизации имеет плотность распределения  $p(w) = (b - a)e^{-(b - a)w}$ . Учитывая линейный порядок обработки пакетов, искажение  $i$ -го межпакетного интервала можно представить разностью двух случайных величин  $\Delta_i = w_i - w_{i-1}$ . Плотность распре-

деления  $\Delta_i$  для стационарного режима работы сети в этом случае имеет следующий вид:

$$d(\Delta) = \begin{cases} \frac{(b-a)e^{-(b-a)\Delta}}{2}, & \text{если } \Delta \geq 0, \\ \frac{(b-a)e^{(b-a)\Delta}}{2}, & \text{если } \Delta < 0. \end{cases}$$

Энтропия канального шума при демодуляции межпакетного интервала определяется плотностью искажений для кодирующих межпакетных интервалов  $d(\cdot)$  и может быть вычислена следующим образом:

$$H(d(\Delta)) = \log_2 \left[ \frac{b-a}{2} \right] - 1.$$

На выходе в канал связи, вне зависимости от порядка  $i$ , плотность распределения межпакетных интервалов СШС с учетом задержки в криптомаршрутизаторе  $y = x + \Delta$  имеет вид:

$$f(y) = \begin{cases} \frac{a(b-a)^2}{(b-2a)b} e^{-ay} - \frac{a(b-a)}{2(b-2a)} e^{-(b-a)y}, & \text{если } y \geq 0, \\ \frac{a(b-a)}{2b} e^{(b-a)y}, & \text{если } y < 0. \end{cases}$$

Исходя из общих положений теории информации, ввиду независимости процессов генерации нагрузки и обработки пакетов в сетевых устройствах СШС, общая информативность межпакетного интервала на внешнем интерфейсе криптомаршрутизатора может быть оценена следующим образом:

$$I(a, b) = P[y > 0][H(f(y)) - H(d(\Delta))] = \\ = - \left[ 1 - \frac{a}{2b} \right] [H(f(y)) - H(n(\Delta))].$$

Необходимость включения множителя  $P[y > 0]$  вызвана тем, что случаи, когда  $y < 0$ , в рамках рассматриваемого алгоритма обработки пакетов физически нереализуемы, т. е. отрицательные значения  $y$  на выходе сетевого устройства всегда преобразуются в межпакетный интервал "нулевой" длительности (задержка пакета не может быть отрицательной). На рис. 2 приведены оценки для пропускной способности канала утечки  $C = aI(a, b)$  при размере пакета 1500 байт, при пропускной способности внешней линии связи, соответственно, 20 (кривая 1), 30 (кривая 2) и 40 (кривая 3) Мбит/с.

Входная нагрузка изменяется в диапазоне 0...800 пакетов, обеспечивая коэффициенты загрузки внешнего канала, соответственно, не выше 0,5; 0,3; 0,25.

С учетом приведенной выше модели, при обосновании выбора топологии

VPN-сети и схемы маршрутизации необходимо базироваться на следующих общих принципах.

Агрегация телекоммуникационной нагрузки приводит к более эффективному использованию ресурсов связи СШС и одновременно к повышению степени защищенности сети, однако ухудшает ВВХ. Наиболее рельефно противоречие в требованиях иллюстрируется сравнением характеристик полностью связанной топологии (максимальная плотность числа виртуальных каналов на информационное направление) и топологии "звезда" (минимальное число виртуальных каналов). Для выбора топологии с учетом требований защищенности и эффективности необходимо классифицировать сетевую нагрузку по степени доверия, предсказуемости (синхронный или асинхронный трафик), а также по степени временной прозрачности протокола (критичности к ухудшению ВВХ). С учетом особенностей обработки трафика, влияющих на формирование и пропускную способность канала утечки информации, целесообразно выделять следующие классы (типы) нагрузки:

- доверенная нагрузка, создаваемая абонентскими средствами СШС, которые построены на базе сертифицированного программного обеспечения, функционирующего в режиме замкнутой программной среды;

- недоверенная регулярная нагрузка, например, данные регистрационных систем, систем контроля состояния, видеонаблюдения (без использования компрессии), например передача видеоданных в формате mjpeg;

- недоверенная, не критичная к ухудшению ВВХ нагрузка, например, сервисы передачи данных, электронная почта;

- недоверенная критичная к ухудшению ВВХ нагрузка, например, пакетная телефонная связь с компрессией, передача команд и сигналов оперативно-технического управления.

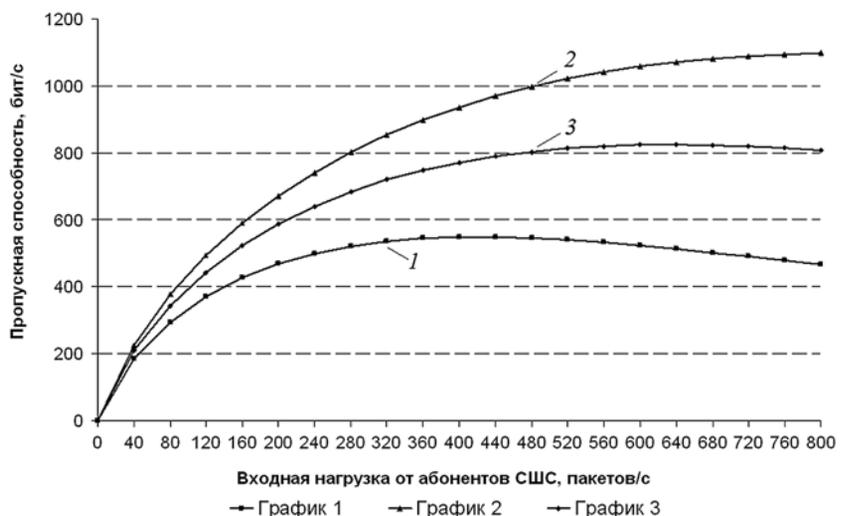


Рис. 2. Пример оценки пропускной способности

Относительно выделенных классов телекоммуникационной нагрузки можно сформулировать общие правила выбора топологии виртуальных каналов и схемы маршрутизации.

*Изоляция сетевого трафика различных источников нагрузки в разных виртуальных каналах, позволяющий снизить пропускную способность канала утечки за счет уменьшения параметра  $a$  (то есть исключения из общего сетевого потока доверенной нагрузки).* Данный метод необходимо применять, если в СШС имеются доверенные источники нагрузки и обеспечивается в диапазоне рабочих нагрузок  $[0..0,3]b$  уменьшение пропускной способности канала утечки, в среднем пропорциональное доверенной нагрузке (выводимой из общего потока, направляемого в виртуальный канал).

*Агрегация источников однородной нагрузки в общем виртуальном канале с одновременным снижением значения параметра  $b$ .* Данный метод, очевидно, можно применять при наличии в сети источников нагрузки, не чувствительных к ухудшению ВВХ. На практике рабочие значения интенсивности  $a$ , оптимальные (с точки зрения нарушителя) для передачи данных по СК (без учета вопросов помехоустойчивости передачи), находятся в диапазоне  $[0,3b..0,95b]$ .

В диапазоне значений интенсивности  $[0,95b..1,0b]$  имеет место резкое падение (до нуля) пропускной способности канала утечки, однако при этом сеть оказывается работающей в режиме перегрузки (возможны потери пакетов).

Механизмы маскировки межпакетных интервалов для постоянной скорости виртуального канала целесообразны для использования на потоках данных, формируемых источниками регулярной нагрузки, которые чувствительны к ухудшению ВВХ.

Механизмы адаптивной маскировки межпакетных интервалов для переменной скорости виртуального канала [3], [4] целесообразны для передачи нерегулярной нагрузки, чувствительной к ухудшению ВВХ.

### 3. Выбор топологии виртуальных каналов

Далее для простоты диапазон рабочих нагрузок  $a_{\min} \dots a_{\max}$  будем называть совместным с пропускной способностью информационного направления  $b_{\max}$ , если пропускная способность канала утечки (при заданном значении  $b_{\max}$ ) не превышает нормированных (задаются в ВВХ СШС или в соответствующей нормативной документации) значений.

1. Выбирается базовая конфигурация виртуальных каналов СШС (исходя из обеспечения требований к ВВХ, отказоустойчивости, живучести, других тактико-технических показателей сети).

2. Выбирается схема маршрутизации телекоммуникационной нагрузки в сети, обеспечивающая привязку источников нагрузки (абонентов) к соот-

ветствующему виртуальному каналу по каждому объекту (узлу) ССС, где это требуется.

3. Исходя из заданной абонентской емкости (числа источников нагрузки) и интенсивности формируемой им (абонентом) нагрузки, по каждому виртуальному каналу  $V$  оценивается общая интенсивность трафика в виде диапазона значений  $a_{\min}^V \dots a_{\max}^V$ .

4. С учетом категории безопасности объекта СШС оценивается совместность  $a_{\min}^V \dots a_{\max}^V$  и  $b_{\max}^V$ .

5. В случае, когда имеет место совместность по всем виртуальным каналам СШС  $\{V\}$ , выбранная конфигурация обеспечивает выполнение требований безопасности. В противном случае для каждого из несовместных виртуальных каналов ( $V_I$ ) выбирается один из следующих вариантов действий (приведены в порядке очередности рассмотрения).

- Оценивается подмножество виртуальных каналов сети  $\{V\}/V_I$ , обеспечивающих доведение (до получателя) телекоммуникационной нагрузки через пограничные сетевые устройства при удалении несовместного виртуального канала ( $V_I$ ) из исходной топологии СШС. При наличии такой возможности (при условии, что выполняются требования по ВВХ в целом) несовместный канал удаляется, вносятся соответствующие изменения в схему маршрутизации.

- Доверенные источники телекоммуникационной нагрузки выделяются в отдельную виртуальную подсеть, изолированную (на сетевом уровне) от недоверенных источников нагрузки (например, путем конфигурации над соответствующими сетевыми интерфейсами  $V_{\text{LAN}}$  или за счет разделения подсетей через пакетный фильтр). Соответственно, исходный (несовместный) виртуальный канал  $V_I$  разделяется на два подканала (выделенная полоса пропускания пропорционально делится между двумя виртуальными подканалами), один из которых ( $V_I^I$ ) обеспечивает передачу доверенного, а другой ( $V_I^U$ ) — недоверенного тра-

фика. Оценивается совместность подканала  $V_I^U$ , при наличии которой в схему маршрутизации вносятся соответствующие изменения. Предполагается, что обслуживание нагрузки в сетевых устройствах осуществляется по алгоритму "взвешенная справедливая очередь". Подканал  $V_I^I$  при этом считается защищенным (доверенным) и исключается из дальнейшего анализа. Описанное правило может применяться за пределами диапазона значений эффективной передачи  $(0,35..0,95)b$ , где влияние параметров  $a$  и  $b$  (если не рассматривать ситуацию, когда сетевое устройство работает в режиме перегрузки) на пропускную способность канала утечки не является

одинаковым, что можно увидеть из анализа графиков на рис. 2. Снижение  $a$  (в отличие от  $b$ ) практически приводит к пропорциональному снижению пропускной способности канала утечки.

- Если виртуальный канал  $V_I^u$  вновь оказался несовместным, оценивается возможность уменьшения значения параметра  $b_{\max}$  до минимального значения, при котором данный виртуальный канал обеспечивает совместность без выхода ВВХ в область недопустимых значений. При наличии такой возможности корректируется параметр  $b$  до соответствующего значения.
- При наличии на выбранном направлении регулярной нагрузки (синхронного трафика) виртуальный канал  $V_I^u$  разделяется на два подканала ( $V_{IS}^u, V_{IA}^u$ ). Источники синхронной нагрузки изолируются в отдельных сегментах сети. Передача синхронной нагрузки осуществляется в канал  $V_{IS}^u$ , на котором настраивается "трафик шейпер" с соответствующим параметром  $b$  (определяется периодичностью передачи данных). Нагрузка по указанному подканалу, а также выделенная для канала полоса пропускания не учитываются в дальнейшем анализе пропускной способности канала утечки для подканала  $V_{IA}^u$ .
- Для передачи остальной (асинхронной) нагрузки выделяется подканал  $V_{IA}^u$ . Оценивается совместность виртуального данного подканала. При соответствии нормам защиты подканал  $V_{IA}^u$  включается в активную топологию виртуальных каналов СШС.
- При несовместности подканала  $V_{IA}^u$  с нормами защиты на указанном подканале включается функция адаптивной маскировки межпакетных интервалов с параметрами конфигурации, соответствующими категории безопасности СШС.

*Примечание:* "трафик шейпер" в описанной выше схеме обеспечивает нормализацию трафика, т. е. устранение пульсаций, что достигается за счет предварительной буферизации пакетов с последующим "опорожнением" буфера с заданной (для данного сетевого интерфейса) скоростью.

### Заключение

На текущий момент в России требования по защите информации в СШС от утечки по рассмат-

риваемому скрытому каналу в общем виде (без задания числовых показателей и без привязки к конкретному типу объекта информационных технологий) определены, например, в ГОСТ Р 53113.1—2008, ГОСТ Р 53113.2—2009. На практике требования к числовым значениям показателей защищенности категорированных СШС задаются заказчиком СШС в явном виде с учетом положений ряда НМД ФСБ России и результатов проведения специального инженерного анализа таких изделий.

Учитывая, что информационная безопасность СШС обеспечивается на уровне наиболее "слабого" звена в системе защитных мер, для обоснования показателей защищенности СШС от утечки информации по рассматриваемому скрытому каналу методически правильно воспользоваться его аналогией с техническими каналами утечки информации.

В рамках приятия такой аналогии целесообразно обеспечить привязку пропускной способности канала утечки ( $C$ ) к соответствующим нормам защиты от технических каналов (отношение сигнал/шум на границе контролируемой зоны для объектов информатизации разных категорий защиты). Для корректного вычисления параметра  $C$  при этом необходимо выбрать схему модуляции, что является определенной проблемой, поскольку такая схема должна при прочих равных условиях обеспечивать максимальную пропускную способность технического канала [5].

Поскольку передача трафика с кодированными межпакетными интервалами осуществляется одновременно с трафиком, формируемым другими источниками телекоммуникационной нагрузки (рассматриваемым как помеха), имеет место также и аналогия способа учета структурных помех, вызванных электромагнитным фоном объекта [6] при проведении специальных исследований объектов информатизации.

### Список литературы

1. **Тарасюк М. В.** Демаскирующие признаки трафика категорированных сетей в задачах технической разведки // Информатика и космос. 2010. № 3.
2. **Руководящий** документ Гостехкомиссии при президенте РФ "Концепция защиты информации от НСД в СВТ и АС", утвержден решением Гостехкомиссии при президенте РФ от 30 марта 1992 г.
3. **Тарасюк М. В., Исаченко Ю. С.** Подавление модуляции длин дейтаграмм в сетях шифрованной связи // Информационные технологии. 2010. № 1. С. 12—16.
4. **Исаченко Ю. С., Тарасюк М. В.** Асинхронный метод маскировки междейтаграммных интервалов в VPN-сети // Защита информации. Инсайд. 2010. № 4.
5. **Тарасов И. В., Тарасюк М. В.** Выбор норм эффективности защиты от утечки информации по скрытым каналам модуляции трафика в средствах сетевого шифрования // Информационные технологии. 2006. № 8. С. 16—19.

Г. Ч. Набибекова, зав. отд., e-mail: gulnarara58@mail.ru,  
Институт Информационных Технологий НАН Азербайджана, Баку, Азербайджан

## Об одном методе фаззификации атрибутов хранилища данных в системах поддержки принятия решений в сфере внешней политики

*Представлен процесс фаззификации атрибутов хранилища данных системы поддержки принятия решений в сфере внешней политики. С этой целью выбранный алгоритм CLARANS был применен для кластеризации данных хранилища. С помощью медоидов, полученных в результате кластеризации, были выведены значения функции принадлежности. Рассматривается случай, когда число терм-множеств, создаваемых на различных измерениях куба, различно.*

**Ключевые слова:** OLAP-куб, фаззификация, нечеткое множество, лингвистическая переменная, терм-множество, кластеризация, хранилище данных, атрибут, медоид

G. Ch. Habibakova

## A Fuzzification Method of Attributes of Data Warehouse of Decision Support System in the Foreign Policy

*The paper presents a fuzzification process of attributes of data warehouse of decision support system in the foreign policy. To this end, the chosen algorithm CLARANS was applied for clustering of data warehouse. The values of membership function were derived with medoids, which were obtained after clustering. The paper deals with the case, where the number of term-sets generated on different dimensions are various.*

**Keywords:** OLAP-cube, fuzzification, fuzzy-set, linguistic variable, term-set, clustering, data warehouse, attribute, medoid

### Введение

В последнее время руководители учреждений стали интересоваться хранилищем данных (ХД) как важнейшим инструментом управления, входящим в число наиболее значимых элементов инфраструктуры предприятия [1].

Являясь основным элементом систем поддержки принятия решений (СППР), ХД представляет собой единый и достаточно большой репозиторий данных, используемых для принятия решений. Согласно Б. Инмону [2], ХД — это "предметно-ориентированное, привязанное ко времени и неизменяемое собрание данных для поддержки процесса принятия управленческих решений". ХД строят путем консолидации данных из множества гетерогенных источников данных, которые преобразованы в единый формат и содержат данные как в детализированном, так и в агрегированном виде. Агрегирование данных многомерного ХД проводят по атрибутам (измерениям) для облегчения принятия

решений. Детализированные и агрегированные данные хранятся в многомерном OLAP-кубе, что обеспечивает быстрый ответ пользователям, осуществляющим запрос в ХД.

Инструменты OLAP, который является ключевым элементом ХД, обеспечивают различные операции, такие как свертка, развертка, сечение, срез, ранжирование, сводные таблицы. Обычно эти операции проводят над точными данными, и в результате их действия пользователи получают ответ в числовом выражении. Однако в процессе принятия решений можно столкнуться с множеством нечетких задач, вследствие чего в запросах к базам данных (БД), которые пытается формулировать исследователь, часто могут присутствовать неточности и неопределенности. Кроме того, для лиц, принимающих решения, важен и полезен бывает не сам результат запроса, а то, является ли этот результат *хорошим*, *средним* или *плохим*, т. е. им важна качественная сторона результата [3, 4]. Все это требует, чтобы СППР, использующая нечеткие и неопределенные

рассуждения, для оказания помощи руководителям в принятии решений поддерживала соответствующую технологию представления знаний [4].

### 1. Поликубическая OLAP-модель с нечетким OLAP-кубом

Эта проблема является актуальной и при решении аналогичных задач в сфере внешней политики (ВП).

Отметим, что в основе ХД, спроектированного для сферы ВП, лежит информация о зарубежных поездках сотрудников государственного органа (ГО) [5].

Для более эффективного принятия управленческих решений аналитикам необходимо иметь в своем распоряжении целостную картину развития ГО по различным направлениям. Поэтому разрабатываемая СППР в сфере ВП носит многоцелевой характер, т. е. с ее помощью возможно решение задачи управления в различных аспектах — управление финансовыми, кадровыми, техническими ресурсами и т. д. С этой целью из ХД выделяются витрины данных (ВД), ориентированные на решение подзадач, сопутствующих основной задаче. Вследствие этого, кроме гиперкуба (или вместо гиперкуба), в котором все показатели определяются одним и тем же набором измерений, по каждой подзадаче в соответствии с ее запросами строится свой OLAP-куб. Таким образом, на основании одного ХД можно построить несколько OLAP-кубов. В первом случае OLAP-модель называется гиперкубической, а во втором случае — поликубической [6].

На рис. 1 представлена поликубическая OLAP-модель, в которой из ХД выделены три ВД.

В качестве примера создания нечеткого OLAP-куба рассмотрим подзадачу управления кадровыми ресурсами, включающую такие вопросы, как анализ кадровых ресурсов, их размещение, обучение, планирование продвижения сотрудников и т. д. При решении такой подзадачи для лиц, принимающих решение, при формировании запроса к БД часто бывает важен не конкретно возраст сотрудника, а то, является ли он *молодым*, *средним* или *пожилым*, либо, например, не конкретно стаж работы, а является он *малым*, *средним* или *продолжительным*. Кроме того, его может интересовать не точная дата мероприятия, в котором сотрудник принимал участие, а то, когда произошло это событие — *давно* или *недавно*. В этом случае ХД должно обеспечивать ответы на запросы, использующие лингвистические термины, которые являются терм-множествами соответствующей лингвистической переменной. Здесь следует отметить, что главное преимущество ХД перед остальными типами источников данных — это наличие семантического слоя, который дает возможность пользователю оперировать терминами предметной об-

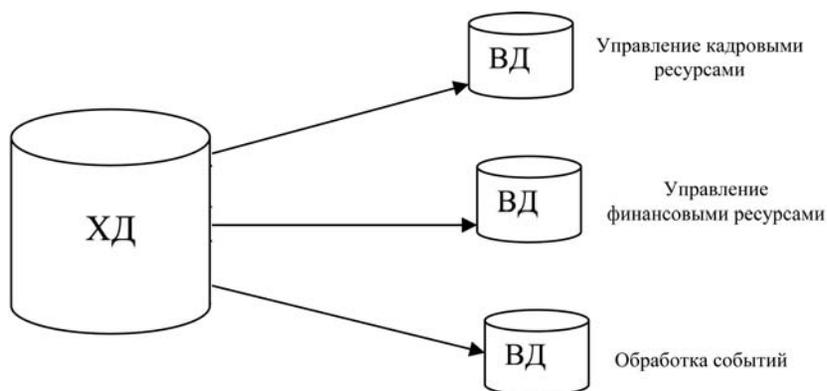


Рис. 1. Поликубическая OLAP-модель

ласти для формирования аналитических запросов к хранилищу [7].

В работе поставлена цель фаззифицировать измерения OLAP-куба в целях разработки нечеткого OLAP-куба в поликубической модели OLAP на основе информации ХД в СППР в сфере ВП.

Для достижения поставленной цели были решены следующие задачи.

- Исследованы алгоритмы кластеризации PAM (Partitioning Around Medoids), CLARA (Clustering Large Applications) и CLARANS (Clustering Large Applications based upon RANdomized Search), сделан их сравнительный анализ, на основании которого в качестве используемого был выбран алгоритм CLARANS.
- В ХД был применен алгоритм кластеризации CLARANS с указанием требуемого числа кластеров.
- В целях фаззификации атрибутов AGE (возраст), EXPERIENCE (стаж работы), DATE (дата) и получения нечеткого OLAP-куба для них с помощью медоидов (медоид — центр кластера, являющийся одной из его точек) полученных кластеров были выведены значения функции принадлежности.

### II. Фаззификация измерения OLAP-куба

Для решения данной проблемы обратимся к получившему в настоящее время широкое распространение аппарату теории нечетких множеств в задачах поиска информации [8].

Нечеткое множество, включая функцию принадлежности, являющуюся его характеристикой, может быть задано экспертом. Но в некоторых случаях функцию принадлежности удобнее задать аналитической формулой и для наглядности изобразить графически. Для задания функций принадлежности существуют различные типовые формы кривых, среди которых для решения нашей задачи выберем треугольную и трапециевидальную кусочно-линейные формы. В общем случае треугольная функция принадлежности определяется тремя числами, а трапециевидальная — четырьмя, хотя их число может и

меняться в зависимости от условий задачи. Для определения чисел, с помощью которых будет задана функция принадлежности, будем основываться на методе, описанном в работе [3], адаптируя его к нашей задаче, в которой по условию число термножеств на различных измерениях различно.

