

# ВЕСТНИК ТОМСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА

УПРАВЛЕНИЕ,  
ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА  
И ИНФОРМАТИКА

TOMSK STATE UNIVERSITY  
JOURNAL OF CONTROL AND COMPUTER SCIENCE

---

*Научный журнал*

---

2018

№ 45

Зарегистрирован в Федеральной службе по надзору в сфере  
массовых коммуникаций, связи и охраны культурного наследия  
(свидетельство о регистрации ПИ № ФС 77-29497 от 27 сентября 2007 г.)

Подписной индекс в объединённом каталоге «Пресса России» 44031

Журнал включен в «Перечень рецензируемых научных изданий,  
в которых должны быть опубликованы основные научные результаты диссертаций  
на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук»,  
Высшей аттестационной комиссии

## Founder – Tomsk State University

### EDITORIAL BOARD

**Alexander Gortsev** – Editor-in-Chief, Doctor of Sc., Prof., Head of the Operation Research Department, Tomsk State University. Tel: +72822529599  
**Valery Smagin** – Deputy Editor-in-Chief, Doctor of Sc., Prof. of the Operation Research Department Tomsk State University. Tel: +72822529599  
**Lyudmila Nezhelskaya** – Executive Editor, Doctor. of Sc., Prof. of the Operation Research Department Tomsk State University.  
E-mail: vestnik\_uvti@mail.tsu.ru

**Sergey Vorobeychikov** – Doctor of Sc., Prof. of the Higher Mathematics and Mathematical Modeling Department Tomsk State University  
**Vladimir Vishnevsky** – Doctor of Sc., Prof. Head of the laboratory Institute of Control Sciences of Russian Academy of Sciences (Moscow, Russia).  
**Gennady Koshkin** – Doctor of Sc., Prof. of the Theoretical Cybernetics Department Tomsk State University  
**Yury Kostyuk** – Doctor of Sc., Prof. of the Theoretical Informatics Department Tomsk State University  
**Anjela Matrosova** – Doctor of Sc., Prof., Head of the Programming Department Tomsk State University  
**Anatoly Nazarov** – Doctor of Sc., Prof., Head of the Probability Theory and Mathematical Statistics Department Tomsk State University  
**Konstantin Samouylov** – Doctor of Sc., Prof., Head of the Applied Probability and Informatics Department RUDN University (Moscow, Russia)  
**Eugene Semenkina** – Doctor of Sc., Prof. System Analysis and Operations Research Department Reshetnev Siberian State Aerospace University (Krasnoyarsk, Russia)  
**Sergey Sushchenko** – Doctor of Sc., Prof., Head of the Applied of Information Department, Director of the Institute of Applied Mathematics and Computer Science Tomsk State University  
**Mais Farkhadov** – Doctor of Sc., Head of the laboratory Institute of Control Sciences of Russian Academy of Sciences (Moscow, Russia).  
**Gurami Tsitsiasvili** – Doctor of Sc., Prof., Chief researcher Institute for Applied Mathematics Far Eastern Branch of RAS, Prof. Far Eastern Federal University (Vladivostok, Russia)  
**Editorial address:**  
Institute of Applied Mathematics and Computer Science,  
unit of Applied Mathematics  
National Research Tomsk State University  
36 Lenina Avenue, Tomsk, 634050  
Telephone / fax: +72822529599  
E-mail: vestnik\_uvti@mail.tsu.ru

### EDITORIAL COUNCIL

**Ana Rosa Cavalli**  
PhD, Prof.  
University VII  
Paris, France  
**Vladimir Dombrovskii**  
Doctor of Sc., Prof.  
Tomsk State University  
Russia  
**Alexander Dudin**  
Doctor of Sc., Prof.  
Belarusian State University  
Minsk, Republic Belorussia

**Gilbert Saporta**  
PhD, Prof.  
Pierre and Marie Curie  
University, Paris, France  
**Raimund Ubar**  
Doctor of Sc., Prof.  
University of Technology  
Tallinn, Estonia  
**Reindert Nobel**  
Doctor of Sc., Associate Prof.  
Vrije University of Amsterdam  
Netherlands

**Enco Orsingher**  
PhD, Prof.  
University of Rome  
Italy  
**Paolo Prinetto**  
Prof.  
Politecnico Institute  
Torino, Italy

**Nina Yevtushenko**  
Doctor of Sc., Prof.  
Tomsk State University  
Russia  
**Yervant Zorian**  
PhD, Fellow & Chief Architect, Synopsys, Mountain View, CA, USA

### JOURNAL INFO

Tomsk State University Journal of Control and Computer Science is an independent peer-reviewed research journal that welcomes submissions from across the world. Tomsk State University Journal of Control and Computer Science is issued four times per year, and can be subscribed to in the Russian Press Joint Catalogue (Subscription Index 44031). The publication in the journal is free of charge and may be in Russian or in English. The topics of the journal are the following:

- control of dynamical systems,
- mathematical modeling,
- data processing,
- informatics and programming,
- discrete function and automation,
- designing and diagnostics of computer systems.

Rules of registration articles are given in a site:  
<http://journals.tsu.ru/informatics/>  
ISSN 2311-2085 (Online), ISSN 1998-8605 (Print).

## Учредитель – Томский государственный университет

### РЕДАКЦИОННАЯ КОЛЛЕГИЯ

**Горцев Александр Михайлович** – гл. редактор, проф., д-р техн. наук, зав. кафедрой исследования операций ТГУ. Тел. +72822529599  
**Смагин Валерий Иванович** – зам. гл. редактора, проф., д-р техн. наук, проф. кафедры исследования операций ТГУ. Тел. +72822529599  
**Нежелская Людмила Алексеевна** – ответственный секретарь, доц., д-р физ.-мат. наук, проф. кафедры исследования операций ТГУ  
E-mail: vestnik\_uvti@mail.tsu.ru

**Воробейчиков Сергей Эрикович** – д-р физ.-мат. наук, проф. кафедры высшей математики и математического моделирования ТГУ  
**Вишневский Владимир Миронович** – проф., д-р техн. наук, зав. лабораторией Института проблем управления РАН (г. Москва)  
**Кошкин Геннадий Михайлович** – проф., д-р физ.-мат. наук, проф. кафедры теоретической кибернетики ТГУ  
**Костюк Юрий Леонидович** – проф., д-р техн. наук, проф. кафедры теоретической информатики ТГУ  
**Матросова Анжела Юрьевна** – проф., д-р техн. наук, зав. кафедры программирования ТГУ  
**Назаров Анатолий Андреевич** – проф., д-р техн. наук, зав. кафедрой теории вероятностей и математической статистики ТГУ  
**Самуйлов Константин Евгеньевич** – проф., д-р техн. наук, зав. кафедрой прикладной информатики и теории вероятностей РУДН (г. Москва)  
**Семенкин Евгений Станиславович** – проф., д-р техн. наук, проф. каф. системного анализа и исследования операций, СГАУ им. акад. М.Ф. Решетнева (г. Красноярск)  
**Сущенко Сергей Петрович** – проф., д-р техн. наук, зав. кафедрой прикладной информатики, директор института прикладной математики и компьютерных наук ТГУ  
**Фархадов Маис Паша Оглы** – д-р техн. наук, зав. лабораторией Института проблем управления РАН (г. Москва)  
**Цициашвили Гурами Шалвович** – проф., д-р физ.-мат. наук, гл. науч. сотр. Института прикладной математики ДВО РАН, проф. ДВФУ (г. Владивосток)

**Адрес редакции и издателя:** 634050, Томск, пр. Ленина, 36  
Национальный исследовательский Томский государственный университет,  
Институт прикладной математики и компьютерных наук,  
отделение прикладной математики  
Телефон / факс: +72822529599  
E-mail: vestnik\_uvti@mail.tsu.ru

### РЕДАКЦИОННЫЙ СОВЕТ

**Ана Роза Кавалли**  
д-р философии, проф.  
Университет VII,  
Париж, Франция  
**Владимир Домбровский**  
д-р техн. наук, проф.  
ТГУ, Томск, Россия  
**Александр Дудин**  
д-р физ.-мат. наук, проф.  
БГУ, Минск,  
Республика Беларусь

**Жильберт Сапорта**  
д-р философии, проф.  
Университет им. Пьера и Марии,  
Кюри, Париж, Франция  
**Раймонд Убар**  
д-р, проф.  
Технологический университет,  
Таллинн, Эстония  
**Рейндерт Нобель**  
д-р, доцент  
Свободный университет,  
Амстердам, Нидерланды

**Енцо Орзингер**  
д-р философии, проф.  
Римский университет,  
Италия  
**Паоло Принетто**  
проф.  
Политехнический институт,  
Турин, Италия

**Нина Евтушенко**  
д-р техн. наук, проф.  
ТГУ, Томск, Россия  
**Ервант Зориан**  
д-р философии, гл. науч. сотр.  
фирмы «Синописис», США

### О ЖУРНАЛЕ

Журнал «Вестник Томского государственного университета. Управление, вычислительная техника и информатика» выходит ежеквартально и распространяется по подписке. Статьи публикуются на русском и английском языках. Тематика публикаций журнала:

- управление динамическими системами,
- математическое моделирование,
- обработка информации,
- информатика и программирование,
- дискретные функции и автоматы,
- проектирование и диагностика вычислительных систем.

Журнал входит в систему Российского Индекса Научного Цитирования (РИНЦ). Правила оформления статей приведены на сайте:  
<http://journals.tsu.ru/informatics/>  
ISSN 2311-2085 (Online), ISSN 1998-8605 (Print).

СОДЕРЖАНИЕ

УПРАВЛЕНИЕ  
ДИНАМИЧЕСКИМИ СИСТЕМАМИ

- Домбровский В.В., Пашинская Т.Ю.**  
Управление с прогнозированием распределенными  
стохастическими гибридными системами  
с мультипликативными шумами при ограничениях..... 4

ОБРАБОТКА ИНФОРМАЦИИ

- Галилейская А.А., Лисовская Е.Ю.**  
Асимптотический анализ многофазной  
бесконечнолинейной ресурсной системы  
массового обслуживания  
с входящим MMPP потоком..... 13
- Dmitriev Yu.G., Koshkin G.M.**  
Estimation of present value  
of whole life annuity using information  
about expectation of life..... 22
- Нежелская Л.А., Сидорова Е.Ф.**  
Оптимальная оценка состояний обобщенного  
синхронного потока событий второго порядка  
в условиях неполной наблюдаемости..... 30
- Цициашвили Г.Ш., Осипова М.А.,  
Самуйлов К.Е., Гайдамака Ю.В.**  
Распределение ресурсов в многоканальной  
системе массового обслуживания с блокировкой  
на основе синергетических эффектов..... 42
- Шкленник М.А., Моисеев А.Н.**  
Исследование потоков заявок в двухфазной  
системе массового обслуживания с неограниченным  
числом приборов и повторными обращениями..... 48

ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

- Арыков С.Б.**  
Решение прикладных задач в системе  
параллельного программирования Аспект..... 59
- Kulyukin V.A., Mukherjee S., Burkatovskaya Yu.B.**  
Classification of audio samples by convolutional  
networks in audio beehive monitoring..... 68
- Сорокин М.С., Захарова Г.Б.**  
Интеллектуальный программный инструментарий  
для подготовки новостных  
телевизионных программ..... 76

ДИСКРЕТНЫЕ ФУНКЦИИ И АВТОМАТЫ

- Поттосин Ю.В.**  
Параллельная декомпозиция системы  
частичных булевых функций..... 83
- Сведения об авторах..... 92**

CONTENTS

CONTROL  
OF DYNAMICAL SYSTEMS

- Dombrovskii V.V., Pashinskaya T.Yu.**  
Model predictive control of distributed stochastic  
hybrid systems with multiplicative noises under  
constraints..... 4

DATA PROCESSING

- Galileyskaya A.A., Lisovskaya E.Yu.**  
Asymptotic analysis  
of resource infinite-server  
queueing tandem  
with MMPP arrivals..... 13
- Dmitriev Yu.G., Koshkin G.M.**  
Estimation of present value  
of whole life annuity using information  
about expectation of life..... 22
- Nezhel'skaya L.A., Sidorova E.F.**  
Optimal estimate of the states of a generalized  
synchronous flow of second-order events under  
conditions of incomplete observability..... 30
- Tsitsiashvili G.Sh., Osipova M.A.,  
Samouylov K.E., Gaidamaka Yu.V.**  
The allocation of resources  
in multichannel loss queueing system  
based on synergistic effects..... 42
- Shklennik M.A., Moiseev A.N.**  
Analysis of customers flows  
in the infinite-server  
queueing tandem with feedback..... 48

INFORMATICS AND PROGRAMMING

- Arykov S.B.**  
Solution of applied problems in the parallel  
programming system aspect..... 59
- Kulyukin V.A., Mukherjee S., Burkatovskaya Yu.B.**  
Classification of audio samples by convolutional  
networks in audio beehive monitoring..... 68
- Sorokin M.S., Zakharova G.B.**  
The intelligent software toolkit  
for production of news TV-programs  
discrete function and automaton..... 76

DISCRETE FUNCTION AND AUTOMATONS

- Pottosin Yu.V.**  
Parallel decomposition of a system  
of partial boolean functions..... 83
- Brief information about the authors..... 92**

## УПРАВЛЕНИЕ ДИНАМИЧЕСКИМИ СИСТЕМАМИ

УДК 519

DOI: 10.17223/19988605/45/1

**В.В. Домбровский, Т.Ю. Пашинская**

### УПРАВЛЕНИЕ С ПРОГНОЗИРОВАНИЕМ РАСПРЕДЕЛЕННЫМИ СТОХАСТИЧЕСКИМИ ГИБРИДНЫМИ СИСТЕМАМИ С МУЛЬТИПЛИКАТИВНЫМИ ШУМАМИ ПРИ ОГРАНИЧЕНИЯХ

В работе рассматривается задача синтеза оптимальных стратегий управления с прогнозирующей моделью для класса распределенных гибридных систем с мультипликативными шумами, состоящих из подсистем, с учетом ограничений на управляющие переменные. Параметры каждой из подсистем изменяются в соответствии с эволюцией марковских цепей, состояния которых взаимосвязаны между собой.

**Ключевые слова:** управление с прогнозирующей моделью; распределенные гибридные системы; векторная односвязная цепь Маркова; мультипликативные шумы; ограничения.

Современные системы управления, как правило, представляют собой сложные иерархические системы, состоящие из взаимодействующих подсистем неоднородной непрерывно-дискретной природы. В частности, инвестиционный портфель представляет собой сложную стохастическую нестационарную динамическую систему и может содержать рискованные финансовые активы разных классов, динамика доходностей которых меняется скачкообразно в соответствии с эволюцией состояний взаимосвязанных марковских цепей, характеризующих, например, состояние различных секторов экономики или различных финансовых рынков [1]. Такие системы относятся к классу распределенных гибридных систем [2, 3].

Гибридные системы с марковскими скачкообразными параметрами нашли широкое признание и применение в практике управления многими реальными процессами [2, 4]. Эффективным подходом к управлению гибридными системами в присутствии ограничений является метод управления с прогнозирующей моделью [3]. В работах [5–9] рассматриваются задачи управления с прогнозированием для гибридных систем, параметры которых изменяются в соответствии с эволюцией одномерной марковской цепи, с учетом ограничений на управления. В работе [10] предложен метод синтеза стратегий управления с прогнозированием для взаимосвязанных гибридных систем с марковскими скачками при условии, что от состояния марковской цепи зависит только матрица управления каждой из подсистем.

В настоящей работе рассматриваются распределенные гибридные системы с мультипликативными шумами. Параметры каждой из подсистем изменяются в соответствии с эволюцией марковских цепей, состояния которых взаимосвязаны между собой. Получены уравнения синтеза оптимальных стратегий управления с прогнозирующей моделью с учетом явных ограничений на управляющие переменные.

#### 1. Постановка задачи

Пусть система состоит из совокупности подсистем, состояния которых описываются уравнениями

$$\begin{aligned} x^{(q)}(k+1) = & A^{(q)}[\alpha^{(q)}(k+1), k+1]x^{(q)}(k) + B_0^{(q)}[\alpha^{(q)}(k+1), k+1]u^{(q)}(k) + \\ & + \sum_{j=1}^n B_j^{(q)}[\alpha^{(q)}(k+1), k+1]w_j^{(q)}(k+1)u^{(q)}(k), \quad q = 1, 2, \dots, s, \end{aligned} \quad (1)$$

где  $x^{(q)}(k)$  –  $n_x^{(q)}$ -мерный вектор состояния  $q$ -й подсистемой,  $u^{(q)}(k)$  –  $n_u^{(q)}$ -мерный вектор управления  $q$ -й подсистемой;  $A^{(q)}[\alpha^{(q)}(k), k]$ ,  $B_j^{(q)}[\alpha^{(q)}(k), k]$ ,  $j = 0, \dots, n$  – матрицы соответствующих размерностей;  $w_j^{(q)}(k)$  – последовательность белых шумов с нулевым средним и единичной дисперсией;  $\alpha^{(q)}(k)$  – скалярная однородная цепь Маркова с конечным множеством состояний  $\{1, 2, \dots, v_q\}$ ;  $\alpha^{(q)}(k)$  и  $w_j^{(q)}(k)$  независимы. Таким образом, каждая из подсистем может находиться в  $v_q$  состояниях, определяемых скалярным случайным процессом с дискретным множеством значений (состояний).

Между подсистемами существует взаимосвязь: состояние цепи  $\alpha^{(q)}(k)$   $q$ -й подсистемы ( $q = 1, 2, \dots, s$ ) в  $k$ -й момент времени зависит от состояний цепей  $\alpha^{(r)}(k-1)$  ( $r = 1, 2, \dots, s$ ) в момент времени  $k-1$ . Таким образом, динамика системы в целом зависит от дискретного векторного случайного процесса  $\alpha(k) = [\alpha^{(1)}(k), \alpha^{(2)}(k), \dots, \alpha^{(s)}(k)]^T$  с конечным множеством состояний  $\{q, j_q\}$  ( $q = 1, 2, \dots, s$ ;  $j_q = 1, 2, \dots, v_q$ ) и дискретным временем. Случайный процесс  $\alpha(k)$  представляет собой векторную односвязную цепь Маркова.

Для векторной цепи вероятности перехода за один шаг определяются в виде:

$$P_{i_1, \dots, i_s; j_1, \dots, j_s} = P\{\alpha_1(k+1) = \alpha_{1j_1}, \dots, \alpha_s(k+1) = \alpha_{sj_s} / \alpha_1(k) = \alpha_{1i_1}, \dots, \alpha_s(k) = \alpha_{si_s}\}, \sum_{j_1, \dots, j_s} P_{i_1, \dots, i_s; j_1, \dots, j_s} = 1$$

с начальным распределением

$$P_{j_1, \dots, j_s} = P\{\alpha_1(0) = j_1, \dots, \alpha_s(0) = j_s\}, (j_1 = \overline{1, v_1}; \dots; j_s = \overline{1, v_s}), \sum_{j_1, \dots, j_s} P_{j_1, \dots, j_s} = 1.$$

Предполагается, что состояние векторной марковской цепи в момент времени  $k$  доступно наблюдению.

На управляющие воздействия каждой из подсистем накладываются ограничения:

$$u_{\min}^{(q)}(k) \leq S^{(q)}(k)u^{(q)}(k) \leq u_{\max}^{(q)}(k), q = \overline{1, s}, \quad (2)$$

где  $S^{(q)}(k)$  – матрицы соответствующих размерностей.

Необходимо определить закон управления системой, состоящей из подсистем вида (1), при ограничениях (2) из условия минимума критерия со скользящим горизонтом управления

$$J(k+m/k) = M\left\{\sum_{q=1}^s \sum_{i=1}^m (x^{(q)}(k+i))^T R_1^{(q)}(k+i)x^{(q)}(k+i) - R_2^{(q)}(k+i)x^{(q)}(k+i) + \right. \\ \left. + (u^{(q)}(k+i-1/k))^T R^{(q)}(k+i-1)u^{(q)}(k+i-1/k) / x^{(q)}(k), \alpha(k)\right\}, \quad (3)$$

где  $u^{(q)}(k+l/k)$ ,  $l = \overline{0, m-1}$  – последовательность прогнозирующих управлений  $q$ -й подсистемой,  $u^{(q)}(k) = u^{(q)}(k/k)$ ,  $M\{a/b\}$  – оператор условного математического ожидания,  $R_1^{(q)}(k+i) \geq 0$ ,  $R_2^{(q)}(k+i) \geq 0$ ,  $R^{(q)}(k+i) > 0$  – весовые матрицы соответствующих размерностей,  $m$  – горизонт прогноза,  $k$  – текущий момент времени.

## 2. Синтез стратегий управления с прогнозированием

Стратегии управления с прогнозированием определяются по следующему правилу. На каждом шаге  $k$  минимизируем функционал (3) по последовательности прогнозирующих управлений  $u^{(q)}(k/k), \dots, u^{(q)}(k+m-1/k)$ ,  $q = 1, 2, \dots, s$ , зависящих от состояния подсистемы в момент времени  $k$ , при ограничениях (2). В качестве управления в момент времени  $k$  берем  $u^{(q)}(k) = u^{(q)}(k/k)$ . Тем самым получаем управление  $q$ -й подсистемой  $u^{(q)}(k)$  как функцию состояний  $x^{(q)}(k)$  и  $\alpha(k)$ , т.е. управление с обратной связью. Чтобы получить управление  $u^{(q)}(k+1)$  на следующем шаге, процедура повторяется для следующего момента  $k+1$  и т.д.

Если цепи Маркова  $\alpha^{(q)}(k)$  ( $q = 1, \dots, s$ ) независимы между собой (состояния подсистем не зависят от состояний других подсистем), т.е. представляют собой однородные скалярные цепи Маркова, то каждая из них допускает следующее представление в пространстве состояний [11]:

$$\theta^{(q)}(k+1) = P^{(q)}\theta^{(q)}(k) + v^{(q)}(k+1), \quad (4)$$

где  $\theta^{(q)}(k) = [\delta(\alpha^{(q)}(k), 1), \dots, \delta(\alpha^{(q)}(k), v_q)]^T$ ,  $\delta(\alpha^{(q)}(k), j)$  – функция Кронекера ( $j = 1, \dots, v_q$ );  $P^{(q)}$  – матрица переходных вероятностей для  $q$ -й цепи;  $v^{(q)}(k)$  – мартигал разность.

Обобщим соотношение (4) для скалярных цепей на случай векторных однородных цепей Маркова.

Введем мультииндексы  $i = (i_1, i_2, \dots, i_s)$ ,  $j = (j_1, j_2, \dots, j_s)$ . Тогда матрицу вероятностей перехода за один шаг векторной цепи Маркова можно представить в виде  $P = (P_{ij})$ , где

$$P_{ij} = P_{i_1, \dots, i_s; j_1, \dots, j_s}; (i_1 = \overline{1, v_1}; \dots; i_s = \overline{1, v_s}; j_1 = \overline{1, v_1}; \dots; j_s = \overline{1, v_s}).$$

Матрица  $P$  обладает свойством:

$$\sum_j P_{ij} = 1, \forall i.$$

Введем вектор  $\theta(k) = [\delta(\alpha(k), 1), \dots, \delta(\alpha(k), v)]^T$ ,  $v = v_1 \times v_2 \times \dots \times v_s$ . Значение вектора  $\theta(k)$  соответствует комбинации состояний одномерных цепей Маркова.