**А. Выбор алгоритма кластеризации.** Как известно, основная цель кластеризации заключается в разделении исследуемого множества объектов на группы "похожих" объектов, называемых кластерами. Классификация алгоритмов кластеризации дана в работе [9]. Две основные категории — это иерархические алгоритмы и неиерархические алгоритмы. В целях выбора эффективного алгоритма были рассмотрены неиерархические алгоритмы, основанные на методе разбиения, а именно на методе *k-medoids*, так как медоид — это центр кластера, который принадлежит кластеру, что является для нас главным условием. Кроме того, методы *k-medoids* являются устойчивыми к наличию выбросов и могут работать достаточно эффективно с большими наборами данных.

Алгоритмы, основанные на методе разбиений, проходят два основных этапа [10]:

- начальный шаг, на котором  $k$  объектов выбраны в качестве медоидов;
- оценочный шаг, на котором происходит попытка минимизировать целевую функцию, обычно основанную на сумме общего расстояния между невыбранными объектами (значениями) и их медоидами, т. е.

$$D(r, s) = \sum_{j=1}^n d(r_j, s_j),$$

где  $s_j \in S$  ( $S$  — множество объектов (значений) для кластеризации) и  $d(r_i, s_j) < d(r_c, s_j), \forall r_i, r_c \in R$  ( $R$  — множество объектов (значений) из  $S$ , выбранных в качестве медоидов),  $r_i \neq r_c$ .

Чем меньше сумма расстояний между медоидом и всеми другими объектами соответствующего ему кластера, тем лучше кластеризация.

Были рассмотрены три известных алгоритма, основанных на методе *k-medoids*: PAM, CLARA и CLARANS.

PAM является одним из первых алгоритмов, основанных на методе *k-medoids*. В работе [11] представлен основной принцип процесса кластеризации PAM, заключающийся в переборе всех объектов, которые в данный момент не являются медоидами, для вычисления расстояния от них до выбранных медоидов. Алгоритм PAM приводит к высокому качеству кластеров, но поскольку он пробует все возможные комбинации, он эффективен для небольших наборов данных. И в связи с его вычислительной сложностью его применение непрактично для кластеризации больших наборов данных.

Вычислительная сложность алгоритма PAM явилась мотивацией для разработки алгоритма CLARA — алгоритма кластеризации на основе выборки. Его принцип работы также можно найти в работе [11]. CLARA выделяет из множества данных несколько выборок, применяет PAM на каждой выборке и находит ее медоиды. Поскольку выборка проведена случайным образом, то медоиды выборки можно считать медоидами всего набора данных. Чтобы выбрать лучшие приближения, CLARA создает несколько выборок и на выходе выдает лучшую кластеризацию. В работе [11] также экспериментально показано, что пять выборок размером  $(40 + 2k)$  дают удовлетворительные результаты.

CLARANS был разработан в рамках пространственного анализа данных. При поиске лучшего медоида на шаге оценки CLARANS случайным образом выбирает объекты из  $(n - k)$  объектов ( $n$  — число объектов в множестве;  $k$  — заданное число кластеров, или число медоидов). Число объектов, перебираемых на этом этапе, ограничено параметром *maxneighbor*, заданным пользователем. Если после *maxneighbor* попыток не будет найдено лучшее решение, то локальный оптимум считается достигнутым. Процедура продолжается до тех пор, пока не найдены *numloc* локальных оптимумов [12]. Алгоритм CLARANS, использующий стратегию случайного поиска, улучшает алгоритмы PAM и CLARA с точки зрения эффективности (сложность вычислений или времени) и эффекта кластеризации (среднее искажение на расстояниях) соответственно.

**В. Использование медоидов для вывода функции принадлежности.** Понятия "математическая теория нечетких множеств" (*fuzzy sets*) и "нечеткая логика" (*fuzzy logic*) впервые были предложены в работе [13]. Основной причиной появления новой теории стало наличие нечетких и приближенных рассуждений при описании человеком процессов, систем, объектов. В работах [14, 15] также дана информация о теории нечетких множеств, описание нечетких операторов, различные комбинации которых используются в операциях на нечетком OLAP-кубе.

Напомним основные определения. Характеристикой нечеткого множества является функция принадлежности (*membership function*). Обозначим через  $\mu(x)$  степень принадлежности элемента  $x$  к нечеткому множеству, представляющую собой обобщение понятия характеристической функции обычного множества. Тогда нечетким множеством  $S$  называется множество упорядоченных пар вида  $S = \{\mu(x)/x\}$ , при этом  $\mu(x)$  может принимать любые значения в интервале  $[0, 1]$ ,  $x \in X$ . При этом значение  $\mu(x) = 0$  означает отсутствие принадлежности к множеству  $X$ ;  $\mu(x) = 1$  — полную принадлежность к множеству  $X$ .

Под средой поиска информации будем понимать пару запрос—база данных. С точки зрения четко-

**Возможные варианты нечеткости в среде поиска информации**

№ варианта	Запрос	База данных
1	Четкий	Четкая
2	Четкий	Нечеткая
3	Нечеткий	Четкая
4	Нечеткий	Нечеткая

сти—нечеткости среды поиска информации могут быть четыре варианта (см. таблицу) [8].

Как видно из таблицы, наиболее общим является вариант 4.

Рассмотрим ХД СППР сферы внешней политики, в котором выделена ВД по подзадаче управления кадровыми ресурсами.

Пусть в целях создания нечеткого OLAP-куба на основании этой ВД из всего множества атрибутов надо фаззифицировать три атрибута — AGE (возраст), EXPERIENCE (стаж работы) и DATE (дата). При этом другие атрибуты, например LOCATION (место проведения мероприятия), могут оставаться четкими.

*Первый шаг.* Применим алгоритм CLARANS в БД, имеющей  $n$  атрибутов, задав число кластеров  $k = 3$ . В результате будут получены три кластера с центрами (медоидами) в точках  $m_1, m_2, m_3$ .

*Второй шаг.* Определим числа, на основании которых будет задана функция принадлежности для двух атрибутов — AGE (возраст) и EXPERIENCE (стаж работы). С этой целью каждый медоид представим как вектор  $m_i = \{a_{ij}, \dots, a_{ij}\}$ , где  $i = \overline{1, 3}$ ,  $j = \overline{1, 2}$ .

Для нашего случая имеем:

$$m_1 = \{a_{11}, a_{12}\}; m_2 = \{a_{21}, a_{22}\}; m_3 = \{a_{31}, a_{32}\}.$$

Множество  $\{a_{1j}, a_{2j}, a_{3j}\}$ ,  $j = \overline{1, 2}$ , представляет значения трех медоидов  $j$ -го атрибута, который предстоит фаззифицировать.

*Третий шаг.* На этом шаге будет проводиться фаззификация атрибута. В результате будут получены значения принадлежности всех значений  $j$ -го атрибута в каждом нечетком множестве.

Пусть  $J$  — множество значений  $j$ -го атрибута;  $F_{ij}(x)$  — функция принадлежности для получения значений принадлежности всех значений  $j$ -го атрибута в  $i$ -м нечетком множестве.

Для всех  $x \in J$  выполняем:

1) для первого нечеткого множества

(т. е. при  $i = 1$ )

$$F_{1j}(x) = 1.0 \quad \text{if } x \leq a_{1j}.$$

$$F_{1j}(x) = (x - a_{2j}) / (a_{1j} - a_{2j}) \quad \text{if } a_{1j} < x < a_{2j}.$$

$$F_{1j}(x) = 0 \quad \text{if } x \geq a_{2j}.$$

2) для второго нечеткого множества

(т. е. при  $i = 2$ ):

$$F_{2j}(x) = 0 \quad \text{if } x \leq a_{1j}.$$

$$F_{2j}(x) = (x - a_{1j}) / (a_{2j} - a_{1j}) \quad \text{if } a_{1j} < x < a_{2j}.$$

$$F_{2j}(x) = 1.0 \quad \text{if } x = a_{2j}.$$

$$F_{2j}(x) = (x - a_{3j}) / (a_{2j} - a_{3j}) \quad \text{if } a_{2j} < x < a_{3j}.$$

$$F_{2j}(x) = 0 \quad \text{if } x \geq a_{3j}.$$

3) для третьего нечеткого множества

(т. е. при  $i = 3$ ):

$$F_{3j}(x) = 0 \quad \text{if } x \leq a_{2j}.$$

$$F_{3j}(x) = (x - a_{2j}) / (a_{3j} - a_{2j}) \quad \text{if } a_{2j} < x < a_{3j}.$$

$$F_{3j}(x) = 1 \quad \text{if } x \geq a_{3j}.$$

Отметим, что после фаззификации полученные три кластера станут тремя терм-множествами. Для атрибута EXPERIENCE это {малый, средний, продолжительный}, а для атрибута AGE это {молодой, средний, пожилой}. При этом атрибуты EXPERIENCE и AGE становятся лингвистическими переменными.

На рис. 2 и рис. 3 дано графическое изображение терм-множеств соответствующих атрибутов. Отметим, что номер  $j$  следует закреплять за соответствующим измерением. В данном случае  $j = 1$  — измерение AGE,  $j = 2$  — измерение EXPERIENCE.

Атрибут DATE будет разбит на два терм-множества (недавно и давно), поэтому для его фаззификации на первом шаге применим в БД алгоритм CLARANS, задав число кластеров  $k = 2$ . В результате будут получены два кластера с центрами (медоидами) в точках  $m_1, m_2$ . На втором шаге определяются числа, на основании которых будет задана функция принадлежности для этого атрибута. В этом случае имеем  $m_1 = \{a_1\}$ ,  $m_2 = \{a_2\}$ , т. е. две точки на

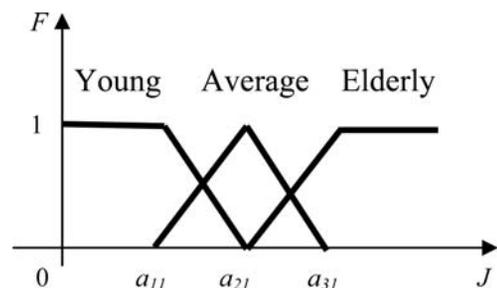


Рис. 2. Графическое изображение лингвистической переменной AGE

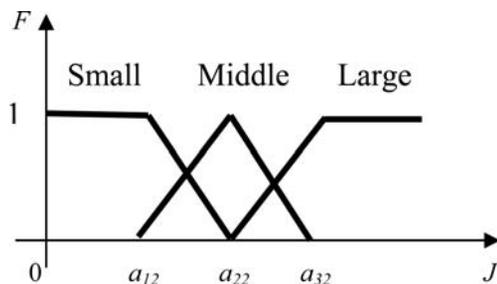


Рис. 3. Графическое изображение лингвистической переменной EXPERIENCE

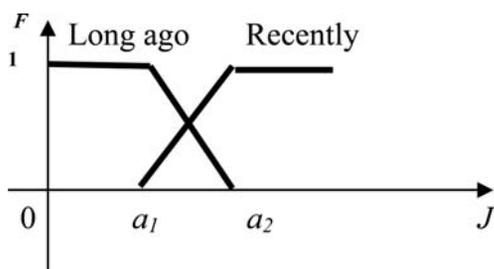


Рис. 4. Графическое изображение лингвистической переменной DATE

одном измерении для задания функции принадлежности.

На третьем шаге проводится соответствующая фаззификация, в результате которой будут получены два терм-множества. На рис. 4 показано графическое изображение лингвистической переменной DATE.

В результате фаззификации измерений куба рассматриваемой СППР можно выполнять нечеткие запросы, используя термины *молодой*, *зрелый*, *пожилой* (касательно возраста), *малый*, *средний*, *продолжительный* (касательно стажа работы), *недавно* и *давно* (касательно даты мероприятия) в сочетании с четкими значениями базы данных, таких как место проведения мероприятия, название мероприятия и др.

Приведем примеры типовых запросов, которые могут сыграть не последнюю роль в управлении кадровыми ресурсами.

1. Сколько *молодых* сотрудников посетили недавно Турцию?

2. Выдать список сотрудников, имеющих *продолжительный* стаж работы, которые посетили *недавно* Турцию в связи с мероприятиями в медицинской сфере.

3. В каких странах *молодые* сотрудники прошли *недавно* тренинги в военной сфере?

4. Выдать список *молодых* и *зрелых* сотрудников, прошедших тренинги в США в сфере ИКТ и т. д.

### III. Практическое применение

Следует отметить, что включение разрабатываемой СППР в программу электронного правительства способно обеспечить информационную поддержку проводимой в стране внешней политики. В настоящее время эту систему, включая описанный выше подход разработки нечеткого OLAP-куба, планируется реализовать в Управлении международных связей (УМС) Национальной академии наук Азербайджана (НАНА) на основе Распоряжения Президиума НАНА об информатизации Центрального Аппарата Президиума НАНА, куда как структурное подразделение включено и УМС. Отметим, что на данный момент вся документация, связанная с зарубежными поездками сотрудников НАНА, хра-

нится в бумажном или электронном виде в архивах различных отделов — отделе кадров, бухгалтерии, научно-организационном отделе и т. д., т. е. хранение этой информации носит неорганизованный, бессистемный характер. В целях внедрения разрабатываемой системы будет использован минимум 2,2 ГГц на платформе Microsoft Windows 7. Для анализа OLAP будет использована Microsoft Analysis Server, 2008. Алгоритм CLARANS и задание функций принадлежности, а также Front-end (интерфейс пользователя) планируется реализовать в Object Pascal (в среде Delphi). В качестве базы данных будет использован SQL-сервер, язык запросов — T-SQL. Набор данных будет содержать информацию за последние 12 лет, что приблизительно составляет 16 000—20 000 записей. При таких условиях время отклика на запрос пользователя к данной системе ожидается очень коротким — не более 2 с. Измерениями могут быть *Data*, *Event* и *Persons*, таблицей фактов — *costs* (финансовые затраты на мероприятие). Измерение *Event* состоит из атрибутов: ID; название мероприятия; дата проведения мероприятия; сфера, к которой оно относится; город; страна; численность населения; текущая информация о стране; документы, содержащие международные (в которые включена данная страна) или двусторонние контракты, договоры, соглашения и т. п. или информация об этих документах (т. е. их название, дата заключения, срок действия и т. д.) и др. Измерение *Data* в качестве атрибутов содержит: ID; день (может отсутствовать); название месяца и год поездки (не путать с датой проведения мероприятия). Измерение *Persons* в качестве атрибутов содержит подробную информацию о командированном сотруднике: ID; имя, фамилию, год рождения, место работы, должность, стаж работы и т. д.

### Заключение

Обычное ХД, объединяющее данные из различных источников, обеспечивает лицам, принимающим решения, возможность делать анализ точных данных. Однако в процессе принятия решений можно столкнуться с множеством нечетких задач, вследствие чего в запросах к базам данных, которые пытается формулировать исследователь, часто могут присутствовать неточности и неопределенности. В этих случаях необходимо, чтобы СППР для оказания помощи руководителям в принятии решений поддерживала соответствующую технологию представления знаний. В статье показан метод фаззифицирования измерений OLAP-куба в целях разработки нечеткого OLAP-куба в поликубической OLAP-модели на основе информации ХД в СППР в сфере внешней политики. В статье рассматривается случай, когда число создаваемых на измерениях куба терм-множеств разное.

## Список литературы

1. **Арсентьев А.** Хранилища данных становятся инфраструктурным компонентом № 1 // CNews аналитика. 2010. [Электронный ресурс]. URL: <http://retail.cnews.ru/reviews/free/BI2010/articles/articles6.shtml>
2. **Immon W.** Building the Data Warehouse, Second eds. New York: John Wiley & sons Inc., 1996.
3. **Kumar K. N. V. N. P., Radha K. P., De S. K.** Fuzzy OLAP Cube for Qualitative Analysis. Intelligent Sensing and Information Processing, 2005 // Proc. of 2005 International Conference on. 2005. P. 290–295. URL: [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=1529464&tag=1](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1529464&tag=1)
4. **Wang S.** Application of Decision Support System in E-government // Proc. of International Conference "Management and Service Science", 2009. MASS'09. URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5302532>
5. **Nabibakova G.** Development of Data Warehouse Architecture in the Informational Analytical Systems of Decision-Making Support in the Field of Foreign Policy // International Journal of Ubiquitous Computing and Internationalization. December 2011. Vol. 3, N 2. P. 15–18.
6. **Каширин И. Ю., Семченков С. Ю.** Интерактивная аналитическая обработка данных в современных OLAP-системах // Бизнес-информатика. 2009. № 2(08).
7. **Паклин Н. Б., Орешков В. И.** Бизнес-аналитика: от данных к знаниям. СПб.: ПИТЕР, 2010.
8. **Рыжов А. П.** Модели поиска информации в нечеткой среде. М.: ЦПИ МГУ, 2004.
9. **Чубукова И. А.** Data Mining: учеб. пособие. М.: БИНОМ. Лаборатория знаний, 2006. 382 с.
10. **Varioni M. C. N., Razente H. L., Traina A. J. M., Jr C. T.** Art efficient approach to scale up k-medoid based algorithms in large databases. XXI Brazilian Symposium on Databases. 2006. <http://www.lbd.dcc.ufrmg.br:8080/colecoes/sbbd/2006/018.pdf>
11. **Kaufman L., Rousseeuw P. J.** Finding Groups in Data: an Introduction to Cluster Analysis. John Wiley & Sons, 2005.
12. **Ng R. T., Han J.** Clarans: A method for clustering objects for spatial data mining // IEEE Transactions on Knowledge and Data Engineering (TKDE). 2002. 14(5). P. 1003–1016.
13. **Zadeh L.** Fuzzy sets // Information and Control. 1965. N 8. P. 338–353.
14. **Klir G. J., Yuan B.** Fuzzy sets and fuzzy logic: Foundations and applications. New Jersey: Prentice Hall, 1995. 256 p. URL: <http://www.bookadda.com/books/fuzzy-set-theory-foundations-applications-g-j-0133410587-9780133410587>
15. **Рыжов А. П.** Элементы теории нечетких множеств и измерения нечеткости. М.: Диалог-МГУ, 1998.

УДК 004.652.5

**Н. Т. Фисун**, д-р техн. наук, проф., зав. каф., e-mail: [ntfis@kma.mk.ua](mailto:ntfis@kma.mk.ua)

**Г. В. Горбань**, аспирант,

Черноморский государственный университет им. П. Могилы, г. Николаев, Украина

## Модели и методы построения системы OLAP для объектно-ориентированных баз данных

*Рассмотрены аспекты построения системы OLAP в объектно-ориентированных СУБД, предложена схема связи фактов с измерениями в объектно-ориентированной модели данных и диаграмма метаклассов для реализации OLAP при использовании метаданных.*

**Ключевые слова:** базы данных, OLAP, реляционные БД, объектно-ориентированные БД, таблица фактов, измерение, мера, многомерный массив, класс, экземпляр класса, свойство, ссылка, связь, метаданные, нотация IDEF1

**N. T. Fisun, G. V. Gorbanj**

## Models and Methods for Building OLAP System for Object-Oriented Databases

*Aspects of building OLAP systems in an object-oriented DBMS are discussed in the article. A scheme of relationship of facts with dimensions in the object-oriented data model and metaclass diagram for OLAP implementation using metadata are proposed.*

**Keywords:** database OLAP, relational database, object-oriented database, fact table, dimension, measure, multidimensional array, class, instance of the class, property, reference, relationship, metadata, notation IDEF1

### Введение

В настоящее время в системах поддержки принятия решений (СППР) важное место занимают технологии оперативного (OLAP) и интеллектуаль-

ного (Data Mining) анализа данных [1, 2]. Данные технологии уже реализованы в некоторых коммерческих системах управления базами данных (СУБД), таких как Microsoft SQL Server, Oracle и других, которые представляют собой реляционные СУБД.

Реляционные базы данных (БД) имели успех и на данный момент являются самыми распространенными. Это обосновано тем, что данные в них сохраняются в табличном формате и доступны для каждого, кто знаком со структурированным языком запросов (SQL). Однако со временем информационные системы предъявляют такие требования к обработке данных, с которыми реляционные БД уже неспособны справиться. Примером этого может послужить хранение изображений, звуковых файлов и видеозаписей. Существенным недостатком реляционных БД является то, что хранение реальных структур данных в виде столбцов и строк таблиц является довольно непростой задачей, поскольку при декомпозиции данных может получиться довольно большое количество таблиц, отношения между которыми сложно использовать и запомнить.

Одним из альтернативных к реляционному подходу хранения данных является объектный подход. По сравнению с реляционным подходом данный подход появился несколько позже, первые публикации, касающиеся объектно-ориентированных БД, датируются примерно серединой 1980-х годов. Объектный подход позволяет наиболее естественным способом записать объекты в БД и обеспечить их хранение и действия с ними.

### Некоторые аспекты построения системы OLAP в объектно-ориентированных базах данных

Основой любой системы OLAP является куб, представляющий собой упорядоченный многомерный массив и строящийся на фактических данных, которые могут находиться в нескольких таблицах фактов и обычно имеют несколько измерений [2, 3].

Основными реализациями систем OLAP являются MOLAP (многомерный OLAP), ROLAP (реляционный OLAP) и HOLAP (гибридный OLAP), использующий как многомерные, так и реляционные БД. В MOLAP таблица фактов содержит все комбинации значений всех измерений и соответствующие им значения мер. В отличие от MOLAP таблица фактов в ROLAP не содержит все возможные варианты комбинаций измерений, она содержит уникальный составной ключ, который объединяет первичные ключи таблиц измерений.

Типичными схемами связи таблицы фактов с таблицами измерений являются схемы "звезда" и "снежинка". В схеме "звезда" таблица фактов является денормализованной, поскольку данная схема не позволяет учесть возможные уровни иерархии измерений. Для этого применяется схема "снежинка", главная особенность которой заключается в хранении информации об одном измерении в несколь-

ких связанных таблицах. Однако главный недостаток такой схемы — это усложнение в структуре базы данных, которая в данном случае будет содержать достаточно много таблиц фактов.

При сравнении реляционной модели данных с объектной моделью между ними можно найти достаточно много аналогий. Так, реляционное понятие таблицы соответствует классу, строка таблицы — экземпляру класса, столбец — свойству класса, внешний ключ — ссылке на объект и т. д. [4]. Таким образом, можно сделать вывод, что таблица фактов в объектной модели данных представляет собой класс, свойствами которого являются меры и ссылки на таблицы измерений, последние, в свою очередь, также являются классами.

Структуру таблицы фактов и таблиц измерений для объектной модели данных можно показать на примере базы данных некоторой торговой сети, имеющей магазины в различных регионах страны, в которые поставляют товары различных категорий.

Факты про продажу определенного товара в определенном магазине за определенный месяц фиксируются в таблице фактов, имеющей название "Продажи". Соответственно, в данной системе будет три измерения: "Дата", "Магазины" и "Товары". Мерой в таблице фактов является доход за продажу. При этом каждое из измерений имеет несколько уровней иерархии. Таким образом, нижним уровнем иерархии "Дата" является "Месяц", выше находятся уровни "Квартал" и "Год". В свою очередь, для иерархий "Магазины" и "Товары" нижними уровнями являются соответственно "Магазин" и "Товар", а верхними — "Регион" (каждый магазин находится в определенном регионе) и "Категория товара" (каждый товар можно отнести к определенной категории). На рис. 1 показана схема вышеуказанной базы данных в нотации IDEF1 [5].

В проекте стандарта объектных БД определен язык ODL (*Object Definition Language* — язык определения объектов), синтаксис которого аналогичен синтаксису C++ и JAVA, однако точно с ними не совпадает [6].

Описание класса "Продажи" на языке ODL будет выглядеть следующим образом:

```
class FactSales
{
    attribute Month factmonth; // ссылка
    на объект класса "Месяц"
    attribute Shop factshop; // ссылка на
    объект класса "Магазин"
    attribute Goods factgoods; // ссылка
    на объект класса "Товар"
    attribute float profit; // доход (ме-
    ры таблицы фактов)
}
```

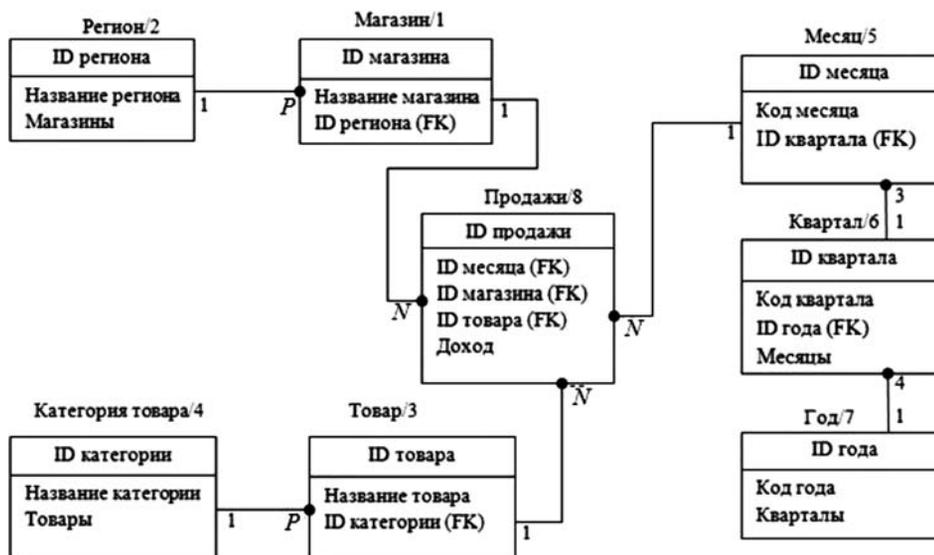


Рис. 1. Схема базы данных торговой сети

Таким образом, в таблице фактов содержатся ссылки на экземпляры классов измерений нижнего уровня иерархии. В свою очередь, классы измерений в своем описании содержат атрибуты связи с экземплярами классов более высоких уровней иерархии. Например, для класса "Месяц":

```

Class Month
  attribute string CodeMonth; //
  код месяца в базе данных (например,
  "01-2013" — январь 2013 года)
  relationship Quarter quarter
    inverse Quarter::month; // описа-
    ние связи с классом "Квартал" (ме-
    сяц всегда принадлежит к определен-
    ному кварталу, а свою очередь квар-
    тал имеет 3 месяца)
}
    
```

В классе "Квартал" также имеется указанная связь с классом "Месяц", а также связь с классом "Год":

```

Class Quarter
{
  attribute string CodeQuarter; // код
  месяца в базе данных (например, "1
  кв.-2013" — 1 квартал 2013 года)
  relationship ser<Month> months
    inverse Month::quarter; // описание
    связи с классом "Месяц", в данном
    случае атрибут представляет множе-
    ство экземпляров класса "Месяц", по-
    скольку в квартале более чем 1 мес
  relationship Year year
    inverse Year::quarters // описание
    связи с классом "Год", в котором так-
    же определена соответственная связь
}
    
```

```

Class Year
{
  attribute integer CodeYear;
  relationship set<Quarter> quarters
    inverse Quarter::year; // соответ-
    ствующая связь с классом "Квартал"
}
    
```

Соответствующие связи между собой имеют классы и других измерений. В отличие от таблицы фактов, которая содержит ссылки на экземпляры классов измерений, классы измерений содержат атрибуты связи с классами измерений более высокого уровня для того, чтобы не нарушить целостность базы данных.

При агрегировании данных на более высоком уровне к измерениям можно обращаться с помощью точечного синтаксиса, применяемого в объектной модели данных, например:

Month. Quarter. CodeQuarter — код кварта-  
ла, в который входит месяц;

Month. Quarter. Year. CodeYear — год, в ко-  
торый входит месяц (доступ через объект класса  
"Квартал").

С помощью точечного синтаксиса можно также  
обратиться к значениям более высокого уровня ие-  
рархии более низкому, например:

Year. Quarter [n]. CodeQuarter — код *n*-го  
квартала года;

Year. Quarter [n]. Month [m]. CodeMonth —  
код *m*-го месяца *n*-го квартала года.

Таким образом, становится возможным агреги-  
рование по измерениям с уровнями иерархий  
большого количества. При этом нет избыточности  
данных, которая возникает при ненормализиро-  
ванности таблицы фактов в MOLAP и схеме "звезда"

в ROLAP. Вместе с тем агрегирование по более высоким уровням иерархии не будет таким затратным по времени, как при использовании схемы "снежинка" в ROLAP. Так, объектная модель данных может решить две проблемы при проектировании OLAP, а именно проблему избыточности данных, возникающую при использовании денормализованных таблиц, и проблему увеличения числа операций при нормализации базы данных, что приводит к снижению скорости работы системы в целом.

### Использование метаданных при построении OLAP-системы в объектно-ориентированных СУБД

Ниже рассматриваются вопросы проектирования и реализации системы OLAP для объектно-ориентированных СУБД. Данная задача является одной из задач научного исследования, касающегося интеграции информационных технологий баз данных, баз знаний и Data Mining. Так, с использованием объектной модели данных были разработаны OLAP-кубы для трех различных баз данных: торговой компании, продажи автомобилей и компьютерной сети. Для построения кубов использовался алгоритм многопозиционного агрегирования многомерного массива [7]. Все три базы данных относятся к совершенно разным предметным областям. Поэтому возникла задача создания OLAP-системы, которую можно было бы использовать для произвольной объектной базы данных. При этом возникает одна острая проблема: OLAP-средство не может работать с произвольной объектной базой данных в связи со статическим встраиванием схемы базы данных в объектное приложение [8]. Поэтому можно сделать вывод, что для реализации

технологий OLAP (а затем и Data Mining) необходимо использовать метаданные. В работе [9] предлагается описание следующих метаданных, представляющих собой метаклассы: объект, простой тип, сложный тип, атрибут, параметр, связь между классами. На их основе в данной работе предлагается добавить такие метаклассы: пакет, таблица фактов, измерение, мера, куб. Диаграмма данных метаклассов представлена на рис. 2.

Метакласс "Таблица фактов" вводится потому, что таблица фактов является специфической таблицей, в которой должны быть отображены измерения и меры для построения OLAP-куба. В данном метаклассе хранится информация про название соответствующей таблицы фактов, ее измерения, меры и кубы.

```

Class FactTable
{
    attribute string Name; // название
    таблицы фактов
    attribute date CreatioDate; // дата
    создания
    relationship Package ParentPackage
        inverse Package::FactTables; //
    связь с метаклассом "Пакет", пакет
    (схема данных) может иметь несколь-
    ко таблиц фактов
    relationship set<Dimension> Dimen-
    sions
        inverse Dimension::facttable //
    связь с метаклассом "Измерение"
    relationship set<Measure> Measures
        inverse Measure::facttable //
    связь с метаклассом "Мера"
}
    
```

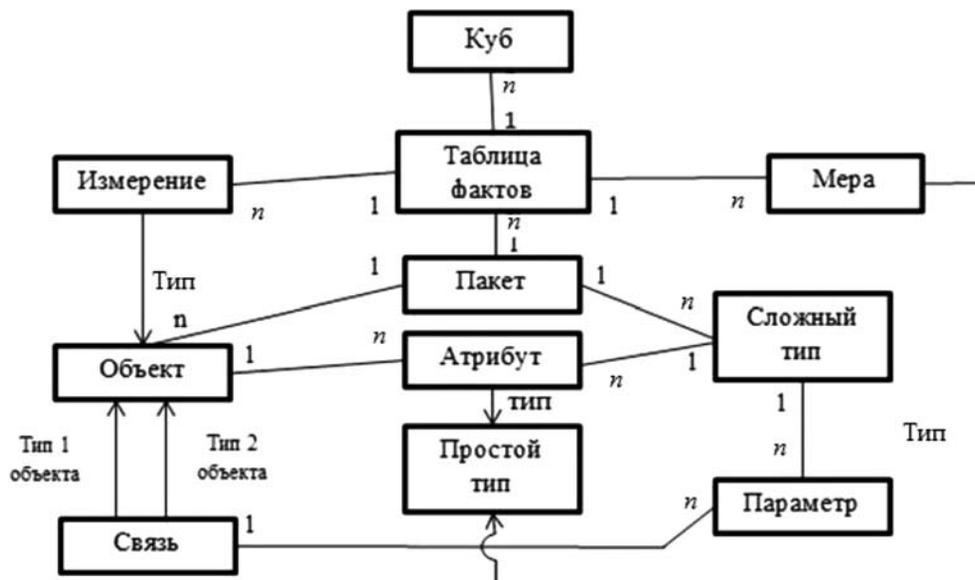


Рис. 2. Диаграмма метаклассов OLAP-системы в объектно-ориентированной СУБД

Метакласс "Измерение" хранит информацию о наборе доменов, по которым будет строиться многомерное пространство. Измерение относится к определенной таблице фактов, его типом будет некий предметный класс, определенный в системе. Также в данном метаклассе хранится число значений измерения, список значений и специальный параметр шага, который используется при выполнении алгоритма многопозиционного агрегирования многомерного массива при построении OLAP-куба.

```
Class Dimension
{
    attribute string Name; // название измерения
    attribute Object Type; // тип измерения
    attribute integer Number; // количество значений измерения (а именно количество экземпляров класса данного измерения)
    attribute set<String> Values; // список значения измерения
    relationship FactTable facttable
        inverse FactTable::Dimensions;
}
```

Метакласс "Мера" хранит информацию об атрибуте таблицы фактов, по которому будет происходить агрегирование. Таблица фактов может иметь несколько мер. Типом меры является некоторый простой тип (целое число, вещественное число и т. д.), определенный в системе. Также существуют различные функции агрегирования: сумма, количество, среднее значение, минимум, максимум и т. д. Данный метакласс содержит информацию о названии меры, ее тип, функции агрегирования.

```
Class Measure
{
    attribute string Name; // название меры
    attribute SimpleType Type; // тип меры
    attribute string AggrFunc; // функция агрегирования
    relationship FactTable facttable
        inverse FactTable::measures;
}
```

Последний из добавленных метаклассов "Куб" содержит информацию о названии OLAP-куба, дате его создания и таблице фактов, на основе которой построен данный куб,

```
Class Cube
{
    attribute string Name; // название куба
    attribute date CreationDate; // дата создания куба
    relationship FactTable facttable
        inverse FactTable::Cubes;
}
```

## Заключение

На данный момент апробированы алгоритмы построения кубов OLAP для трех баз данных с различными предметными областями [10], также реализована часть представленной на вышеуказанной диаграмме системы, представляющая подсистему проектирования базы данных посредством метаклассов и ведения данных путем создания, удаления и модификации экземпляров классов, спроектированных с помощью метаданных. Создание метаклассов позволит существенно упростить проектирование OLAP-приложений в среде объектных СУБД. Ведется также работа по реализации подсистемы OLAP в данной системе. В дальнейшем планируется проектирование подсистемы интеллектуального анализа данных (Data Mining), представляющей алгоритмы поиска ассоциативных правил в OLAP-кубах.

## Список литературы

1. Барсегян А. А., Куприянов М. С., Степаненко В. В., Холод И. И. Методы и модели анализа данных: OLAP и Data Mining. СПб.: БХВ-Петербург, 2004.
2. Паклин Н. Б., Орешков В. И. Бизнес-аналитика: от данных к знаниям: учеб. пособие. СПб.: Питер, 2010.
3. Харинатх С. и др. Microsoft SQL Server Analysis Services 2008 и MDX для профессионалов: пер. с англ. М.: Вильямс, 2010.
4. Кирстен В., Иррингер М., Кюн М., Рериг Б. Постреляционная СУБД Cache 5. Объектно-ориентированная разработка приложений. — 3-е изд., перераб. и дополн. — М.: Бином-Пресс, 2008.
5. Вендров А. М. CASE-технологии. Современные методы и средства проектирования информационных систем. URL: <http://www.citforum.ru>
6. Харрингтон Д. Проектирование объектно-ориентированных баз данных: пер. с англ. М.: ДМК Пресс, 2001.
7. Кудрявцев Ю. Обзор алгоритмов MOLAP. ВМиК МГУ. 2008. URL: [http://www.citforum.ru/consulting/BI/molap\\_overview/node2.shtml](http://www.citforum.ru/consulting/BI/molap_overview/node2.shtml).
8. Коновалов А. В. Анализ данных в объектно-ориентированных СУБД // Искусственный интеллект (Донецк). 2000. № 2. С. 74—81.
9. Труб И. И. СУБД Cache: работа с объектами. М.: ДИАЛОГ-МИФИ, 2006.
10. Фісун М. Т., Горбань Г. В. Аналіз особливостей об'єктної та багатовимірної моделей даних в СКБД Cache // Вестник Херсонского национального технического университета. 2011. № 2(41). С. 116—124.

# ОБРАБОТКА ИЗОБРАЖЕНИЙ IMAGE PROCESSING

УДК 004.9;519.25

Н. А. Соломещ, аспирант, МФТИ, Россия, Долгопрудный

И. В. Браиловский, канд. физ.-мат. наук, принципиальный инженер, Lab126, Калифорния, США  
e-mail: Solomeshch.Natalya@gmail.com, brailovs@lab126.com

## Анализ и оптимизация работы вейвлетного видеокодека в условиях передачи видеопотока по нестабильному каналу связи

*Исследуется зависимость качества закодированного видеопотока от уровня помех в канале связи в случае видеокодирования на основе вейвлетного преобразования. Для решения поставленной задачи проводились вычислительные эксперименты на базе кодека с открытым исходным кодом Schrodinger. Результаты исследования позволяют адаптировать вейвлетный видеокодек для передачи видеопотока оптимального качества по нестабильному каналу.*

**Ключевые слова:** видеокодирование, вейвлет-преобразование, каналы связи с помехами, битрейт, функция скорости искажений, метрика изображений, SSIM, устойчивость к ошибкам

N. A. Solomeshch, I. V. Brailovskiy

## Analysis of Video Quality of Wavelet Video Codec for Unreliable Communicational Channel

*This paper examines dependency of video quality from the level of interference in communicational channel in case of wavelet based video codec. To analyze this dependency authors conducted computational experiments based on open source wavelet-based video codec Schrodinger. The results of this research allows adapting parameters of the wavelet video codec for minimizing visual quality degradations for streaming over unreliable communicational channel.*

**Keywords:** video coding, wavelet transform, communicational channels with interference, bit rate, rate distortion function, image and video quality, SSIM, robustness

### Введение

Передача видеопотока по цифровым каналам связи является важной задачей, представляющей большой теоретический и прикладной интерес [1]. Такой способ передачи информации имеет ряд особенностей, в частности непостоянную, в общем случае, пропускную способность такого канала связи. В настоящее время предложено большое число способов повышения устойчивости кодирования и восстановления данных при наличии ошибок на основе дискретного косинусного преобразования [2, 3]. Недостатком подхода использования стандартных кодеков является слабая в целом защищенность кодека от потерь. Во-первых, ошибка в одном бите может быть критичной с точки зрения восстановления изображения для части или даже для целого кадра, поскольку часто используемое энтропийное кодирование — бинарное арифмети-

ческое кодирование — очень чувствительно к ошибкам. Во-вторых, даже в случае, когда используется более устойчивое кодирование переменной длины вместо арифметического кодирования, данные разной значимости перемешаны в потоке. В стандартных видеокодеках, таких как AVC, SVC, HEVC [4, 5], векторы движения чередуются с данными, относящимися к текущему блоку. Потеря битов, относящихся к векторам движения, влечет за собой потерю большого объема визуальной информации. В-третьих, в стандартных кодеках отсутствует непрерывная масштабируемость по качеству/битрейту. Это означает, что в одном потоке данных может содержаться только фиксированное число уровней качества/разных битрейтов, в таких кодеках, как SVC, редко создается больше чем шесть уровней качества. Поэтому в случае потерь в канале связи старшие уровни SVC теряются, что приводит к резким скачкам качества изображения.

В отличие от кодеков, использующих дискретно косинусное преобразование, кодирование видео-последовательности с помощью вейвлет-преобразований имеет ряд преимуществ. Например, в JPEG2000 [6] потеря уровня ведет к потере всех последующих данных с более высокими частотами в пределах одной структуры, все низкие частоты остаются, что позволяет восстанавливать изображение. Из битового потока JPEG2000 можно терять последние блоки данных и получать правильный JPEG2000 немного более низкого качества и меньшего размера (что означает непрерывную масштабируемость по качеству и размеру). Целый ряд исследований указывает на более высокое качество видеопотока, закодированного JPEG2000, нежели JPEG [7]. Единственным недостатком JPEG2000 по сравнению с JPEG является его высокая вычислительная сложность. Но поскольку вычислительные мощности постоянно растут, эта особенность JPEG2000 постепенно перестает быть большим недостатком.

Видеокодеки, построенные на базе вейвлет-преобразований, отстают по качеству от стандартных кодеков, таких как AVC или HEVC. Однако тот факт, что JPEG2000 превосходит JPEG, а также ряд сравнений кодеков на базе вейвлетов со стандартными кодеками на базе DCT [8] внушают некоторый оптимизм по поводу эволюции качества кодеков на базе вейвлет-преобразований [9]. Стоит отметить, что в каналах связи, где потери неизбежны, наиболее важным параметром является устойчивость видеопоследовательности к потерям.

В настоящей работе авторы решают следующие задачи:

- выявить зависимость качества вейвлетного кодека при потерях пакетов в канале связи;
- определить допустимый уровень потерь для приемлемого субъективного качества видеопотока, закодированного с использованием вейвлетов;
- выработать рекомендации для параметров передачи и кодирования видеокодека с помощью вейвлет-преобразований, передаваемого по нестабильному каналу связи.

Для решения поставленных задач авторы использовали метод вычислительных экспериментов.

### **Модель канала связи и потери информации**

Одна из самых распространенных и простых моделей канала связи — это модель, в которой ошибки возникают равномерно и равновероятно в любом месте видеопотока. В этом случае данные разделяются на пакеты одинакового размера и равного приоритета. Потеря одного пакета в общем случае приводит к полной потере изображения в пределах одной структуры. Несмотря на распространенность такой модели, она является далеко не самой точной для реальных потерь в пакетных сетях. В условиях реальных потерь при динамиче-

ски нестабильной ширине канала связи у отправляющей стороны есть возможность проиндексировать пакеты на важные и второстепенные пакеты. Важные пакеты доставляются с очень высокой вероятностью.

Следует отметить, что одним из принципиальных аспектов кодирования с помощью вейвлет-преобразований является возможность выделения основного видеопотока и нескольких второстепенных, которые дополняют качество основного. Основной видеопоток в этой модели передается гарантированно за счет сравнительно малых размеров основных пакетов, второстепенные передаются в зависимости от ширины канала и могут иногда теряться (при этом происходит потеря качества). Таким образом, появляется возможность получать непрерывный видеопоток при динамическом изменении ширины канала.

Однако в рамках данного подхода можно использовать различные способы разбиения видеопотока на пакеты с различной степенью важности. Выбор функций вейвлет-преобразования играет важную роль с точки зрения качества декодированного видеопотока. Однако необходимо исследовать более широкий спектр критериев качества видеоконтента.

### **Разложение видеоряда на высоко- и низкочастотные компоненты**

Для симуляционной модели авторы рассматривали несколько вариантов, в частности, рассмотрели проект Motion Jpeg 2000 как первый шаг. Однако в ходе тестовых запусков была выявлена проблема формирования контекста для энтропийного энкодера. По этой причине, а также вследствие того, что Motion Jpeg 2000 поддерживает кодирование кадров типа intra, авторы решили использовать вейвлетный видеокодек Dirac [10], а именно его вторую реализацию — Schrodinger. Schrodinger осуществляет сжатие с помощью вейвлетов вместо дискретного косинусного преобразования, применяющегося в большинстве стандартных кодеков (таких как H.264 или VC-1).

В работе авторами было рассмотрено шесть вейвлет-преобразований, таких как LeGall 5\_3 [11], Daubechies 9\_7 [12], Haar1 [1], Haar0 [1], Fidelity [9], Deslauriers Dubuc 13\_7 [13], используемая глубина преобразований была выбрана равной пяти.

Напомним некоторые основные свойства вейвлет-преобразований, необходимые для дальнейшего изложения. Вейвлет-преобразование одномерного сигнала — это его представление в виде обобщенного ряда или интеграла Фурье по системе базисных функций, сконструированных из исходного вейвлета  $\psi(t)$ , обладающего определенными свойствами за счет операций сдвига во времени  $b$  и изменения временного масштаба  $a$ . Множитель  $\frac{1}{\sqrt{a}}$

обеспечивает независимость нормы этих функций от масштабирующего числа  $a$ :

$$\psi_{mk}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) = \frac{1}{\sqrt{2^m}} \psi(2^{-m}t - k), \quad (1)$$

где  $m$  и  $k$  — целые числа,  $a = 2^m$ ,  $b = k2^m$ . В этом случае плоскость  $a$ ,  $b$  превращается в соответствующую сетку  $m$ ,  $k$ . Параметр  $n$  называется параметром масштаба.

В случае двумерного вейвлет-преобразования используют двумерный аналог формулы (1):

$$\frac{1}{\sqrt{a_1 a_2}} \psi\left(\frac{x-b_1}{a_1}, \frac{x-b_2}{a_2}\right), \quad a = 2^m, \quad b = k2^m = ka, \quad (2)$$

где  $a_1$  и  $a_2$ ,  $b_1$  и  $b_2$  — значения  $a$  и  $b$  по каждому измерению. Двумерный кратномасштабный анализ строится как тензорное произведение одномерных. При таком подходе исходные вейвлеты будут сформированы следующим образом:

$$\phi(x, y) = \phi(x)\phi(y);$$

$$\psi_{LH}(x, y) = \phi(x)\psi(y), \quad \psi_{HL}(x, y) = \psi(x)\phi(y); \quad (3)$$

$$\psi_{HH}(x, y) = \psi(x)\psi(y), \quad (4)$$

где индексы  $H$  (*High*) и  $L$  (*Low*) означают реализацию фильтров ВЧ (высокочастотных) и НЧ (низкочастотных) составляющих.

Тогда двумерные вейвлетные базисные функции запишутся в виде

$$2^{-m}\phi(2^{-m}x - k)\phi(2^m y - l), \quad 2^{-m}\phi(2^{-m}x - k)\psi(2^m y - l);$$

$$2^{-m}\psi(2^{-m}x - k)\phi(2^m y - l), \quad 2^{-m}\psi(2^{-m}x - k)\psi(2^m y - l).$$

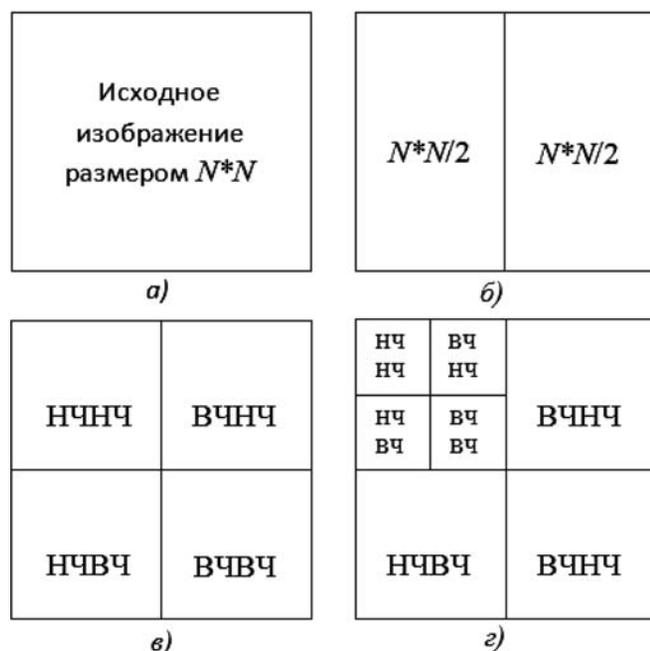


Рис. 1. Схема работы прямого вейвлета

Таким образом, на двумерной плоскости происходит анализ по горизонтали, вертикали и диагонали с одинаковым разрешением в соответствии с тремя приведенными выше вейвлетами [14].