Тогда для многомерной цепи можно записать представление в пространстве состояний, аналогичное (4):

$$\theta(k+1) = P\theta(k) + v(k+1). \quad (5)$$

С учетом (5) уравнения для подсистем (1) можно представить в следующем виде:

$$\begin{aligned} x^{(q)}(k+1) = & A^{(q)}[\theta^{(q)}(k+1), k+1]x^{(q)}(k) + B_0^{(q)}[\theta^{(q)}(k+1), k+1]u^{(q)}(k) + \\ & + \sum_{j=1}^n B_j^{(q)}[\theta^{(q)}(k+1), k+1]w_j^{(q)}(k+1)u^{(q)}(k), \quad q = \overline{1, s}, \end{aligned} \quad (6)$$

где

$$A^{(q)}[\theta(k), k] = \sum_{i=1}^v \theta_i(k) A^{(i)(q)}(k), \quad (7)$$

$$B_j^{(q)}[\theta(k), k] = \sum_{i=1}^v \theta_i(k) B_j^{(i)(q)}(k), \quad j = \overline{0, n},$$

здесь  $\theta_i(k)$  ( $i = 1, 2, \dots, v$ ) – компоненты вектора  $\theta(k)$ ;  $\{A^{(i)(q)}\}$ ,  $\{B_j^{(i)(q)}\}$  ( $i = 1, \dots, v$ ) – множества значений матриц  $A^{(q)}[\alpha(k), k]$  и  $B_j^{(q)}[\alpha(k), k]$ .

Критерий (3) примет вид:

$$\begin{aligned} J(k+m/k) = & M \left\{ \sum_{q=1}^s \sum_{i=1}^m (x^{(q)}(k+i))^T R_1^{(q)}(k+i) x^{(q)}(k+i) - R_2^{(q)}(k+i) x^{(q)}(k+i) + \right. \\ & \left. + (u^{(q)}(k+i-1/k))^T R^{(q)}(k+i-1) u^{(q)}(k+i-1/k) / x^{(q)}(k), \theta(k) \right\}. \end{aligned} \quad (8)$$

**Теорема.** Векторы прогнозирующих управлений

$$U^{(q)}(k) = \left[ \left( u^{(q)}(k/k) \right)^T, \left( u^{(q)}(k+1/k) \right)^T, \dots, \left( u^{(q)}(k+m-1/k) \right)^T \right]^T, \quad q = \overline{1, s},$$

минимизирующие критерий (3) при ограничениях вида (2), на каждом шаге  $k$  определяется из решения задачи квадратичного программирования с критерием вида

$$Y(k+m/k) = \left[ 2x^T(k)G(k) - F(k) \right] U(k) + U^T(k)H(k)U(k) \quad (9)$$

при ограничениях

$$U_{\min}^{(q)}(k) \leq \bar{S}^{(q)}(k)U^{(q)}(k) \leq U_{\max}^{(q)}(k). \quad (10)$$

Оптимальное управление для  $q$ -й подсистемы равно

$$u^{(q)}(k) = \left[ I_{n_{u^{(q)}}} \quad 0_{n_{u^{(q)}}} \quad \dots \quad 0_{n_{u^{(q)}}} \right] U^{(q)}(k),$$

где  $I_{n_{u^{(q)}}}$  – единичная матрица размерности  $n_{u^{(q)}}$ ,  $0_{n_{u^{(q)}}}$  – квадратная нулевая матрица размерности  $n_{u^{(q)}}$ ,

$$U(k) = \left[ \left( U^{(1)}(k) \right)^T, \left( U^{(2)}(k) \right)^T, \dots, \left( U^{(s)}(k) \right)^T \right]^T,$$

$$\bar{S}^{(q)}(k) = \text{diag}(S^{(q)}(k), \dots, S^{(q)}(k+m-1)),$$

$$U_{\min}^{(q)}(k) = [(u_{\min}^{(q)}(k))^T, \dots, (u_{\min}^{(q)}(k+m-1))^T]^T, U_{\max}^{(q)}(k) = [(u_{\max}^{(q)}(k))^T, \dots, (u_{\max}^{(q)}(k+m-1))^T]^T,$$

$H(k)$ ,  $G(k)$ ,  $F(k)$  – блочные матрицы вида

$$H(k) = \text{diag}(H^{(1)}(k), H^{(2)}(k), \dots, H^{(s)}(k)), \quad (11)$$

$$G(k) = \begin{bmatrix} G^{(1)}(k) & G^{(2)}(k) & \dots & G^{(s)}(k) \end{bmatrix}, F(k) = \begin{bmatrix} F^{(1)}(k) & F^{(2)}(k) & \dots & F^{(s)}(k) \end{bmatrix}, \quad (12)$$

$$H^{(q)}(k) = \begin{bmatrix} H_{11}^{(q)}(k) & H_{12}^{(q)}(k) & \dots & H_{1m}^{(q)}(k) \\ H_{21}^{(q)}(k) & H_{22}^{(q)}(k) & \dots & H_{2m}^{(q)}(k) \\ \dots & \dots & \dots & \dots \\ H_{m1}^{(q)}(k) & H_{m2}^{(q)}(k) & \dots & H_{mm}^{(q)}(k) \end{bmatrix}, \quad q = \overline{1, s}, \quad (13)$$

$$G^{(q)}(k) = \begin{bmatrix} G_1^{(q)}(k) & G_2^{(q)}(k) & \dots & G_m^{(q)}(k) \end{bmatrix}, F^{(q)}(k) = \begin{bmatrix} F_1^{(q)}(k) & F_2^{(q)}(k) & \dots & F_m^{(q)}(k) \end{bmatrix}, \quad (14)$$

блоки которых равны

$$H_{tt}^{(q)}(k) = \sum_{i_t=1}^{\nu} \sum_{j=0}^n \left( B_j^{(i_t)(q)}(k+t) \right)^T Q^{(i_t)(q)}(k) B_j^{(i_t)(q)}(k+t) + R^{(q)}(k+t-1), \quad (15)$$

$$H_{tf}^{(q)}(k) = \sum_{i_t=1}^{\nu} \sum_{i_{t+1}=1}^{\nu} \dots \sum_{i_f=1}^{\nu} \left( B_0^{(i_t)(q)}(k+t) \right)^T \left( A^{(i_{t+1})(q)}(k+t+1) \right)^T \dots \left( A^{(i_f)(q)}(k+f) \right)^T \times \quad (16)$$

$$\times Q^{(i_t, \dots, i_f)(q)}(k) B_0^{(i_f)(q)}(k+f), f > t,$$

$$H_{tf}^{(q)}(k) = (H_{tt}^{(q)}(k))^T, f < t, \quad (17)$$

$$G_t^{(q)}(k) = \sum_{i_1=1}^{\nu} \dots \sum_{i_t=1}^{\nu} \left( A^{(i_1)(q)}(k+1) \right)^T \dots \left( A^{(i_t)(q)}(k+t) \right)^T Q^{(i_1, i_2, \dots, i_t)(q)}(k) B_0^{(i_t)(q)}(k+t), \quad (18)$$

$$F_t^{(q)}(k) = \sum_{i_t=1}^{\nu} Q_2^{(i_t)(q)}(k) B_0^{(i_t)(q)}(k+t). \quad (19)$$

Матрицы  $Q^{(i_t, \dots, i_f)(q)}(k)$ ,  $Q_2^{(i_t, \dots, i_f)(q)}(k)$ ,  $(t, f = \overline{1, m})$  определяются уравнениями:

$$Q^{(i_t, \dots, i_f)(q)}(k) = \Theta^{(i_t, \dots, i_f)}(k) R_1^{(q)}(k+f) + \sum_{i_{f+1}=1}^{\nu} \left( A^{(i_{f+1})(q)}(k+f+1) \right)^T Q^{(i_t, \dots, i_{f+1})(q)}(k) A^{(i_{f+1})(q)}(k+f+1), t = \overline{1, m-2}, t < f < m, \quad (20)$$

$$Q^{(i_t)(q)}(k) = E_{i_t} P^t \theta(k) R_1^{(q)}(k+t) + \sum_{i_{t+1}=1}^{\nu} \left( A^{(i_{t+1})(q)}(k+t+1) \right)^T Q^{(i_t, i_{t+1})(q)}(k) A^{(i_{t+1})(q)}(k+t+1), t = \overline{1, m-1}, \quad (21)$$

$$Q_2^{(i_t, \dots, i_f)(q)}(k) = R_2^{(q)}(k+f) \Theta^{(i_t, \dots, i_f)}(k) + \sum_{i_{s+1}=1}^{\nu} Q_2^{(i_t, \dots, i_{s+1})(q)}(k) A^{(i_{s+1})(q)}(k+f+1), \quad (22)$$

$$t = \overline{1, m-2}, t < f < m,$$

$$Q_2^{(i_t)(q)}(k) = R_2^{(q)}(k+t) E_{i_t} P^t \theta(k) + \sum_{i_{t+1}=1}^{\nu} Q_2^{(i_t, i_{t+1})(q)}(k) A^{(i_{t+1})(q)}(k+t+1), t = \overline{1, m-1}, \quad (23)$$

с начальными условиями

$$Q^{(i_m)(q)}(k) = E_{i_m} P^m \theta(k) R_1^{(q)}(k+m),$$

$$Q^{(i_t, \dots, i_m)(q)}(k) = \Theta^{(i_t, \dots, i_m)}(k) R_1^{(q)}(k+m), t = \overline{1, m-1},$$

$$Q_2^{(i_m)(q)}(k) = R_2^{(q)}(k+m)E_{i_m}P^m\theta(k),$$

$$Q_2^{(i_1, \dots, i_m)(q)}(k) = R_2^{(q)}(k+m)\Theta^{(i_1, \dots, i_m)}(k), t = \overline{1, m-1},$$

где

$$\Theta^{(i_1, \dots, i_f)}(k) = P_{i_f, i_{f-1}} P_{i_{f-1}, i_{f-2}} \dots P_{i_2, i_1} \theta_{i_1}(k+t | k), t = \overline{1, m-1}, f > t,$$

$\theta_{i_t}(k+t | k)$  – компонента вектора

$$\theta(k+t | k) = E\{\theta(k+t) | \theta(k)\} = P^t \theta(k),$$

$$E_{i_t} = [0, \dots, 0, 1, 0, \dots, 0]_{1 \times v}, i_t = \overline{1, v}, t = \overline{1, m}.$$

**Доказательство.** Критерий (8) и уравнение (6) можно представить в следующем виде:

$$J(k+m/k) = \sum_{q=1}^s M \left\{ (X^{(q)}(k+1))^T \Delta_1^{(q)}(k+1) X^{(q)}(k+1) - \right. \\ \left. - \Delta_2^{(q)}(k+1) X^{(q)}(k+1) + (U^{(q)}(k))^T \Delta^{(q)}(k) U^{(q)}(k) / x^{(q)}(k), \theta(k) \right\}, \quad (24)$$

$$X^{(q)}(k+1) = \Psi^{(q)}[\Xi(k+1), k+1] x^{(q)}(k) + \Phi^{(q)}[\Xi(k+1), k+1] U^{(q)}(k) + \\ + \Lambda^{(q)}[\Xi(k+1), W^{(q)}(k+1), k+1] U^{(q)}(k), \quad (25)$$

где

$$X^{(q)}(k+1) = \begin{bmatrix} x^{(q)}(k+1) \\ x^{(q)}(k+2) \\ \dots \\ x^{(q)}(k+m) \end{bmatrix}, U^{(q)}(k) = \begin{bmatrix} u^{(q)}(k/k) \\ u^{(q)}(k+1/k) \\ \dots \\ u^{(q)}(k+m-1/k) \end{bmatrix}, \Xi(k+1) = \begin{bmatrix} \theta(k+1) \\ \theta(k+2) \\ \dots \\ \theta(k+m) \end{bmatrix}, W^{(q)}(k+1) = \begin{bmatrix} w^{(q)}(k+1) \\ w^{(q)}(k+2) \\ \dots \\ w^{(q)}(k+m) \end{bmatrix},$$

$$\Psi^{(q)}[\Xi(k+1), k+1] = \begin{bmatrix} A^{(q)}[\theta(k+1), k+1] \\ A^{(q)}[\theta(k+2), k+2] A^{(q)}[\theta(k+1), k+1] \\ \dots \\ A^{(q)}[\theta(k+m), k+m] \dots A^{(q)}[\theta(k+2), k+2] A^{(q)}[\theta(k+1), k+1] \end{bmatrix},$$

$$\Phi^{(q)}[\Xi(k+1), k+1] = \begin{bmatrix} B_0^{(q)}[\theta(k+1), k+1] \\ A^{(q)}[\theta(k+2), k+2] B_0^{(q)}[\theta(k+1), k+1] \\ \dots \\ A^{(q)}[\theta(k+m), k+m] \dots A^{(q)}[\theta(k+2), k+2] B_0^{(q)}[\theta(k+1), k+1] \\ 0_{n_x^{(q)} \times n_u^{(q)}} \quad \dots \quad 0_{n_x^{(q)} \times n_u^{(q)}} \\ B_0^{(q)}[\theta(k+2), k+2] \quad \dots \quad 0_{n_x^{(q)} \times n_u^{(q)}} \\ \dots \quad \dots \quad \dots \\ A^{(q)}[\theta(k+m), k+m] \dots A^{(q)}[\theta(k+3), k+3] B_0^{(q)}[\theta(k+2), k+2] \quad \dots \quad B_0^{(q)}[\theta(k+m), k+m] \end{bmatrix},$$

$$\Lambda^{(q)}[\Xi(k+1), W^{(q)}(k+1), k+1] = \begin{bmatrix} \sum_{j=1}^n B_j^{(q)}[\theta^{(q)}(k+1), k+1] w_j^{(q)}(k+1) \\ A^{(q)}[\theta(k+2), k+2] \sum_{j=1}^n B_j^{(q)}[\theta^{(q)}(k+1), k+1] w_j^{(q)}(k+1) \\ \dots \\ A^{(q)}[\theta(k+m), k+m] \dots A^{(q)}[\theta(k+2), k+2] \sum_{j=1}^n B_j^{(q)}[\theta^{(q)}(k+1), k+1] w_j^{(q)}(k+1) \end{bmatrix}$$



$$\begin{aligned}
 & \begin{bmatrix} 0_{n_x^{(q)} \times n_u^{(q)}} & \dots & \dots \\ \sum_{j=1}^n B_j^{(q)} [\theta^{(q)}(k+2), k+2] w_j^{(q)}(k+2) & \dots & \dots \\ \dots & \dots & \dots \\ A^{(q)}[\theta(k+m), k+m] \dots A^{(q)}[\theta(k+3), k+3] \sum_{j=1}^n B_j^{(q)} [\theta^{(q)}(k+2), k+2] w_j^{(q)}(k+2) & \dots & \dots \\ \dots & 0_{n_x^{(q)} \times n_u^{(q)}} & \dots \\ \dots & 0_{n_x^{(q)} \times n_u^{(q)}} & \dots \\ \dots & \dots & \dots \\ \dots & \sum_{j=1}^n B_j^{(q)} [\theta^{(q)}(k+m), k+m] w_j^{(q)}(k+m) & \dots \end{bmatrix}, \\
 & \Delta^{(q)}(k) = \text{diag}(R^{(q)}(k+0), R^{(q)}(k+1), \dots, R^{(q)}(k+m-1)), \\
 & \Delta_1^{(q)}(k+1) = \text{diag}(R_1^{(q)}(k+1), R_1^{(q)}(k+2), \dots, R_1^{(q)}(k+m)), \\
 & \Delta_2^{(q)}(k+1) = \begin{bmatrix} R_2^{(q)}(k+1) & R_2^{(q)}(k+2) & \dots & R_2^{(q)}(k+m) \end{bmatrix}.
 \end{aligned}$$

Приведем (24)–(25) к виду:

$$J(k+m/k) = M \left\{ X^T(k+1) \Delta_1(k+1) X(k+1) - \Delta_2(k+1) X(k+1) + U^T(k) \Delta(k) U(k) / x(k), \theta(k) \right\}, \quad (26)$$

$$X(k+1) = \Psi[\Xi(k+1), k+1] x(k) + \Phi[\Xi(k+1), k+1] U(k) + \Lambda[\Xi(k+1), W(k+1), k+1] U(k), \quad (27)$$

где

$$\begin{aligned}
 X(k) &= [(X^{(1)}(k))^T, (X^{(2)}(k))^T, \dots, (X^{(s)}(k))^T]^T, x(k) = [(x^{(1)}(k))^T, (x^{(2)}(k))^T, \dots, (x^{(s)}(k))^T]^T, \\
 \Delta(k+1) &= \text{diag}(\Delta^{(1)}(k+1), \Delta^{(2)}(k+1), \dots, \Delta^{(s)}(k+1)), \\
 \Delta_1(k+1) &= \text{diag}(\Delta_1^{(1)}(k+1), \Delta_1^{(2)}(k+1), \dots, \Delta_1^{(s)}(k+1)), \Delta_2(k+1) = \begin{bmatrix} \Delta_2^{(1)}(k+1) & \Delta_2^{(2)}(k+1) & \dots & \Delta_2^{(s)}(k+1) \end{bmatrix}, \\
 U(k) &= [(U^{(1)}(k))^T, (U^{(2)}(k))^T, \dots, (U^{(s)}(k))^T]^T, \\
 \Psi[\Xi(k), k] &= \left[ \left( \Psi^{(1)}[\Xi(k), k] \right)^T, \dots, \left( \Psi^{(s)}[\Xi(k), k] \right)^T \right]^T, \\
 \Phi[\Xi(k), k] &= \text{diag}(\Phi^{(1)}[\Xi(k), k], \dots, \Phi^{(s)}[\Xi(k), k]), \\
 \Lambda[\Xi(k), W(k), k] &= \text{diag}(\Lambda^{(1)}[\Xi(k), W^{(1)}(k), k], \dots, \Lambda^{(s)}[\Xi(k), W^{(s)}(k), k]).
 \end{aligned}$$

С учетом (27) представим (26) в следующем виде:

$$\begin{aligned}
 J(k+m/k) &= M \{ U^T(k) \Delta(k) U(k) + x^T(k) \Psi^T[\Xi(k+1), k+1] \Delta_1(k+1) \Psi[\Xi(k+1), k+1] x(k) + \\
 &+ [2x^T(k) \Psi^T[\Xi(k+1), k+1] \Delta_1(k+1) - \Delta_2(k+1)] \Phi[\Xi(k+1), k+1] U(k) - \Delta_2(k+1) \Psi[\Xi(k+1), k+1] x(k) + \\
 &+ U^T(k) \Phi^T[\Xi(k+1), k+1] \Delta_1(k+1) \Phi[\Xi(k+1), k+1] U(k) + \\
 &+ U^T(k) \Lambda^T[\Xi(k+1), W(k+1), k+1] \Delta_1(k+1) \Lambda[\Xi(k+1), W(k+1), k+1] U(k) / x(k), \theta(k) \}. \quad (28)
 \end{aligned}$$

Определим матрицы

$$\begin{aligned}
 H(k) &= M \left\{ \Phi^T[\Xi(k+1), k+1] \Delta_1(k+1) \Phi[\Xi(k+1), k+1] + \right. \\
 &+ \Lambda^T[\Xi(k+1), W(k+1), k+1] \Delta_1(k+1) \Lambda[\Xi(k+1), W(k+1), k+1] / \theta(k) \left. \right\} + \Delta(k), \\
 G(k) &= M \left\{ \Psi^T[\Xi(k+1), k+1] \Delta_1(k+1) \Phi[\Xi(k+1), k+1] / \theta(k) \right\}, \\
 F(k) &= \Delta_2(k+1) M \left\{ \Phi[\Xi(k+1), k+1] / \theta(k) \right\}.
 \end{aligned}$$

Матрицы  $H(k)$ ,  $G(k)$ ,  $F(k)$  можно представить в блочном виде (11)–(14):

$$H(k) = \text{diag}\left(H^{(1)}(k), H^{(2)}(k), \dots, H^{(s)}(k)\right),$$

$$G(k) = \begin{bmatrix} G^{(1)}(k) & G^{(2)}(k) & \dots & G^{(s)}(k) \end{bmatrix}, F(k) = \begin{bmatrix} F^{(1)}(k) & F^{(2)}(k) & \dots & F^{(s)}(k) \end{bmatrix},$$

где

$$H^{(q)}(k) = \begin{bmatrix} H_{ij}^{(q)} \end{bmatrix}, i, j = \overline{1, m},$$

$$G^{(q)}(k) = \begin{bmatrix} G_1^{(q)}(k) & G_2^{(q)}(k) & \dots & G_m^{(q)}(k) \end{bmatrix}, F^{(q)}(k) = \begin{bmatrix} F_1^{(q)}(k) & F_2^{(q)}(k) & \dots & F_m^{(q)}(k) \end{bmatrix}, q = \overline{1, s}.$$

Используя представление матриц  $A^{(q)}[\theta(k), k]$ ,  $B_j^{(q)}[\theta(k), k]$  в виде (7) и уравнение (5), получим, что блоки матриц  $H^{(q)}(k)$ ,  $G^{(q)}(k)$ ,  $F^{(q)}(k)$  удовлетворяют уравнениям (15)–(23).

Таким образом, имеем задачу минимизации критерия (28) при ограничениях (2), которая эквивалентна задаче квадратичного программирования с критерием

$$Y(k + m / k) = \left[ 2x^T(k)G(k) - F(k) \right] U(k) + U^T(k)H(k)U(k)$$

при ограничениях (10).

### Заключение

Предложен метод синтеза стратегий прогнозирующего управления для гибридных взаимосвязанных систем с марковскими скачками и мультипликативными шумами. Данный подход позволяет в явном виде учесть ограничения на управления. Алгоритм синтеза прогнозирующей стратегии включает решение последовательности задач квадратичного программирования.

### ЛИТЕРАТУРА

1. Billio M., Pelizzon L. Value-at-Risk: a multivariate switching regime approach // Journal of Empirical Finance. 2000. No. 7. P. 531–554.
2. Teel A.R., Subbaram A., Sferlazza A. Stability analysis for stochastic hybrid systems: a survey // Automatica. 2014. V. 50, No. 10. P. 2435–2456.
3. Mayne D.Q. Model predictive control: Recent developments and future promise // Automatica. 2014. V. 50, No. 12. P. 2967–2986.
4. Costa O.L.V., Fragoso M.D., Marques R.P. Discrete-time Markov jump linear systems. New York : Springer, 2005. 286 p.
5. Tonne J., Jilg M., Stursberg O. Constrained Model Predictive Control of High Dimensional Jump Markov Linear Systems // Proc. American Control Conference. Palmer House Hilton. July 1–3, Chicago, IL, USA, 2015. P. 2993–2998.
6. Dombrovskii V.V., Obyedko, T.Yu. Predictive control of systems with Markovian jumps under constraints and its application to the investment portfolio optimization // Automation and remote control. 2011. V. 72, No. 5. P. 989–1003.
7. Dombrovskii V.V., Obyedko T.Yu., Samorodova M. Model predictive control of constrained Markovian jump nonlinear stochastic systems and portfolio optimization under market frictions // Automatica. 2018. V. 87, No. 1. P. 61–68.
8. Patrinos P., Soparasakis P., Sarimveis H., Bemporad A. Stochastic model predictive control for constrained discrete-time Markovian switching systems // Automatica. 2014. V. 50, No. 10. P. 2504–2514.
9. Blackmore L., Bektasov A., Ono M., Williams B.C. Robust optimal predictive control of jump Markov linear systems using particles // Hybrid systems: Comput. and Control / A. Bemporad, A. Bicchi, G. Buttazzo, eds. New York : Springer Verlag, 2007. V. 4416: Lecture Notes in Computer Science. P. 104–117.
10. Домбровский В.В., Обьедко Т.Ю. Управление с прогнозированием взаимосвязанными гибридными системами с марковскими скачками при ограничениях // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2012. № 3 (20). С. 5–12.
11. Elliott R.J., Aggoun L., Moore J.B. Hidden Markov models: Estimation and control. Berlin, Heidelberg, New York : Springer, 1995. 382 p.