На рис. 1 можно видеть графическую иллюстрацию работы прямого вейвлет-преобразования. Предположим, что имеем изображение размером  $N \times N$  (рис. 1, а). Первоначально каждая из  $N$  строк изображения фильтруется на НЧ и ВЧ составляющие. В результате получается два изображения размером  $N \times N/2$  (рис. 1, б). Далее каждый столбец делится точно так же, в итоге получается четыре изображения размером  $N/2 \times N/2$  (рис. 1, в): НЧ по горизонтали и вертикали (НЧНЧ), ВЧ по горизонтали и вертикали (ВЧВЧ), НЧ по горизонтали и ВЧ по вертикали (НЧВЧ) и ВЧ по горизонтали и НЧ по вертикали (ВЧНЧ). Первое из указанных выше изображений делится аналогичным образом на следующем уровне преобразования (рис. 1, г) и т. д.

Двумерное дискретное вейвлет-преобразование в нашем примере сохраняет большую часть значимых коэффициентов в верхнем левом углу — остальные три четверти состоят в основном из нулевых или близких к нулю значений. Исходя из локальности преобразований получается, что любой элемент построен на основе четырех элементов исходного входного изображения.

### Разбиение видеопотока на пакеты и описание типов потерь

Человеческое восприятие более чувствительно к ошибкам на низких частотах, чем к потерям высокочастотных данных. Кроме того, в общем случае концентрация человеческого внимания более высока на центре кадра [15]. Таким образом, ошибки, вызванные потерями данных в областях, расположенных по краю изображения, менее заметны, нежели потери низкочастотных данных в центре кадра. Так как низкие частоты будут гарантированно переданы, и они наиболее значимы с точки зрения качества, то саббенды, содержащие низкочастотные коэффициенты, были расположены в центре кадра, а высокочастотные — по краям изображения. На рис. 2 отображена схема деления одного видеофрейма по соответствующим саббендам, симметрично отраженная относительно центра кадра. Таким образом, в саббендах в центре кадра сохранены низкочастотные коэффициенты, и вероятность их потери меньше, так как они меньшего размера, нежели саббенды по краям изображения, содержащие высокочастотные коэффициенты.

В процессе кодирования каждый видеофрейм разделялся на саббенды, закодированные различными вейвлетами. Каждый уровень содержит по три саббенда, кроме базового уровня. Базовый уровень содержит только один саббенд из низкочастотных коэффициентов (рис. 3). Каждый саббенд был по-

мещен в отдельный пакет для целей дальнейшей передачи по каналу связи.

Так как основной целью исследования является анализ скорости падения качества закодированного видеопотока в случае сужения информационного канала с частичной потерей пакетов (каждый пакет содержал один саббенд), то был проведен анализ влияния объема потерянного видеоконтента на качество декодированной последовательности. В исследовании рассматривали тестовые последовательности в разрешении CIF, такие как Bus, Foreman, News, Stefan, Motion, на пяти битрейтах от 0,6 до 2,2 Мбит. Исследование базировалось на универсальных сценариях использования при передаче пакетов по сети.

В сценарии № 1 все уровни были сохранены. В сценарии № 2 рассматривалась потеря всех данных пятого уровня (высокочастотные коэффициенты), в № 4 — потеря четвертого и пятого уровней, в сценарии № 5 потеряны данные трех уровней: пятого, четвертого и третьего. В сценарии № 6 сохранялся только базовый (низкочастотный) уровень. В сценарии № 3 происходили потери только третьего и пятого уровней, однако базовый уровень, второй и четвертый были сохранены (табл. 1).

### Результаты

Для определения наиболее устойчивого типа вейвлет-преобразований в условиях сетевой передачи пакетов каждый саббенд был помещен в отдельный пакет. Было оценено качество выходного потока для шести сценариев потери данных. Качество определялось референсной метрикой SSIM [16].

В табл. 2 представлено среднее значение SSIM при использовании шести вейвлет-преобразований на шести сценариях использования. В случае отсутствия потерь и 100 %-й передачи видеоконтента среднее значение SSIM составляет 0,93. В сценарии № 2 (потери пакетов только пятого уровня) мы видим уменьшение SSIM в среднем до 0,91. Обратим внимание, что при работе со сценарием № 4 качество выходного потока значительно уменьшается до 0,77. При работе по сценарию № 5 можно увидеть тенденцию уменьшения качества до 0,64, что яв-

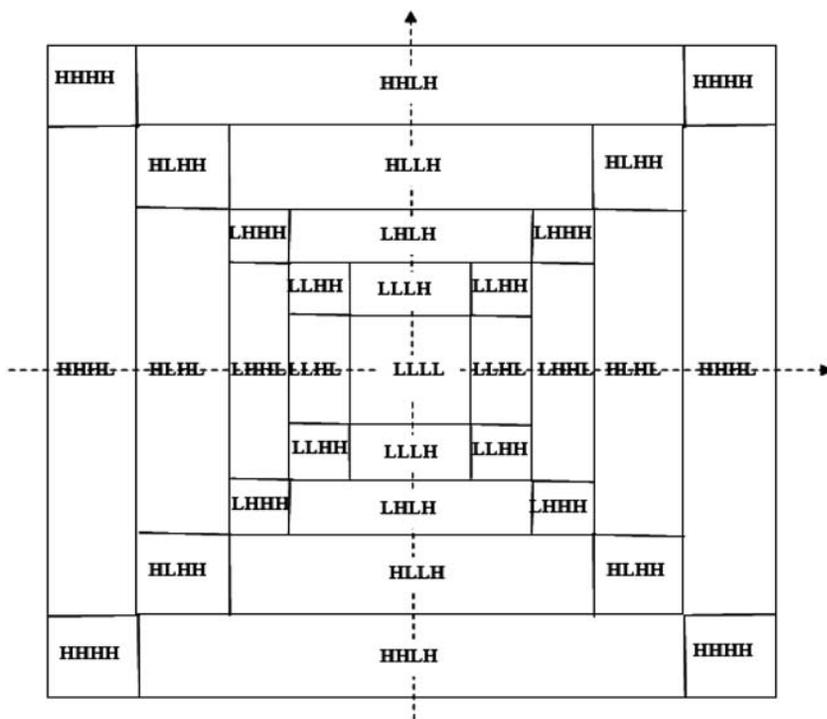


Рис. 2. Разделение видеокадра по саббендам для вейвлет-преобразования глубиной, равной 5

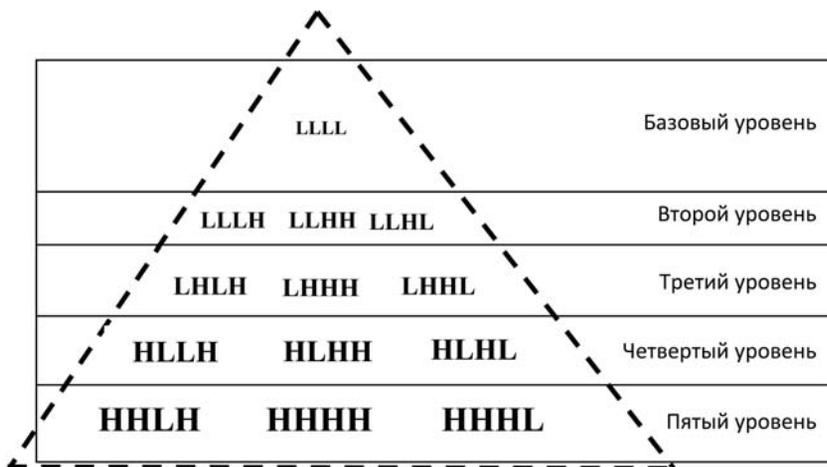


Рис. 3. Видеокадр, разложенный на пять уровней в процессе кодирования

Описание рассмотренных сценариев

Таблица 1

Сценарии	Уровни потерь				
	Базовый	Второй	Третий	Четвертый	Пятый
№ 1	Сохранено	Сохранено	Сохранено	Сохранено	Сохранено
№ 2	Сохранено	Сохранено	Сохранено	Сохранено	Потеряно
№ 3	Сохранено	Сохранено	Потеряно	Сохранено	Потеряно
№ 4	Сохранено	Сохранено	Сохранено	Потеряно	Потеряно
№ 5	Сохранено	Сохранено	Потеряно	Потеряно	Потеряно
№ 6	Сохранено	Потеряно	Потеряно	Потеряно	Потеряно

ляется низким показателем качества работы видеокодека при больших потерях данных. В рамках сценария № 3 происходили потери только третьего и пятого уровней — SSIM = 0,85, однако значение метрики качества выше, чем в сценариях № 4 и № 5. При рассмотрении крайнего случая (сценарий № 6) значение SSIM уменьшается до 0,58, что является очень низким показателем качества. Несколько

сценариев можно объединить, например сценарии № 2 и № 3, так как при потере только пятого или третьего и пятого уровней сценарии дают близкий результат по качеству выходного видеопотока — не различимый на субъективном уровне (рис. 4, см. вторую сторону обложки). Итоговый размер объема потерянных пакетов также был учтен, как один из ключевых факторов качества работы кодека.

Таблица 2

Среднее значение SSIM при использовании различных вейвлет-преобразований для шести сценариев использования

Тип вейвлета	Сценарии					
	№ 1	№ 2	№ 3	№ 4	№ 5	№ 6
Deslauriers Dubuc 13_7	0,9360664	0,9128304	0,8589536	0,7753336	0,6449512	0,5811292
Fidelity	0,9360672	0,9128304	0,8589532	0,7753356	0,6449516	0,5811292
Haar_0	0,936336	0,9122776	0,8589532	0,7764496	0,646854	0,5807832
Haar_1	0,936326	0,9059844	0,8589532	0,7732536	0,6395988	0,5698104
Daubechies 9_7	0,9380484	0,9159908	0,8589532	0,7830224	0,6554416	0,5905244
LeGall 5_3	0,9360664	0,9128304	0,8589548	0,7753336	0,644952	0,5811292

Таблица 3

Процентное соотношение среднего объема сохраненных пакетов при использовании различных вейвлет-преобразований для шести сценариев использования

Тип вейвлета	Сценарии					
	№ 1	№ 2	№ 3	№ 4	№ 5	№ 6
Des. Dub. 13_7	100	96,26	95,38	64,80	27,19	9,32
Fidelity	100	96,26	95,38	64,80	27,19	9,32
Haar_0	100	96,19	93,86	59,11	22,52	7,66
Haar_1	100	99,42	93,98	64,28	23,35	6,67
Daubechies 9_7	100	96,69	95,58	70,44	32,26	12,06
LeGall 5_3	100	96,26	95,37	64,80	27,19	9,32

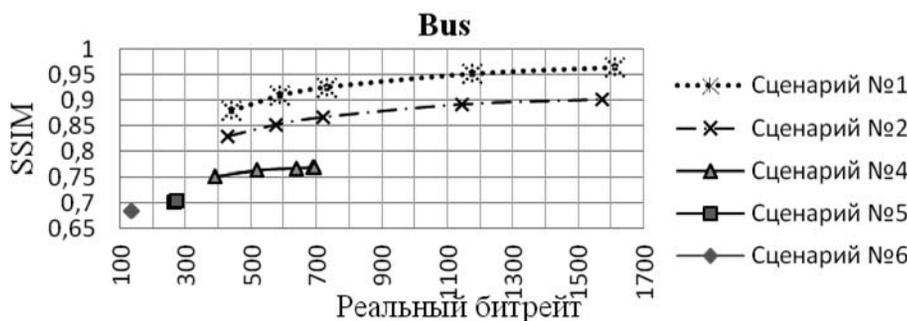


Рис. 5. Зависимость SSIM от реального битрейта на последовательности Bus для различных сценариев использования с применением вейвлета Daubechies 9\_7

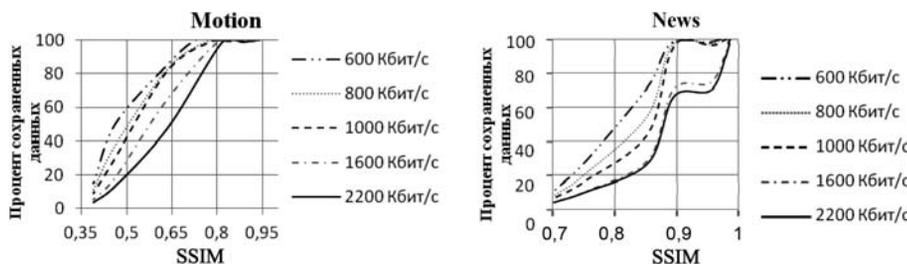


Рис. 6. Зависимость SSIM от объема потерянных данных

На рис. 5 представлена зависимость SSIM от реального битрейта на последовательности Bus. Значение SSIM больше в случае отсутствия потерь или при использовании сценария № 2. При работе по сценарию № 6 или № 5 размер сохраненных пакетов настолько мал (табл. 3), что уже нет разницы, на каком битрейте изначально была закодирована видеопоследовательность.

Проанализировав рис. 4, можно сделать вывод о том, что при потерях уровней третьего и второго качество выходного потока имеет приблизительно похожий низкий уровень качества, т. е. видеопоследовательность размыта примерно одинаково, и, с точки зрения субъективного анализа, изображение в обоих случаях имеет низкое качество. Из этого можно сделать вывод о том, что базовый, второй и четвертый уровни желательно передавать без потерь. Однако гарантированно передается только базовый уровень. Для уменьшения вероятности потерь пакетов, содержащих саббенды с низкочастотными коэффициентами, необходимо минимизировать их размеры, так как пакеты с меньшим размером входят с большей вероятностью, нежели большие пакеты.

Несмотря на то что значения SSIM для различных вейвлет-преобразований достаточно близки, они различаются в разрядах после запятой, и этого достаточно для того, чтобы проводить сравнение. Если добавить субъективное сравнение, то такой анализ подтверждает, что вейвлет-преобразование типа Daubechies 9\_7 справляется с задачей лучше, нежели Haar 1/0 (см. рис. 4).

Мы также рассмотрели зависимость SSIM от объема потерянных данных на пяти последо-

вательностях для шести битрейтов. Опираясь на полученные данные, можно предположить, что качество видеоконтента с большим объемом движения (bus, motion, stefan) более чувствительно к потерям, нежели статические последовательности типа "говорящие головы" (news и foreman). На рис. 6 приведены графики зависимости SSIM от объема потерянных данных на одном вейвлет-преобразовании.

### Заключение

Решена задача выявления и анализа качества вейвлетного кодера в условиях потерь пакетов в канале связи. На основе этого анализа выработан ряд рекомендаций по использованию вейвлетного видеокодера при потоковой передаче видеоряда. В частности, предлагается выделить наиболее приоритетные пакеты, содержащие низкочастотные вейвлет-коэффициенты.

Были проведены анализ и сравнение шести различных типов вейвлет-преобразований. С небольшим превосходством по качеству победило преобразование Daubechies 8\_7, хотя следует отметить, что остальные преобразования не сильно различаются по качеству за исключением вейвлетов на базе Хаар.

Также были рассмотрены пять тестовых последовательностей на пяти битрейтах. Сложность кодирования в условиях потерь существенно зависит от интенсивности движения, поэтому качество падает на последовательностях с более сложным движением. Следует отметить, что качество на 0,8 Мбит/с часто незначительно уступает качеству на 1 Мбит/с, однако, как правило, превосходит 0,6 Мбит/с. В связи с этим оптимальным выглядит использование битрейта 0,8 Мбит/с при кодировании вейвлетным видеокодером видеопотока.

Анализ сценариев потерь указывает на то, что при потере более двух уровней коэффициентов качество видеопотока становится трудным для человеческого восприятия. Поэтому авторы рекомендуют выделить основные пакеты таким образом, чтобы как минимум три уровня (см. рис. 3) передавались гарантированно. Передача четырех уровней увеличивает качество получаемого видеоряда, однако требует дополнительных 50 % от размера передаваемого файла.

Развитием настоящего исследования может быть применение предложенной методики для изучения более эффективных и современных методов сжатия видеопоследовательностей, таких как 3D-вейвлеты, улучшенные схемы компенсации движения для вейвлетных кодеков и т. д.

Другим направлением развития может быть анализ альтернативных метрик оценки качества видеоряда, таких как метрика на базе нейронных се-

тей [17], узконаправленные метрики, например метрика размытости — Blur [18], и т. д. Перспективно было бы также изучить зависимости между используемым вейвлет-преобразованием и типом видеоконтента.

### Список литературы

1. Rao K. R., Bojkovic Z. S., Milovanovic D. A. *Multimedia Communication Systems: Techniques, Standards and Networks*. NJ, USA: Prentice-Hall, 2002.
2. Katz, B., Greenberg S., Yarkoni N., Blaunstein N. & Giladi R. New error-resilient scheme based on FMO and dynamic redundant slices allocation for wireless video transmission // *IEEE Transactions on Broadcasting*. 2007. V. 53. Is. 1. P. 308—319.
3. Stockhammer T., Zia W. *Error Resilient Coding and Decoding Strategies for Video Communications // Multimedia over IP and Wireless Networks: Compression, Networking, and Systems*. Elsevier Academic Press, Match 2007. P. 13—58.
4. ITU-T Recommendation H.264/ISO/IEC 14496-10: Audiovisual and Multimedia Systems Infrastructure of audiovisual services — Coding of moving video; Advanced video coding for generic audiovisual services. Telecommunication Standardization Sector of ITU. International Telecommunication Union, Mar. 2005.
5. Sullivan G., Ohm J., Han W., and Wiegand T. Overview of the high efficiency video coding (HEVC) standard // *IEEE Transactions on Circuits and Systems for Video Technology*, Dec. 2012. V. 22. P. 1649—1668.
6. International Standard ISO/IEC 15444:2000 ITU-T Recommendation T.800. Information technology — JPEG 2000 image coding system. Telecommunication Standardization Sector of ITU, 2003.
7. Farzad E., Matthieu C., Stefan W. JPEG vs. JPEG2000: An objective comparison of image encoding quality // *Proc. SPIE Applications of Digital Image Processing*. Denver, 2004. V. 5558. P. 300—308.
8. Dunic E., Mustra M., Grgic S. Comparison of Dirac and H.264/AVC Coding Quality Using Objective Video Quality Measures // *Systems, Signals and Image Processing, IWSSIP 16th International Conference*, 2009. Chalkida. P. 1—4.
9. Zhang Y., Zafar S. Motion-compensated wavelet transform coding for color video compression // *IEEE Transactions on Circuits and Systems for Video Technology*, September. 1992. V. 2, Is. 3. P. 285—296.
10. Dirac Specification Version 2.2.3 Issued: September 23. 2008. URL: <http://diracvideo.org/download/specification/dirac-spec-lat-test.pdf>
11. Le Gall D., Tabatabai A. Sub-band coding of digital images using symmetric short kernellters and arithmetic coding techniques // *Proc. ICASSP1988, IEEE, New York*, April 1988. P. 761—764.
12. Daubechies I. *Ten Lectures on Wavelets*. Philadelphia: SIAM, 1992. V. 61.
13. Deslauriers G., Dubuc S. Interpolation Dyadique // *Fractals: dimensions non entieres et applications*. Paris, New York, Barcelone, 1987. P. 44—45. URL: <http://www.worldcat.org/title/fractals-dimensions-non-entieres-et-applications/oclc/799162931>
14. Яковлев А. Н. Введение в вейвлет-преобразования. Новосибирск: Изд-во НГТУ. 2003. 104 с.
15. Judd T., Durand F. et al. A Benchmark of Computational Models of Saliency to Predict Human Fixations // *Tech. rep., MIT*. Jan. 2012.
16. Wang Z., Bovik A., Sheikh H., Simoncelli F. Image quality assessment: From error visibility to structural similarity // *IEEE Transactions on Image Processing*. 2004. V. 13, Is. 4. P. 600—612.
17. Saad M., Bovik A. Blind image quality assessment: A natural scene statistics approach in the DCT domain // *IEEE Transactions on Image Processing*, 2012. Vol. 21, Is. 8. P. 3339—3352.
18. Crete F., Dolmiere T., Ladret P., Nicolas M. The Blur Effect: Perception and Estimation with a New No-Reference Perceptual Blur Metric // *Proceedings of SPIE, Human Vision and Electronic Imaging XII, San Jose*. 2007. V. 6492.

ЖУРНАЛ В ЖУРНАЛЕ



# НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

№ 1  
ЯНВАРЬ  
2014

**Главный редактор:**

ГАЛУШКИН А.И.

**Редакционная коллегия:**

АВЕДЬЯН Э.Д.  
БАЗИАН Б.Х.  
БЕНЕВОЛЕНСКИЙ С.Б.  
БОРИСОВ В.В.  
ГОРБАЧЕНКО В.И.  
ЖДАНОВ А.А.  
ЗЕФИРОВ Н.С.  
ЗОЗУЛЯ Ю.И.  
КРИЖИЖАНОВСКИЙ Б.В.  
КУДРЯВЦЕВ В.Б.  
КУЛИК С.Д.  
КУРАВСКИЙ Л.С.  
РЕДЬКО В.Г.  
РУДИНСКИЙ А.В.  
СИМОРОВ С.Н.  
ФЕДУЛОВ А.С.  
ЧЕРВЯКОВ Н.И.

**Иностранные  
члены редколлегии:**

БОЯНОВ К.  
ВЕЛИЧКОВСКИЙ Б.М.  
ГРАБАРЧУК В.  
РУТКОВСКИЙ Л.

**Редакция:**

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

**Савченко А. В.**

Сегментация речи в задачах автоматического обнаружения и распознавания голосовых команд . . . . . 53

**Аведьян Э. Д., Луганский В. Э.**

Подход к задаче заполнения числовых пропусков в таблицах и строках, основанный на многослойной нейронной сети и нейронной сети СМАС . . . . . 58

**Сирота Е. А.**

Прогнозирование нестационарных временных последовательностей на основе иерархических многослойных нейронных сетей . . . . . 67

А. В. Савченко, канд. техн. наук, доц.,

Национальный исследовательский университет, Высшая школа экономики — Нижний Новгород,  
e-mail: avsavchenko@hse.ru

## Сегментация речи в задачах автоматического обнаружения и распознавания голосовых команд<sup>1</sup>

*Рассмотрена задача автоматического выделения голосовых команд из непрерывного речевого потока и их последующего распознавания. Для обнаружения в слогах границ гласных звуков используется основанный на идее вероятностной нейронной сети с проверкой однородности алгоритм фонемной сегментации речи. Предложено считать речевой сигнал частью голосовой команды, если доля длительности выделенного в нем однородного сегмента по отношению к длительности всего слога превышает фиксированный порог. Экспериментально показано, что для четкого слогового произношения команд такой подход позволяет в 90 % случаев выделить голосовую команду, состоящую из более чем двух слогов.*

**Ключевые слова:** автоматическое распознавание речи, сегментация речи, вероятностная нейронная сеть с проверкой однородности, система голосового управления, метод фонетического кодирования слов

A. V. Savchenko

## Speech Segmentation in the Problems of Automatic Voice Control Detection and Recognition

*The problem of automatic detection and recognition of the voice commands in the continuous speech in control application are investigated. The phoneme segmentation algorithm on the basis of the probabilistic neural network with homogeneity testing is applied to extract a vowel from the syllable. The speech signal is proposed to be marked as a part of the voice control command if the ratio of durations of the homogeneous segment and the whole syllable exceeds a fixed threshold. The experimental research shows that the proposed approach extracts the control command containing more than two syllables in 90 % cases if the commands are pronounced in isolated syllable mode.*

**Keywords:** automatic speech recognition, speech segmentation, probabilistic neural network with homogeneity testing, voice control system, words phonetic coding method

### Введение

Разработка систем голосового управления [1, 2] является одним из наиболее перспективных и прогрессирующих направлений в области современных речевых технологий [3]. В последние годы появились такие успешные коммерческие продукты, как Apple SIRI [4], Microsoft Voice Command, Windows Speech Recognition [5], Nuance Dragon Naturally Speaking [6] и др., служащие, в первую очередь, для управления персональным компьютером или мобильным телефоном. Развитие подобных систем связано [7] с их применением в разнообразных областях человеческой деятельности, таких как интеллектуальное управление бытовыми приборами ("умный дом"), построение интерактивных конференц-залов и учебных классов [8], системы тревоги (экспресс-оповещение специальных служб спасения об авариях на транспорте, на производстве и

в быту) [7], удаленное управление техникой (в том числе военной) и др. Известно [1, 7], что перед разработчиками таких систем стоят следующие основные проблемы: недостаточная точность алгоритмов распознавания речи для произвольных словарей команд, а также проблема ложных тревог [7] — при работе в режиме непрерывного времени даже невысокая вероятность ошибки первого рода [9] приводит к большому числу (за фиксированный период времени, например сутки) ложных обнаружений команд (ключевых слов) и выполнению автоматизированной системой ассоциированных с этими командами действий.

Как показано в работе [1], можно построить достаточно надежную вычислительно эффективную систему фонемного распознавания русской речи [10] с автоматически перенастраиваемым словарем и быстрой настройкой на голос пользователя, если искусственно ввести требование к пользователю на четкое слоговое произношение команд [1]. Между тем до настоящего времени еще не рассматривали возможности такой системы для выделения голо-

<sup>1</sup> Исследование осуществлено в рамках Программы "Научный фонд НИУ ВШЭ" в 2013—2014 гг., проект № 12-01-0003.

совых команд из непрерывного потока речи [7]. Исследованиям в этом актуальном направлении и посвящена настоящая работа. В ней предложен новый подход к детектированию ключевых слов, основанный на предварительной фонемной сегментации [12, 13] выделенных слогов с использованием вероятностной нейронной сети с проверкой однородности [14, 15] и сопоставлением с некоторым фиксированным порогом отношения длительности выделенного таким образом гласного звука к длительности всего слога. Представленные результаты рассчитаны на широкий круг специалистов в области речевых технологий.

### Метод фонетического кодирования-декодирования в задаче распознавания голосовых команд

Пусть заданы множество из  $L > 1$  команд  $\{X_l\}$ , где  $l = \overline{1, L}$  — номер слова-эталона, и фонетическая база данных (ФБД) из  $R$  эталонных фонем  $\{x_r^*\}$ ,  $r = \overline{1, R}$ . Каждой фонеме поставлен в соответствие номер класса  $c(r) \in \{1, \dots, C\}$ , где  $C$  — число различных классов. Вначале [1] эталонная команда разбивается на последовательность гласных фонем  $X_l = \{c_{l,1}, c_{l,2}, \dots, c_{l,L_l}\}$ . Здесь  $L_l$  — число фонем в  $l$ -й команде,  $c_{l,j}$  — номер класса  $j$ -й гласной фонемы  $l$ -го слова. Задача состоит в том, чтобы речевому сигналу  $X$  поставить в соответствие наиболее близкое к нему слово-эталон.

Для решения задачи сигнал  $X$  разбивается на  $T$  непересекающихся частей  $\{x(t)\}$ ,  $t = \overline{1, T}$ , длиной  $\tau = 0,01\dots 0,013$  с. Далее каждый сегмент  $x(t) = [x_1(t)\dots x_\Lambda(t)]$  (здесь  $\Lambda = \tau F$  — число отсчетов в сигнале  $x(t)$ ,  $F$  — частота дискретизации (в герцах) отождествляется с гласным звуком из ФБД, ближайшим в смысле сильно коррелированного с субъективными оценками близости речевых сигналов (MOS, mean opinion score) [3] рассогласования Итакуры—Саито [16] между авторегрессионными оценками спектральной плотности мощности сигналов  $x(t)$  и эталона  $x_r^*$ :

$$v(t) = \arg \min_{r \in \{1, \dots, R\}} \rho_{IS}(x(t), x_r^*). \quad (1)$$

Далее предполагаем, что входная команда  $X$  разбита на  $N$  слогов, а границы  $n$ -го слога определены с точностью до номера сегмента  $(t_1^{(n)}, t_2^{(n)})$ . Тогда  $n$ -му слогу поставим в соответствие последовательность частот  $\mu^{(n)}(c)$ ,  $c = \overline{1, C}$ :

$$\mu^{(n)}(c) = \frac{1}{t_2^{(n)} - t_1^{(n)} + 1} \sum_{t=t_1^{(n)}}^{t_2^{(n)}} \delta(c(v(t)) - c), \quad (2)$$

где  $\delta(x)$  — дискретная дельта-функция. Решение принимается в пользу слова  $X^*$ , такого, что

$$X^* = \arg \min_{X_r, L_r=N, l=\overline{1, L}} \sum_{n=1}^N \mu^{(n)}(c_{l,n}). \quad (3)$$

Система выражений (1)–(3) и определяет метод фонетического кодирования-декодирования [17] к распознаванию голосовых команд [1]. Здесь для принятия решения в выражении (2) используются сегменты всего слога целиком. В настоящее время в практике автоматического распознавания речи более распространен подход, в котором в речевом сигнале предварительно выполняется сегментация [12] — выделение регулярных (стационарных) участков, которые далее сопоставляются со звуками из ФБД. Рассмотрим эту задачу более подробно.

### Фонемная сегментация речи на основе вероятностной нейронной сети с проверкой однородности

Задача сегментации  $n$ -го слога состоит в выделении границ четких фонем, определенных с точностью до номера сегмента  $(t_1^{(n)}(k), t_2^{(n)}(k))$ , где  $k = \overline{1, K_n}$  — номер фонемы;  $K_n$  — число выделенных фонем [18]. Решение этой задачи может быть проведено путем адаптивной процедуры сопоставления каждого сегмента  $x(t)$  с предыдущим выделенным регулярным участком сигнала  $x^{(n)}(k)$  [12]. Согласно одному подходу к распознаванию образов [19, 20] с неизвестными заранее распределениями классов, задачу следует сводить к проверке статистической однородности сегмента  $x(t)$  и сигнала  $x^{(n)}(k)$ :

$$W_0: \text{сигналы } x(t) \text{ и } x^{(n)}(k) \text{ имеют одинаковое распределение,} \quad (4)$$

против альтернативы

$$W_1: \text{сигналы } x(t) \text{ и } x^{(n)}(k) \text{ неоднородны.}$$

Предположим [15], что сигналы  $x(t)$  и  $x^{(n)}(k)$  порождены авторегрессионным процессом порядка  $p$ , имеют гауссовское распределение с нулевым средним и неизвестными автокорреляционными матрицами  $K(t)$  и  $K^{(n)}(k)$  соответственно, а векторы  $x_i(t) = [x_i(t) \dots x_{i+p}(t)]^T$ ,  $i = \overline{1, \Lambda - p}$ , независимы. Тогда оптимальное решение [20] задачи (4) принимается по объединенной выборке  $x^{(n)}(k; t) = [x^{(n)}(k)x(t)]$  из условия

$$W_0: \frac{\sup_{K^{(n)}(k)} \sup_{K(t)} f(x^{(n)}(k; t) | W_0)}{\sup_{K^{(n)}(k)} \sup_{K(t)} f(x^{(n)}(k; t) | W_1)} > \lambda_0 = \text{const},$$

где  $f(\mathbf{x}|W_j)$ ,  $j \in \{0, 1\}$ , — функция правдоподобия сигнала  $\mathbf{x}$  при справедливости гипотезы  $W_j$ ; символ  $\sup$  — верхняя граница функции на множестве допустимых распределений. Предполагая независимость сигналов  $\mathbf{x}(t)$  и  $\mathbf{x}^{(n)}(k)$ , выполнив логарифмирование последнего выражения, запишем

$$W_0: \sup_{K^{(n)}(k), K(t)} [\ln f(\mathbf{x}^{(n)}(k)|W_0) + \ln f(\mathbf{x}(t)|W_0)] - \sup_{K^{(n)}(k)} \ln f(\mathbf{x}^{(n)}(k)) - \sup_{K(t)} \ln f(\mathbf{x}(t)) < \ln \lambda_0,$$

где  $f(\mathbf{x})$  — функция правдоподобия (распределение) сигнала. Если обозначить число сегментов в участке  $\mathbf{x}^{(n)}(k)$  за  $\Delta t^{(n)}(k)$ , то, следуя известной [19, 20] методике вычислений и учитывая, что верхняя граница достигается при равенстве автокорреляционной матрицы ее максимально правдоподобной оценке [19], запишем

$$\sup_{K^{(n)}(k), K(t)} \ln f(\mathbf{x}^{(n)}|W_0) = -\frac{1}{2} (\Lambda \cdot \Delta t^{(n)}(k) - p) (\ln(2\pi) + \ln|\hat{K}(t)| + p + I_{KL}(\hat{K}(t); \hat{K}_x)).$$

Здесь  $I_{KL}(\hat{K}(t); \hat{K}_x)$  — дивергенция Кульбака—Лейблера [21] между оценками автокорреляционных матриц  $\hat{K}(t)$  и  $\hat{K}_x$  текущего сегмента  $\mathbf{x}(t)$  и сигнала  $\mathbf{x}^{(n)}(k; t)$ , соответственно. В итоге получаем следующее решающее правило:

$$W_0: (\Lambda \cdot \Delta t^{(n)}(k) - p) I_{KL}(\hat{K}^{(n)}; \hat{K}_x) + (\Lambda - p) I_{KL}(\hat{K}(t); \hat{K}_x) < -2 \ln \lambda_0.$$

Как известно [20, 22], рассогласование Кульбака—Лейблера между гауссовскими сигналами с точностью до константного множителя совпадает с рассогласованием Итакуры—Саито. Поэтому, предполагая, что  $\Lambda \gg p$ , окончательно получим следующий критерий для присоединения анализируемого сегмента  $\mathbf{x}(t)$  к текущему регулярному участку  $\mathbf{x}^{(n)}(k)$ :

$$\rho_{IS}(\mathbf{x}^{(n)}(k), \mathbf{x}^{(n)}(k; t)) + \frac{1}{\Delta t^{(n)}(k)} \rho_{IS}(\mathbf{x}(t), \mathbf{x}^{(n)}(k; t)) < \rho_1, \quad (5)$$

где порог  $\rho_1 = \text{const}$  подбирается экспериментально. Если условие (5) не выполняется и длительность текущего однородного участка  $\Delta t^{(n)}(k)$  достаточно велика (больше фиксированного порога), то сигнал  $\mathbf{x}^{(n)}(k)$  добавляется в список выделенных однородных участков, а  $\mathbf{x}(t)$  объявляется как новый,  $(k+1)$ -й, регулярный сегмент. Процедура повторяется для всех сегментов слога  $t = \overline{t_1^{(n)}, t_2^{(n)}}$ . Тогда итоговое решение в задаче распознавания гласной

фонемы в слоге принимается не по слогу целиком (2), а только по сегментам выделенных однородных участков:

$$\mu^{(n)}(c) = \frac{1}{t_2^{(n)} - t_1^{(n)} + 1} \sum_{k=1}^{K_n} \left[ (t_2^{(n)}(k) - t_1^{(n)}(k) + 1) \times \sum_{t=t_1^{(n)}(k)}^{t_2^{(n)}(k)} \delta(c(v(t)) - c) \right]. \quad (6)$$

Дальнейшее развитие предложенного правила сегментации (5) может быть связано с его использованием в голосовом управлении в присутствии посторонних шумов [2, 7]. В указанном приложении для метода фонетического кодирования-декодирования (1), (3), (6) за счет послогового произношения [1, 11] возникает возможность различать команды от непрерывной речи, сравнивая с порогом долю длительности выделенных в сигнале однородных сегментов по отношению к длительности анализируемого участка сигнала (слога):

$$\frac{1}{t_2^{(n)} - t_1^{(n)} + 1} \sum_{k=1}^{K_n} (t_2^{(n)}(k) - t_1^{(n)}(k) + 1) \geq \delta_0 = \text{const}. \quad (7)$$

Действительно, обычная непрерывная речь не содержит пауз между слогами (и иногда словами), поэтому указанная доля (7) будет мала, особенно при повышении порога на минимальную длительность (в сегментах) четкой фонемы. В таком случае, как будет продемонстрировано далее в ходе экспериментального исследования, получится с достаточной надежностью идентифицировать речевые команды в речевом потоке и, как следствие, повысить надежность всей системы голосового управления.

### Результаты экспериментальных исследований

Рассмотрим применение предложенного правила сегментации (5) и критерия обнаружения команды в речевом потоке (7) в задаче распознавания слов русской речи для следующих предметных областей:

- наименования 200 блюд из меню одной пиццерии (далее "Меню");
- названия 1913 лекарств (далее "Лекарства");
- список из 1830 крупных городов России вместе с включающими их областями, например, "Семенов (Нижегородская)", "Сочи (Краснодарский край)", "Казань (Татарстан)", но все города Московской области указываются без наименования области, например, "Москва", "Истра" (далее "Города").

Словарь формировался автоматически по списку из текстовых представлений слов/словосочетаний [1, 23]. Речь — в формате моно, частота дискретизации 8000 Гц, 16 бит — на отсчет записыва-

лась с помощью микрофона из гарнитуры A4Tech HS с функцией шумоподавления (*noise-canceling close-speaking microphone*), нелинейные искажения (клиппирование, автоматическая регулировка усиления и пр.) и реверберация отсутствуют, отношение сигнал/шум 20 дБ. В процессе распознавания слоги выделялись простейшим амплитудным детектором паузы, определенной как сигнал с малой амплитудой длительностью не менее 70 мс, минимальная длительность выделяемого слога — 120 мс. Далее каждый выделенный слог членился на последовательность пересекающихся сегментов длительностью  $\tau = 0,02$  с ( $\Lambda = 160$  отсчетов). Для сопоставления сигналов применялась дивергенция Итакуры—Саито между их спектральными плотностями мощности, оцененными на основе авторегрессионной модели порядка  $p = 20$  с использованием метода Берга и рекурсивной процедуры Левинсона—Дурбина [24]. Значение порога сегментации  $\rho_1 = 0,03$  было подобрано экспериментально. Для распознавания команд использовали программный комплекс фонетического декодирования слов [23].

ФБД состояла из всех гласных звуков русского языка, при этом звуки "а" и "я", "е" и "э", "е" и "о", "и" и "ы", "у" и "ю" были объединены в  $C = 5$  кластеров [25]. Тестирование точности распознавания проводила группа дикторов разного возраста, трое мужчин и две женщины (всего пять дикторов), каждый диктор произнес по две реализации каждого слова из словаря.

Вначале сопоставлялись оригинальный метод фонетического кодирования—декодирования (1)—(3) и его модификация (1), (3), (6) с предварительным выделением стационарных сегментов в слоге на основе правила сегментации (5). Исследовались два режима: дикторозависимый (каждый диктор изолированно произнес 10 гласных звуков, которые и

использовались в качестве его персональной ФБД) и дикторонезависимый (в качестве ФБД для всех дикторов применялось множество из 10 гласных, произнесенных одним из дикторов — мужчиной). Результаты распознавания для трех словарей ("Меню", "Лекарства", "Города") приведены в табл. 1 в следующем формате: усредненная по дикторам вероятность ошибки  $\pm$  ее среднее квадратичное отклонение.

Далее был проведен эксперимент детектирования команд в непрерывном потоке речи. В качестве исходных звуковых данных непрерывной речи использовали 200 фраз из базы данных ТИМТ [26], каждая из них длительностью 10...15 с. Минимальная длительность паузы между слогами команды установлена равной 0,8 с, максимальную длительность слога не ограничивали. Зависимость среднего числа выделенных слов от параметров  $\delta_0$  и числа слогов  $N$  показана в табл. 2. В табл. 3 сведены оценки вероятностей ошибки первого рода (ложная тревога — в непрерывной речи из базы ТИМТ была выделена команда) и второго рода (пропуск/неверное определение слогов команды в записанных дикторами сигналах из первого эксперимента для словаря "Меню") критерия выделения команд (7) с числом слогов  $N \geq 3$  из звукового потока, содержащего только речь и паузы между слогами (т. е. паузы в начале и конце фраз отсутствовали).

По результатам проведенного экспериментального исследования можно сделать следующие выводы. Во-первых, точность классификации (см. табл. 1) при использовании предварительной сегментации (5), (6) слогов оказалась выше, чем для оригинального метода фонетического кодирования—декодирования (1)—(3) как для дикторозависимого, так и для дикторонезависимого режимов, причем сразу для нескольких словарей. Во-вторых,

Таблица 1

Оценка вероятности ошибки APP, %

Режим	Система	Словарь		
		"Меню"	"Лекарства"	"Города"
Дикторозависимый	Распознавание слога целиком (1)—(3)	6,5 ± 4,2	5,6 ± 5,5	11,3 ± 6,4
	Предварительная сегментация слога (1), (3), (5), (6)	5,6 ± 4,1	4,3 ± 4	10 ± 6,5
Дикторонезависимый	Распознавание слога целиком (1)—(3)	9,3 ± 5,1	38,9 ± 10	26,7 ± 11
	Предварительная сегментация слога (1), (3), (5), (6)	6,5 ± 4,3	24 ± 9,5	20,8 ± 8,8

Таблица 2

Среднее число выделенных критерием (7) слов во фразе длительностью 5—15 с

Число слогов $N$	Порог $\delta_0$				
	0	0,1	0,2	0,3	0,4
$N = 1$	0,6	1,8	2,1	0,85	0,1
$N = 2$	1,3	1,4	1,1	0,35	0,05
$N \geq 3$	0,9	0,6	0,3	0,15	0,01

Таблица 3

Оценки вероятностей ошибки первого и второго рода при выделении из речи команд с числом слогов  $N \geq 3$  по критерию (7), %

Ошибка	Порог $\delta_0$				
	0	0,1	0,2	0,3	0,4
Первого рода	81	29	14	6	0,1
Второго рода	1,9	8,3	9,2	11,8	33,3

критерий (7) показал себя достаточно эффективным средством для выделения команд голосового управления с послоговым произношением из непрерывного потока речи, если число слогов в команде не слишком мало. Так (см. табл. 2), при  $\delta_0 = 0,3$  число детектированных слов с числом слогов  $N \geq 3$  составило в среднем 0,15 слова на одну фразу. И в-третьих, изменение порога  $\delta_0$  позволяет осуществить выбор между ошибкой пропуска голосовой команды и ошибкой принятия некоторой последовательности непрерывной речи как команды (см. табл. 3).

### Заключение

В результате проведенного исследования на основе вероятностной нейронной сети с проверкой однородности [14] предложен новый критерий (7) для выделения команд из речевого потока для систем голосового управления. Экспериментально продемонстрировано, что такой подход в сочетании с выражением (6) позволяет не только с достаточной степенью надежности детектировать голосовые команды (см. табл. 2, 3), но и повысить точность распознавания (см. табл. 1) по сравнению с оригинальным методом фонетического кодирования—декодирования (1)—(3).

Наиболее перспективным качеством с точки зрения практического применения предложенного подхода (1), (3), (5)—(7) является отсутствие запрета на использование в обычной речи слов/фраз из рабочего словаря. Действительно, для традиционных систем голосового управления появление в речи ключевой фразы приводит к автоматическому выполнению системой ассоциированного с ней действия. Если же искусственно ввести требование к четкому слоговому произношению команд [1], то у пользователя появляется возможность выбора: если система должна выполнить определенное действие, то ключевая фраза должна произноситься по слогам, в противном же случае она произносится в обычном темпе.

### Список литературы

1. **Савченко А. В.** Адаптивный алгоритм распознавания речи на основе метода фонетического декодирования слов в задаче голосового управления // Информационные технологии. 2013. № 4. С. 34—39.
2. **Tan B.** A Distributed Speech Remote Control System Based on Web Service and Automatic Speech Recognition, Electrical Power Systems on Computers // Lecture Notes in Electrical Engineering. 2011. Vol. 99. S. 771—778.
3. **Benesty J., Sondh M., Huang Y. (eds.)** Springer Handbook of Speech Recognition. New York: Springer, 2008. 1159 p.
4. **Pat. US 0026678 A1, Int. CI G10L21/00 (2006/01)** Intelligent automated assistant / Inventor: Gruber T., Cheyer A., Kittaus D., Guzzoni D., Brigham C., Giuli R., Bastea-Forte M., Saddler H., Assignee: Apple Inc., Pub. Date 19.01.2012.
5. **Du J., Liu P., Soong F., Zhou J.-L., Wang R.-H.,** Noisy Speech Recognition Performance of Discriminative HMMs // Chi-

nese Spoken Language Processing LNCS/LNAI. 2006. Vol. 4274. P. 358—369.

6. **Pat. US 8175883 B2, Int. CI G10L21/00 (2206.01)** Speech recognition system and method / Inventor: Grant R., Gregor P. Assignee: Nuance Communications Inc., Pub. Date 08.05.2012.
7. **Савченко В. В.** Решение проблемы ложных тревог в системах голосового управления на основе метода фонетического декодирования слов // Известия вузов России. Радиоэлектроника. 2012. Вып. 6. С. 7—15.
8. **Юсупов Р. М., Ронжин А. Л.** От умных приборов к интеллектуальному пространству // Вестник Российской академии наук. 2010. Т. 80, № 1. С. 45—51.
9. **Левин Е. К., Рагузин П. С., Татарникова М. Ю.** Оценка вероятности ложного срабатывания системы автоматического распознавания голосовых команд // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. 2011. № 5. С. 7—12.
10. **Козлов А. В., Саввина Г. В., Шелепов В. Ю.** Система фонемного распознавания отдельно произносимых слов // Искусственный интеллект. 2003. № 1. С. 156—165.
11. **Cerf-Danon H., Derouault A.-M., Elbeze M., Merialdo B.** Speech recognition in French with a very large dictionary // Proc. of EUROSpeech. 1989. P. 2150—2153.
12. **Цыплихин А. И., Сорокин В. Н.** Сегментация речи на кардинальные элементы // Информационные процессы. 2006. Т. 6, № 3. С. 177—207.
13. **Вишнякова О. А., Лавров Д. Н.** Автоматическая сегментация речевого сигнала на базе дискретного вейвлет-преобразования // Математические структуры и моделирование. 2011. Вып. 23. С. 43—48.
14. **Savchenko A. V.** Statistical Recognition of a Set of Patterns Using Novel Probability Neural Network, Proc. of International Workshop on Artificial Neural Networks and Pattern Recognition, LNCS/LNAI, 2012. Vol. 7477. P. 93—103.
15. **Savchenko A. V.** Adaptive segmentation of piecewise stationary stochastic processes based on the statistical testing for homogeneity // Proc. of XXXI International Seminar on Stability Problems for Stochastic Models. 2013. P. 59—61.
16. **Itakura F., Saito S.** An analysis-synthesis telephony based on the maximum likelihood method // Proc. of International Congress on Acoustics. 1968. P. —5-5, 17—20.
17. **Савченко В. В., Савченко А. В.** Метод фонетического декодирования слов в информационной метрике Кульбака—Лейблера для систем автоматического анализа и распознавания речи с повышенным быстродействием // Информационно-управляющие системы. 2013. № 2. С. 7—12.
18. **Qiao Y., Shimomura N., Minematsu N.** Unsupervised optimal phoneme segmentation: Objectives, algorithm and comparisons // Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing. 2008. P. 3989—3992.
19. **Боровков А. А.** Математическая статистика: дополнительные главы. М.: Наука, 1984.
20. **Савченко В. В.** Различение случайных сигналов в частотной области // Радиотехника и электроника. 1997. Т. 42, № 4. С. 426—431.
21. **Kullback S.** Information Theory and Statistics. New York: Dover Publications, 1997. 399 p.
22. **Basseville M.** Distance measures for signal processing and pattern recognition // Signal Processing. 1989. Vol. 18. P. 349—369.
23. **Савченко А. В.** Программный комплекс фонетического декодирования слов в информационной метрике Кульбака—Лейблера // Системы управления и информационные технологии. 2013. № 1(51). С. 21—28.
24. **Марпл С. Л.-мл.** Цифровой спектральный анализ и его приложения. М.: Мир, 1990. 584 с.
25. **Савченко А. В., Акатьев Д. Ю.** Методика формирования фонетической базы данных диктора из непрерывного потока разговорной речи на основе адаптивного алгоритма сегментации фонем и когнитивной кластерной модели речевых единиц // Информационные системы и технологии. 2013. № 3. С. 5—12.
26. **База данных TIMIT.** URL: <http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC93S1> (дата обращения: 23.05.2013).