Поступила в редакцию 5 апреля 2018 г.

Dombrovskii V.V., Pashinskaya T.Yu. (2018) MODEL PREDICTIVE CONTROL OF DISTRIBUTED STOCHASTIC HYBRID SYSTEMS WITH MULTIPLICATIVE NOISES UNDER CONSTRAINTS. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitel'naya tekhnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 45. pp. 4–12

DOI: 10.17223/19988605/45/1

We consider the following complex Markov jump linear system composed of interconnected subsystems

$$\begin{aligned} x^{(q)}(k+1) = & A^{(q)}[\alpha^{(q)}(k+1), k+1]x^{(q)}(k) + B_0^{(q)}[\alpha^{(q)}(k+1), k+1]u^{(q)}(k) + \\ & + \sum_{j=1}^n B_j^{(q)}[\alpha^{(q)}(k+1), k+1]w_j^{(q)}(k+1)u^{(q)}(k), q = \overline{1, s}, \end{aligned} \quad (1)$$

where  $x^{(q)}(k)$  is the  $n_x^{(q)}$ –dimensional subsystem state vector,  $u^{(q)}(k)$  is the  $n_u^{(q)}$ –dimensional subsystem control vector;  $A^{(q)}[\alpha^{(q)}(k), k]$ ,  $B_j^{(q)}[\alpha^{(q)}(k), k]$ ,  $j = 0, \dots, n$  are matrices of appropriate dimensions;  $w_j^{(q)}(k)$  is a sequence of white noises;  $\alpha^{(q)}(k)$  denotes a time-invariant Markov chain taking values in a finite set of states  $\{1, 2, \dots, v_q\}$ .

These subsystems interact in the following way. The state of Markov chain  $\alpha^{(q)}(k)$  of  $q$ th subsystem ( $q = 1, 2, \dots, s$ ) at the moment  $k$  depends on states of Markov chains  $\alpha^{(r)}(k-1)$  ( $r = 1, 2, \dots, s$ ) at the moment  $k-1$ . Thus, the complex system dynamics depends on a discrete-time vector stochastic process  $\alpha(k) = [\alpha^{(1)}(k), \alpha^{(2)}(k), \dots, \alpha^{(s)}(k)]^T$  taking values in a finite set of states  $\{q, j_q\} (q = 1, 2, \dots, s; j_q = 1, 2, \dots, v_q)$ . The process  $\alpha(k)$  is a simple connected Markov chain with the transition probability matrix

$$P_{i_1, \dots, i_s; j_1, \dots, j_s} = P\{\alpha_1(k+1) = \alpha_{1j_1}, \dots, \alpha_s(k+1) = \alpha_{sj_s} / \alpha_1(k) = \alpha_{1i_1}, \dots, \alpha_s(k) = \alpha_{si_s}\}, \sum_{j_1, \dots, j_s} P_{i_1, \dots, i_s; j_1, \dots, j_s} = 1,$$

and the initial distribution

$$P_{j_1, \dots, j_s} = P\{\alpha_1(0) = j_1, \dots, \alpha_s(0) = j_s\}, (j_1 = \overline{1, v_1}; \dots, j_s = \overline{1, v_s}), \sum_{j_1, \dots, j_s} P_{j_1, \dots, j_s} = 1.$$

It is assumed that the Markov chain  $\alpha(k)$  is observable at the moment  $k$ .

The following constraints are imposed on each subsystem control effects

$$u_{\min}^{(q)}(k) \leq S^{(q)}(k)u^{(q)}(k) \leq u_{\max}^{(q)}(k), q = \overline{1, s}, \quad (2)$$

where  $S^{(q)}(k)$  is the matrix of corresponding dimension.

For control of system (1) we synthesize the strategies with a predictive control model according to the following rule. At each step  $k$  we minimize the following quadratic objective with receding horizon

$$\begin{aligned} J(k+m/k) = & M\left\{\sum_{q=1}^s \sum_{i=1}^m (x^{(q)}(k+i))^T R_1^{(q)}(k+i)x^{(q)}(k+i) - R_2^{(q)}(k+i)x^{(q)}(k+i) + \right. \\ & \left. + (u^{(q)}(k+i-1/k))^T R^{(q)}(k+i-1)u^{(q)}(k+i-1/k) / x^{(q)}(k), \alpha(k)\right\}, \end{aligned}$$

on trajectories of system (1) over the sequence of predictive controls  $u^{(q)}(k+l/k)$ ,  $l = \overline{0, m-1}$ , depending on the subsystem state at the current time  $k$  under constraints (2);  $R_1^{(q)}(k+i) \geq 0$ ,  $R_2^{(q)}(k+i) \geq 0$ ,  $R^{(q)}(k+i) > 0$  are weigh matrices of appropriate dimensions;  $m$  is the prediction horizon;  $k$  is the current moment. The synthesis of predictive control strategies is reduced to the sequence of quadratic programming tasks.

Keywords: model predictive control; distributed hybrid systems; vector simple connected Markov chain; multiplicative noises; constraints.

*PASHINSKAYA Tatiana Yurievna* (Candidate of Physical and Mathematical Sciences, National Research Tomsk State University, Russian Federation).

E-mail: tatyana.obedko@mail.ru

*DOMBROVSKII Vladimir Valentinovich* (Doctor of Technical Sciences, Professor, National Research Tomsk State University, Russian Federation).

E-mail: dombrovs@ef.tsu.ru

## REFERENCES

1. Billio, M. & Pelizzon, L. (2000) Value-at-Risk: a multivariate switching regime approach. *Journal of Empirical Finance*. 7. pp. 531–554. DOI: 10.1016/S0927-5398(00)00022-0
2. Teel, A.R., Subbaram, A. & Sferlazza, A. (2014) Stability analysis for stochastic hybrid systems: A survey. *Automatica*. 50(10). pp. 2435–2456.
3. Mayne, D.Q. (2014) Model predictive control: Recent developments and future promise. *Automatica*. 50(12). pp. 2967–2986. DOI: 10.1016/j.automatica.2014.10.128
4. Costa, O.L.V., Fragoso, M.D. & Marques, R.P. (2005) *Discrete-time Markov jump linear systems*. Springer: New York.
5. Tonne, J., Jilg, M. & Stursberg, O. (2015) Constrained Model Predictive Control of High Dimensional Jump Markov Linear Systems. *Proc. American Control Conference*. Palmer House Hilton, July 1–3, 2015. Chicago, IL, USA. pp. 2993–2998. DOI: 10.1109/ACC.2015.7171190
6. Dombrovskii, V.V. & Obyedko, T.Yu. (2011) Predictive control of systems with Markovian jumps under constraints and its application to the investment portfolio optimization. *Automation and Remote Control*. 72(5). pp. 989–1003. DOI: 10.1134/S0005117911050079

7. Patrinos, P., Soparasakis, P., Sarimveis, H. & Bemporad, A. (2014) Stochastic model predictive control for constrained discrete-time Markovian switching systems. *Automatica*. 50(10). pp. 2504–2514. DOI: 10.1016/j.automatica.2014.08.031
8. Dombrovskii, V.V., Obyedko, T.Yu. & Samorodova, M. (2018) Model predictive control of constrained Markovian jump nonlinear stochastic systems and portfolio optimization under market frictions. *Automatica*. 87(1). pp. 61–68. DOI: 10.1016/j.automatica.2017.09.018
9. Blackmore, L., Bektassov, A., Ono, M. & Williams, B.C. (2007) Robust optimal predictive control of jump Markov linear systems using particles. In: Bemporad, A., Bicchi, A. & Buttazzo, G. (eds) *Hybrid Systems: Computation and Control*. . Vol. 4416. New York: Springer-Verlag. pp. 104–117.
10. Dombrovskii, V.V. & Obyedko, T.Yu. (2012) Model predictive control of interconnected hybrid systems with Markov jumps under constraints. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 3(20). pp. 5–12. (In Russian).
11. Elliott, R.J., Aggoun, L. & Moore, J.B. (1995) *Hidden Markov Models: Estimation and Control*. Berlin, Heidelberg, New York: Springer.

## ОБРАБОТКА ИНФОРМАЦИИ

УДК 519.872

DOI: 10.17223/19988605/45/2

А.А. Галилейская, Е.Ю. Лисовская

АСИМПТОТИЧЕСКИЙ АНАЛИЗ МНОГОФАЗНОЙ БЕСКОНЕЧНОЛИНЕЙНОЙ  
РЕСУРСНОЙ СИСТЕМЫ МАССОВОГО ОБСЛУЖИВАНИЯ  
С ВХОДЯЩИМ ММРП ПОТОКОМ

Рассматривается бесконечнолинейная ресурсная система массового обслуживания ММРП/(GI/∞)<sup>M</sup>. Решается задача исследования 2M-мерного процесса чисел занятых приборов и суммарных объемов занятого ресурса на фазах системы в стационарном режиме функционирования. С помощью метода асимптотического анализа получена асимптотическая характеристическая функция распределения вероятностей исследуемого многомерного процесса, которая совпадает с характеристической функцией многомерного гауссовского распределения.

**Ключевые слова:** многофазная система массового обслуживания; объем ресурса; метод асимптотического анализа; гауссовская аппроксимация.

Математические модели бесконечнолинейных систем массового обслуживания представляют большой интерес для практического применения в области телекоммуникаций, например для моделирования беспроводных сетей связи с целью оптимизации существующих и проектирования новых сетей [1–2]. Кроме того, такие системы важны при моделировании инженерных устройств, где необходимо вычислить достаточный объем буфера для хранения данных.

Так как необходимо учитывать объем передаваемой информации, то в связи с этим актуальной является разработка новых ресурсных моделей, сформулированных в терминах систем массового обслуживания (СМО), которые бы позволили оценить объемы занятого ресурса [3–4].

К сожалению, аналитические результаты получены лишь для систем с входящим простейшим потоком заявок и экспоненциальным временем обслуживания, которые не всегда соответствуют реальным сетям [5–7].

В настоящее время появляется все больше работ, посвященных исследованию многофазных систем массового обслуживания, которые состоят из нескольких последовательно соединенных подсистем, причем входящий поток каждой последующей подсистемы является выходящим для предыдущей, кроме последней.

## 1. Постановка задачи

Рассмотрим M-фазную ресурсную СМО с неограниченным числом приборов и неограниченным объемом предоставляемого ресурса на каждой фазе. На вход системы поступает ММРП-поток заявок, управляемый цепью Маркова  $k(t) = 1, 2, \dots, K$ , которая задается матрицей инфинитезимальных характеристик  $\mathbf{Q} = \|q_{ij}\|$  размера  $K \times K$ , и диагональной матрицей условных интенсивностей  $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \dots, \lambda_k\}$ . Поступающее требование занимает любой свободный прибор на первой фазе, где обслуживается в течение случайного времени  $\tau_1 \geq 0$  с функцией распределения  $B_1(x) = P\{\tau_1 < x\}$  и формирует запрос на предоставление случайного объема ресурса  $v \geq 0$  с функцией распределения  $G(y) = P\{v < y\}$ . По окончании обслуживания на первой фазе заявка освобождает тот же объем ресурса, мгновенно переходит на вторую фазу, где обслуживается в течение случайного времени  $\tau_2 \geq 0$  с функцией распределения

$B_2(x) = P\{\tau_2 < x\}$  и занимает тот же объем ресурса, как и на первой фазе. И так далее; после окончания обслуживания на  $M$ -й фазе заявка покидает систему и освобождает занимаемый ресурс.

Пусть  $i_m(t)$  – число заявок на  $m$ -й фазе в момент времени  $t$ ,  $V_m(t)$  – суммарный объем занятого ресурса на  $m$ -й фазе в момент времени  $t$ , где  $m = \overline{1, M}$  – номер фазы.

Поставим задачу нахождения характеристик многомерного случайного процесса  $\{\mathbf{i}(t), \mathbf{V}(t)\} = \{i_1(t), \dots, i_M(t), V_1(t), \dots, V_M(t)\}$ . Отметим, что исследуемый процесс не является марковским. Для его исследования применим метод многомерного динамического просеивания [8].

Изобразим  $M$  параллельных осей времени, пронумерованных от 0 до  $M$  (рис. 1). Ось под номером 0 будет отображать события входящего потока, ось под номером 1 будет соответствовать первому просеянному потоку, ось под номером 2 – второму и т.д., ось под номером  $M$  соответствует  $M$ -му просеянному потоку.

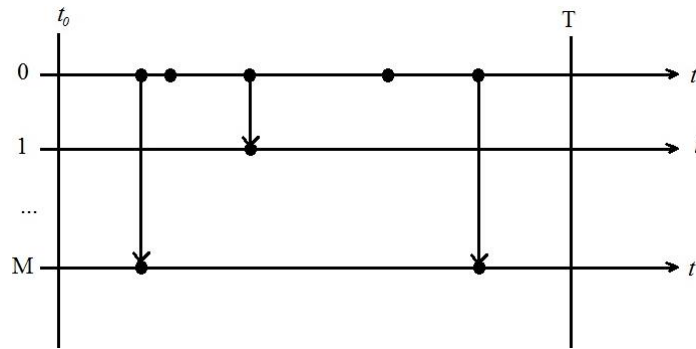


Рис. 1. Просеивание входящего потока

Пусть имеется набор функций  $S_m(t)$ ,  $m = \overline{1, M}$ , значения которых лежат в диапазоне  $[0, 1]$  и обладают свойством

$$\sum_{m=1}^M S_m(t) \leq 1,$$

для любых  $t$ .

Событие входящего потока может просеяться только на одну из осей либо не просеяться ни на одну. Вероятность того, что заявка входящего потока, поступившая в систему в момент времени  $t > t_0$ , сформирует событие потока на  $m$ -й оси, то есть к моменту времени  $T$  будет находиться на обслуживании на  $m$ -й фазе, равна  $S_m(t) = B_{m-1}^*(T-t) - B_m^*(T-t)$ , где  $B_m^*(\tau) = (B_{m-1} \cdot B_m)(\tau)$  – свертка функций распределения  $B_{m-1}(x)$ ,  $B_m(x)$  длительности обслуживания на фазах системы. Вероятность того, что заявка не сформирует событие ни на одной из осей, равна  $S_0(t) = 1 - \sum_{m=1}^M S_m(t)$ , т.е. к моменту времени  $T$  заявка закончит обслуживание на всех фазах и покинет систему.

Обозначим  $n_m(t)$  – число событий, наступивших на  $m$ -й оси просеянного потока до момента  $t$ ,  $W_m(t)$  – суммарный объем занятого ресурса просеянными заявками на  $m$ -й оси.

Как показано в [8], многомерное распределение вероятностей числа заявок на фазах системы в момент времени  $T$  совпадает с многомерным распределением вероятностей числа просеянных заявок на соответствующие оси:

$$P\{\mathbf{i}(T) = \mathbf{m}\} = P\{\mathbf{n}(T) = \mathbf{m}\}$$

для любых  $\mathbf{m} = [m_1, \dots, m_M]$ . Нетрудно показать, что для исследуемого процесса  $\{\mathbf{i}(t), \mathbf{V}(t)\}$  справедливо

$$P\{\mathbf{i}(T) = \mathbf{m}, \mathbf{V}(T) < \mathbf{z}\} = P\{\mathbf{n}(T) = \mathbf{m}, \mathbf{W}(T) < \mathbf{z}\} \quad (1)$$

для любых  $\mathbf{m}=[m_1, \dots, m_M]$  и любых  $\mathbf{z}=[z_1, \dots, z_M]$ . Следует отметить, что неравенства  $\mathbf{V}(T) < \mathbf{z}$ ,  $\mathbf{W}(T) < \mathbf{z}$  подразумевают поэлементное сравнение векторов, т.е.  $W_1(T) < z_1$  и т.д. Будем использовать равенство (1) для исследования процесса  $\{\mathbf{i}(t), \mathbf{V}(t)\}$  с помощью исследования процесса  $\{\mathbf{n}(t), \mathbf{W}(t)\}$ .

## 2. Система дифференциальных уравнений Колмогорова

Добавим компоненту  $k(t)$  – состояние управляющей цепи Маркова в момент времени  $t$ , к процессу  $\{\mathbf{n}(t), \mathbf{W}(t)\}$ , тогда полученный многомерный процесс будет являться марковским. Введем обозначение для его распределения вероятностей:

$$P(k, \mathbf{n}, \mathbf{w}, t) = P\{k(t) = k, \mathbf{n}(t) = \mathbf{n}, \mathbf{W}(t) < \mathbf{w}\}.$$

Для этого распределения составим  $\Delta t$ -методом прямую систему дифференциальных уравнений Колмогорова. По формуле полной вероятности запишем:

$$\begin{aligned} P(k, \mathbf{n}, \mathbf{w}, t + \Delta t) = & P(k, \mathbf{n}, \mathbf{w}, t)(1 - \lambda_k \Delta t)(1 + q_{kk} \Delta t) + P(k, \mathbf{n}, \mathbf{w}, t) \lambda_k \Delta t S_0(t) + \\ & + \sum_{m=1}^M \lambda_k \Delta t S_m(t) \int_0^{w_m} P(k, \mathbf{n} - \mathbf{e}_m, \mathbf{w} - \mathbf{y}_m, t) dG(y) + \sum_{v \neq k} q_{vk} \Delta t P(v, \mathbf{n}, \mathbf{w}, t) + o(\Delta t). \end{aligned} \quad (2)$$

Из (2) получаем систему дифференциальных уравнений Колмогорова

$$\begin{aligned} \frac{\partial P(k, \mathbf{n}, \mathbf{w}, t)}{\partial t} = & -\lambda_k (S_0(t) - 1) P(k, \mathbf{n}, \mathbf{w}, t) + \\ & + \sum_{v \neq k} q_{vk} \Delta t P(v, \mathbf{n}, \mathbf{w}, t) + \sum_{m=1}^M \lambda_k S_m(t) \int_0^{w_m} P(k, \mathbf{n} - \mathbf{e}_m, \mathbf{w} - \mathbf{y}_m, t) dG(y) \end{aligned} \quad (3)$$

для  $k = 1, \dots, K$ ;  $\mathbf{n}=[n_1, \dots, n_M]$ ,  $\mathbf{w}=[w_1, \dots, w_M]$ .

Начальное условие для решения  $P(k, \mathbf{n}, \mathbf{w}, t)$  в момент времени  $t_0$  определим в виде:

$$P(k, \mathbf{n}, \mathbf{w}, t_0) = \begin{cases} r(k), \mathbf{n} = \mathbf{w} = \mathbf{0}, \\ 0, \text{ иначе.} \end{cases}$$

где  $r(k)$  – стационарное распределение вероятностей состояний цепи Маркова  $k(t)$ .

Введем частичные характеристические функции вида:

$$h(k, \mathbf{u}, \mathbf{v}, t) = \sum_{n_1=0}^{\infty} e^{ju_1 n_1} \int_0^{\infty} e^{jv_1 w_1} \dots \sum_{n_M=0}^{\infty} e^{ju_M n_M} \int_0^{\infty} e^{jv_M w_M} P(k, \mathbf{n}, d\mathbf{w}, t),$$

где  $j = \sqrt{-1}$  – мнимая единица.

Тогда (3) можем переписать в виде следующей системы уравнений:

$$\frac{\partial h(k, \mathbf{u}, \mathbf{v}, t)}{\partial t} = \sum_v q_{vk} h(k, \mathbf{u}, \mathbf{v}, t) + \lambda_k \sum_{m=1}^M S_m(t) h(k, \mathbf{u}, \mathbf{v}, t) \left( e^{ju_m} G^*(v_m) - 1 \right),$$

для  $k = 1, \dots, K$ , где

$$G^*(v) = \int_0^{\infty} e^{jvy} dG(y).$$

Перепишем эту систему в виде матричного уравнения:

$$\frac{\partial \mathbf{H}(\mathbf{u}, \mathbf{v}, t)}{\partial t} = \mathbf{H}(\mathbf{u}, \mathbf{v}, t) \left[ \Lambda \sum_{m=1}^M S_m(t) \left( e^{ju_m} G^*(v_m) - 1 \right) + \mathbf{Q} \right] \quad (4)$$

с начальным условием

$$\mathbf{H}(\mathbf{u}, \mathbf{v}, t_0) = \mathbf{r}, \quad (5)$$

где

$$\mathbf{H}(\mathbf{u}, \mathbf{v}, t) = [h(1, \mathbf{u}, \mathbf{v}, t), \dots, h(K, \mathbf{u}, \mathbf{v}, t)],$$

$\mathbf{r} = [r(1), \dots, r(K)]$  – вектор стационарного распределения вероятностей состояний управляющей цепи Маркова  $k(t)$ , удовлетворяющий системе

$$\begin{cases} \mathbf{r}\mathbf{Q} = \mathbf{0}, \\ \mathbf{r}\mathbf{e} = 1, \end{cases} \quad (6)$$

и  $\mathbf{e}$  – единичный вектор-столбец.

### 3. Метод асимптотического анализа

Для решения задачи (4)–(5) воспользуемся методом асимптотического анализа [8] в условии растущей интенсивности входящего потока и предельно частых изменений состояний цепи Маркова [9–12].

Подставим в уравнение (4)  $\Lambda = N\Lambda_1$  и  $\mathbf{Q} = N\mathbf{Q}_1$ , где  $N \rightarrow \infty$  – некоторый параметр, который используется для асимптотического анализа. Тогда можно записать

$$\frac{1}{N} \frac{\partial \mathbf{H}(\mathbf{u}, \mathbf{v}, t)}{\partial t} = \mathbf{H}(\mathbf{u}, \mathbf{v}, t) \left[ \Lambda_1 \sum_{m=1}^M S_m(t) \left( e^{ju_m} G^*(v_m) - 1 \right) + \mathbf{Q}_1 \right] \quad (7)$$

с начальным условием (5).

**Теорема 1.** Асимптотическая характеристическая функция первого порядка многомерного случайного процесса  $\{\mathbf{n}(t), \mathbf{W}(t)\}$  имеет вид:

$$\mathbf{H}(\mathbf{u}, \mathbf{v}, t) = \mathbf{r} \exp \left\{ N \lambda \sum_{m=1}^M (ju_m + jv_m a_1) \int_{t_0}^t S_m(\tau) d\tau \right\},$$

где  $\lambda = \mathbf{r}\Lambda_1\mathbf{e}$  – интенсивность входящего потока,  $a_1$  – средний объем занимаемого одной заявкой ресурса.

*Доказательство.* Обозначим

$$\frac{1}{N} = \varepsilon, \mathbf{u} = \varepsilon \mathbf{x}, \mathbf{v} = \varepsilon \mathbf{y}, \mathbf{H}(\mathbf{u}, \mathbf{v}, t) = \mathbf{F}_1(\mathbf{x}, \mathbf{y}, t, \varepsilon). \quad (8)$$

Перепишем задачу (7)–(5) с учетом введенных обозначений:

$$\varepsilon \frac{\partial \mathbf{F}_1(\mathbf{x}, \mathbf{y}, t, \varepsilon)}{\partial t} = \mathbf{F}_1(\mathbf{x}, \mathbf{y}, t, \varepsilon) \left[ \Lambda_1 \sum_{m=1}^M S_m(t) \left( e^{j\varepsilon x_m} G^*(\varepsilon y_m) - 1 \right) + \mathbf{Q}_1 \right] \quad (9)$$

с начальным условием

$$\mathbf{F}_1(\mathbf{x}, \mathbf{y}, t_0, \varepsilon) = \mathbf{r}. \quad (10)$$

Найдем асимптотическое решение задачи (9)–(10) в два этапа.

*Этап 1.* Подставляя в (9)  $\varepsilon = 0$ , получим

$$\mathbf{F}_1(\mathbf{x}, \mathbf{y}, t) \mathbf{Q}_1 = \mathbf{0}.$$

Сравнивая это уравнение с первым в системе (6), можно сделать вывод, что его можно записать в виде:

$$\mathbf{F}_1(\mathbf{x}, \mathbf{y}, t) = \mathbf{r} \Phi_1(\mathbf{x}, \mathbf{y}, t), \quad (11)$$

где  $\Phi_1(\mathbf{x}, \mathbf{y}, t)$  – некоторая скалярная функция, которая удовлетворяет условию:

$$\Phi_1(\mathbf{x}, \mathbf{y}, t_0) = 1. \quad (12)$$

*Этап 2.* Умножим (9) на вектор  $\mathbf{e}$ , подставим (11), учитывая разложение

$$e^{j\varepsilon x} = 1 + j\varepsilon x + O(\varepsilon^2), \quad (13)$$

разделим результаты на  $\varepsilon$  и произведем предельный переход при  $\varepsilon \rightarrow 0$ . Тогда, учитывая, что  $\mathbf{Q}_1\mathbf{e} = \mathbf{0}$  и  $\mathbf{r}\mathbf{e} = 1$ , для функции  $\Phi_1(\mathbf{x}, \mathbf{y}, t)$  получим следующее дифференциальное уравнение:

$$\frac{\partial \Phi_1(\mathbf{x}, \mathbf{y}, t)}{\partial t} = \Phi_1(\mathbf{x}, \mathbf{y}, t) \lambda \sum_{m=1}^M S_m(t) (jx_m + jy_m a_1). \quad (14)$$



Проинтегрировав уравнение (14) от  $t_0$  до  $t$ , учитывая начальное условие (12), получим

$$\Phi_1(\mathbf{x}, \mathbf{y}, t) = \exp \left\{ \lambda \sum_{m=1}^M (jx_m + jy_m a_1) \int_{t_0}^t S_m(\tau) d\tau \right\}.$$

Подставляя это выражение в (11) и выполняя замены, обратные к (8), получим

$$\begin{aligned} \mathbf{H}(\mathbf{u}, \mathbf{v}, t) &= \mathbf{F}_1(\mathbf{x}, \mathbf{y}, t, \varepsilon) \approx \mathbf{F}_1(\mathbf{x}, \mathbf{y}, t) = \\ &= \mathbf{r} \exp \left\{ \lambda \sum_{m=1}^M (jx_m + jy_m a_1) \int_{t_0}^t S_m(\tau) d\tau \right\} = \mathbf{r} \exp \left\{ N\lambda \sum_{m=1}^M (ju_m + jv_m a_1) \int_{t_0}^t S_m(\tau) d\tau \right\}. \end{aligned}$$

Теорема доказана.

**Теорема 2.** Асимптотическая характеристическая функция второго порядка многомерного случайного процесса  $\{\mathbf{n}(t), \mathbf{W}(t)\}$  имеет вид:

$$\begin{aligned} \mathbf{H}(\mathbf{u}, \mathbf{v}, t) &\approx \exp \left\{ N\lambda \sum_{m=1}^M (ju_m + jv_m a_1) \int_{t_0}^t S_m(\tau) d\tau + N \sum_{m=1}^M \frac{(ju_m)^2}{2} \left( \lambda \int_{t_0}^t S_m(\tau) d\tau + \kappa \int_{t_0}^t S_m^2(\tau) d\tau \right) + \right. \\ &+ N \sum_{m=1}^M \frac{(jv_m)^2}{2} \left( \lambda a_2 \int_{t_0}^t S_m(\tau) d\tau + \kappa a_1 \int_{t_0}^t S_m(\tau) d\tau \right) + N \sum_{m=1}^M ju_m jv_m \left( \lambda a_1 \int_{t_0}^t S_m(\tau) d\tau + \kappa a_1 \int_{t_0}^t S_m(\tau) d\tau \right) + \\ &+ N \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M ju_m ju_l \kappa \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau + N \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jv_m jv_l \kappa a_1^2 \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau + \\ &\left. + N \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M ju_m jv_l \kappa a_1 \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau \right\}, \end{aligned}$$

где  $\kappa = 2\mathbf{g}(\bar{\Lambda} - \lambda\mathbf{I})\mathbf{e}$ ,  $a_2$  – второй начальный момент случайной величины с функцией распределения  $G(y)$ .

*Доказательство.* Представим функцию  $\mathbf{H}(\mathbf{u}, \mathbf{v}, t)$  в виде:

$$\mathbf{H}(\mathbf{u}, \mathbf{v}, t) = \mathbf{H}_2(\mathbf{u}, \mathbf{v}, t) \exp \left\{ N\lambda \sum_{m=1}^M (ju_m + jv_m a_1) \int_{t_0}^t S_m(\tau) d\tau \right\}, \quad (15)$$

получим уравнение относительно функции  $\mathbf{H}_2(\mathbf{u}, \mathbf{v}, t)$ :

$$\begin{aligned} \frac{1}{N} \frac{\partial \mathbf{H}_2(\mathbf{u}, \mathbf{v}, t)}{\partial t} + \mathbf{H}_2(\mathbf{u}, \mathbf{v}, t) \left[ N\lambda \sum_{m=1}^M (ju_m + jv_m a_1) S_m(t) \right] &= \\ = \mathbf{H}_2(\mathbf{u}, \mathbf{v}, t) \left[ \Lambda_1 \sum_{m=1}^M (e^{ju_m} G^*(v_m) - 1) S_m(t) + \mathbf{Q}_1 \right]. \end{aligned} \quad (16)$$

Выполним здесь следующие замены:

$$\varepsilon^2 = \frac{1}{N}, \mathbf{u} = \varepsilon \mathbf{x}, \mathbf{v} = \varepsilon \mathbf{y}, \mathbf{H}_2(\mathbf{u}, \mathbf{v}, t) = \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon). \quad (17)$$

С использованием этих обозначений уравнение (16) перепишется в виде:

$$\begin{aligned} \varepsilon^2 \frac{\partial \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon)}{\partial t} + \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon) \lambda \sum_{m=1}^M (j\varepsilon x_m + j\varepsilon y_m a_1) S_m(t) &= \\ = \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon) \left[ \Lambda_1 \sum_{m=1}^M S_m(t) (e^{j\varepsilon x_m} G^*(\varepsilon y_m) - 1) + \mathbf{Q}_1 \right]. \end{aligned} \quad (18)$$

Найдем асимптотическое, при  $\varepsilon \rightarrow 0$ , решение этой задачи, т.е.  $\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t) = \lim_{\varepsilon \rightarrow 0} \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon)$ .

*Этап 1.* Выполним предельный переход при  $\varepsilon \rightarrow 0$  в (18), получим:

$$\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon) \mathbf{Q}_1 = 0.$$

Представим  $\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t)$  в виде:

$$\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t) = \mathbf{r}\Phi_2(\mathbf{x}, \mathbf{y}, t), \quad (19)$$

где  $\Phi_2(\mathbf{x}, \mathbf{y}, t)$  – некоторая скалярная дифференцируемая функция, удовлетворяющая условию  $\Phi_2(\mathbf{x}, \mathbf{y}, t_0) = 1$ .

Этап 2. Решение уравнения (18) запишем в виде разложения:

$$\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon) = \Phi_2(\mathbf{x}, \mathbf{y}, t) \left\{ \mathbf{r} + \mathbf{g} \sum_{m=1}^M (j\varepsilon x_m + j\varepsilon y_m a_1) S_m(t) \right\} + O(\varepsilon^2), \quad (20)$$

где  $\mathbf{g}$  – некоторая вектор-строка. Подставим разложение (20) в (18), используя разложение (13), получим матричное уравнение для вектора  $\mathbf{g}$ :

$$\mathbf{g}\mathbf{Q}_1 = \mathbf{r}(\lambda\mathbf{I} - \Lambda_1).$$

Этап 3. Домножим (18) на вектор  $\mathbf{e}$ , используя (20) и разложение

$$e^{j\varepsilon x} = 1 + j\varepsilon x + \frac{(j\varepsilon x)^2}{2} + O(\varepsilon^3),$$

в результате несложных преобразований получаем

$$\begin{aligned} \frac{\partial \Phi_2(\mathbf{x}, \mathbf{y}, t)}{\partial t} = \Phi_2(\mathbf{x}, \mathbf{y}, t) & \left[ \sum_{m=1}^M \frac{(jx_m)^2}{2} (\lambda S_m(t) + \kappa S_m^2(t)) + \right. \\ & + \sum_{m=1}^M \frac{(jy_m)^2}{2} (\lambda a_2 S_m(t) + \kappa a_1 S_m^2(t)) + \sum_{m=1}^M jx_m jy_m (\lambda a_1 S_m(t) + \kappa a_1 S_m^2(t)) + \\ & \left. + \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jx_m jx_l \kappa S_m(t) S_l(t) + \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jy_m jy_l \kappa a_1^2 S_m(t) S_l(t) + \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jx_m jy_l \kappa a_1 S_m(t) S_l(t) \right], \end{aligned}$$

где  $\kappa = 2\mathbf{g}(\Lambda_1 - \lambda\mathbf{I})\mathbf{e}$ .

Решение этого уравнения с учетом начального условия, имеет вид:

$$\begin{aligned} \Phi_2(\mathbf{x}, \mathbf{y}, t) = \exp & \left\{ \sum_{m=1}^M \frac{(jx_m)^2}{2} \left( \lambda \int_{t_0}^t S_m(\tau) d\tau + \kappa \int_{t_0}^t S_m^2(\tau) d\tau \right) + \right. \\ & + \sum_{m=1}^M \frac{(jy_m)^2}{2} \left( \lambda a_2 \int_{t_0}^t S_m(\tau) d\tau + \kappa a_1 \int_{t_0}^t S_m^2(\tau) d\tau \right) + \\ & + \sum_{m=1}^M jx_m jy_m \left( \lambda a_1 \int_{t_0}^t S_m(\tau) d\tau + \kappa a_1 \int_{t_0}^t S_m^2(\tau) d\tau \right) + \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jx_m jx_l \kappa \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau + \\ & \left. + \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jy_m jy_l \kappa a_1^2 \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau + \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jx_m jy_l \kappa a_1 \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau \right\}, \end{aligned}$$

подставляя его в (19), получаем

$$\begin{aligned} \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon) = \mathbf{r} \exp & \left\{ \sum_{m=1}^M \frac{(jx_m)^2}{2} \left( \lambda \int_{t_0}^t S_m(\tau) d\tau + \kappa \int_{t_0}^t S_m^2(\tau) d\tau \right) + \right. \\ & \left. + \sum_{m=1}^M \frac{(jy_m)^2}{2} \left( \lambda a_2 \int_{t_0}^t S_m(\tau) d\tau + \kappa a_1 \int_{t_0}^t S_m^2(\tau) d\tau \right) + \right. \end{aligned}$$

$$\begin{aligned}
 & + \sum_{m=1}^M jx_m jy_m \left( \lambda a_1 \int_{t_0}^t S_m(\tau) d\tau + \kappa a_1 \int_{t_0}^t S_m^2(\tau) d\tau \right) + \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jx_m jx_l \kappa \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau + \\
 & + \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jy_m jy_l \kappa a_1^2 \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau + \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jx_m jy_l \kappa a_1 \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau \Bigg\}.
 \end{aligned}$$

Выполним замены, обратные к (17) и (15), запишем приближенное равенство для характеристической функции  $\mathbf{H}(\mathbf{u}, \mathbf{v}, t)$ :

$$\begin{aligned}
 \mathbf{H}(\mathbf{u}, \mathbf{v}, t) \approx \exp \Bigg\{ & N \lambda \sum_{m=1}^M (ju_m + jv_m a_1) \int_{t_0}^t S_m(\tau) d\tau + N \sum_{m=1}^M \frac{(ju_m)^2}{2} \left( \lambda \int_{t_0}^t S_m(\tau) d\tau + \kappa \int_{t_0}^t S_m^2(\tau) d\tau \right) + \\
 & + N \sum_{m=1}^M \frac{(jv_m)^2}{2} \left( \lambda a_2 \int_{t_0}^t S_m(\tau) d\tau + \kappa a_1 \int_{t_0}^t S_m^2(\tau) d\tau \right) + N \sum_{m=1}^M ju_m jv_m \left( \lambda a_1 \int_{t_0}^t S_m(\tau) d\tau + \kappa a_1 \int_{t_0}^t S_m^2(\tau) d\tau \right) + \\
 & + N \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M ju_m ju_l \kappa \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau + N \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M jv_m jv_l \kappa a_1^2 \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau + \\
 & + \sum_{m=1}^M \sum_{\substack{l=1 \\ l \neq m}}^M ju_m jv_l \kappa a_1 \int_{t_0}^t S_m(\tau) S_l(\tau) d\tau \Bigg\}.
 \end{aligned}$$

**Следствие.** Асимптотическое стационарное распределение вероятностей  $2M$ -мерного процесса чисел занятых приборов и суммарных объемов занятых ресурсов на фазах системы  $\text{MPP}/(\text{GI}/\infty)^M$  является  $2M$ -мерным гауссовским распределением с параметрами:

– вектор математических ожиданий  $\mathbf{Y} = N\lambda [\mathbf{Y}_1 \quad \mathbf{Y}_2 \quad \dots \quad \mathbf{Y}_L]$ , где  $\mathbf{Y}_i = [1 \quad a_1] \int_0^\infty (B_{i-1}^*(\tau) - B_i^*(\tau)) d\tau$ ,

– ковариационная матрица  $\mathbf{W} = N(\lambda \mathbf{W}^{(1)} + \kappa \mathbf{W}^{(2)})$ , где

$$\begin{aligned}
 \mathbf{W}^{(1)} &= \begin{bmatrix} \mathbf{K}_1^{(1)} & 0 & \dots & 0 \\ 0 & \mathbf{K}_2^{(1)} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \mathbf{K}_L^{(1)} \end{bmatrix}, \quad \mathbf{W}^{(2)} = \begin{bmatrix} \mathbf{K}_1^{(2)} & \mathbf{K}_{12}^{(2)} & \dots & \mathbf{K}_{1L}^{(2)} \\ \mathbf{K}_{21}^{(2)} & \mathbf{K}_2^{(2)} & \dots & \mathbf{K}_{2L}^{(2)} \\ \dots & \dots & \dots & \dots \\ \mathbf{K}_{L1}^{(2)} & \mathbf{K}_{L2}^{(2)} & \dots & \mathbf{K}_L^{(2)} \end{bmatrix}, \\
 \mathbf{K}_i^{(1)} &= \begin{bmatrix} 1 & a_1 \\ a_1 & a_1^2 \end{bmatrix} \int_0^\infty (B_{i-1}^*(\tau) - B_i^*(\tau)) d\tau, \quad \mathbf{K}_i^{(2)} = \begin{bmatrix} 1 & a_1 \\ a_1 & a_1^2 \end{bmatrix} \int_0^\infty (B_{i-1}^*(\tau) - B_i^*(\tau))^2 d\tau, \\
 \mathbf{K}_{ij}^{(2)} &= \begin{bmatrix} 1 & a_1 \\ a_1 & a_1^2 \end{bmatrix} \int_0^\infty (B_{i-1}^*(\tau) - B_i^*(\tau))(B_{j-1}^*(\tau) - B_j^*(\tau)) d\tau.
 \end{aligned}$$

### Заключение

В работе построена математическая модель процесса изменения чисел занятых приборов и суммарных объемов занятых ресурсов на фазах бесконечнолинейной многофазной ресурсной системы массового обслуживания  $\text{MPP}/(\text{GI}/\infty)^M$ . С помощью метода асимптотического анализа при условии растущей интенсивности входящего потока доказано, что асимптотическое распределение вероятностей чисел занятых приборов и суммарных объемов занятых ресурсов на фазах системы является многомерным гауссовским.

# ЛИТЕРАТУРА

1. Тихоненко О.М. Моделирование процессов и систем обработки информации : курс лекций. Минск: Белорус. гос. ун-т, 2008. 148 с.
2. Вихрова О.Г., Сопин Э.С. Анализ показателей качества сети LTE с помощью систем массового обслуживания с ограниченным ресурсом и случайными требованиями // Современные информационные технологии и ИТ-образование. 2015. Т. 2, № 11. С. 185–191.
3. Morozov E.V., Potakhina L.V. Speed-up estimation of a system with random volume customers // Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2016) : материалы 19-й междунар. науч. конф. М. : Моск. ун-т дружбы народов, 2016. С. 334–336.
4. Наумов В.А., Саймулов К.Е., Самуйлов А.К. О суммарном объеме ресурсов, занимаемых обслуживаемыми заявками // Автоматика и телемеханика. 2016. № 8. С. 125–135.
5. Sengupta B. The Spatial Requirement of M/G/1 Queue or: How to Design for Buffer Space Modeling and Performance Evaluation Methodology // Lect. Notes Contr. Inf. Sci. / F. Baccelli, G. Fayolle, eds. Berlin, 1984. V. 60. P. 547–562.
6. Тихоненко О.М. Распределение суммарного объема сообщений в системах массового обслуживания с групповым поступлением // Автоматика и телемеханика. 1987. № 11. С. 111–120.
7. Тихоненко О.М. Распределение суммарного объема сообщений в однолинейной системе массового обслуживания с групповым поступлением // Автоматика и телемеханика. 1985. № 11. С. 78–83.
8. Моисеев А.Н., Назаров А.А. Бесконечнолинейные системы и сети массового обслуживания. Томск : Изд-во НТЛ, 2015. 240 с.
9. Назаров А.А., Моисеева С.П. Метод асимптотического анализа в теории массового обслуживания. Томск : Изд-во НТЛ, 2006. 112 с.
10. Лисовская Е.Ю., Моисеева С.П. Асимптотический анализ системы MMPP/GI $\infty$  с обслуживанием требований случайного объема // Труды Томского государственного университета. Сер. физико-математическая. Т. 299: Математическое и программное обеспечение информационных, технических и экономических систем : материалы IV Междунар. молодежной науч. конф. Томск, 20–21 мая 2016 г. Томск : Издательский Дом ТГУ, 2016. С. 99–104.
11. Лисовская Е.Ю., Моисеева С.П. Исследование бесконечнолинейной системы массового обслуживания требований случайного объема с входящим MMPP-потокм // Информационные технологии и математическое моделирование (ИТММ-2016) : материалы XV Междунар. конф. им. А.Ф. Терпугова. Катунь, 12–16 сентября 2016 г. Томск : Изд-во Том. ун-та, 2016. Ч. 1. С. 77–82.
12. Лисовская Е.Ю., Моисеева С.П. Суммарный объем заявок в бесконечнолинейной системе массового обслуживания с рекуррентным входящим потоком // Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2016) : материалы Девятнадцатой междунар. науч. конф., 21–25 ноября 2016 г. М. : РУДН, 2016. С. 313–325.

Поступила в редакцию 29 декабря 2017 г.

Galileyskaya A.A., Lisovskaya E.Yu. (2018) ASYMPTOTIC ANALYSIS OF RESOURCE INFINITE-SERVER QUEUEING TANDEM WITH MMPP ARRIVALS. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitel'naya tekhnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 45. pp. 13–21

DOI: 10.17223/19988605/45/2

In this work, the MMPP/(GI $\infty$ )<sup>M</sup> queueing system with unlimited number of servers and with unlimited amount of resource is studied. The arrival process is a MMPP. Service times on each server on  $m^{\text{th}}$  phase are i.i.d. with distribution function  $B_m(x)$  ( $m = 1, M$ ). All customers form request for a random resource  $v$  with distribution function  $G(y) = P\{v < y\}$  on each phase. After the service end on  $M^{\text{th}}$  phase the customer leaves system and sets free the occupied resource.

Consider  $2M$ -dimensional stochastic process  $\{\mathbf{i}(t), \mathbf{V}(t)\} = \{i_1(t), \dots, i_M(t), V_1(t), \dots, V_M(t)\}$ , where  $i_m(t)$  and  $V_m(t)$  denote the numbers of customers and the total resources at the  $m^{\text{th}}$  phase in the system at time  $t$ , respectively.

We proposed the dynamic screening method for its investigation. Note that this method exactly determines the characteristics of the process  $\mathbf{V}(t)$  since the screened process contains only those customers, which do not finish the service at the moment  $T$ .

The system of Kolmogorov differential equations is derived. By using the partial characteristic function, we obtained the main equation:

$$\frac{\partial \mathbf{H}(\mathbf{u}, \mathbf{v}, t)}{\partial t} = \mathbf{H}(\mathbf{u}, \mathbf{v}, t) \left[ \Lambda \sum_{m=1}^M S_m(t) \left( e^{j\mu_m G^*(v_m)} - 1 \right) + \mathbf{Q} \right]$$

with the initial condition

$$\mathbf{H}(\mathbf{u}, \mathbf{v}, t_0) = \mathbf{r}.$$

To solve this equation, the method of asymptotic analysis is proposed under the condition of an infinitely growing arrival rate. We proved that in steady state regime the characteristic function of the customers numbers and the total resources at the system phases corresponds to a  $2M$ -dimensional Gaussian distribution with parameters:

$$\text{— expectations vector } \mathbf{Y} = N\lambda[\mathbf{Y}_1 \quad \mathbf{Y}_2 \quad \dots \quad \mathbf{Y}_L], \text{ where } \mathbf{Y}_i = [1 \quad a_1] \int_0^\infty (B_{i-1}^*(\tau) - B_i^*(\tau)) d\tau,$$

– covariance matrix  $\mathbf{W} = N(\lambda \mathbf{W}^{(1)} + \kappa \mathbf{W}^{(2)})$ , where

$$\mathbf{W}^{(1)} = \begin{bmatrix} \mathbf{K}_1^{(1)} & 0 & \dots & 0 \\ 0 & \mathbf{K}_2^{(1)} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \mathbf{K}_L^{(1)} \end{bmatrix}, \quad \mathbf{W}^{(2)} = \begin{bmatrix} \mathbf{K}_1^{(2)} & \mathbf{K}_{12}^{(2)} & \dots & \mathbf{K}_{1L}^{(2)} \\ \mathbf{K}_{21}^{(2)} & \mathbf{K}_2^{(2)} & \dots & \mathbf{K}_{2L}^{(2)} \\ \dots & \dots & \dots & \dots \\ \mathbf{K}_{L1}^{(2)} & \mathbf{K}_{L2}^{(2)} & \dots & \mathbf{K}_L^{(2)} \end{bmatrix},$$

$$\mathbf{K}_i^{(1)} = \begin{bmatrix} 1 & a_1 \\ a_1 & a_2 \end{bmatrix} \int_0^\infty (B_{i-1}^*(\tau) - B_i^*(\tau)) d\tau, \quad \mathbf{K}_i^{(2)} = \begin{bmatrix} 1 & a_1 \\ a_1 & a_2 \end{bmatrix} \int_0^\infty (B_{i-1}^*(\tau) - B_i^*(\tau))^2 d\tau,$$

$$\mathbf{K}_{ij}^{(2)} = \begin{bmatrix} 1 & a_1 \\ a_1 & a_2 \end{bmatrix} \int_0^\infty (B_{i-1}^*(\tau) - B_i^*(\tau))(B_{j-1}^*(\tau) - B_j^*(\tau)) d\tau.$$

Keywords: queueing tandem; random resource; method of asymptotic analysis; Gaussian approximation.