Э. Д. Аведьян, д-р техн. наук, гл. науч. сотр., e-mail: avedian@mail.ru,  
 В. Э. Луганский, зам. директора, e-mail: lugansky@inevm.ru  
 ФГАНУ "Центр информационных технологий и систем органов исполнительной власти"  
 (ФГАНУ ЦИТиС), г. Москва

## Подход к задаче заполнения числовых пропусков в таблицах и строках, основанный на многослойной нейронной сети и нейронной сети СМАС<sup>1</sup>

*Проблема обнаружения и восстановления пропущенных данных сопутствует многим практическим задачам. Приводится краткий обзор современных работ по данной проблеме, в которых описаны возможные приложения и широкий спектр подходов к ее решению. Из этого обзора следует, что нейросетевые методы наиболее перспективны в решении задачи обнаружения и восстановления пропущенных данных. Показывается, что такая задача особенно эффективно решается с помощью нейронной сети СМАС (НС СМАС). Приведено ее краткое описание и алгоритм решения. Приведенные результаты компьютерного моделирования показывают, что НС СМАС решает задачу заполнения пропусков в таблицах с точностью, которая зависит от свойств восстанавливаемых данных.*

**Ключевые слова:** заполнение пропусков в таблицах и строках, многослойная нейронная сеть, нейронная сеть СМАС, алгоритм, компьютерное моделирование

E. D. Aved'yan, V. E. Lugansky

## An Approach to the Problem of Filling Gaps in Tables and Lines Based on the Multilayer and CMAC Neural Networks

*In this paper, is claimed that the problem of detection and filling gaps arises in many practical applications. A brief review of recent works on this problem is given. These works describe the possible applications and a wide range of approaches to their solution. From this review, it follows that the approach based on the multilayer neural network have great potential in solving this problem. We show that this problem is especially effectively solved using neural network CMAC. We give a brief description of this neural network and an algorithm for solving the problem of detection and recovery of missing data. The simulation results show that the neural network CMAC solves the problem of filling gaps in the table with the precision that depends on the properties of recovered data.*

**Keywords:** filling gaps in tables and lines, neural network CMAC, algorithm, simulation results

### Введение

Проблема обнаружения и восстановления пропущенных числовых данных сопутствует многим практическим задачам. Существует множество причин возникновения пропусков в обрабатываемых данных, среди которых можно назвать выход из строя регистрирующей аппаратуры, плохие условия наблюдений, например сильная облачность или туман, различные правила формирования данных

в разные периоды времени, наконец, просто потеря части данных. Чаще всего эта задача возникает при обработке одномерных процессов, заданных строкой данных, и при обработке двумерных процессов, представляющих, например, изображения фрагментов местности, заданных в виде таблицы данных. К таким задачам относят, в первую очередь, задачи обработки метеорологических наблюдений и спутниковой съемки местности. Эти задачи возникают также при обработке данных в геофизике, в жидкостной хроматографии — масс-спектрометрии, в социологических и экономических системах, при обработке двумерных изображений различных сцен и других данных. Интерес к этой тематике возник в прошлом веке и по-прежнему достаточно велик в настоящее время.

Ниже приведен краткий обзор работ по данной тематике, целью которого является демонстрация возможных приложений, в которых возникает та-

<sup>1</sup>Материалы данной статьи создавались в соответствии с основными системными подходами разрабатываемой в ФГАНУ ЦИТиС НИР "Исследование и разработка макета программного комплекса интеллектуальной системы поддержки принятия решений с использованием больших массивов показателей в среде единого информационного пространства социально-экономической сферы", финансируемой Минобрнауки России в рамках реализации Федеральной целевой программы "Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007—2013 годы".

кая задача, и широкий спектр подходов к ее решению. В приведенных ниже статьях имеется большое число ссылок на работы по тематике заполнения пропусков в таблицах и строках.

В работах [1—4] рассматриваются задачи восстановления пропущенных метеорологических данных и методы их решения. Так, в работе [1] для восстановления пропущенных данных, полученных с метеорологических станций северо-восточного региона Италии Венето, использован регрессионный подход. Здесь для восстановления пропусков применен динамический выбор данных. Авторы утверждают, что при правильном выборе настраиваемых параметров восстановление пропущенных данных выполняется с большой точностью. В работе [2] рассмотрено 15 современных методов восстановления пропущенных данных применительно к метеорологическим реальным европейским данным. Для выполнения сравнительного анализа здесь использованы различные сценарии создания искусственных пропущенных данных: очень короткие пропуски, короткие пропуски, средние пропуски, длинные пропуски и смешанные пропуски. Отмечается, что восстановление пропущенных данных с помощью трех нейросетевых методов оказалось лучшим, хотя и незначительно, среди всех исследованных методов. Здесь была использована стандартная искусственная нейронная сеть и искусственные нейронные сети с байесовской регуляризацией и с переквантованием и сглаживанием. Различные подходы для восстановления пропущенных метеорологических данных обсуждаются в работе [3], где для восстановления конкретных данных использован регрессионный подход. В качестве перспективного метода для улучшения оценки пропущенных данных авторы видят нейросетевой подход. В работе [4] применен калмановский подход для восстановления пропущенных метеорологических данных при оценке ежедневной, месячной и годовой потребности в воде. Полученные авторами результаты дают основание утверждать, что используемый ими подход следует рекомендовать для применения в задачах оценки объемов испаряемой воды, когда в регистрируемых данных имеются пропуски.

Работы [5, 6] посвящены задачам восстановления данных в двумерных изображениях. В работе [5] предложен эмпирический метод восстановления пропущенных данных спутниковой съемки местности, который авторы назвали интерполяционным методом окрестности подобных пикселей. В основе данного метода лежит пороговая процедура нахождения пикселей, подобных пропущенному пикселю. Подобные пиксели находятся в прямоугольной окрестности пропущенного пикселя. Оценка значения пропущенного пикселя выполняется линейным взвешиванием значений подобных пикселей с весами, для которых предложена вычислительная процедура. В заключение утверждается,

что предложенный метод решает задачу с высокой точностью. В работе [6] рассмотрена задача заполнения пропусков в двумерных изображениях. Восстановление пропущенных данных выполняется с помощью двумерного рекурсивного фильтра.

В работе [7] решена задача восстановления пропущенных данных в звуковых файлах. Предложенный здесь метод основан на применении спектрального анализа. В работе [8] описан метод восстановления пропущенных данных в жидкостной хроматографии — масс-спектрометрии. В основе предложенного подхода лежит интерполяционный метод ближайших данных. Отмечается, что, несмотря на простоту предложенного алгоритма, восстановление пропущенных данных выполняется с большой точностью. В работе [9] разработана и теоретически обоснована методика экспериментального сравнения эффективности различных алгоритмов заполнения пропусков в данных, которая апробирована на "типичных" социологических данных. Здесь приведен большой обзор работ российских и зарубежных ученых по заполнению пропусков в рядах и таблицах. В работе [10] использован метод аналогий для заполнения пропусков в таблице. В соответствии с этим методом выбирается ближайшая к пропущенному элементу строка или столбец, в которых нет пропущенных элементов. Оценкой пропущенного элемента является соответствующий элемент ближайшей строки или столбца. В работе [11] выполнен анализ моделей и методов, предназначенных для восстановления пропущенных данных. Предложен эволюционный метод, базирующийся на композиции использования нейронной сети и генетического алгоритма. Технология заполнения пропусков не требует выполнения ограничений, связанных с линейностью модели, распределением параметров и других. Применительно к задаче восстановления пропущенных данных в работе [12] проведено сравнение методов сглаженных сплайнов [13], сингулярного спектрального анализа [14] и метода Хока—Кэмпфера (метод главных компонент сглаженного спектрального анализа) [15] с предложенным в работе методом минимизации средней абсолютной ошибки между доступными данными и предлагаемой моделью этих данных. Сравнительный анализ методов проведен на реальных данных с модельными пропусками. Авторы подчеркивают, что выбор конкретного метода для оценки пропущенных данных сильно зависит от природы обрабатываемого сигнала и типа распределения пропусков.

Из анализа представленных результатов краткого обзора следует, что в настоящее время существует большое число методов решения задач заполнения пропусков в таблицах и рядах. Как правило, авторы этих методов подчеркивают, что их подходы дают лучшие результаты по сравнению с известными при решении конкретных задач обработки данных.

В ряде работ имеются утверждения, что нейросетевые методы наиболее перспективны в решении данной задачи.

Перечислим основные причины, порождающие большое число методов решения рассматриваемой задачи.

1. Эффективная обработка для таблицы и ряда в общем случае достигается различными методами.
2. Различные размерности обрабатываемых данных: число строк и столбцов и их длины.
3. Разный характер обрабатываемых данных: статистические свойства, природа возникновения данных и др.
4. Виды пропусков в обрабатываемых данных: отдельные пропуски или пропуски областей, статистические свойства пропусков и др.
5. Сюда можно отнести и субъективные причины, связанные с областью научных интересов авторов разрабатываемых методов.

Таким образом, в системе обработки данных с возможными пропусками должно быть несколько методов решения данной задачи. Среди этих методов обязательно должны быть и нейросетевые методы, о качественных возможностях которых говорилось выше в кратком обзоре. Нейросетевые методы базируются на различных вариантах многослойных нейронных сетей (МНС), которые после цикла обучения на непропущенных данных позволяют оценить значения пропущенных данных. Альтернативой для МНС [16] является нейронная сеть СМАС (НС СМАС), которая по своей структуре в большинстве случаев наилучшим образом подходит к задаче заполнения пропусков в таблицах и строках, поскольку: 1) входные переменные НС СМАС в силу своей природы принимают только целочисленные значения; 2) нейронная сеть СМАС обучается значительно быстрее МНС.

Следует учесть, что теоретически МНС может восстановить дифференцируемую нелинейную функцию многих переменных с любой заданной точностью [17], тогда как НС СМАС решает эту задачу только для ограниченного класса функций. Поэтому в ряде случаев, с точки зрения требования точности решения заполнения пропущенных данных, предпочтение может быть отдано МНС. Следовательно, в системе заполнения пропусков в таблицах, строках и более сложных образованиях должны присутствовать инструменты, базирующиеся как на НС СМАС, так и на МНС.

### **1. Нейросетевой подход к решению задачи заполнения пропусков числовых данных в таблицах и строках**

Эта задача относится к задаче оценивания функции  $y(x)$  в точке  $x$  по ее значениям, заданным в соседних точках, а именно к задаче интерполяции либо экстраполяции двумерных данных в случае с таблицами и одномерных данных в случае

с рядами. К особенностям рассматриваемой задачи относится тот факт, что аргументы  $x$  функции  $y(x)$  принимают целочисленные значения, а именно  $x = n, n = 1, \dots, N, y(n)$  — для рядов и  $x = (x_1, x_2), x_1 = n, n = 1, \dots, N, x_2 = m, m = 1, \dots, M, y(n, m)$  — для таблиц.

Следует различать два основных подхода к решению задачи заполнения таблиц и рядов с пропусками: подходы, основанные на локальной и глобальной интерполяции или экстраполяции. В первом подходе строится аппроксимирующая функция по значениям функции  $y(x)$ , которые лежат в окрестности пропущенной точки, и значение аппроксимирующей функции в пропущенной точке принимается за пропущенное значение. Второй подход основан на нахождении аппроксимирующей функции для всей таблицы или всего ряда по непропущенным значениям и вычислении пропущенных значений функции по значениям аппроксимирующей функции в пропущенных точках. Большое число методов заполнения пропусков в таблицах и строках порождено большим числом существующих методов аппроксимации функций, которые, в свою очередь, порождены видом аппроксимирующих функций (степенные, тригонометрические, специально подобранные) и функционалом приближения аппроксимируемой и аппроксимирующей функций.

В качестве аппроксимирующих функций, как отмечалось выше, могут быть использованы и обучаемые искусственные нейронные сети. Основной алгоритм заполнения пропусков в таблицах и строках с помощью нейронных сетей состоит из следующих шагов.

*Шаг 1.* Анализ таблицы или строки в целях принятия решения о локальном или глобальном подходе к задаче заполнения пропущенных данных.

*Шаг 2.* Анализ таблицы или строки в целях автоматического выделения координат пропущенных и непропущенных данных.

*Шаг 3.* Формирование обучающей выборки для нейронной сети из непропущенных данных  $\{X_{exist}, Y_{exist}\}$  — множество, состоящее из координат непропущенных данных  $X_{exist}$  и соответствующих им значений из таблицы или строки  $Y_{exist}$ . Формирование множества координат пропущенных данных  $X_{gap}$ .

*Шаг 4.* Обучение нейронной сети с помощью части множества непропущенных данных  $\{X_{exist}, Y_{exist}\}$ , в котором удалена первая точка непропущенных данных  $x_{exist}(1), y_{exist}(1)$ . Вычисление оценки  $\tilde{y}_{exist}(1)$  непропущенного значения  $y_{exist}(1)$  с помощью обученной нейронной сети и абсолютного значения относительной ошибки оценки  $\varepsilon_{exist}(1) = abs(y_{exist}(1) - \tilde{y}_{exist}(1)/y_{exist}(1))$ .

**Шаг 5.** Повторение шага 4 при последовательном удалении либо всех, либо части непропущенных данных с вычислением оценок удаленных данных и абсолютных значений относительных ошибок их оценивания. Вычисление усредненной ошибки заполнения пропусков в таблице или строке по вычисленным абсолютным значениям относительных ошибок.

**Шаг 6.** Если оценка ошибки заполнения пропусков удовлетворяет требованиям точности заполнения, то провести обучение на всем множестве непропущенных данных и вычислить оценки пропущенных данных последовательной подачей на вход обученной нейронной сети координат пропущенных данных  $X_{gap}$  и заполнить пропуски вычисленными данными.

**Шаг 7.** Если оценка заполнения пропусков не удовлетворяет требованиям точности заполнения, то перейти к другой конфигурации нейронной сети и для нее повторить вычисления шагов 4–6, определив для новой конфигурации точность заполнения.

**Шаг 8.** Шаг 7 повторять до тех пор, пока либо будет достигнута требуемая точность заполнения пропусков, либо будет выбран вариант конфигурации сети с минимально достигнутым абсолютным значением относительной ошибки заполнения из всех исследованных вариантов.

**Шаг 9.** После выбора оптимального варианта конфигурации сети провести ее обучение на всем множестве непропущенных данных  $\{X_{exst}, Y_{exst}\}$  и по множеству координат пропущенных данных  $X_{gap}$  оценить значения пропущенных данных и внести их в таблицу.

Так вкратце выглядит алгоритм заполнения пропусков данных в таблицах и строках, основанный на нейросетевом подходе. Его реализация затруднена для МНС в силу большого числа вариантов возможных конфигураций МНС: число слоев, число нейронов в каждом слое, вид функции активации, связность между слоями. С этой точки зрения НС СМАС имеет преимущества в силу своей относительно малой вариантности по сравнению с МНС. В настоящей работе программная реализация алгоритма нейросетевого заполнения пропусков в таблицах и строках выполнена на НС СМАС. Поэтому, прежде чем перейти к описанию этого алгоритма заполнения пропусков в таблицах и строках, приведем краткое описание структуры и алгоритма функционирования НС СМАС независимо от решаемой ею задачи.

## 2. Структура и алгоритм функционирования нейронной сети СМАС

Нейронная сеть СМАС берет свое название от первых букв ее английского названия — *Cerebellar Model Articulation Controller* (мозжечковая модель суставного регулятора) [18, 19]. В основу этой ней-

ронной сети положена нейрофизиологическая модель мозжечка для управления роботом-манипулятором. Подробное описание нейронной сети СМАС можно найти в статьях [20, 21]. НС СМАС находит применение в самых различных областях. Далеко не полное перечисление этих областей приведено в работе [22]. Реализация НС СМАС для решения задачи распознавания образов описана в работе [23].

Основное назначение нейронных сетей — запоминание и воспроизведение функций многих переменных. Наиболее существенными отличиями НС СМАС от других нейронных сетей являются следующие:

- аргументы запоминаемой и воспроизводимой функций принимают только дискретные значения;
- нелинейное преобразование аргументов функции происходит неявно с помощью алгоритма вычисления адресов ячеек ассоциативной памяти, в которых хранятся числа, определяющие значение функции.

Областью определения функции  $N$  переменных  $u(x)$  является целочисленная  $N$ -мерная сетка:

$$X = \{x^{(1)} = \overline{1, x_{\max}^{(1)}}; x^{(2)} = \overline{1, x_{\max}^{(2)}}; \dots; x^{(N)} = \overline{1, x_{\max}^{(N)}}\}. \quad (1)$$

В НС СМАС каждый входной  $N$ -мерный сигнал  $x$  (аргумент функции) возбуждает, или делает активными, ровно  $\rho^*$  ячеек памяти, суммарное содержимое которых равно значению запоминаемой функции. Каждому входному  $N$ -мерному вектору  $x$  однозначно соответствует  $\rho^*$ -мерный вектор активных номеров ячеек памяти  $m$ . Параметр  $\rho^*$  (обобщающий параметр) играет очень важную роль, его значение определяет разрешающую способность НС СМАС и требуемый объем памяти нейронной сети. Алгоритм вычисления номеров активных ячеек памяти устроен так, что чем ближе друг к другу расположены входные вектора, тем больше у них общих ячеек памяти — именно это и придает НС СМАС интерполяционные свойства. Алгоритм вычисления номеров активных ячеек памяти здесь не приводится, поскольку его выражение можно найти в русскоязычных работах [20, 21, 23], а вывод — в работе [22].

Значение запомненной в НС СМАС функции  $\tilde{u}(x)$  равно сумме содержимого активных ячеек памяти. Введем следующие переменные:  $w[n]$  —  $M$ -мерный вектор памяти сети, вычисленный на  $n$ -м шаге обучения, каждая  $j$ -я компонента которого  $w_j[n]$ ,  $j = \overline{1, M}$ , соответствует содержимому  $j$ -й ячейки памяти НС СМАС;  $a(x)$  —  $M$ -мерный ассоциативный вектор, однозначно связанный с вектором аргументов  $x$  посредством вектора  $m$  номеров активных ячеек памяти по следующему правилу: элементы вектора  $a_j$ ,  $j = \overline{1, M}$ , номера которых сов-

падают с номерами активных ячеек памяти, равны единице, все остальные элементы вектора  $a$  равны нулю. Тогда в соответствии с правилом функционирования нейронной сети НС СМАС ее выход  $\tilde{y}(x[n])$  при заданном входном векторе  $x[n]$  равен скалярному произведению векторов  $a(x[n])$  и  $w[n]$ :

$$\tilde{y}(x[n]) = a^T(x)w(n), \quad (2)$$

т. е. выход равен сумме содержимого активных ячеек памяти сети.

Алгоритм обучения нейронной сети НС СМАС функционирует следующим образом.

Пусть после  $(n - 1)$ -го измерения значения запоминаемой функции  $y(x)$  и соответствующих значений вектора аргументов  $x$ ,  $(y(x[i]), x[i]), i = \overline{1, n-1})$ , был вычислен вектор памяти  $w[n - 1]$ . Тогда на следующем  $n$ -м шаге после измерения значения функции  $y[n] \equiv y(x[n])$  при известном значении аргумента  $x[n]$  сначала с помощью алгоритма нелинейного преобразования аргументов вычисляются номера активных ячеек памяти, далее вычисляется предсказываемое нейронной сетью значение функции  $\tilde{y}[n] \equiv \tilde{y}(x[n])$ , равное сумме содержимого активных ячеек памяти. Вычисляются ошибка предсказания  $\varepsilon[n] = y[n] - \tilde{y}[n]$  и значение коррекции  $\Delta w[k] = \gamma^* \varepsilon[n] / \rho^*$ ,  $0 < \gamma < 2$ , которое прибавляется к содержимому активных ячеек памяти. Неактивные ячейки коррекции не подвергаются. Аналитическая форма алгоритма обучения имеет следующий вид:

$$w[n] = w[n - 1] + \gamma^* \frac{y[n] - a^T(x[n])w[n - 1]}{a^T(x[n])a(x[n])} a(x[n]), \quad (3)$$

$$w[0] = w_0, \quad n = 1, 2, \dots$$

Обученная нейронная сеть НС СМАС способна предсказывать значения функции, которые ей не были предъявлены при обучении. Для этого необходимо на вход сети подать вектор аргументов запрашиваемого значения функции, нейронная сеть по данному вектору вычислит адреса активных ячеек обученной памяти, сумма содержимого которых и будет значением предсказываемой функции.

### 3. Программная реализация алгоритма заполнения пропусков в таблицах и строках с помощью НС СМАС

Рассмотрим, как каждый из шагов нейросетевого алгоритма решения задачи заполнения пропусков числовых данных в таблицах и строках (разд. 1) интерпретируется применительно к НС СМАС.

*Шаг 1. Анализ таблицы или строки в целях принятия решения о локальном или глобальном подходе к задаче заполнения пропущенных данных.*

Этот шаг непосредственно не связан с парадигмой нейронной сети. Его реализация предполагает применение всех перечисленных ниже шагов 2—9.

При глобальном подходе для таблицы или строки выполняется процедура заполнения пропусков, вычисляется оценка ошибки заполнения и принимается решение о допустимости решения задачи. При локальном подходе таблица или строка разбивается на ряд подтаблиц или подстрок, для каждой из которых выполняется процедура заполнения пропусков, вычисляется оценка ошибки заполнения для каждой из подтаблиц или подстрок, общая оценка ошибки, полученные оценки сравниваются с глобальным подходом и принимается окончательное решение. Эта процедура ресурсоемкая, она существенно зависит от характера пропусков в таблице и требует отдельного исследования. Далее предполагается, что решение о подходе принято, и фактически необходимо уметь работать с отдельной таблицей или строкой, независимо от того, таблица это или подтаблица, или строка это или подстрока.