*GALILEYSKAYA Anastasiya Alexandrovna* (National Research Tomsk State University, Tomsk, Russian Federation).

E-mail: lusta.nastya@mail.ru

*LISOVSKAYA Ekaterina Yurievna* (National Research Tomsk State University, Tomsk, Russian Federation).

E-mail: ekaterina\_lisovs@mail.ru

## REFERENCES

1. Tikhonenko, O.M. (2008) *Modelirovanie protsessov i sistem obrabotki informatsii* [Modelling of processes and information processing systems]. Minsk: BSU.
2. Vikhrova, O.G. & Sopin, E.S. (2015) Analiz pokazateley kachestva seti LTE s pomoshch'yu sistem massovogo obsluzhivaniya s ogranichennym resursom i sluchaynymi trebovaniyami [Analysis of LTE network quality indicators with the help of queueing systems with limited resource and random requirements]. *Sovremennye informatsionnye tekhnologii i IT-obrazovanie – Modern Information Technologies and IT-Education*. 11. pp. 185–191.
3. Morozov, E.V. & Potakhina, L.V. (2016) Speed-up estimation of a system with random volume customers. *Raspredelennye komp'yuternye i telekommunikatsionnye seti: upravlenie, vychislenie, svyaz' (DCCN-2016)* [Distributed computer and communication networks: control, computation, communications (DCCN-2016)]. Proc. of the 19th International Conference. Moscow, 2016. pp. 547–562.
4. Naumov, V.A., Samuilov, K.E. & Samuilov, E.K. (2016) On the total amount of resources occupied by serviced customers. *Automation and Remote Control*. 77(8). pp. 125–135. DOI: 10.1134/S0005117916080087
5. Sengupta, B. (1984) The Spatial Requirement of M/G/1 Queue or: How to Design for Buffer Space Modeling and Performance Evaluation Methodology. In: Baccelli, F. & Fayolle, G. (eds) *Lecture Notes in Control and Information Sciences*. Vol. 60. Berlin: Springer. pp. 547–562.
6. Tikhonenko, O.M. (1987) Raspredelenie summarnogo ob'yema soobshcheniy v sistemakh massovogo obsluzhivaniya s gruppovym postupleniem [Distribution of the total volume of messages in a queueing system with batch arrival]. *Avtomatika i telemekhanika – Automation and Remote Control*. no. 11. pp. 111–120.
7. Tikhonenko, O.M. (1985) Raspredeleniye summarnogo ob'yema soobshcheniy v odnolineynoy sisteme massovogo obsluzhivaniya s gruppovym postupleniem [Distribution of the total volume of messages in a single-server queueing system with batch arrival]. *Avtomatika i telemekhanika – Automation and Remote Control*. 11. pp. 78–83.
8. Moiseev, A.N. & Nazarov, A.A. (2015) *Beskonechnoлинейные системы и сети массового обслуживания* [Queueing systems and networks with infinite number of servers]. Tomsk: NTL.
9. Nazarov, A.A. & Moiseeva, S.P. (2006) *Metod asimptoticheskogo analiza v teorii massovogo obsluzhivaniya* [Method of asymptotic analysis in queueing theory]. Tomsk: NTL.
10. Lisovskaya, E.Yu. & Moiseeva, S.P. (2016) [Asymptotic analysis MMPP|GI| $\infty$  queue with random customers capacity]. *Matematicheskoe i programnoe obespechenie informatsionnykh, tekhnicheskikh i ekonomicheskikh sistem* [Mathematical and software information, technical and economic systems]. Proc. of the 4th International Youth Conference. Tomsk, May 20–21, 2016. Tomsk: Tomsk State University. pp. 99–104. (In Russian).
11. Lisovskaya, E.Yu. & Moiseeva, S.P. (2016) [Study of infinite-server queue with random customers capacity with the MMPP arrives]. *Informatsionnye tekhnologii i matematicheskoe modelirovanie (ITMM-2016)* [Information technology and mathematical modeling (ITMM 2016)]. Proc. of the 15th International Conference named after A.F. Terpugov. Vol. 1. Tomsk: Tomsk State University. pp. 77–82. (In Russian).
12. Lisovskaya, E.Yu. & Moiseeva, S.P. (2016) [The total volume of customers in the infinite-server queueing system with renewal input process]. *Raspredelennye komp'yuternye i telekommunikatsionnye seti: upravlenie, vychislenie, svyaz' (DCCN-2016)* [Distributed computer and communication networks: control, computation, communications (DCCN-2016)]. Proc. of the 19th International Conference. Moscow: PFUR. pp. 313–325. (In Russian).

**Yu.G. Dmitriev, G.M. Koshkin**

# **ESTIMATION OF PRESENT VALUE OF WHOLE LIFE ANNUITY USING INFORMATION ABOUT EXPECTATION OF LIFE**

*This research was supported by "The Tomsk State University competitiveness improvement programme" under grant No. 8.1.37.2018.*

The paper deals with the estimation problem of the actuarial present value of the continuous whole life annuity using auxiliary information about the expectation of life. Nonparametric estimators of life annuity are constructed by individuals' death moments. It is shown that the usage of such auxiliary information can often provide the mean squared error (MSE) smaller than that of standard estimators. An adaptive estimator is also proposed. The asymptotic normality of all these estimators is proved.

**Keywords:** nonparametric estimation; whole life annuity; auxiliary information; mean squared error; asymptotic normality.

The idea of life annuity in accordance with [1. P. 170] is this: from the moment  $t = 0$  an individual once a year begins to get a certain money, which we take as the unit of money, and payments are made only for the lifetime of an individual. It is known that the calculation of the characteristics of life annuity is based on the characteristics of the respective type of insurance. Thus, the average total cost of the present continuous annuity is defined by the following formula (see [1. P. 184]):

$$\bar{a}_x(\delta) = \frac{1 - \bar{A}_x}{\delta},$$

where  $\bar{A}_x$  is a net premium (the average of the present value of a single sum of money in the insurance lifetime at the age  $x$ ),  $\delta$  is a force of interest. Let  $x$  be an individual's age on the moment of payments start,  $X$  be his lifetime,  $T_x = X - x$  be his future lifetime. Let us introduce the random variable

$$z(x) = \frac{1 - e^{-\delta T_x}}{\delta}, T_x > 0. \quad (1)$$

Then, by averaging the random variable  $z(x)$  (1), we get the formula of the whole life annuity (see [2–4]):

$$\bar{a}_x(\delta) = E(z(x)) = \frac{1}{\delta} \left( 1 - \frac{\Phi(x, \delta)}{S(x)} \right), \quad (2)$$

where  $E$  is the symbol of the mathematical expectation,  $S(x) = P(X > x)$  is a survival function,

$$\Phi(x, \delta) = e^{\delta x} \int_x^{\infty} e^{-\delta t} dF(t),$$

$F(x) = P(X \leq x) = 1 - S(x)$  is a distribution function.

## **1. Estimation of Annuity**

Suppose we have a random sample  $X_1, \dots, X_N$  of  $N$  individuals' lifetimes. Now, separately estimate the numerator and denominator in (2). The substitution of unknown function  $S(x)$  for its nonparametric estimator

$$S_N(x) = \frac{1}{N} \sum_{i=1}^N I(X_i > x),$$

where  $I(A)$  is the indicator of an event  $A$ , gives us the following estimators of the whole life annuity:

$$\bar{a}_x^N(\delta) = \frac{1}{\delta} \left( 1 - \frac{e^{\delta x}}{S_N(x) \cdot N} \sum_{i=1}^N \exp(-\delta X_i) I(X_i > x) \right) = \frac{1}{\delta} \left( 1 - \frac{\Phi_N(x, \delta)}{S_N(x)} \right), \quad (3)$$

$$\Phi_N(x, \delta) = \frac{e^{\delta x}}{N} \sum_{i=1}^N \exp(-\delta X_i) I(X_i > x).$$

## 2. Bias and MSE of $\bar{a}_x^N(\delta)$

In this section, we will obtain the principal term of the asymptotic MSE and the bias convergence rate of the estimator (3). Now introduce the notation according to [5]:  $t_N = (t_{1N}, t_{2N}, \dots, t_{sN})^T$  is an  $s$ -dimensional vector with components  $t_{jN} = t_{jN}(x) = t_{jN}(x; X_1, \dots, X_N)$ ,  $j = \overline{1, s}$ ,  $x \in R^\alpha$ ,  $R^\alpha$  is the  $\alpha$ -dimensional Euclidean space;  $H(t): R^s \rightarrow R^1$  is a function, where  $t = t(x) = (t_1(x), \dots, t_s(x))^T$  is an  $s$ -dimensional bounded vector function;  $N_s(\mu, \sigma)$  is the  $s$ -dimensional normally distributed random variable with a mean vector and covariance matrix  $\sigma = \sigma(x)$ ;  $\nabla H(t) = (H_1(t), \dots, H_s(t))^T$ ,  $H_j(t) = \frac{\partial H(z)}{\partial z_j} \Big|_{z=t}$ ,  $j = \overline{1, s}$ ,  $\Rightarrow$  is the symbol of convergence in distribution (weak convergence);  $\|x\|$  is the Euclidean norm of a vector  $x$ ,  $\mathfrak{N}$  is the set of natural numbers.

**Definition 1.** The function  $H(t): R^s \rightarrow R^1$  and the sequence  $\{H(t_N)\}$  are said to belong to class  $N_{v,s}(t; \gamma)$ , provided that:

1) there exists an  $\varepsilon$ -neighborhood

$$\sigma = \{z: |z_i - t_i| < \varepsilon, i = \overline{1, s}\},$$

in which the function  $H(z)$  and all its partial derivatives up to order  $v$  are continuous and bounded;

2) for any values of variables  $X_1, \dots, X_N$  the sequence  $\{H(t_N)\}$  is dominated by a numerical sequence  $C_0 d_N^\gamma$ , such that  $d_N \uparrow \infty$ , as  $N \rightarrow \infty$ , and  $0 \leq \gamma < \infty$ .

**Theorem 1 [5].** Let the conditions

1)  $H(z), \{H(t_N)\} \in N_{2,s}(t; \gamma)$ ,

2)  $E\|t_N - t\|^i = O(d_N^{-i/2})$

hold for all  $i \in \mathfrak{N}$ . Then, for every  $k \in \mathfrak{N}$ ,

$$\left| E[H(t_N) - H(t)]^k - E[\nabla H(t) \cdot (t_N - t)]^k \right| = O(d_N^{-(k+1)/2}). \quad (4)$$

Note, if in formula (4)  $k = 1$ , we obtain the principal term of the bias for  $H(t_N)$ , and at  $k = 2$ , we have the principal term of the MSE.

**Theorem 2.** If  $S(x) > 0$  and  $S(t)$  is continuous at  $x$ , then

1) for the bias of (3), the following relation holds:

$$\left| b(\bar{a}_x^N(\delta)) \right| = \left| E(\bar{a}_x^N(\delta) - \bar{a}_x(\delta)) \right| = O(N^{-1});$$

2) the MSE of (3) is given by the formula

$$u^2(\bar{a}_x^N(\delta)) = E(\bar{a}_x^N(\delta) - \bar{a}_x(\delta))^2 = \frac{\Phi(x, 2\delta) - \Phi^2(x, \delta) / S(x)}{N\delta^2 S^2(x)} + O(N^{-3/2}).$$

**Proof.** For the estimator  $\bar{a}_x^N(\delta)$  (3) in the notation of Theorem 1, we have:  $s = 2$ ;

$$\begin{aligned} t_N &= (t_{1N}, t_{2N})^T = (\Phi_N(x, \delta), S_N(x))^T; \quad d_N = N; \quad t = (t_1, t_2)^T = (\Phi(x, \delta), S(x))^T; \\ H(t) &= \frac{1}{\delta} \left( 1 - \frac{t_1}{t_2} \right) = \frac{1}{\delta} \left( 1 - \frac{\Phi(x, \delta)}{S(x)} \right) = \bar{a}_x(\delta); \quad H(t_N) = \frac{1}{\delta} \left( 1 - \frac{\Phi_N(x, \delta)}{S_N(x)} \right) = \bar{a}_x^N(\delta); \\ \nabla H(t) &= (H_1(t), H_2(t))^T = \left( -\frac{1}{\delta S(x)}, \frac{\Phi(x, \delta)}{\delta S^2(x)} \right)^T \neq 0. \end{aligned}$$

The sequence  $\{H(t_N)\}$  satisfies the condition 1) of Theorem 1 with  $C_0 = \frac{2}{\delta}$  and  $\gamma = 0$ . Indeed,

$$\begin{aligned} |H(t_N)| &= \frac{1}{\delta} \left| 1 - \frac{\Phi_N(x, \delta)}{S_N(x)} \right| \leq \frac{1}{\delta} \left( 1 + \frac{\Phi_N(x, \delta)}{S_N(x)} \right) \leq \frac{1}{\delta} \left( 1 + \frac{e^{\delta x} \sum_{i=1}^N \exp(-\delta X_i) \mathbf{I}(X_i > x)}{\sum_{i=1}^N \mathbf{I}(X_i > x)} \right) \leq \\ &\leq \frac{1}{\delta} \left( 1 + \frac{e^{\delta x} e^{-\delta x} \sum_{i=1}^N \mathbf{I}(X_i > x)}{\sum_{i=1}^N \mathbf{I}(X_i > x)} \right) = \frac{2}{\delta}. \end{aligned}$$

Further, the function  $H(t)$  satisfies the condition 1) in view of  $t_2 = S(x) > 0$ , the condition 2) due to Lemma 3.1 [6], provided that  $E\{\mathbf{I}^i(X > x)\} = S(x) \leq 1$ ,  $E\{e^{i\delta x} e^{-i\delta X} \mathbf{I}^i(X > x)\} \leq e^{i\delta x} e^{-i\delta x} S(x) = S(x) \leq 1$  for all  $i \in \mathfrak{R}$ .

We know that  $S_N(x)$  is an unbiased and consistent estimator of  $S(x)$ . Show that  $\Phi_N(x, \delta)$  is an unbiased estimator of the functional  $\Phi(x, \delta)$ :

$$E\Phi_N(x, \delta) = \frac{e^{\delta x}}{N} E \left\{ \sum_{i=1}^N \exp(-\delta X_i) \mathbf{I}(X_i > x) \right\} = \Phi(x, \delta).$$

Now, calculate the variance of  $\Phi_N(x, \delta)$ :

$$D\Phi_N(x, \delta) = D \left\{ \frac{e^{\delta x}}{N} \sum_{i=1}^N \mathbf{I}(X_i > x) e^{-\delta X_i} \right\} = \frac{e^{2\delta x}}{N^2} \sum_{i=1}^N D \left\{ \mathbf{I}(X_i > x) e^{-\delta X_i} \right\} = \frac{1}{N} \left( \Phi(x, 2\delta) - \Phi^2(x, \delta) \right).$$

The ratio of two unbiased estimators can have a bias. Considering that all the conditions of Theorem 1 are fulfilled and  $E(t_N - t) = 0$ , in accordance with (4) we get the order of the bias of  $\bar{a}_x^N(\delta)$ :

$$\left| E(\bar{a}_x^N(\delta) - \bar{a}_x(\delta)) - E[\nabla H(t)(t_N - t)] \right| = \left| E(\bar{a}_x^N(\delta) - \bar{a}_x(\delta)) \right| = O(N^{-1}).$$

Find the components of the covariance matrix  $\sigma(\bar{a}_x(\delta)) = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix}$  for the statistics  $\Phi_N(x, \delta), S_N(x)$ :

$$\begin{aligned} \sigma_{11} &= ND\{\Phi_N(x, \delta)\} = \Phi(x, 2\delta) - \Phi^2(x, \delta); \quad \sigma_{22} = ND\{S_N(x)\} = S(x)(1 - S(x)); \quad \sigma_{12} = \sigma_{21} = \\ &= N \text{cov}(S_N(x), \Phi_N(x, \delta)) = N \left( E\{S_N(x)\Phi_N(x, \delta)\} - E\{S_N(x)\}E\{\Phi_N(x, \delta)\} \right) = (1 - S(x))\Phi(x, \delta). \end{aligned}$$

Using the previous results on the bias and the covariance matrix, we obtain

$$\begin{aligned} u^2(\bar{a}_x^N(\delta)) &= E[\nabla H(t)(t_N - t)]^2 + O(N^{-3/2}) = \frac{\tilde{C}(\bar{a}_x(\delta))}{N} + O(N^{-3/2}), \\ \tilde{C}(\bar{a}_x(\delta)) &= H_1^2(t)\sigma_{11} + H_2^2(t)\sigma_{22} + 2H_1(t)H_2(t)\sigma_{12} = \frac{\Phi(x, 2\delta) - \Phi^2(x, \delta) / S(x)}{\delta^2 S^2(x)}. \end{aligned} \tag{5}$$

The proof is completed.



### 3. Asymptotic Normality of $\bar{a}_x^N(\delta)$

To find the limit distribution of (3), we need the following two Theorems.

**Theorem 3** (The usual central limit theorem) [7, Appendix 5]. If  $\xi_1, \xi_2, \dots, \xi_N, \dots$  is a sequence of independent and identically distributed  $s$ -dimensional vectors,  $E\{\xi_k\} = 0$ ,  $\sigma(x) = E\{\xi_k^T \xi_k\}$ ,  $t_N = \frac{1}{N} \sum_{k=1}^N \xi_k$ , then, as  $N \rightarrow \infty$ ,  $\sqrt{N} t_N \Rightarrow N_s(0, \sigma(x))$ .

**Theorem 4** (asymptotic normality of  $H(t_N)$ ) [2]. Let

- 1)  $\sqrt{N} \cdot t_N \Rightarrow N_s\{\mu, \sigma(x)\}$ ;
- 2)  $H(z)$  be differentiable at the point  $\mu$ ,  $\nabla H(\mu) \neq 0$ .

Then

$$\sqrt{N}(H(t_N) - H(\mu)) \Rightarrow N_1 \left( \sum_{j=1}^s H_j(\mu) \mu_j, \sum_{p=1}^s \sum_{j=1}^s H_j(\mu) \sigma_{jp} H_p(\mu) \right).$$

**Theorem 5.** Under the conditions of Theorem 2

$$\sqrt{N}(\bar{a}_x^N(\delta) - \bar{a}_x(\delta)) \Rightarrow N_1 \left( 0, \frac{\Phi(x, 2\delta) - \Phi^2(x, \delta)/S(x)}{\delta^2 S^2(x)} \right).$$

**Proof.** In the notation of Theorem 3, we have  $s = 2$ ,  $\sigma(x) = \sigma(\bar{a}_x(\delta))$ . Thus,

$$\sqrt{N} \{(\Phi_N(x, \delta), S_N(x)) - (\Phi(x, \delta), S(x))\} \Rightarrow N_2((0, 0), \sigma(\bar{a}_x(\delta))).$$

The function  $H(z)$  is differentiable at the point  $t = (\Phi(x, \delta), S(x))$  and  $\nabla H(t) \neq 0$ . Consequently, all the conditions of Theorem 4 hold, and using (5), we have  $\sqrt{N}(\bar{a}_x^N(\delta) - \bar{a}_x(\delta)) \Rightarrow N_1(0, \tilde{C}(\bar{a}_x(\delta)))$ .

Theorem 5 is proved.

### 4. Construction of Estimators Using Expected Lifetime

Suppose we know the expected lifetime

$$EX = a. \quad (6)$$

The estimator by making use of such information according to [8–17] can be taken in the following form:

$$\bar{a}_x^N(\delta, \lambda) = \frac{1}{\delta} \left( 1 - \frac{\Phi_N(x, \delta)}{S_N(x)} - \lambda(\bar{x} - a) \right), \quad (7)$$

where  $\bar{x} = \frac{1}{N} \sum_{i=1}^N X_i$  is an estimator of  $a$ , parameter  $\lambda$  we will find minimizing the principal term of the asymptotic MSE of  $\bar{a}_x^N(\delta, \lambda)$  (7). The estimator (7) combines the available empirical information containing in (3) and prior information (6).

For the estimator  $\bar{a}_x^N(\delta, \lambda)$  in the notation of Theorem 1, we have:  $s = 3$ ,

$$t_N = (t_{1N}, t_{2N}, t_{3N})^T = (\Phi_N(x, \delta), S_N(x), \bar{x})^T; \quad d_N = N; \quad t = (t_1, t_2, t_3)^T = (\Phi(x, \delta), S(x), a)^T;$$

$$H(t) = H(t_1, t_2, t_3) = \frac{1}{\delta} \left( 1 - \frac{t_1}{t_2} - \lambda(t_3 - a) \right) = \frac{1}{\delta} \left( 1 - \frac{\Phi(x, \delta)}{S(x)} - \lambda(a - a) \right) = \bar{a}_x(\delta);$$

$$H(t_N) = \frac{1}{\delta} \left( 1 - \frac{\Phi_N(x, \delta)}{S_N(x)} - \lambda(\bar{x} - a) \right) = \bar{a}_x^N(\delta, \lambda);$$

$$\nabla H(t) = (H_1(t), H_2(t), H_3(t))^T = \left( -\frac{1}{\delta S(x)}, \frac{\Phi(x, \delta)}{\delta S^2(x)}, -\frac{\lambda}{\delta} \right)^T \neq 0.$$

### 5. Bias and MSE of $\bar{a}_x^N(\delta, \lambda)$

Arguing as in Section 1, it is easy to show that the sequence  $\{H(t_N)\}$  satisfies the condition 1) of Theorem 1 with  $C_0 = \frac{2+|\lambda|(\omega+a)}{\delta}$ ,  $\omega < \infty$  is the limiting age, and  $\gamma = 0$ ; also, the statistic  $t_N$  satisfies the condition 2) due to Lemma 3.1 [6], provided that  $EX^i \leq \omega^i < \infty$  for all  $i \in \mathbb{N}$ . From here, for the bias of (7) we obtain the following result:

$$\left| E(\bar{a}_x^N(\delta, \lambda) - \bar{a}_x(\delta)) - E[\nabla H(t)(t_N - t)] \right| = \left| E(\bar{a}_x^N(\delta, \lambda) - \bar{a}_x(\delta)) \right| = \left| b(\bar{a}_x^N(\delta, \lambda)) \right| = O(N^{-1}).$$

Now, find the covariance matrix  $\sigma(\bar{a}_x(\delta, \lambda)) = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix}$  for statistics  $\Phi_N(x, \delta)$ ,  $S_N(x)$ ,  $\bar{x}$ :

$$\sigma_{33} = ND\{\bar{x}\} = D(X); \quad \sigma_{13} = \sigma_{31} = N \text{cov}(\bar{x}, \Phi_N(x, \delta)) = C_1(x, \delta) - a\Phi(x, \delta);$$

$$\sigma_{23} = \sigma_{32} = N \text{cov}(S_N(x), \bar{x}) = C_2(x) - aS(x), \quad \text{where } C_1(x, \delta) = e^{\delta x} \int_x^\infty e^{-\delta u} u dF(u), \quad C_2(x) = \int_x^\infty u dF(u), \quad \text{and } \sigma_{11},$$

$\sigma_{12}$ ,  $\sigma_{21}$ ,  $\sigma_{22}$  are defined in Section 1. Using (5), the above results for the bias and the covariance matrix  $\sigma(\bar{a}_x(\delta, \lambda))$ , we obtain:

$$\begin{aligned} u^2(\bar{a}_x^N(\delta, \lambda)) &= E[\nabla H(t)(t_N - t)]^2 + O(N^{-3/2}) = \frac{\tilde{C}(\bar{a}_x(\delta, \lambda))}{N} + O(N^{-3/2}), \\ \tilde{C}(\bar{a}_x(\delta, \lambda)) &= \sum_{p=1}^3 \sum_{j=1}^3 H_j(t) \sigma_{jp} H_p(t) = H_1^2(t) \sigma_{11} + H_2^2(t) \sigma_{22} + H_3^2(t) \sigma_{33} + 2H_1(t) H_2(t) \sigma_{12} + \\ &+ 2H_1(t) H_3(t) \sigma_{13} + 2H_2(t) H_3(t) \sigma_{23} = \tilde{C}(\bar{a}_x(\delta)) + \frac{\lambda^2 \sigma_{33}}{\delta^2} - \frac{2\lambda H_1 \sigma_{13}}{\delta} - \frac{2\lambda H_2 \sigma_{23}}{\delta} = \\ &= \tilde{C}(\bar{a}_x(\delta)) + \lambda^2 Q_1 - 2\lambda Q_2, \end{aligned} \quad (8)$$

where  $Q_1 = \frac{\sigma_{33}}{\delta^2} > 0$ ,  $Q_2 = \frac{H_1 \sigma_{13} + H_2 \sigma_{23}}{\delta}$ . Then the minimum of  $\tilde{C}(\bar{a}_x(\delta, \lambda))$  with respect to  $\lambda$  is achieved at  $\lambda_0 = \frac{Q_2}{Q_1}$ . Such  $\lambda_0$  minimizes the principal term of MSE  $u^2(\bar{a}_x^N(\delta, \lambda))$ , and this minimum is as follows:

$$\frac{\tilde{C}(\bar{a}_x(\delta, \lambda_0))}{N} = \frac{1}{N} \left( \tilde{C}(\bar{a}_x(\delta)) - \frac{Q_2^2}{Q_1} \right) < \frac{\tilde{C}(\bar{a}_x(\delta))}{N}. \quad (9)$$

### 6. Bias, MSE, and Asymptotic Normality of $\bar{a}_x^N(\delta, \lambda_0)$

In accordance with (9), the estimator

$$\bar{a}_x^N(\delta, \lambda_0) = \frac{1}{\delta} \left( 1 - \frac{\Phi_N(x, \delta)}{S_N(x)} - \lambda_0(\bar{x} - a) \right) \quad (10)$$

will be called the optimal (in the mean square sense) estimator. The non-negative quantity  $\frac{Q_2^2}{NQ_1}$  in (9) determines the decrease of the principal term of MSE for the optimal estimator by using auxiliary information (6).

**Theorem 6.** If  $S(x) > 0$  and  $S(t)$  is continuous at  $x$ , then

1) for the bias of (10), the following relation holds:

$$\left| b(\bar{a}_x^N(\delta, \lambda_0)) \right| = O(N^{-1});$$

2) the MSE of (10) is given by the formula

$$u^2(\bar{a}_x^N(\delta, \lambda_0)) = E(\bar{a}_x^N(\delta, \lambda_0) - \bar{a}_x(\delta))^2 = \frac{\tilde{C}(\bar{a}_x(\delta, \lambda_0))}{N} + O(N^{-3/2}),$$

where  $\tilde{C}(\bar{a}_x(\delta, \lambda_0))$  is defined by the formula (9).

**Theorem 7.** Under the conditions of Theorem 2

$$\sqrt{N}(\bar{a}_x^N(\delta, \lambda_0) - \bar{a}_x(\delta)) \Rightarrow N_1(0, \tilde{C}(\bar{a}_x(\delta, \lambda_0))).$$

**Proof.** The statements of Theorems 6 and 7 follow from Theorems 1 and 4 with the usage of the arguments of Sections 3–5.

## 7. Adaptive Estimator

The statistic  $\bar{a}_x^N(\delta, \lambda_0)$  can be used as an estimator for  $\bar{a}_x(\delta)$  if we know  $\lambda_0$ ; otherwise, it is required to construct an adaptive estimator. We need a more detailed formula for  $\lambda_0$ :

$$\lambda_0 = \frac{1}{S(x)DX} \left[ \frac{\Phi(x, \delta)}{S(x)} (C_2(x) - aS(x)) - C_1(x, \delta) + a\Phi(x, \delta) \right]. \quad (11)$$

Using (11), we consider the following adaptive estimator:

$$\bar{a}_x^N(\delta, \hat{\lambda}_0) = \frac{1}{\delta} \left( 1 - \frac{\Phi_N(x, \delta)}{S_N(x)} - \hat{\lambda}_0(\bar{x} - a) \right) \quad (12)$$

with

$$\hat{\lambda}_0 = \frac{1}{s^2 S_N(x)} \left[ \frac{\Phi_N(x, \delta)}{S_N(x)} (\hat{C}_2(x) - aS_N(x)) - \hat{C}_1(x, \delta) + a\Phi_N(x, \delta) \right], \quad (13)$$

where  $s^2 = \frac{1}{N-1} \sum_{i=1}^N (X_i - \bar{x})^2$  is an unbiased estimator of the variance  $DX$ ,

$$\hat{C}_2(x) = N^{-1} \sum_{i=1}^N X_i I(X_i > x), \quad \hat{C}_1(x, \delta) = N^{-1} \sum_{i=1}^N e^{-\delta X_i} X_i I(X_i > x).$$

**Theorem 8.** Under the conditions of Theorem 2,

$$\sqrt{N}(\bar{a}_x^N(\delta, \hat{\lambda}_0) - \bar{a}_x(\delta)) \Rightarrow N_1(0, \tilde{C}(\bar{a}_x(\delta, \lambda_0))).$$

**Proof.** The following equality holds:

$$\sqrt{N}(\bar{a}_x^N(\delta, \hat{\lambda}_0) - \bar{a}_x(\delta)) = \sqrt{N}(\bar{a}_x^N(\delta, \lambda_0) - \bar{a}_x(\delta)) + R_N,$$

where  $R_N = \delta^{-1}(\lambda_0 - \hat{\lambda}_0)\sqrt{N}(\bar{x} - a)$ . All the estimators, used in (13), converge almost surely to their true values according to the strong law of large numbers (the Second Theorem of Kolmogorov [18]). Thus, from the First Continuity Theorem of Borovkov [7], estimator  $\hat{\lambda}_0$  converges almost surely to  $\lambda_0$ . Based on the central limit theorem  $\sqrt{N}(\bar{x} - a) \Rightarrow N_1(0, DX)$ , we retrieve  $R_N \Rightarrow 0$ . Now, the statement of Theorem 8 is proved by making use of Theorem 7.

## Conclusion

The paper deals with the problem of estimating the present values of the continuous whole life annuity using auxiliary information about the expectation of life. It is shown that the usage of such auxiliary information can often provide the MSE smaller than that of standard estimators. We proved the results on asymptotic properties of the proposed estimators: unbiasedness, consistency and normality. Also, the main parts of the asymptotic MSEs of the estimators were found. An adaptive estimator is constructed; such estimator is equivalent (in the sense of asymptotic distribution) to the estimator with the optimal weight coefficient  $\lambda_0$ .

Note that the improved estimators of life annuities (3), (10) and (12) can be obtained by substituting of empirical survival functions by the smooth empirical survival functions (cf. [19–32]).

## REFERENCES

1. Falin, G.I. (2002) *Mathematical foundations of the theory of life insurance and pension schemes*. Moscow: Ankil Publ.
2. Koshkin, G.M. & Gubina, O.V. (2016) Estimation of the present values of life annuities for the different actuarial models. *Proc. The Second International Symposium on Stochastic Models, in Reliability Engineering, Life Science, and Operations Management*. SMRLO 2016. Beer Sheva, Israel, February 15–18, 2016. Conference Publishing Services The Institute of Electrical and Electronics Engineers, Inc. 2016. pp. 506–510.
3. Bowers, N., Gerber, H., Hickman, J., Jones, D. & Nesbitt, C. (1986) *Actuarial mathematics*. Itasca: Society of Actuaries.
4. Gerber, H. (1997) *Life insurance mathematics*. 3rd ed. New York: Springer-Verlag.
5. Koshkin, G.M. (1999) Deviation moments of the substitution estimator and its piecewise smooth approximations. *Siberian Mathematical Journal*. 40(3). pp. 515–527. DOI: 10.1007/BF02679759
6. Ibragimov, I.A. & Khasminskii, R.Z. (1981) *Statistical estimation: asymptotic theory*. Berlin-New York: Springer.
7. Borovkov, A.A. (1998) *Mathematical statistics*. New York: Gordon & Breach.
8. Dmitriev, Yu.G. & Koshkin, G.M. (1987) On the use of a priori information in nonparametric regression estimation. *IFAC Proc. Series*. 2. pp. 223–228.
9. Dmitriev, Yu.G. & Tarasenko, P.F. (2014) The use of a priori information in the statistical processing of experimental data. *Russian Physics Journal*. 35(9). pp. 888–893. DOI: 10.1007/BF00560063
10. Dmitriev, Yu.G., Tarassenko, P.F. & Ustinov, Y.K. (2014) On estimation of linear functional by utilizing a prior guess. In: Dudin, A., Nazarov, A., Yakupov, R., Gortsev, A. (eds) *Information Technologies and Mathematical Modelling. ITMM 2014. Communications in Computer and Information Science*. Vol 487. Springer, Cham. pp. 82–90.
11. Dmitriev, Yu.G. & Koshkin, G.M. (2017) On distribution functionals estimation with auxiliary information. *Applied Methods of Statistical Analysis*. pp. 9–18.
12. Dmitriev, Yu.G. & Koshkin, G.M. (2018) Nonparametric estimators of probability characteristics using unbiased prior conditions. *Statistical Papers*. pp. 1559–1575. DOI 10.1007/s00362-018-1044-7.
13. Levit, B.Ya. (1975) Conditional estimation of linear functionals. *Problems of Information Transmission*. 1(4). pp. 291–302.
14. Owen, A.B. (1991) Empirical likelihood for linear models. *The Annals of Statistics*. 19. pp. 1725–1747.
15. Qin, J. & Lawless, J. (1994) Empirical likelihood and general estimating equations. *The Annals of Statistics*. 22. pp. 300–325.
16. Tarima, S. & Pavlov, D. (2006) Using auxiliary information in statistical function estimation. *ESAIM: Probability and Statistics*. 10. pp. 11–23. DOI: 10.1051/ps:2005019
17. Zhang, B. (1995) M-estimation and quantile estimation in the presence of auxiliary information. *Journal of Statistical Planning and Inference*. 44. pp. 77–94. DOI: 10.1016/0378-3758(94)00040-3
18. Rao, C.R. (1965) *Linear statistical methods and its applications*. New York: Willey.
19. Nadaraya, E.A. (1964) Some new estimates of distribution function. *Theory of Probability and its Applications*. 9(3). pp. 497–500. DOI: 10.1137/1109069
20. Azzalini, A. (1981) A note on the estimation of a distribution function and quantiles by a kernel method. *Biometrika*. 68(1). pp. 326–328. DOI: 10.1093/biomet/68.1.326
21. Reiss, R.-D. (1981) Nonparametric estimation of smooth distribution functions. *Scand. J. Statist.* 8. pp. 116–119.
22. Falk, M. (1983) Relative efficiency and deficiency of kernel type estimators of smooth distribution functions. *Statistica Neerlandica*. 37. pp. 73–83. DOI: 10.1111/j.1467-9574.1983.tb00802.x
23. Swanepoel, J.W.H. (1988) Mean integrated squared error properties and optimal kernels when estimating a distribution function. *Communication in Statistics – Theory and Methods*. 17(11). pp. 3785–3799. DOI: 10.1080/03610928808829835
24. Jones, M.C. (1990) The performance of kernel density functions in kernel distribution function estimation. *Statistics and Probability Letters*. 9(2). pp. 129–132. DOI: 10.1016/0167-7152(92)90006-Q
25. Shirahata, S. & Chu, I.S. (1992) Integrated squared error of kernel-type estimator of distribution function. *Annals of the Institute of Statistical Mathematics*. 44(3). pp. 579–591. DOI: 10.1007/BF00050707
26. Sarda, P. (1993) Smoothing parameter selection for smooth distribution functions. *Journal of Statistical Planning and Inference*. 35. pp. 65–75. DOI: 10.1016/0378-3758(93)90068-H
27. Altman, N. & Leger, C. (1995) Bandwidth selection for kernel distribution function estimation. *Journal of Statistical Planning and Inference*. 46. pp. 195–214. DOI: 10.1016/0378-3758(94)00102-2
28. Bowman, A., Hall, P. & Prvan, T. (1998) Trust bandwidth selection for the smoothing of distribution functions. *Biometrika*. 85(4). pp. 799–808.
29. Chu, I.S. (1995) Bootstrap smoothing parameter selection for distribution function estimation. *Math. Japon.* 41(1). pp. 189–197.
30. Shao, Y. & Xiang, X. (1997) Some extensions of the asymptotics of a kernel estimator of a distribution function. *Statistics & Probability Letters*. vol. 34. pp. 301–308.
31. Una-Alvarez, J., Gonzalez-Manteiga, W. & Cadarso-Suarez, C. (2000) Kernel distribution function estimation under the Koziol–Green model. *Journal of Statistical Planning and Inference*. 87. pp. 199–219. DOI: 10.1016/S0378-3758(99)00192-5

32. Koshkin, G.M. (2014) Smooth estimators of the reliability functions for non-restorable elements. *Russian Physics Journal*. 57(5). pp. 672–681. DOI: 10.1007/s11182-014-0290-y

Received: March 29, 2018

Dmitriev Yu.G., Koshkin G.M. (2018) ESTIMATION OF PRESENT VALUE OF WHOLE LIFE ANNUITY USING INFORMATION ABOUT EXPECTATION OF LIFE. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitel'naja tehnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 45. pp. 22–29

DOI: 10.17223/19988605/45/3

Дмитриев Ю.Г., Кошкин Г.М. ОЦЕНИВАНИЕ ПОЖИЗНЕННОЙ РЕНТЫ С ИСПОЛЬЗОВАНИЕМ ИНФОРМАЦИИ О СРЕДНЕЙ ПРОДОЛЖИТЕЛЬНОСТИ ЖИЗНИ. *Вестник Томского государственного университета. Управление, вычислительная техника и информатика*. 2018. № 45. С. 22–29.

Рассматривается проблема оценивания актуарной непрерывной пожизненной ренты с использованием дополнительной информации о средней продолжительности жизни. По данным продолжительностей жизни индивидуумов строятся непараметрические оценки пожизненной ренты. Показано, что использование дополнительной информации приводит к среднеквадратической ошибке, меньшей, чем у стандартной оценки. Также предлагается адаптивная оценка. Показано, что адаптивная оценка эквивалентна в смысле асимптотического распределения оптимальной оценке. Доказана асимптотическая нормальность всех оценок.

Ключевые слова: непараметрическая оценка; пожизненная рента; дополнительная информация; среднеквадратическая ошибка, асимптотическая нормальность.

*DMITRIEV Yury Glebovich* (Doctor of Physics and Mathematics, Head of the Department of Theoretical Cybernetics of National Research Tomsk State University, Russian Federation).

E-mail: dmit70@mail.ru

*KOSHKIN Gennady Mikhailovich* (Doctor of Physics and Mathematics, Professor of the Department of Theoretical Cybernetics of National Research Tomsk State University, Russian Federation).

E-mail: kgm@mail.tsu.ru

Л.А. Нежелская, Е.Ф. Сидорова

## ОПТИМАЛЬНАЯ ОЦЕНКА СОСТОЯНИЙ ОБОБЩЕННОГО СИНХРОННОГО ПОТОКА СОБЫТИЙ ВТОРОГО ПОРЯДКА В УСЛОВИЯХ НЕПОЛНОЙ НАБЛЮДАЕМОСТИ

Рассматривается задача оптимальной оценки состояний обобщенного синхронного потока второго порядка. Определяются апостериорные вероятности состояний потока, учитывающие эффект непродлевающегося мертвого времени. Предлагается алгоритм оптимального оценивания состояний, основанный на критерии максимума апостериорной вероятности. Приводятся численные результаты статистических экспериментов, полученные с использованием расчетных аналитических формул и аппарата имитационного моделирования.

**Ключевые слова:** обобщенный синхронный поток событий второго порядка; состояние потока; оптимальная оценка состояний; непродлевающееся мертвое время; критерий максимума апостериорной вероятности.

На современном этапе развития компьютерной техники и информационных технологий происходит усложнение структуры телекоммуникационных систем связи, глобальных компьютерных сетей и спутниковых сетей связи, для которых характерны наиболее сложные и актуальные исследования в сфере систем массового обслуживания (СМО) [1, 2]. Исследованию входящих случайных потоков событий, математической моделью которых, в частности, являются дважды стохастические потоки событий, посвящены работы [3–8].

При изучении последних выделяют два основных класса задач – оценивание состояний потока [9–11] и оценивание его параметров [12–15] по наблюдаемым моментам наступления событий.

В работах [16, 17] решена задача оптимального оценивания состояний обобщенного синхронного потока событий второго порядка при его полной наблюдаемости. Однако наблюдаемое событие может вызвать период мертвого времени, т.е. повлечь за собой недоступность наблюдению в этот период последующих событий потока [18, 19]. Таким образом, в настоящей статье, являющейся непосредственным развитием работ [16, 17], решается задача об оптимальной оценке состояний потока в условиях его частичной наблюдаемости, т.е. при наличии непродлевающегося мертвого времени, по методу максимума апостериорной вероятности, которая представляет наиболее полную информацию о потоке событий, содержащуюся в наблюдаемой выборке, и обеспечивает минимум полной (безусловной) вероятности вынесения ошибочного решения о его состоянии [20, 21].

### 1. Постановка задачи

Рассматривается обобщенный синхронный дважды стохастический поток событий второго порядка (далее – поток), сопровождающий случайный процесс которого  $\lambda(t)$  является кусочно-постоянным с двумя состояниями  $S_1$  и  $S_2$ . Далее под  $i$ -м состоянием  $\lambda(t)$  понимается состояние  $S_i$ ,  $i = 1, 2$ .

Длительность интервала между событиями потока в  $i$ -м состоянии определяется случайной величиной  $\eta_i = \min(\xi_i^{(1)}, \xi_i^{(2)})$ , где случайная величина  $\xi_i^{(1)}$  распределена по закону  $F_i^{(1)}(t) = 1 - e^{-\lambda_i t}$ , случайная величина  $\xi_i^{(2)}$  – по закону  $F_i^{(2)}(t) = 1 - e^{-\alpha_i t}$ ,  $i = 1, 2$ ;  $\xi_i^{(1)}$  и  $\xi_i^{(2)}$  – независимые случайные величины. В момент наступления события потока в зависимости от того, какое значение приняла случайная величина  $\eta_i$ ,  $i = 1, 2$ , процесс  $\lambda(t)$  переходит из  $i$ -го состояния в  $j$ -е,  $i \neq j$ , или остается

в  $i$ -м состоянии,  $i = j$ , с вероятностью  $P_1^{(1)}(\lambda_j | \lambda_i)$  либо  $P_1^{(2)}(\lambda_j | \lambda_i)$ ,  $i, j = 1, 2$ . При этом  $P_1^{(1)}(\lambda_j | \lambda_i) + P_1^{(1)}(\lambda_i | \lambda_i) = 1$ ,  $P_1^{(2)}(\lambda_j | \lambda_i) + P_1^{(2)}(\lambda_i | \lambda_i) = 1$ ,  $i, j = 1, 2$ ,  $i \neq j$ . Таким образом, длительность интервала между событиями потока в  $i$ -м состоянии процесса  $\lambda(t)$  является случайной величиной с функцией распределения  $F_i(t) = 1 - e^{-(\lambda_i + \alpha_i)t}$ ,  $i = 1, 2$ .

В дальнейшем принимается, что имеет место состояние  $S_i$ , если  $\lambda(t) = \lambda_i$ ,  $i = 1, 2$  ( $\lambda_1 > \lambda_2 \geq 0$ ).

Подчеркнем, что  $\lambda(t)$  – принципиально ненаблюдаемый случайный процесс (скрытый марковский процесс), матрицы инфинитезимальных характеристик которого имеют вид:

$$D_0 = \begin{bmatrix} -(\lambda_1 + \alpha_1) & 0 \\ 0 & -(\lambda_2 + \alpha_2) \end{bmatrix},$$

$$D_1 = \begin{bmatrix} \lambda_1 P_1^{(1)}(\lambda_1 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_1 | \lambda_1) & \lambda_1 P_1^{(1)}(\lambda_2 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_2 | \lambda_1) \\ \lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2) & \lambda_2 P_1^{(1)}(\lambda_2 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_2 | \lambda_2) \end{bmatrix}.$$

Диагональные элементы матрицы  $D_0$  имеют смысл интенсивностей выхода процесса  $\lambda(t)$  из своих состояний, взятые с противоположным знаком, недиагональные элементы – интенсивностей переходов из состояния в состояние без наступления события. Элементами матрицы  $D_1$  выступают интенсивности переходов процесса  $\lambda(t)$  из состояния в состояние с наступлением события потока.

Имеет место ситуация недоступности наблюдению всех событий потока – каждое зарегистрированное в момент времени  $t_k$  событие порождает время фиксированной длительности  $T$  (мертвое время), в течение которого другие события исходного потока теряются. Наступившие в течение мертвого времени события не вызывают продления его периода, т.е. рассматривается непродлевающееся мертвое время, по окончании которого первое наступившее событие вновь вызывает период ненаблюдаемости длительности  $T$  и т.д.