*Шаг 2. Анализ строки в целях автоматического выделения координат пропущенных и непропущенных данных.*

Программа, выполняющая анализ таблицы или строки, зависит от способа организации данных и пропусков в таблице или строке. Способов организации данных и пропусков много, так что каждый способ порождает свою программу анализа. В настоящей работе принята следующая организация таблицы или строки.

Числовая таблица или строка представляет собой текстовый файл, в конце которого находится признак конца файла. Каждая таблица состоит из строк, в конце каждой из которых имеется признак конца строки. Предполагается, что таблица имеет  $N$  строк и  $M$  столбцов —  $N \times M$ -таблица (в последующем строка является таблицей с одной строкой —  $1 \times M$ -таблица). Столбцы строки разделены символами табуляции, так что число символов табуляции в строке всегда равно  $M - 1$ . Числа, составляющие строку данных, записываются между символами табуляции. Если в первом столбце строки имеется число, то оно записывается перед первым символом табуляции в строке. Отсутствие числа между  $(k - 1)$ -м и  $k$ -м символами табуляции,  $k = \overline{2, M - 1}$ , означает наличие пропуска в  $k$ -м столбце анализируемой строки. Если строка начинается с символа табуляции, это означает, что имеет место пропуск в первом столбце, если же строка кончается символом табуляции, это означает, что имеет место пропуск в последнем столбце.

Результаты функционирования алгоритма будут иллюстрироваться на примере обработки табл. 1, содержащей 14 строк и 12 столбцов. В табл. 1 имеется 30 пропусков, что составляет около 18 % всех элементов таблицы. Обработка таблицы ведется

Таблица 14x12, содержащая 30 пропусков (около 18% всех элементов таблицы)

11.860	15.801		23.683	27.624	31.565		39.447	43.388	47.329	51.270	
13.934	17.875	21.816		29.698	33.639	37.580	41.521	45.462	49.403	53.344	57.285
	19.949	23.890	27.831	31.772	35.713	39.654	43.595		51.477	55.418	59.359
18.082	22.023		29.905	33.846	37.787	41.728	45.669	49.610	53.551	57.492	
20.156	24.097	28.038	31.979	35.920	39.861		47.743	51.684		59.566	63.507
22.230	26.171	30.112			41.935	45.876		53.758	57.699	61.640	65.581
	28.245	32.186	36.127	40.068	44.009	47.950	51.891	55.832	59.773		67.655
11.860	15.801		23.683	27.624	31.565		39.447	43.388	47.329	51.270	
13.934	17.875	21.816		29.698	33.639	37.580	41.521	45.462	49.403	53.344	57.285
	19.949	23.890	27.831	31.772	35.713	39.654	43.595		51.477	55.418	59.359
18.082	22.023		29.905	33.846	37.787	41.728	45.669	49.610	53.551	57.492	
20.156	24.097	28.038	31.979	35.920	39.861		47.743	51.684		59.566	63.507
22.230	26.171	30.112			41.935	45.876		53.758	57.699	61.640	65.581
	28.245	32.186	36.127	40.068	44.009	47.950	51.891	55.832	59.773		67.655

построчно начиная с первой строки. Программа заполнения пропусков реализована на языке *Object Pascal*. Первая строка считывается тремя способами: как строка, как последовательность символов и как последовательность чисел. Эта информация запоминается и подвергается анализу для обнаружения пропусков и числовых данных. В табл. 2 представлена печать первой запомненной строки как строки, как последовательности числовых данных и как последовательности символов. Отметим, что рамки табл. 1 и 2 не являются результатами печати, а введены после выполнения программы для наглядности изложения. Программа, выполняющая анализ первой строки символов, при обнаружении двух следующих друг за другом кодов символов табуляции (код 9) определяет номер столбца, в котором имеется пропуск. Нетрудно видеть, что в соответствии с правилом формирования таблицы программа обнаружит пропуски в 3-м, 7-м и 12-м столбцах первой строки, т. е. будут сформированы координаты пропусков первой строки табл. 1:

$$X_{gap}(1) = (1,3), (1,7), (1,12). \tag{4}$$

Из остальных девяти столбцов будут сформированы координаты непропущенных данных первой строки и соответствующие им значения в виде:

$$\{X_{exist}(1), Y_{exist}(1)\} = \{x = (1, 1), (1,2), (1,4), (1,5), (1,6), (1,8), (1,9), (1,10), (1,11); y = (11.860, 15.801, 23.683, 27.624, 31.565, 39.447, 43.388, 47.329, 51.270)\}. \tag{5}$$

**Шаг 3. Формирование обучающей выборки для нейронной сети из непропущенных данных  $\{X_{exist}, Y_{exist}\}$  — множество, состоящее из координат непропущенных данных  $X_{exist}$  и соответствующих им значений из**

**таблицы или строки  $Y_{exist}$ . Формирование множества координат пропущенных данных  $X_{gap}$ .**

Применение описанной в шаге 2 процедуры ко всем последующим строкам таблицы позволяет сформировать множество непропущенных данных  $\{X_{exist}, Y_{exist}\}$  из данных вида (5) и множество координат пропущенных данных  $X_{gap}$  из данных вида (4). Для табл. 1 множество непропущенных данных состоит из 138 пар данных (координата и значение таблицы в этой координате), а множество координат пропущенных данных состоит из 30 элементов.

**Шаг 4. Обучение нейронной сети с помощью части множества непропущенных данных  $\{X_{exist}, Y_{exist}\}$ , в котором удалена первая точка непропущенных данных  $x_{exist}(1), y_{exist}(1)$ . Вычисление оценки  $\tilde{y}_{exist}(1)$  непропущенного значения  $y_{exist}(1)$  с помощью обученной нейронной сети и абсолютного значения относительной ошибки оценки**

$$\varepsilon_{exist}(1) = abs(y_{exist}(1) - \tilde{y}_{exist}(1)/y_{exist}(1)).$$

Задается квадратная область определения аргументов (1), на которой происходит обучение и которая покрывает всю таблицу:

$$X = \{x^{(1)} = \overline{1, x_{max}^{(1)}}; x^{(2)} = \overline{1, x_{max}^{(2)}}\},$$

Таблица 2

Результаты печати первой строки

Строка																							
11.860	15.801		23.683	27.624	31.565		39.447	43.388	47.329	51.270													
Строка чисел																							
11.860	15.801	23.683	27.624	31.565	39.447	43.388	47.329	51.270															
Строка кодов символов (65 символов)																							
49	49	46	56	54	48	9	49	53	46	56	48	49	9	9	50	51	46	54	56	51	9	50	55
46	54	50	52	9	51	49	46	53	54	53	9	9	51	57	46	52	52	55	9	52	51	46	51
56	56	9	52	55	46	51	50	57	9	53	49	46	50	55	48	9							

где

$$x_{\max}^{(1)} = x_{\max}^{(2)} = 2^k + 1, \\ k = \text{Trunc}(\log_2(\text{Max}(N, M))) + 1. \quad (6)$$

В выражении (6)  $\text{Max}$  — наибольшее из двух чисел ( $N$  — число строк,  $M$  — число столбцов),  $\text{Trunc}(a)$  — целая часть числа  $a$ . Для табл. 1  $x_{\max}^{(1)} = x_{\max}^{(2)} = 17$ . Задается длина обучающей последовательности в виде

$$N_{\text{learn}} = N \times M \times N_{\text{cycle}}, \quad (7)$$

где число циклов обучения  $N_{\text{cycle}} \geq 4$ . При  $N_{\text{cycle}} = 4$  получается приемлемая точность заполнения, с увеличением числа циклов обучения  $N_{\text{cycle}}$  точность заполнения обычно повышается. Оценка времени обучения следует из результатов, представленных в работе [24]. Параметр  $\rho^*$  имеет следующий вид:

$$\rho^* = 2^m, \quad m = 1, \dots, k, \quad (8)$$

где для рассматриваемой табл. 1 параметр  $k$  вычисляется в соответствии с выражением (6) ( $k = 4$ ).

Обучение начинается при значении параметра  $\rho^* = 2$  с помощью выборки  $\{X_{\text{exist}}, Y_{\text{exist}}\}$ , в которой удалена первая точка обучения  $x_{\text{exist}}(1)$ ,  $y_{\text{exist}}(1)$ . Из данной укороченной выборки случайным образом извлекаются точки, по которым в соответствии с алгоритмом (3) происходит обучение (корректировка вектора памяти) НС СМАС. После завершения процесса обучения вычисляется оценка удаленной первой точки  $\hat{y}_{\text{exist}}(1)$  в соответствии с формулой (2) и абсолютное значение относительной ошибки.

**Шаг 5.** Повторение шага 4 при последовательном удалении либо всех, либо части непропущенных данных с вычислением оценок удаленных данных и абсолютных значений относительных ошибок их оценивания. Вычисление усредненной ошибки заполнения пропусков в таблице или строке по вычисленным абсолютным значениям относительных ошибок.

Реализация этого шага позволяет вычислить и запомнить абсолютные значения относительных ошибок для всех удаляемых из процесса обучения данных и вычислить усредненную оценку ошибки заполнения таблицы в целом:

$$E_{\text{indirct}}(\rho^*) = \sum_{i=1}^{N_{\text{exist}}} \varepsilon_{\text{exist}}(i) / N_{\text{exist}} \quad (9)$$

**Шаг 6.** Если оценка ошибки заполнения пропусков удовлетворяет требованиям точности заполнения, то провести обучение на всем множестве непропущенных данных и вычислить оценки пропущенных данных последовательной подачей на вход обученной нейронной сети координат пропущенных данных  $X_{\text{gap}}$  и заполнить пропуски вычисленными данными.

Реализация этого шага осуществляется последовательной подачей на вход обученной НС СМАС

координат пропущенных данных  $X_{\text{gap}}$ , вычислением оценок пропущенных данных в соответствии с формулой (2) и занесением их значений в пропущенные координаты таблицы.

**Шаг 7.** Если оценка заполнения пропусков не удовлетворяет требованиям точности заполнения, то перейти к другой конфигурации нейронной сети и для нее повторить вычисления шагов 4, 5, определив для новой конфигурации точность заполнения.

Для НС СМАС переход к новой конфигурации означает выбор нового значения обобщающего параметра  $\rho^* = 4$  в соответствии с формулой (8) при  $m = 2$  и вычисления усредненной ошибки заполнения  $E_{\text{indirct}}(\rho^*)$ .

**Шаг 8.** Шаг 7 повторять до тех пор, пока будет либо достигнута требуемая точность заполнения пропусков, либо выбран вариант конфигурации сети с минимально достигнутым значением усредненной ошибки заполнения из всех исследованных вариантов.

В результате выполнения шага 8 будут проведены расчеты усредненных ошибок для всех возможных значений  $\rho^* = 2, 4, 8, 16$  и выбор оптимального значения  $\rho^*$ .

**Шаг 9.** После выбора оптимального варианта конфигурации сети провести ее обучение на всем множестве непропущенных данных  $\{X_{\text{exist}}, Y_{\text{exist}}\}$  и по множеству координат пропущенных данных  $X_{\text{gap}}$  оценить значения пропущенных данных и внести их в таблицу.

Комментарии к этому шагу отражены в предыдущих шагах.

#### 4. Результаты экспериментальных исследований

В этом разделе представлены результаты заполнения пробелов в двух модельных таблицах (табл. 1 и 5).

В первом эксперименте представлены результаты заполнения пробелов табл. 1, для которой исходной послужила табл. 3 без пробелов. Табл. 3 состоит из двух одинаковых подтаблиц: верхняя половина табл. 3 (строки 1—7) полностью повторяет ее нижнюю часть (строки 8—14). Каждая подтаблица представляет числовые значения куска наклонной плоскости. Разрыв между плоскостями проходит между строками 7 и 8. Первая подтаблица, в которой случайным образом удалены 15 элементов, составила как верхнюю, так и нижнюю часть табл. 1. Поскольку исходная табл. 3 без пропусков доступна, то программа, кроме восстановления пропущенных данных и вычисления описанной выше косвенной оценки ошибки заполнения  $E_{\text{indirct}}(\rho^*)$  (9) пропусков, вычисляла и точную ошибку заполнения  $E_{\text{dirct}}(\rho^*)$  в виде усредненных абсолютных значений относительных разностей между доступными пропущенными данными и их оценками. Последнее возможно только в модельных экспериментах. В соответствии с шагом 8 алгоритма определено оптимальное значение параметра  $\rho^* = 16$ .

Исходная таблица, соответствующая табл. 1

11.860	15.801	19.742	23.683	27.624	31.565	35.506	39.447	43.388	47.329	51.270	55.211
13.934	17.875	21.816	25.757	29.698	33.639	37.580	41.521	45.462	49.403	53.344	57.285
16.008	19.949	23.890	27.831	31.772	35.713	39.654	43.595	47.536	51.477	55.418	59.359
18.082	22.023	25.964	29.905	33.846	37.787	41.728	45.669	49.610	53.551	57.492	61.433
20.156	24.097	28.038	31.979	35.920	39.861	43.802	47.743	51.684	55.625	59.566	63.507
22.230	26.171	30.112	34.053	37.994	41.935	45.876	49.817	53.758	57.699	61.640	65.581
24.304	28.245	32.186	36.127	40.068	44.009	47.950	51.891	55.832	59.773	63.714	67.655
11.860	15.801	19.742	23.683	27.624	31.565	35.506	39.447	43.388	47.329	51.270	55.211
13.934	17.875	21.816	25.757	29.698	33.639	37.580	41.521	45.462	49.403	53.344	57.285
16.008	19.949	23.890	27.831	31.772	35.713	39.654	43.595	47.536	51.477	55.418	59.359
18.082	22.023	25.964	29.905	33.846	37.787	41.728	45.669	49.610	53.551	57.492	61.433
20.156	24.097	28.038	31.979	35.920	39.861	43.802	47.743	51.684	55.625	59.566	63.507
22.230	26.171	30.112	34.053	37.994	41.935	45.876	49.817	53.758	57.699	61.640	65.581
24.304	28.245	32.186	36.127	40.068	44.009	47.950	51.891	55.832	59.773	63.714	67.655

Таблица 4

Зависимость ошибки заполнения табл. 1 от числа циклов и длины обучающей последовательности, %

$N_{cycle}$	1	2	3	4	5	6	7	8	9	10	50	100
$N_{learn}$	168	336	504	672	840	1008	1176	1334	1512	1680	8400	16800
$E_{indirect}$	6.299	4.727	3.659	3.281	2.958	2.279	2.309	2.123	1.909	1.755	0.524	0.320
$E_{direct}$	9.682	3.127	2.692	2.875	1.779	1.014	1.696	0.709	1.216	0.634	0.005	0.000

Таблица 5

Таблица 14 × 12, содержащая 30 пропусков (около 18% всех элементов таблицы) и соответствующая формуле (10)

3.320	3.749			4.811	5.116	5.404	5.677	5.938	6.188	6.428	
4.124	4.708	5.265		6.310	6.803	7.277	7.732	8.168	8.587	8.986	9.367
	5.563	6.303	7.009	7.671	8.284	8.840	9.333	9.758		10.391	10.596
5.382	6.346		8.102	8.834	9.435	9.889	10.184	10.317	10.292	10.122	
5.912	7.076	8.150	9.060	9.746	10.169		10.178	9.802		8.540	7.801
6.397	7.763	8.963			10.451	10.112		8.535	7.568	6.693	6.056
	8.412	9.702	10.506	10.705	10.301	9.413	8.253	7.090	6.189		5.935
3.320	3.749			4.811	5.116	5.404	5.677	5.938	6.188	6.428	
4.124	4.708	5.265		6.310	6.803	7.277	7.732	8.168	8.587	8.986	9.367
	5.563	6.303	7.009	7.671	8.284	8.840	9.333	9.758		10.391	10.596
5.382	6.346		8.102	8.834	9.435	9.889	10.184	10.317	10.292	10.122	
5.912	7.076	8.150	9.060	9.746	10.169		10.178	9.802		8.540	7.801
6.397	7.763	8.963			10.451	10.112		8.535	7.568	6.693	6.056
	8.412	9.702	10.506	10.705	10.301	9.413	8.253	7.090	6.189		5.935

В табл. 4 представлена зависимость косвенной и точной ошибок заполнения от числа циклов обучения  $N_{cycle}$  и длины обучающей последовательности  $N_{learn}$  (7). Из табл. 4 следует, что использование косвенной оценки ошибки заполнения таблицы правомерно, поскольку косвенные и точные оценки ошибки заполнения таблицы достаточно близки. С увеличением числа циклов обучения ошибки заполнения убывают и при относительно большом числе циклов обучения, больших 50, достигается практически точное заполнение пропусков табл. 1. Этот относительно неожиданный результат связан со свойством НС СМАС точно восстанавливать линейные многомерные функции.

В нелинейных случаях ожидать точного восстановления пропусков в таблице нельзя. Иллюстрацией тому являются результаты восстановления пропусков в табл. 5, которая по своей структуре подобна табл. 1. Данные табл. 5, в отличие от табл. 1, с кусочно-линейными данными, однако представляют значения сложной нелинейной функции двух аргументов:

$$y(n, m) = \sqrt{5,981(n-1) + 3,03(m-1) + 11,023} + 3,1\sin(2\pi(n-1)(m-1)/(NM)),$$

$$n = 1, \dots, 7; m = 1, \dots, 12. \quad (10)$$

Зависимость ошибки заполнения пропусков в табл. 5 от числа циклов и длины обучающей последовательности, %

$N_{cycle}$	1	2	3	4	5	6	7	8	9	10	50	100
$N_{learn}$	168	336	504	672	840	1008	1176	1334	1512	1680	8400	16800
$E_{indirect}$	11.023	10.265	8.842	8.285	8.076	7.547	7.104	7.268	7.041	6.888	5.970	5.868
$E_{direct}$	13.568	10.578	11.167	8.021	10.432	9.829	10.766	8.096	9.541	10.530	7.572	9.705

Оптимальное значение параметра  $\rho^*$  равно 4. В табл. 6, которая по структуре совпадает с табл. 4, представлена зависимость косвенной и точной ошибки заполнения для примера (10). Точности заполнения пробелов в силу нелинейности данных ниже, чем в случае табл. 1. Более того, из табл. 6 также следует, что процесс обучения сходится не к точке, а к области решения и довольно быстро. Последнее означает, что процесс обучения в таких случаях должен быть дополнен обучением по накопленным данным так, как это описано в работе [25].

### Заключение

Результаты, изложенные в настоящей работе, подтверждают вывод о том, что нейросетевой подход к решению задачи заполнения пропусков в таблицах и строках является эффективным инструментом решения данной задачи. Тем не менее он должен рассматриваться не как альтернатива существующим методам, а как дополнение к ним. Повышение точности решения данной задачи с помощью НС СМАС будет обеспечено двумя режимами ее обучения: режимом *on-line* и режимом обучения по накопленным данным.

### Список литературы

1. Tardivo G., Berti A. A Dynamic Method for Gap Filling in Daily Temperature Datasets // *J. Appl. Meteor. Climatol.* 2012. N 51. P. 1079–1086. doi: <http://dx.doi.org/10.1175/JAMC-D-11-0117.1>.
2. Moffat A. M., Papale D., Reichstein M., Hollinger D. Y., Richardson A. D., Barr A. G., Beckstein C., Braswell B. H., Churkina G., Desai A. R., Falge E., Gove J. H., Heimann M., Hui D., Jarvis A. J., Kattge J., Noormets A., Stauch V. J. Comprehensive comparison of gap-filling techniques for eddy covariance net carbon fluxes // *Agricultural and Forest Meteorology.* 2007. V. 147. P. 209–232.
3. Kotsiantis S., Kostoulas A., Lykoudis S., Argiriou A., Menagias K. Filling missing temperature values in weather data banks // *Proc. of 2nd IEEE International Conference on Intelligent Environments.* 5–6 July. 2006. Athens. Greece. V. 1. P. 327–334.
4. Alavi N., Warland J. S., Berg A. A. Filling gaps in evapotranspiration measurements for water budget studies: Evaluation of a Kalman filtering approach // *Agricultural and Forest Meteorology.* 2006. V. 141. P. 57–66.
5. Chen J., Zhu X., Vogelmann J. E., Gao F., Jin S. A simple and effective method for filling gaps in Landsat ETM + SLC-off images // *Remote Sensing of Environment.* 2011. V. 115. P. 1053–1064.
6. Stephen J. D., Marschner R., Garr M., Levoy M. Filling Holes in Complex Surfaces using Volumetric Diffusion // *1st International Symposium on 3D Data Processing Visualization and Transmission (3DPVT 2002).* 19–21 June 2002. Padova, Italy. 2002.
7. Drori I., Fishbach A., Yeshurun Y. Special Sound Gap Filling. URL: <http://www.math.tau.ac.il/~hezy/papers/c32.pdf>.
8. Benton H. P., Want E. J., Ebbels T. M. D. Correction of mass calibration gaps in liquid chromatography-mass spectrometry metabolomics data // *Oxford Journals, Life Sciences Mathematics & Physical Sciences. Bioinformatics.* 2010. V. 26. Is. 19. P. 2488–2489.
9. Зангиева И. К. Сравнительный анализ алгоритмов заполнения пропусков в социологических данных: автореферат диссертации кандидата социологических наук: 22.00.01 // М.: Изд-во Национального исследовательского университета "Высшая школа экономики", 2012. С. 27.
10. Васильев В. И., Шевченко А. И. Восстановление пропущенных данных в эмпирических таблицах // *Искусственный интеллект.* 2003. № 3. С. 317–324.
11. Снитюк В. Е. Эволюционный метод восстановления пропусков в данных // *Сб. трудов VI-й Международной конференции "Интеллектуальный анализ информации".* Киев, 2006. С. 262–271.
12. Musial J. P., Verstraete M. M., Gorbon N. Comparing the effectiveness of recent algorithms to fill and smooth incomplete and noise time series // *Atmospheric Chemistry and Physics Discussion Discuss.* 2011. N 11. OP. 14259–14308. URL: <http://www.atmos-chem-phys-discuss.net/11/14259/2011/doi:10.5194/acpd-11-14259-2011>.
13. Hutchinson M. F., de Hoog F. R. Smoothing Noisy Data with Spline Functions // *Numer. Math.* 1985. 47. P. 99–106.
14. Kondrashov D., Ghil M. Spatio-temporal filling of missing points in geophysical data sets // *Nonlin. Processes Geophys.* 2006. V. 13. P. 151–159. doi: 10.5194/npg-13-151-2006.
15. Hocke K., Kampfer N. Gap filling and noise reduction of unevenly sampled data by means of the Lomb-Scargle periodogram // *Atmos. Chem. Phys.* 2009. V. 9. P. 4197–4206, doi: 10.5194/acp-9-4197-2009.
16. Miller W. T., Glanz F. H., Kraft L. G. СМАС: An associative neural network alternative to backpropagation // *Proc. of the IEEE.* 1990. V. 78, N 10. P. 1561–1567.
17. Колмогоров А. Н. О представлении непрерывных функций несколькими переменных в виде суперпозиции непрерывных функций одного переменного // *Доклады АН СССР.* 1957. Т. 114. № 5. С. 953–956.
18. Albus J. S. A new approach to manipulation control: the cerebellar model articulation controller // *ASME Trans., J. Dynamic Systems. Measurement and Control.* 1975. V. 97, N 3. P. 220–227.
19. Albus J. S. Data storage in the cerebellar model articulation controller (СМАС) // *ASME Trans., J. Dynamic Systems. Measurement and Control.* 1975. V. 97, N 3. P. 228–233.
20. Аведьян Э. Д. Ассоциативная нейронная сеть СМАС. Ч. I. Структура, объем памяти, обучение и базисные функции // *Информационные технологии.* 1997. № 5. С. 6–14.
21. Аведьян Э. Д. Ассоциативная нейронная сеть СМАС. Ч. II. Процессы обучения, ускоренное обучение, влияние помех, ускорение влияния помех в двухслойной сети // *Информационные технологии.* 1997. № 6. С. 16–27.
22. Аведьян Э. Д., Пантюхин Д. В. Алгоритм нелинейного преобразования аргументов в нейронной сети СМАС // *Информационные технологии.* 2011. № 1. С. 64–72.
23. Аведьян Э. Д., Галушкин А. И., Пантюхин Д. В. Ассоциативная нейронная сеть СМАС и ее модификации в задаче распознавания образов // *Информационные технологии.* 2011. № 7. С. 63–71.
24. Aved'yan E. Learning systems. London: Spriger-Verlag, 1995.
25. Аведьян Э. Д., Пантюхин Д. В. Методы подавления помех в нейронной сети СМАС // *Информационные технологии.* 2011. № 6. С. 6–12.