Вариант возникающей ситуации приведен на рис. 1, где  $S_1, S_2$  – состояния процесса  $\lambda(t)$ ,  $t_1, t_2, \dots$  – моменты наступления событий в наблюдаемом потоке, штриховкой обозначены периоды мертвого времени; белыми кружками обозначены наблюдаемые события, черными – ненаблюдаемые.

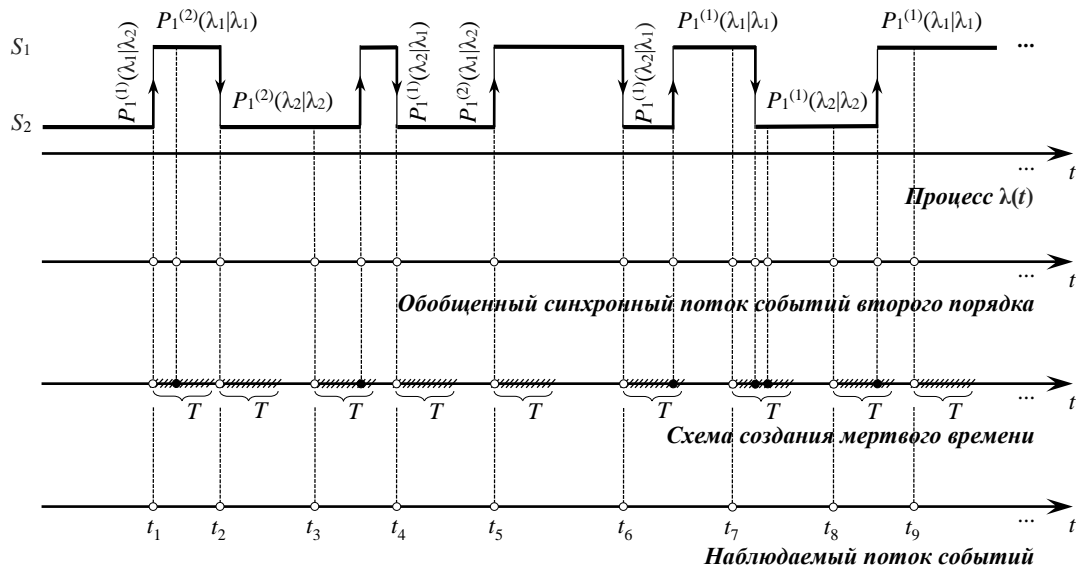


Рис. 1. Формирование наблюдаемого потока событий

Цель настоящего исследования заключается в том, чтобы по реализации моментов наступления событий  $t_1, t_2, \dots$  на  $(0, t)$  определить апостериорные вероятности  $w(\lambda_i | t) = w(\lambda_i | t_1, \dots, t_m, t)$  того,

что в момент окончания наблюдений  $t$  значение процесса  $\lambda(t) = \lambda_i$ ,  $i = 1, 2$ , а также на основании полученных формул для  $w(\lambda_i | t)$  ( $w(\lambda_1 | t) + w(\lambda_2 | t) = 1$ ) разработать алгоритм оценивания состояний потока (процесса  $\lambda(t)$ ).

## 2. Оптимальное оценивание состояний потока событий

Момент вынесения решения  $t$  о состоянии процесса  $\lambda(t)$  принадлежит интервалу  $(t_k, t_{k+1})$ ,  $k = 1, 2, \dots$ , между соседними событиями наблюдаемого потока. Для интервала  $(t_0, t_1)$  момент  $t$  лежит между моментом начала наблюдений  $t_0$  и моментом наступления первого событий потока  $t_1$ . Длительность интервала  $(t_k, t_{k+1})$  есть величина  $\tau_k = t_{k+1} - t_k$ ,  $k = 0, 1, \dots$ . С другой стороны, так как событие, наблюдаемое в момент времени  $t_k$ , порождает период мертвого времени длительности  $T$ , то  $\tau_k = T + \varsigma_k$ , где  $\varsigma_k$  – значение длительности интервала между моментом окончания периода мертвого времени  $t_k + T$  и моментом наступления следующего события потока  $t_{k+1}$ , т.е. имеет место разбиение исходного интервала  $(t_k, t_{k+1})$  на два смежных: первый – полуинтервал  $(t_k, t_k + T]$ , второй – интервал  $(t_k + T, t_{k+1})$ . Подчеркнем, что условия нахождения апостериорной вероятности  $w(\lambda_1 | t)$  первого состояния процесса  $\lambda(t)$  на полуинтервале  $(t_k, t_k + T]$  и интервале  $(t_k + T, t_{k+1})$  принципиально разные – на полуинтервале  $(t_k, t_k + T]$  поток недоступен наблюдению, в то время как на интервале  $(t_k + T, t_{k+1})$  поток наблюдаем. Кроме того, для нахождения вероятности  $w(\lambda_1 | t)$  необходимо точно знать значение длительности мертвого времени  $T$  либо предварительно осуществить его оценку, в противном случае отсутствие такой информации делает невозможным нахождение апостериорной вероятности  $w(\lambda_1 | t)$  в явном виде. Уточним, что в настоящем исследовании предполагается, что значение  $T$  фиксировано и известно точно.

Ситуация полной наблюдаемости исследуемого потока событий, т.е.  $T = 0$ , рассмотрена в работах [16, 17]. В [17] показано, что априорная вероятность  $\pi_1(t | t^0)$  того, что процесс  $\lambda(t)$  в момент времени  $t$  принимает значение  $\lambda_1$  при условии, что функционирование потока началось в момент времени  $t^0$ , определяется дифференциальным уравнением

$$\begin{aligned} \pi_1'(t | t^0) = & -(\lambda_1 + \alpha_1)\pi_1(t | t^0) + \pi_1(t | t^0)(\lambda_1 P_1^{(1)}(\lambda_1 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_1 | \lambda_1)) + \\ & + \pi_2(t | t^0)(\lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2)), \pi_2(t | t^0) = 1 - \pi_1(t | t^0). \end{aligned} \quad (1)$$

Априорная финальная вероятность  $\pi_1$  первого состояния процесса  $\lambda(t)$  является решением уравнения (1) при  $t \rightarrow \infty$  (или  $t^0 \rightarrow -\infty$ ) и определяется явной формулой [17]

$$\begin{aligned} \pi_1 = & [\lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2)] / a, \\ a = & \lambda_1 P_1^{(1)}(\lambda_2 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_2 | \lambda_1) + \lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2). \end{aligned} \quad (2)$$

Апостериорная вероятность первого состояния процесса  $\lambda(t)$  в момент  $t_k$  наступления события потока задается формулой пересчета [Там же]

$$\begin{aligned} w(\lambda_1 | t_k + 0) = & W / [(\lambda_2 + \alpha_2) + w(\lambda_1 | t_k - 0)(\lambda_1 + \alpha_1 - \lambda_2 - \alpha_2)], k = 1, 2, \dots, \\ W = & \lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2) + w(\lambda_1 | t_k - 0) \times \\ & \times [\lambda_1 P_1^{(1)}(\lambda_1 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_1 | \lambda_1) - \lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) - \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2)]. \end{aligned} \quad (3)$$

Апостериорная вероятность первого состояния процесса  $\lambda(t)$  на временных интервалах  $(t_0, t_1)$  и  $(t_k, t_{k+1})$ ,  $k = 1, 2, \dots$ , определяется формулой [Там же]



$$w(\lambda_1 | t) = \frac{w(\lambda_1 | t_k + 0) e^{-(\lambda_1 + \alpha_1 - \lambda_2 - \alpha_2)(t - t_k)}}{1 - w(\lambda_1 | t_k + 0) + w(\lambda_1 | t_k + 0) e^{-(\lambda_1 + \alpha_1 - \lambda_2 - \alpha_2)(t - t_k)}}, \quad (4)$$

$t_k < t < t_{k+1}$ ,  $k = 0, 1, \dots$ ;  $w(\lambda_1 | t_k + 0)$ ,  $k = 1, 2, \dots$ , определена в (3),  $w(\lambda_1 | t_0 + 0) = \pi_1$ ,  $\pi_1$  задается (2).

Обратимся к случаю, когда поток функционирует в условиях недоступности наблюдению всех событий потока, т.е. значение длительности мертвого времени  $T \neq 0$ .

Рассматривается полуинтервал  $(t_k, t_k + T]$ ,  $k = 1, 2, \dots$ , на котором событие потока имеет место в граничной точке  $t_k$ ; на самом полуинтервале события отсутствуют.

**Теорема.** Поведение апостериорной вероятности  $w(\lambda_1 | t)$  для обобщенного синхронного потока событий второго порядка на полуинтервалах  $(t_k, t_k + T]$ ,  $k = 1, 2, \dots$ , определяется формулой

$$w(\lambda_1 | t) = \pi_1 + [w(\lambda_1 | t_k + 0) - \pi_1] e^{-a(t - t_k)}, \quad (5)$$

$t_k < t \leq t_k + T$ ;  $\pi_1$  и  $a$  задаются (2),  $w(\lambda_1 | t_k + 0)$  определяется формулой пересчета (3).

**Доказательство.** На полуинтервале  $(t_k, t_k + T]$ ,  $k = 1, 2, \dots$ , обобщенный синхронный поток событий второго порядка является ненаблюдаемым. Вследствие этого поведение апостериорной вероятности  $w(\lambda_1 | t)$  в течение периода мертвого времени  $T$  аналогично поведению априорной вероятности  $\pi_1(t | t^0)$  первого состояния процесса  $\lambda(t)$ . Отличие заключается в задании начального значения  $w(\lambda_1 | t)$  в момент времени  $t_k$  наступления наблюдаемого события потока. Можно показать, что вероятность  $w(\lambda_1 | t)$  на полуинтервале  $(t_k, t_k + T]$ ,  $k = 1, 2, \dots$ , определяется уравнением (1) с начальным условием  $w(\lambda_1 | t = t_k) = w(\lambda_1 | t_k + 0)$ ,  $k = 1, 2, \dots$ . Интегрируя (1) [22], находим (5). Теорема доказана.

**Утверждение.** На интервале  $(t_k + T, t_{k+1})$ , смежном с временным полуинтервалом  $(t_k, t_k + T]$ , вычисление апостериорной вероятности  $w(\lambda_1 | t)$  осуществляется по формуле

$$w(\lambda_1 | t) = \frac{w(\lambda_1 | t_k + T) e^{-(\lambda_1 + \alpha_1 - \lambda_2 - \alpha_2)(t - t_k - T)}}{1 - w(\lambda_1 | t_k + T) + w(\lambda_1 | t_k + T) e^{-(\lambda_1 + \alpha_1 - \lambda_2 - \alpha_2)(t - t_k - T)}}, \quad (6)$$

$t_k + T < t < t_{k+1}$ ,  $k = 1, 2, \dots$ ; т.е. начальное условие для  $w(\lambda_1 | t)$  привязывается к моменту времени  $t_k + T$ , другими словами, в формуле (4)  $w(\lambda_1 | t_k + 0)$  заменяется на  $w(\lambda_1 | t_k + T)$ , вычисленную по (5) при  $t = t_k + T$ ,  $t_k + T \leq t < t_{k+1}$ ,  $k = 1, 2, \dots$

В момент  $t_k$  наступления очередного события наблюдаемого потока, порождающего период мертвого времени, вероятность  $w(\lambda_1 | t_k + 0)$  рассчитывается по формуле пересчета (3).

Полученные формулы позволяют сформулировать алгоритм расчета апостериорной вероятности  $w(\lambda_1 | t)$  ( $w(\lambda_2 | t) = 1 - w(\lambda_1 | t)$ ) и алгоритм принятия решения о состоянии процесса  $\lambda(t)$  (потока) в произвольный момент времени  $t$ :

1) в момент начала наблюдений за потоком  $t_0 = 0$  задается  $w(\lambda_1 | t_0 + 0) = w(\lambda_1 | t_0 = 0) = \pi_1$ , где  $\pi_1$  вычисляется по формуле (2);

2) по формуле (4) для  $k = 0$  в любой момент времени  $t$ ,  $t_0 < t < t_1$ , где  $t_1$  – момент наблюдения первого события потока, рассчитывается вероятность  $w(\lambda_1 | t)$ , при этом начальное условие  $w(\lambda_1 | t_0 + 0)$  для  $w(\lambda_1 | t)$  определено на предыдущем шаге;

3) в момент времени  $t_1$  по формуле (4) для  $k = 0$  вычисляется  $w(\lambda_1 | t_1) = w(\lambda_1 | t_1 - 0)$ ;

4)  $k$  увеличивается на единицу; с использованием формулы пересчета (3) для апостериорной вероятности для  $k = 1$  рассчитывается  $w(\lambda_1 | t_1 + 0)$  – начальное условие для  $w(\lambda_1 | t)$  в формуле (5);

5) в любой момент  $t$ ,  $t_1 < t < t_1 + T$ , по формуле (5) для  $k = 1$  вычисляется  $w(\lambda_1 | t)$ ;

- 6) по формуле (5) для  $k=1$  рассчитывается  $w(\lambda_1 | t=t_1+T)$  – начальное условие для  $w(\lambda_1 | t)$  на следующем шаге алгоритма;
- 7) для  $k=1$  по формуле (6) вычисляется  $w(\lambda_1 | t)$  в любой момент времени  $t$ ,  $t_1+T < t < t_2$ , где  $t_2$  – момент наблюдения второго события потока;
- 8) в момент времени  $t_2$  по формуле (6) для  $k=1$  рассчитывается  $w(\lambda_1 | t_2) = w(\lambda_1 | t_2 - 0)$ ;
- 9) алгоритм переходит на шаг 4, шаги 4–8 последовательно повторяются для  $k=2$  и т.д.
- Параллельно по ходу вычисления  $w(\lambda_1 | t)$  в любой момент времени  $t$  выносится решение о состоянии процесса  $\lambda(t)$  (потока) по критерию максимума апостериорной вероятности [20].

### 3. Результаты численных расчетов

Для получения численных результатов разработанный алгоритм вычисления вероятности  $w(\lambda_1 | t)$  реализован средствами языка C#, Microsoft Visual Studio 2013, посредством последовательного выполнения двух этапов – имитационного моделирования [23] обобщенного синхронного потока второго порядка с непродлевающимся мертвым временем и вычисления вероятностей  $w(\lambda_1 | t)$ ,  $t_0 \leq t < t_1$ ;  $w(\lambda_1 | t_k + 0)$  и  $w(\lambda_1 | t)$ ,  $t_k < t \leq t_k + T$ ;  $w(\lambda_1 | t)$ ,  $t_k + T < t \leq t_{k+1}$ ,  $k=1, 2, \dots$ , а также определения оценок  $\hat{\lambda}(t)$  траекторий истинного процесса  $\lambda(t)$ .

С целью установления частоты ошибочных решений о состоянии  $\lambda(t)$  по наблюдениям за потоком событий проведен ряд статистических экспериментов, состоящих из следующих этапов:

- 1) для определенного набора параметров  $\lambda_1, \lambda_2, \alpha_1, \alpha_2, P_1^{(1)}(\lambda_j | \lambda_i), P_1^{(2)}(\lambda_j | \lambda_i), i, j=1, 2$ ,  $T$  единиц времени осуществляется имитационное моделирование наблюдаемого потока событий на заданном отрезке времени  $[0, T_m]$  (отдельная  $k$ -я реализация (испытание));
- 2) осуществляется расчет апостериорной вероятности  $w(\lambda_1 | t)$  первого состояния процесса  $\lambda(t)$  на отрезке  $[0, T_m]$  по формулам (2)–(6);
- 3) оценивается траектория процесса  $\lambda(t)$  на  $[0, T_m]$  (оценивание на отрезке  $[0, T_m]$  временных интервалов, когда  $\hat{\lambda}(t)$  принимает то или иное значение);
- 4) осуществляется определение (для  $k$ -го испытания)  $d_k$  – суммарной протяженности интервалов, на которых значение процесса  $\lambda(t)$  не совпадает с его оценкой  $\hat{\lambda}(t)$ ;
- 5) вычисляется доля ошибочных решений  $\hat{p}_k = d_k / T_m$ ;
- 6) производится повторение  $N$  раз ( $k = \overline{1, N}$ ) шагов 1–5 для расчета оценки полной (безусловной) вероятности ошибки принятия решения о состояниях процесса  $\lambda(t)$  на  $[0, T_m]$ .

Результатом выполнения описанного алгоритма является выборка  $\hat{p}_1, \hat{p}_2, \dots, \hat{p}_N$  долей ошибочных решений в  $N$  испытаниях. По этому набору вычисляются выборочное среднее безусловной вероятности ошибочного решения  $\hat{P}_{ош}$  и ее выборочная дисперсия  $\hat{D}_{ош}$ :

$$\hat{P}_{ош} = \frac{1}{N} \sum_{k=1}^N \hat{p}_k, \quad \hat{D}_{ош} = \frac{1}{N-1} \sum_{k=1}^N (\hat{p}_k - \hat{P}_{ош})^2.$$

Первый статистический эксперимент проведен с целью установления времени моделирования  $T_m$ , обеспечивающего удовлетворение свойства стационарности потока событий. Результаты работы имитационной модели получены для значений параметров, приведенных в табл. 1, в условиях полной наблюдаемости потока ( $T=0$  – табл. 2, рис. 2, а) и условиях наличия мертвого времени ( $T=1$  – табл. 3, рис. 2, б). В настоящем эксперименте количество испытаний фиксировано,  $N=100$ .

Таблица 1

Исходные данные для эксперимента

$\lambda_1 = 4, \lambda_2 = 0,25$	$P_1^{(1)}(\lambda_1   \lambda_1) = 0,2$	$P_1^{(1)}(\lambda_2   \lambda_2) = 0,4$	$P_1^{(2)}(\lambda_1   \lambda_1) = 0,3$	$P_1^{(2)}(\lambda_2   \lambda_2) = 0,1$
$\alpha_1 = 4, \alpha_2 = 0,25$	$P_1^{(1)}(\lambda_2   \lambda_1) = 0,8$	$P_1^{(1)}(\lambda_1   \lambda_2) = 0,6$	$P_1^{(2)}(\lambda_2   \lambda_1) = 0,7$	$P_1^{(2)}(\lambda_1   \lambda_2) = 0,9$

Таблица 2

 Результаты первого статистического эксперимента ( $T = 0$ )

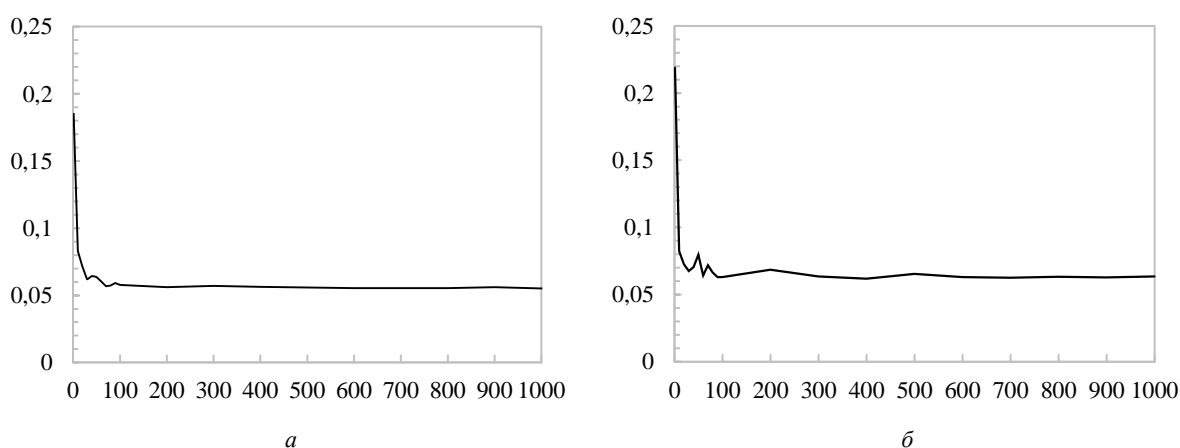
$T_m$	1	10	20	30	40	50	60	70	80	...	1 000
$\hat{P}_{ои}$	0,1849	0,0830	0,0716	0,0617	0,0645	0,0638	0,0603	0,0569	0,0574	...	0,0552
$\hat{D}_{ои} \cdot 10^3$	2,4681	1,6610	0,9100	0,4771	0,3600	0,3790	0,2841	0,1560	0,2101	...	0,0012

Таблица 3

 Результаты первого статистического эксперимента ( $T = 1$ )

$T_m$	1	10	20	30	40	50	60	70	80	...	1 000
$\hat{P}_{ои}$	0,2188	0,0822	0,0726	0,0675	0,0704	0,0897	0,0642	0,0729	0,0646	...	0,0635
$\hat{D}_{ои} \cdot 10^3$	3,1527	2,0270	1,1331	0,7093	0,6412	0,5932	0,3091	0,2780	0,1911	...	0,0021

На рис. 2 отражена зависимость оценки полной вероятности принятия ошибочного решения  $\hat{P}_{ои}$  от времени моделирования  $T_m$  при различных значениях мертвого времени:  $T = 0$  и  $T = 1$ .


 Рис. 2. График зависимости  $\hat{P}_{ои}$  от значения  $T_m$  при  $T = 0$  (а) и  $T = 1$  (б)

Анализ многочисленных результатов эксперимента, включая численные результаты табл. 2–3 и их графическое представление (см. рис. 2), позволяет сделать следующие выводы:

- 1) при увеличении времени моделирования  $T_m$  оценки  $\hat{P}_{ои}$  (в каждом из рассмотренных условий функционирования потока) стабилизируются;
- 2) в случае полной наблюдаемости потока (см. рис. 2, а) оценка полной вероятности ошибки  $\hat{P}_{ои}$  для всех вариантов расчета является достаточно стабильной при  $T_m \geq 100$  ед. времени, в случае наличия мертвого времени (см. рис. 2, б) – при  $T_m \geq 500$ ; как следствие, для дальнейших экспериментов значение времени моделирования выбрано равным  $T_m = 1000$  ед. времени;
- 3) для всех вариантов расчета при  $T_m \geq 100$  ед. времени оценка  $\hat{D}_{ои}$  достаточно мала.

Так как согласно  $D_0$  переход  $\lambda(t)$  из состояния  $i$  в состояние  $j$ ,  $i, j = 1, 2$ , в соответствии с [1, 8] невозможен без наступления события, то во втором статистическом эксперименте исследуется

зависимость  $\hat{P}_{ош}$  и  $\hat{D}_{ош}$  от  $\lambda_1/\lambda_2$  и  $\alpha_1/\alpha_2$ . При  $T=1$ ,  $T_m=1000$ ,  $N=100$  и исходных данных табл. 4 получены результаты, приведенные в табл. 5, 6.