Е. А. Сирота, канд. физ.-мат. наук, доц., e-mail: sirota\_ea@sc.vsu.ru,  
Воронежский государственный университет

## Прогнозирование нестационарных временных последовательностей на основе иерархических многослойных нейронных сетей

*Предлагается методика построения иерархических многослойных нейронных сетей для прогнозирования нестационарных временных последовательностей и проводится анализ возможности ее применения для изучения динамики потребления цельной крови и плазмы лечебными профилактическими учреждениями. Дано обоснование преимуществ работы иерархической нейронной сети перед обычной нейронной сетью при решении задач прогнозирования нестационарных временных последовательностей.*

**Ключевые слова:** иерархическая нейронная сеть, нестационарная временная последовательность, прогнозирование, точность прогноза

E. A. Sirota

## The Unstable Time Sequence Forecasting Based on the Hierarchic Multilayer Neural Network

*The methods of the construction of the hierarchic multilayer neural network for unstable time sequence forecasting are provided. The analysis of the opportunity of using this hierarchic model for blood and plasma consumption process is made. The argumentation of the hierarchic multilayer neural network using advances before usual neural network in solving unstable time sequence forecasting problem is given.*

**Keywords:** hierarchic multilayer neural network, unstable time sequence, forecasting, forecasting accuracy

### Введение

В настоящее время аппарат искусственных нейронных сетей активно используется при решении задач прогнозирования временных рядов и в ряде случаев предоставляет ощутимые преимущества по отношению к традиционным методам, основанным на построении статистических моделей данных. Ряд таких преимуществ обусловлен наличием у нейронных сетей свойств к обобщению информации, возможностью распараллеливания процесса обработки, а также таких свойств, как отказоустойчивость, адаптивность, нелинейность [1]. Для большинства реальных процессов, описываемых как нестационарные случайные последовательности, характерно изменение основных статистических параметров, таких как, например, математическое ожидание или дисперсия, с течением времени. Необходимо отметить, что изменение этих параметров происходит достаточно медленно относительно динамики изменения самого процесса. В этом случае одним из возможных подходов к задаче прогнозирования является использование иерархических нейросетевых архитектур [2], что позволяет учитывать как динамику изменения статистических параметров случайного процесса, так и динамику из-

менения самого процесса. В конечном счете можно ожидать, что применение иерархически организованных нейронных сетей позволит повысить точность прогноза при решении задачи прогнозирования нестационарной случайной последовательности.

Цель настоящей работы — разработка методики построения иерархической нейронной сети для прогнозирования нестационарных временных последовательностей и анализ возможности ее применения для прогнозирования динамики потребления цельной крови и плазмы лечебными профилактическими учреждениями (ЛПУ) в рамках ранее разработанной информационной системы оптимального трансфузионного обеспечения ЛПУ [3].

Под трансфузионным обеспечением лечебных профилактических учреждений понимается обеспечение их компонентами крови, такими как плазма, эритромаасса и тромбоконцентрат, заготавливаемыми региональной "Службой крови" (областной станцией переливания крови). Задача управления запасами крови региональной "Службой крови" подробно описана в работе [4]. Рассмотренная задача представлена как линейно-квадратичная задача управления в дискретном времени. Для решения задачи в работе [4] применяли метод динамического про-

граммирования [5]. Данную математическую модель использовали при разработке информационной системы, практическая апробация которой осуществлялась на базе Государственного учреждения здравоохранения Воронежская областная станция переливания крови (ГУЗ ВОСПК) с перспективой дальнейшего внедрения в другие региональные "Службы крови" Российской Федерации.

Для автоматизации прогнозирования динамики потребления гемокомпонентов ЛПУ с помощью искусственных нейронных сетей в рамках программного комплекса был реализован дополнительный модуль с использованием библиотеки компонентов NeuralBase, предназначенной для программной реализации нейронных сетей. Основным назначением библиотеки является интеграция нейронных сетей в информационные системы для расширения их аналитических возможностей. Реализация нейронных сетей в виде компонентов, наличие открытого кода позволяют легко встраивать их в другие программы. Объектно-ориентированная реализация придает особую гибкость, позволяющую оптимизировать компоненты под конкретные задачи. В частности, был использован расширенный компонент многослойной нейронной сети, обучаемой на основе алгоритма обратного распространения ошибки. В то же время в ходе ранее выполненных исследований было выявлено, что одним из факторов, ограничивающих точность прогноза, является сезонный, т. е. существенно нестационарный характер динамики анализируемого процесса, что определило необходимость дальнейшего развития реализуемого подхода на основе применения иерархической нейронной сети.

### Прогнозирование нестационарной временной последовательности на основе иерархической нейронной сети

В общем случае может рассматриваться задача прогнозирования нестационарной временной последовательности вида

$$y_{t+1} = F(y_t, u_t) + \xi_t, \quad t = 1, 2, \dots, \quad (1)$$

где  $y_{t+1}$  — состояния рассматриваемого процесса в момент времени  $t+1$ ;  $u_t$  — вектор известных дополнительных управляющих параметров, позволяющий конкретизировать условия проведения наблюдений на каждом шаге;  $\xi_t$  — вектор, учитывающий воздействие случайных факторов.

Часто управляющий параметр меняется существенно медленнее, чем основной (таким параметром, например, может быть коэффициент корреляции

случайного процесса), поэтому модель (1) можно представить в виде

$$y_t = F(y_{t-1}, u_{s-1}) + \xi_t, \quad s = \varphi(t) = [t/\Delta N] + 1; \quad t = 1, 2, \dots \quad (2)$$

$$u_s \cong u_i = \text{const}, \quad s = i, \quad (i-1)\Delta N \leq t < i\Delta N,$$

где символ [...] означает целую часть числа;  $\Delta N$  — шаг изменения управляющего параметра.

Относительно переменной  $u_s$  во многих случаях можно предположить, что

$$u_s = \Phi(u_{s-1}) + \tilde{\xi}_s, \quad s = \varphi(t), \quad s = 1, 2, \dots, \quad t = 1, 2, \dots, \quad (3)$$

где  $\tilde{\xi}_s$  — вектор, учитывающий воздействие случайных факторов.

Таким образом, математическая модель исходного процесса может быть представлена как модель двух иерархически вложенных процессов. В качестве примера (2), (3) может служить модель марковского случайного процесса, описываемого рекуррентным уравнением вида

$$y(t) = r(s-1)y(t-1) + \sigma\sqrt{1-r(s-1)^2}\xi(t), \quad t = 1, 2, \dots, \quad (4)$$

где  $\sigma^2$  — дисперсия случайного процесса;  $r(s) = u(s)$  — коэффициент корреляции, компоненты вектора  $\xi$  являются независимыми гауссовскими стандартными величинами. Изменение коэффициента корреляции  $r$  может иметь, например, вид

$$r(s) = \begin{cases} f(s) + 0,5, & 0 < f(s) + 0,5 < 1, \\ 1, & f(s) + 0,5 \geq 1, \\ 0, & 0 \leq f(s) + 0,5, \end{cases} \quad (5)$$

$$f(s) = r_1 r(s-1) + \sigma_1 \sqrt{1-r_1^2} \xi(s), \quad s = 1, 2, \dots,$$

где  $\sigma_1^2$  — дисперсия случайного процесса, описывающего изменение  $r(s)$ ;  $r_1$  — коэффициент корреляции случайного процесса  $r(s)$ ; компоненты вектора  $\tilde{\xi}$  являются независимыми гауссовскими стандартными величинами.

Схема изменения управляющего параметра и основного процесса представлена на рис. 1 (см. третью сторону обложки).

В случае решения задачи прогнозирования для модели (1) с помощью аппарата искусственных нейронных сетей используется нейронная сеть класса многослойный персептрон.

В случае решения задачи прогнозирования для модели (2) будем использовать двухуровневую иерархическую нейронную сеть, архитектура которой представлена на рис. 2.

Первый иерархический уровень (так называемая медленная нейронная сеть) предназначен для про-

гнозирования параметра  $u$  модели (2), который является управляющим для векторов входа второго иерархического уровня. Таким параметром, например, может быть коэффициент корреляции случайного процесса.

В общем случае каждый иерархический уровень сети может иметь несколько скрытых слоев. В нашей задаче каждый иерархический уровень имеет два слоя нейронов.

Для первого иерархического уровня сети вектор входа имеет длину  $m = L$ , т. е. вектор входных значений определяется как  $u = (u_{s-L}, u_{s-L+1}, \dots, u_{s-1})^T$ ,  $s > L$ . Вектор выхода имеет длину  $m = 1$ , и, соответственно, выходной вектор определяется как  $u = u_s$ .

Выходное значение сети первого иерархического уровня  $u = u_s$  является управляющим параметром для сети второго иерархического уровня (так называемой быстрой нейронной сети). Таким образом, вектор входа второго иерархического уровня имеет длину  $m = M + 1$ , а вектор входных значений сети второго уровня, или "быстрой нейронной сети", формируется следующим образом:

$$\bar{x} = (y_{t-M}, y_{t-M+1}, \dots, y_{t-1}, u)^T.$$

Вектор выхода имеет длину  $m = 1$ , и, соответственно, выходной целевой вектор определяется как  $x = x_t$ .

Таким образом, сеть второго иерархического уровня осуществляет прогнозирование с учетом полученного на выходе сети первого уровня управляющего параметра.

Возможны два варианта методики обучения иерархической нейронной сети.

Первый вариант предлагаемой методики обучения иерархической нейронной сети заключается в реализации независимых процедур обучения на каждом уровне при известной математической модели иерархически вложенных процессов. На первом этапе происходит обучение "медленной нейронной сети". Формируем представительные по объему обучающие примеры для построения модели прогнозирования на основе "медленной нейронной сети". Обучающее множество формируется следующим образом. Для фиксированных значений  $t$  и  $L$  формируются входные векторы  $u^{(i)} = (u_{s-L}, u_{s-L+1}, \dots, u_{s-1})^T$ ,  $s > L$ , и требуемые им выходные целевые векторы  $u^{(i)} = u_s^{(i)}$ ,  $i = 1 \dots K$ , где  $K$  — число обучающих примеров.

На втором этапе происходит обучение второго иерархического уровня, или "быстрой нейронной

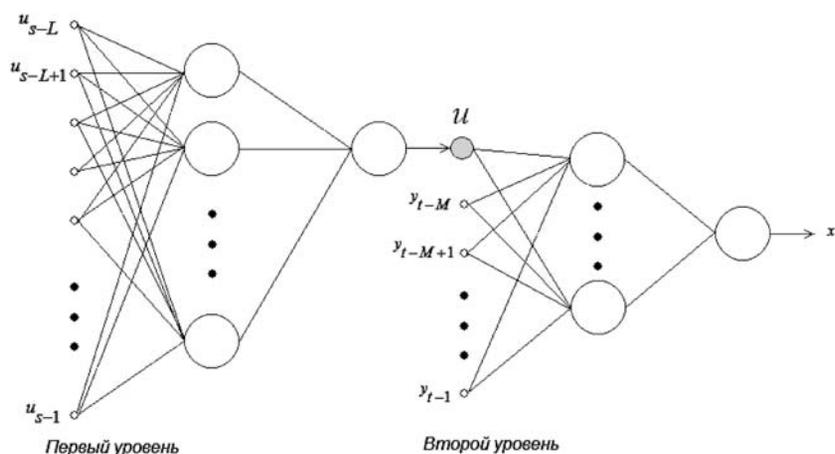


Рис. 2. Структура двухуровневой иерархической нейронной сети

сети". Формирование обучающего множества происходит следующим образом.

Независимо для каждого значения  $u$  ( $u = u_s$ ,  $s = 1, 2, \dots, S$ ), для фиксированных  $t$  и  $M$  формируем входные векторы  $x^{(j)} = (y_{t-M}, y_{t-M+1}, \dots, y_{t-1}, u)^T$ ,  $t > M$ , и требуемые им выходные целевые векторы  $x^{(j)} = x_t^{(j)}$ ,  $j = 1 \dots J$ , где  $J$  — число обучающих примеров.

Второй вариант методики обучения иерархической нейронной сети реализуется в случае, когда модель управляющего параметра неизвестна, но могут быть получены его оценки на основе данных самого случайного процесса. В этом случае обучение "медленной нейронной сети", или первого иерархического уровня, происходит по мере поступления новых статистических оценок управляющего параметра (это может быть, например, оценка коэффициента корреляции случайного процесса или оценка математического ожидания). Обучающее множество для второго уровня получается аналогично тому, как оно формировалось в первом варианте методики обучения иерархической нейронной сети, с той разницей, что вместо управляющего параметра во входном векторе используется его оценка.

В общем случае тестирование алгоритма проводили для модели марковского случайного процесса (4), (5). Использовали первый вариант методики. Второй вариант методики привел к аналогичным результатам.

Для прогнозирования коэффициента корреляции  $r$  с помощью первого иерархического уровня, или "медленной нейронной сети", использовалась сеть "многослойный перцептрон с тремя скрытыми слоями". Объем входного слоя выбран равным размерности входного вектора.

Для формирования обучающего множества первого иерархического уровня, прогнозирующего ко-

**Результаты сравнения работы  
обычной ИНС и иерархической ИНС**

Вид модели	$R^2$	$MAPE$ , %
Обычная ИНС	0,86	5
Иерархическая ИНС (первый вариант методики)	0,93	1,5
Иерархическая ИНС (второй вариант методики)	0,91	1,48

эффицент корреляции, была сформирована последовательность  $N = 1000$  коррелированных значений гауссовской случайной величины при значении  $r_1 = 0,9$ ,  $\sigma_1 = 0,2$ , с использованием рекуррентных уравнений вида (5).

Далее формировалось обучающее множество для второго иерархического уровня, "быстрой нейронной сети". Для каждого заданного значения коэффициента корреляции  $r_i$ ,  $i = 1, \dots, N$ , с использованием (4) формировалось  $N = 1000$  реализаций длины  $M = 100$  (значение  $\sigma = 0,25$ ).

Вектор входных значений сети второго уровня состоит из данных значений случайного процесса  $y_i$ ,  $i = 1 \dots m$ , а также управляющего параметра  $u = r$  — коэффициента корреляции, полученного в результате прогноза с помощью "медленной нейронной сети":

$$\xi = (u, y_1, y_2, \dots, y_m)^T.$$

Для обучения сети использовалась инструментальная среда MATLAB 10. В качестве функций активации входного и скрытого слоя использовалась сигмоидальная функция "logsig", для выходного слоя — линейная функция "purelin". Обучение ИНС проводилось с помощью алгоритма Levenberg—Marquardt ("trainlm").

Для обучения второго иерархического уровня была также выбрана сеть многослойный персептрон с четырьмя скрытыми слоями. В качестве функций активации входного и скрытых слоев использовалась сигмоидальная функция "logsig", для выходного слоя — линейная функция "purelin". Обучение ИНС проводилось с помощью алгоритма Levenberg—Marquardt ("trainlm").

Рис. 3 (см. четвертую сторону обложки) иллюстрирует работу первого иерархического уровня "медленной нейронной сети", "быстрой нейронной сети", обычной нейронной сети.

В качестве показателя эффективности каждой модели использовался коэффициент детерминации

$$R^2 = 1 - \frac{\sum_{t=1}^T (x(t) - \tilde{x}(t))^2}{\sum_{t=1}^T (x(t) - \bar{x})^2}, \quad (6)$$

где  $x(t)$ ,  $\tilde{x}(t)$  — фактические и расчетные значения уровней ряда соответственно,  $\bar{x}$  — среднее значение уровня ряда.

Также для сравнения использовался показатель процентной погрешности прогноза  $MAPE$

$$MAPE = \frac{1}{T} \left| \sum_{t=1}^T \frac{x(t) - \tilde{x}(t)}{x(t)} \right| \cdot 100 \%. \quad (7)$$

$MAPE$  не должен превышать 10—12 %.

Результаты моделирования показаны в табл. 1.

### Применения иерархической нейронной сети в задаче трансфузиологии

В задаче трансфузиологии используется второй вариант методики обучения иерархической нейронной сети, поскольку значения коэффициента корреляции, являющегося управляющим параметром, неизвестны, но могут быть получены его оценки на основе данных о потреблении гемокомпонентов.

В задаче потребления гемокомпонентов входным вектором является вектор коэффициентов корреляции за последние несколько месяцев  $X = (r_1, r_2, \dots, r_n)^T$ , где  $r_i$ ,  $i = 1, \dots, n$ , — коэффициент корреляции объема потребления гемокомпонента в  $i$ -м месяце. Число выходных нейронов определяется числом прогнозируемых периодов (в нашей задаче прогнозируемый период равен единице). Выходное значение сети первого иерархического уровня является управляющим параметром для сети второго иерархического уровня ("быстрой нейронной сети"). Таким образом, вектор входных значений сети второго уровня состоит из данных недельного потребления гемокомпонента за последний месяц, а также управляющего параметра  $u$  — коэффициента корреляции, полученного в результате прогноза с помощью "медленной нейронной сети"  $u = (u, y_1, y_2, \dots, y_m)^T$ , где  $y_i$ ,  $i = 1, \dots, m$ , — объем потребления гемокомпонента в  $i$ -ю неделю. Таким образом, сеть второго иерархического уровня осуществляет прогнозирование объема потребления гемокомпонента на следующую неделю текущего месяца уже с учетом коэффициента корреляции в этом месяце. В общем случае прогнозирование может осуществляться с учетом управляющего вектора статистических моментов, полученного на выходе сети первого уровня. Для прогнозирования значений случайного процесса с учетом коэффициента корреляции была выбрана сеть "многослойный персептрон с тремя скрытыми слоями".

Таблица 2

Результаты сравнения работы обычной ИНС и иерархической ИНС в случае задачи потребления гемокомпонентов

Вид модели	$R^2$	МАРЕ, %
Обычная ИНС	0,89	7,8
Иерархическая ИНС	0,94	5

Рис. 4 (см. четвертую сторону обложки) иллюстрирует работу иерархической и обычной нейронной сети для задачи потребления гемокомпонентов.

Результаты сравнения работы обычной ИНС и иерархической ИНС для задачи потребления гемокомпонентов представлены в табл. 2.

### Заключение

Результаты сравнения моделей, представленные в табл. 1, 2, показывают работоспособность и очевидное преимущество работы иерархической нейронной сети.

В качестве итога подчеркнем тот факт, что использование иерархической нейронной сети в данном случае повышает качество прогнозирования и позволяет модели в целом стать более устойчивой к случайным воздействиям.

*Поддержано Программой стратегического развития ВГУ.*

### Список литературы

1. **Осовский С.** Нейронные сети для обработки информации. М.: Финансы и статистика, 2004. 344 с.
2. **Басканова Т. Ф., Ланкин Ю. П., Комиссаров С. В.** Иерархические нейронные сети как средство решения трудноформулируемых задач искусственного интеллекта // Искусственный интеллект. 2009. № 1. С. 100—110.
3. **Бахметьев А. В., Сирота Е. А., Свекло Л. С.** Информационно-аналитическая система управления производством в "Службе крови" // Вестник службы крови России. 2010. № 1. С. 16—19.
4. **Сирота Е. А.** Информационная система управления производством гемокомпонентов "Службы крови" // Информационные технологии. 2011. № 9. С. 50—53.
5. **Афанасьев В. Н., Колмановский В. Б., Носов В. Р.** Математическая теория конструирования систем управления. М.: Высшая школа, 1989. 447 с.

## ИНФОРМАЦИЯ



# VIII Международная научно-практическая конференция Объектные системы — 2014

10—12 мая 2014, г. Ростов-на-Дону

Конференция посвящена принципам проектирования, реализации и сопровождения объектных систем и включает в себя обсуждение широкого круга проблем.

### Секции научной конференции

- Графические нотации, используемые при объектном проектировании ИС
- Принципы объектного проектирования информационных систем
- Инструменты объектного моделирования
- Теория объектно-ориентированного программирования
- Методы (шаблоны) объектно-реляционного отображения
- Реализация и использование объектных расширений в реляционных СУБД
- Проектирование, разработка и реализация распределенных систем
- Типовые реализации КИС с применением объектных технологий
- Принципы организации и реализации объектных баз данных
- Проблемы реализации объектных СУБД
- Имитационное моделирование объектных систем
- Темпоральные объектные системы
- Проблемы изучения (преподавания) объектных технологий в вузе

Сайт конференции <http://objectsystems.ru>

В журнале освещаются состояние и тенденции развития основных направлений индустрии программного обеспечения, связанных с проектированием, конструированием, архитектурой, обеспечением качества и сопровождением жизненного цикла программного обеспечения, а также рассматриваются достижения в области создания и эксплуатации прикладных программно-информационных систем во всех областях человеческой деятельности.

## Основные тематические направления журнала:

- Общетеоретические вопросы программной инженерии
- Языки и системы программирования, семантика программ
- Инструменты и методы программной инженерии
- Модели, методы и алгоритмы проектирования программных систем
- Системы управления базами данных и знаний
- Модели, методы, алгоритмы и программные инструменты для организации взаимодействия программ и программных систем
- Человеко-машинные интерфейсы, средства визуализации, обработки изображений, системы виртуальной реальности, мультимедийного общения
- Модели и методы создания программных систем для параллельной и распределенной обработки данных
- Верификация и тестирование программного обеспечения
- Оценка качества, стандартизация и сопровождение программных систем
- Безопасность программно-информационных систем
- Модели жизненного цикла программных проектов
- Сопровождение программного обеспечения
- Управление программной инженерией
- Интеллектуальные программные комплексы и системы
- Создание и эксплуатация прикладных программно-информационных систем

*Журнал распространяется только по подписке.*

*Оформить подписку можно через подписные Агентства или непосредственно в редакции журнала.*

**Журнал входит в перечень периодических изданий, рекомендованных ВАК РФ для публикации основных результатов диссертаций на соискание степеней кандидата и доктора наук, а также в базу данных Российского индекса научного цитирования (РИНЦ).**

Подписные индексы по каталогам:

«Роспечать» — 22765; «Пресса России» — 39795

**107076, Москва, Стромынский пер., 4**

**Тел./факс: (499) 269-55-10**

e-mail: [prin@novtex.ru](mailto:prin@novtex.ru)

<http://novtex.ru/pi.html>

---

## Адрес редакции:

107076, Москва, Стромынский пер., 4

Телефон редакции журнала **(499) 269-5510**

E-mail: [it@novtex.ru](mailto:it@novtex.ru)

Дизайнер *Т.Н. Погорелова*. Технический редактор *Е.В. Конова*.

Корректор *Т.В. Пчелкина*.

Сдано в набор 07.11.2013. Подписано в печать 20.12.2013. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ П114. 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.