Таблица 4

Исходные данные для эксперимента

$\lambda_1 = 6$ , $\lambda_1/\lambda_2 = 3; \dots; 192$	$P_1^{(1)}(\lambda_1   \lambda_1) = 0,6$	$P_1^{(1)}(\lambda_2   \lambda_2) = 0,5$	$P_1^{(2)}(\lambda_1   \lambda_1) = 0,3$	$P_1^{(2)}(\lambda_2   \lambda_2) = 0,8$
$\alpha_1 = 6$ , $\alpha_1/\alpha_2 = 3; \dots; 192$	$P_1^{(1)}(\lambda_2   \lambda_1) = 0,4$	$P_1^{(1)}(\lambda_1   \lambda_2) = 0,5$	$P_1^{(2)}(\lambda_2   \lambda_1) = 0,7$	$P_1^{(2)}(\lambda_1   \lambda_2) = 0,2$

Таблица 5

Результаты второго статистического эксперимента (для  $\lambda_1/\lambda_2$ ,  $\alpha_2 = 2$ )

$\lambda_1/\lambda_2$	3	6	12	24	48	96	192
$\hat{P}_{ош}$	0,2038	0,1439	0,1104	0,0914	0,0821	0,0765	0,0745
$\hat{D}_{ош} \cdot 10^3$	0,0045	0,0032	0,0033	0,0022	0,0022	0,0027	0,0022

Таблица 6

Результаты второго статистического эксперимента (для  $\alpha_1/\alpha_2$ ,  $\lambda_2 = 2$ )

$\alpha_1/\alpha_2$	3	6	12	24	48	96	192
$\hat{P}_{ош}$	0,2030	0,1855	0,1737	0,1689	0,1658	0,1641	0,1629
$\hat{D}_{ош} \cdot 10^3$	0,0044	0,0039	0,0042	0,0038	0,0041	0,0039	0,0040

Результаты эксперимента свидетельствуют о лучшем оценивании состояний потока событий при больших значениях отношений  $\lambda_1/\lambda_2$  и  $\alpha_1/\alpha_2$ . Такая тенденция связана с тем фактом, что состояния  $\lambda(t)$  становятся лучше различимы, за счет чего частота принятия ошибочных решений значительно сокращается. Отметим, что данные табл. 5 определяют лучшее в смысле малости оценки полной вероятности ошибки качество оценивания рассматриваемого потока событий, нежели данные табл. 6; другими словами, увеличение значения отношения  $\lambda_1/\lambda_2$  обеспечивает меньшее значение оценки вероятности принятия ошибочного решения, чем увеличение значения отношения  $\alpha_1/\alpha_2$ . Последнее связано с заданным набором параметров, определяющим поток (см. табл. 4). Обратим внимание, что сравнение приведенных в табл. 5–6 результатов статистического эксперимента с результатами аналогичного эксперимента, проведенного для тех же исходных данных в условиях отсутствия мертвого времени,  $T = 0$  [16, 17], позволяет отметить увеличение значения  $\hat{P}_{ош}$  при  $T \neq 0$  для каждого варианта расчета, что естественно в силу потери полезной информации для вынесения решения о состоянии потока за счет наличия мертвого времени.

Предметом рассмотрения в третьем статистическом эксперименте является зависимость  $\hat{P}_{ош}$  от значения длительности  $T$ , при этом рассматриваются ситуации увеличения значений  $\lambda_1$  и  $\alpha_1$ , определяющих интенсивность наступления событий в первом состоянии. В табл. 8–9 приведены результаты, полученные при  $T_m = 1000$ ,  $N = 100$  и значениях параметров, представленных в табл. 7.

Таблица 7

Исходные данные для эксперимента

$\lambda_1 = 4; 6$ , $\lambda_2 = 0,5$	$P_1^{(1)}(\lambda_1   \lambda_1) = 0,2$	$P_1^{(1)}(\lambda_2   \lambda_2) = 0,1$	$P_1^{(2)}(\lambda_1   \lambda_1) = 0,1$	$P_1^{(2)}(\lambda_2   \lambda_2) = 0,2$
$\alpha_1 = 4; 6$ , $\alpha_2 = 0,5$	$P_1^{(1)}(\lambda_2   \lambda_1) = 0,8$	$P_1^{(1)}(\lambda_1   \lambda_2) = 0,9$	$P_1^{(2)}(\lambda_2   \lambda_1) = 0,9$	$P_1^{(2)}(\lambda_1   \lambda_2) = 0,8$

Таблица 8

Результаты третьего статистического эксперимента ( $\lambda_1 = 4, \alpha_1 = 4$ )

$T$	0	0,5	1,0	1,5	2,0	2,5	3,0	3,5
$\hat{P}_{ои}$	0,0871	0,1091	0,1146	0,1178	0,1204	0,1216	0,1223	0,1243
$\hat{D}_{ои} \cdot 10^3$	0,0026	0,0022	0,0022	0,0023	0,0021	0,0025	0,0021	0,0031

Таблица 9

Результаты третьего статистического эксперимента ( $\lambda_1 = 6, \alpha_1 = 6$ )

$T$	0	0,5	1,0	1,5	2,0	2,5	3,0	3,5
$\hat{P}_{ои}$	0,0610	0,0817	0,0856	0,0886	0,0898	0,0910	0,0925	0,0935
$\hat{D}_{ои} \cdot 10^3$	0,0009	0,0014	0,0012	0,0017	0,0017	0,0012	0,0014	0,0016

Построенный согласно данным табл. 8–9 график зависимости оценки вероятности  $\hat{P}_{ои}$  от значений длительности мертвого времени  $T$  приведен на рис. 3.

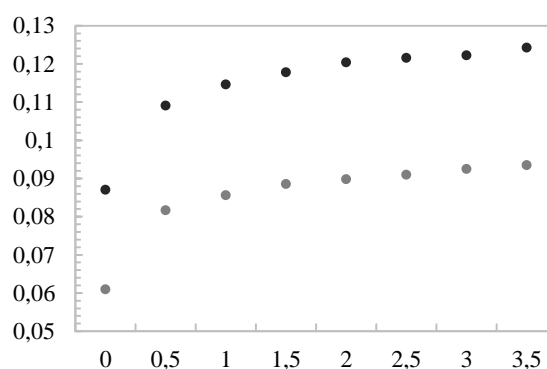
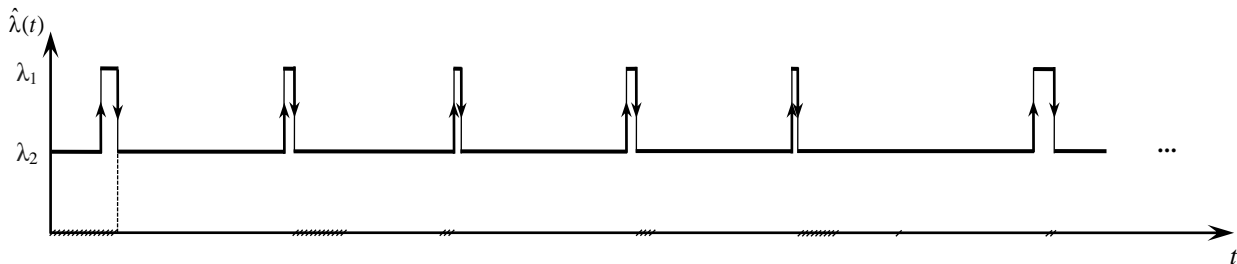


Рис. 3. График зависимости  $\hat{P}_{ои}$  от  $T$  при  $\lambda_1 = \alpha_1 = 4$  – темным маркером, при  $\lambda_1 = \alpha_1 = 6$  – светлым маркером

Численные результаты, отраженные в табл. 8–9, и их графическое представление (см. рис. 3) демонстрируют, что при одном и том же значении  $T$  отмечается тенденция к уменьшению оценки  $\hat{P}_{ои}$  в случае увеличения  $\lambda_1$  и  $\alpha_1$ , так как условия различимости состояний улучшаются, что также полностью согласуется с результатами предыдущего эксперимента. Рассмотрение значений  $\hat{P}_{ои}$  в обоих случаях задания интенсивностей ( $\lambda_1 + \alpha_1 = 8, \lambda_1 + \alpha_1 = 12$ ) в зависимости от  $T = 0; 0,5; \dots; 3,5$  приводит к обнаружению роста  $\hat{P}_{ои}$  с увеличением  $T$ , так как увеличение продолжительности мертвого времени всегда означает уменьшение количества полезной информации, что отрицательно сказывается на качестве оценивания (это также можно видеть на рис. 2). При этом заметим, что поведение оценок  $\hat{P}_{ои}(\lambda_1 + \alpha_1 = 8)$  (темный маркер на рис. 3) и  $\hat{P}_{ои}(\lambda_1 + \alpha_1 = 12)$  (светлый маркер на рис. 3) в целом аналогично.

Ввиду вероятностного механизма смены состояний потока в каждый момент наступления события особый интерес (четвертый статистический эксперимент) представляет рассмотрение частного случая задания вероятностей  $P_1^{(l)}(\lambda_j | \lambda_i)$ ,  $i, j = 1, 2, l = 1, 2$ , определяющих переходы процесса  $\lambda(t)$  из состояния  $S_i$  в состояние  $S_j$ ,  $i, j = 1, 2$ :  $P_1^{(l)}(\lambda_j | \lambda_i) = 0, i = j, P_1^{(l)}(\lambda_j | \lambda_i) = 1, i \neq j$ ; при значениях параметров  $T = 1, T_m = 1000, N = 100, \lambda_1 = 4, \lambda_2 = 0,75, \alpha_1 = 1, \alpha_2 = 0,45$ .

На рис. 4 в качестве иллюстрации приведено поведение оценки  $\hat{\lambda}(t)$  для отдельной реализации исследуемого потока; здесь  $\lambda_1, \lambda_2$  – значения оценки  $\hat{\lambda}(t)$ . Штриховкой отмечены участки ошибочных решений о состоянии процесса  $\lambda(t)$  (потока событий),  $\Delta t = 0,001$ .

Рис. 4. Поведение оценки  $\hat{\lambda}(t)$ 

В четвертом эксперименте выборочное среднее полной вероятности ошибки –  $\hat{P}_{ош} = 0,1821$ , выборочная дисперсия –  $\hat{D}_{ош} = 0,0049 \cdot 10^{-3}$ . Существенное ухудшение качества оценивания (в смысле малости оценки полной вероятности ошибки) по сравнению с результатами, приведенными в [17] в случае  $T = 0$  при том же наборе параметров, объясняется наличием мертвого времени.

### Заключение

В настоящей статье предложен основанный на критерии максимума апостериорной вероятности алгоритм оптимального оценивания состояний обобщенного синхронного потока событий второго порядка, функционирующего в условиях непродлевающегося мертвого времени. Сформулированы алгоритм расчета апостериорной вероятности  $w(\lambda_1 | t)$  и алгоритм принятия решения о состоянии процесса  $\lambda(t)$  в произвольный момент времени  $t$  на основании выборки моментов наступления событий  $t_1, t_2, \dots, t_m$  в наблюдаемом потоке. Проведен ряд статистических экспериментов для установления частоты ошибочных решений о состоянии случайного процесса  $\lambda(t)$ . Численные результаты проведенных на имитационной модели потока экспериментов не противоречат физической интерпретации и иллюстрируют приемлемую оценку полной вероятности ошибки оценивания и достаточно малую выборочную дисперсию, которые обеспечиваются предложенным алгоритмом оптимального оценивания.

### ЛИТЕРАТУРА

1. Дудин А.Н., Клименок В.И. Системы массового обслуживания с коррелированными потоками. Минск : Изд-во Белорус. гос. ун-та, 2000. 175 с.
2. Basharin G.P., Gaidamaka Yu.V., Samouylov K.E. Mathematical theory of teletraffic and its application to the analysis of multi-service communication of next generation networks // Automatic Control and Computer Sciences. 2013. V. 47, No. 2. P. 62–69.
3. Cox D.R. The analysis of non-Markovian stochastic processes by the inclusion of supplementary variables // Proc. of the Cambridge Philosophical Society. 1955. V. 51, No. 3. P. 433–441.
4. Kingman J.F.C. On doubly stochastic Poisson process // Proc. of the Cambridge Philosophical Society. 1964. V. 60, is. 4. P. 923–930.
5. Башарин Г.П., Кокотушкин В.А., Наумов В.А. О методе эквивалентных замен расчета фрагментов сетей связи. Ч. 1 // Известия АН СССР. Техническая кибернетика. 1979. № 6. С. 92–99.
6. Башарин Г.П., Кокотушкин В.А., Наумов В.А. О методе эквивалентных замен расчета фрагментов сетей связи. Ч. 2 // Известия АН СССР. Техническая кибернетика. 1980. № 1. С. 55–61.
7. Neuts M.F. A versatile Markovian point process // Journal of Applied Probability. 1979. V. 16. P. 764–779.
8. Lucantoni D.M. New results on the single server queue with a batch markovian arrival process // Communications in Statistics Stochastic Models. 1991. V. 7. P. 1–46.
9. Горцев А.М., Зуевич В.Л. Оптимальная оценка состояний асинхронного дважды стохастического потока событий с произвольным числом состояний // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2010. № 2. С. 44–65.

10. Горцев А.М., Леонова М.А. Оптимальная оценка состояний обобщенного асинхронного дважды стохастического потока // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2010. № 1. С. 33–47.
11. Bakhholdina M.A., Gortsev A.M. Optimal estimation of the states of modulated semisynchronous integrated flow of events in condition of its incomplete observability // Applied Mathematical Sciences. 2015. V. 9, No. 29–32. P. 1433–1451.
12. Gortsev A.M., Nezhel'skaya L.A. Estimation of parameters of synchronously alternating Poisson stream of events by the moment method // Telecommunications and Radio Engineering. 1996. V. 50, No. 1. P. 56–63.
13. Горцев А.М., Калягин А.А., Нежелская Л.А. Совместная плотность вероятностей длительности интервалов обобщенного полусинхронного потока событий при непродлевающемся мертвом времени // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2014. № 2 (27). С. 19–29.
14. Горцев А.М., Нежелская Л.А. Оценивание длительности мертвого времени и параметров синхронного альтернирующего потока событий // Вестник Томского государственного университета. 2003. № 6. С. 232–239.
15. Горцев А.М., Леонова М.А., Нежелская Л.А. Сравнение МП- и ММ-оценок длительности мертвого времени в обобщенном асинхронном потоке событий // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2013. № 4 (25). С. 32–42.
16. Нежелская Л.А., Сидорова Е.Ф. Алгоритм оптимального оценивания состояний обобщенного синхронного потока событий второго порядка // Труды Томского государственного университета. Серия физико-математическая. Томск : Издательский Дом ТГУ. 2017. Т. 301. С. 105–113.
17. Nezhelskaya L.A., Sidorova E.F. Optimal estimation of the states of synchronous generalized flow of events of the second order under its complete observability // Communications in Computer and Information Science. 2018. V. 912. P. 157–171.
18. Апанасович В.В., Коляда А.А., Чернявский А.Ф. Статистический анализ случайных потоков в физическом эксперименте. Минск : Университетское, 1988. 256 с.
19. Normey-Rico J.E. Control of dead-time process. London : Springer Verlag, 2007. 462 p.
20. Хазен Э.М. Методы оптимальных статистических решений и задачи оптимального управления. М. : Сов. радио, 1968. 256 с.
21. Левин Б.Р. Теоретические основы статистической радиотехники. М. : Сов. радио, 1968. 504 с.
22. Камке Э. Справочник по обыкновенным дифференциальным уравнениям. М. : Наука, 1976. 576 с.
23. Соболев И.М. Численные методы Монте-Карло. М. : Наука, 1973. 312 с.

Поступила в редакцию 4 мая 2018 г.

Nezhel'skaya L.A., Sidorova E.F. (2018) OPTIMAL ESTIMATE OF THE STATES OF A GENERALIZED SYNCHRONOUS FLOW OF SECOND-ORDER EVENTS UNDER CONDITIONS OF INCOMPLETE OBSERVABILITY. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitel'naya tekhnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 45. pp. 30–41

DOI: 10.17223/19988605/45/4

The paper deals with a generalized synchronous flow second-order events accompanying random process  $\lambda(t)$ , which is an unobservable piecewise constant process with two states  $S_1$  and  $S_2$ . Hereinafter, it is understood the  $i$  th state of process  $\lambda(t)$  as the state  $S_i$ ,  $i = 1, 2$ .

The duration of interval between the flow events at the  $i$  th state is determined by random variable  $\eta_i = \min(\xi_i^{(1)}, \xi_i^{(2)})$ , where random variable  $\xi_i^{(1)}$  is distributed according to the law  $F_i^{(1)}(t) = 1 - e^{-\lambda_i t}$ , random variable  $\xi_i^{(2)}$  is distributed according to the law  $F_i^{(2)}(t) = 1 - e^{-\alpha_i t}$ ;  $\xi_i^{(1)}$  and  $\xi_i^{(2)}$  are independent of each other,  $i = 1, 2$ . At the moment when a flow event occurs, process  $\lambda(t)$  transits from the  $i$  th state to the  $j$  th either with probability  $P_1^{(1)}(\lambda_j | \lambda_i)$  or with probability  $P_1^{(2)}(\lambda_j | \lambda_i)$  depending on the value of random variable  $\eta_i$ ,  $i, j = 1, 2$ ,  $i \neq j$ . At the moment when a flow event occurs, process  $\lambda(t)$  stays in the  $i$  th state either with probability  $P_1^{(1)}(\lambda_i | \lambda_i)$  or with probability  $P_1^{(2)}(\lambda_i | \lambda_i)$  depending on the value of random variable  $\eta_i$ ,  $i = 1, 2$ . Wherein,  $P_1^{(1)}(\lambda_j | \lambda_i) + P_1^{(1)}(\lambda_i | \lambda_i) = 1$ ,  $P_1^{(2)}(\lambda_j | \lambda_i) + P_1^{(2)}(\lambda_i | \lambda_i) = 1$ ,  $i, j = 1, 2$ ,  $i \neq j$ . Thus, the duration of interval between the flow events in the  $i$  th state of process  $\lambda(t)$  is a random variable with the exponential distribution function  $F_i(t) = 1 - e^{-(\lambda_i + \alpha_i)t}$ ,  $i = 1, 2$ .

In the sequel it is assumed that the state  $S_1$  (the first state) of random process  $\lambda(t)$  takes place, if  $\lambda(t) = \lambda_1$ , and the state  $S_2$  (the second state) of random process  $\lambda(t)$  takes place, if  $\lambda(t) = \lambda_2$  ( $\lambda_1 > \lambda_2 \geq 0$ ).

We consider the situation, where each event registered at the moment  $t_k$  generates the period of time  $T$ , called a dead time, during which other flow events are lost, and upon its completion, an occurring event also causes the period of non-observability of the

flow. It is necessary to estimate the state of random process  $\lambda(t)$  at the moment  $t$ , having a sample of the moments of occurrence of events  $t_1, t_2, \dots$  in the observed flow.

The algorithm of optimal estimation for the states of a generalized synchronous flow of second-order events with unextendable dead time is as follows:

- 1) at the initial instant  $t_0 = 0$  a priori probability of the first state  $\pi_1$  of the process  $\lambda(t)$  is calculated using explicit expression

$$\pi_1 = \frac{\lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2)}{\lambda_1 P_1^{(1)}(\lambda_2 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_2 | \lambda_1) + \lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2)};$$

- 2)  $k = 0$ , at any moment  $t$  of the interval  $(t_0, t_1)$  a probability  $w(\lambda_1 | t)$  is computed according to formula

$$w(\lambda_1 | t) = \frac{w(\lambda_1 | t_k + 0) e^{-(\lambda_1 + \alpha_1 - \lambda_2 - \alpha_2)(t - t_k)}}{1 - w(\lambda_1 | t_k + 0) + w(\lambda_1 | t_k + 0) e^{-(\lambda_1 + \alpha_1 - \lambda_2 - \alpha_2)(t - t_k)}}, \quad w(\lambda_1 | t_0 + 0) = \pi_1;$$

at the moment  $t_1$  calculations are made for determination  $w(\lambda_1 | t_1) = w(\lambda_1 | t_1 - 0)$  using the same formula;

- 3)  $k$  increases by 1; for  $k = 1$  a posteriori probability is recalculated according to formula

$$w(\lambda_1 | t_k + 0) = W / [(\lambda_2 + \alpha_2) + w(\lambda_1 | t_k - 0)(\lambda_1 + \alpha_1 - \lambda_2 - \alpha_2)],$$

$$W = \lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2) + w(\lambda_1 | t_k - 0)[\lambda_1 P_1^{(1)}(\lambda_1 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_1 | \lambda_1) - \lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) - \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2)];$$

$w(\lambda_1 | t_1 + 0)$  is the initial value for  $w(\lambda_1 | t)$  at the next step;

- 4)  $k = 1$ , at any moment  $t$  of the half-interval  $(t_1, t_1 + T]$  a probability  $w(\lambda_1 | t)$  is calculated according to

$$w(\lambda_1 | t) = \pi_1 + [w(\lambda_1 | t_k + 0) - \pi_1] e^{-a(t - t_k)},$$

$$a = \lambda_1 P_1^{(1)}(\lambda_2 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_2 | \lambda_1) + \lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2);$$

here  $w(\lambda_1 | t = t_1 + T)$  is the initial value for  $w(\lambda_1 | t)$  at the next step of the algorithm;

- 5)  $k = 1$ , at any moment  $t$  of the  $(t_1 + T, t_2)$  a posteriori probability  $w(\lambda_1 | t)$  is calculated according to

$$w(\lambda_1 | t) = \frac{w(\lambda_1 | t_k + T) e^{-(\lambda_1 + \alpha_1 - \lambda_2 - \alpha_2)(t - t_k - T)}}{1 - w(\lambda_1 | t_k + T) + w(\lambda_1 | t_k + T) e^{-(\lambda_1 + \alpha_1 - \lambda_2 - \alpha_2)(t - t_k - T)}};$$

$w(\lambda_1 | t_2) = w(\lambda_1 | t_2 - 0)$  can be found using the same formula when  $t = t_2$ ;

- 6) the algorithm goes to step 3, after that steps 3–5 are repeated sequentially for  $k = 2$  and so on.

Simultaneously with the calculation process of probability  $w(\lambda_1 | t)$ , the estimation of the flow state is made according to criterion of a posteriori probability maximum: if  $w(\lambda_1 | t) \geq w(\lambda_2 | t)$ , then  $\hat{\lambda}(t) = \lambda_1$ , else  $\hat{\lambda}(t) = \lambda_2$ .

The numerical results of described statistical experiments conducted on the simulation model demonstrate the possibility of sufficiently qualitative estimation of the states of a generalized synchronous flow of second-order events in conditions of a constant dead time.

Keywords: generalized synchronous flow of second-order events; flow state; optimal estimation of states; unextendable dead time; criterion of a posteriori probability maximum.

NEZHEL'SKAYA Lyudmila Alekseevna (Doctor of Physical and Mathematical Sciences, Associate Professor, National Research Tomsk State University, Tomsk, Russian Federation).  
E-mail: ludne@mail.ru

SIDOROVA Ekaterina Filippovna (National Research Tomsk State University, Tomsk, Russian Federation).  
E-mail: katusha\_sidorova@mail.ru

## REFERENCES

1. Dudin, A.N. & Klimenok, V.I. (2000) *Sistemy massovogo obsluzhivaniya s korrelirovannymi potokami* [Queueing systems with correlated flows]. Minsk: BSU.
2. Basharin, G.P., Gaidamaka, Yu.V. & Samouylov, K.E. (2013) Mathematical theory of teletraffic and its application to the analysis of multiservice communication of next generation networks. *Automatic Control and Computer Sciences*. 47(2). pp. 62–69.
3. Cox, D.R. (1955) The analysis of non-Markovian stochastic processes by the inclusion of supplementary variables. *Proceedings of the Cambridge Philosophical Society*. 51(3). pp. 433–441. DOI: 10.1017/S0305004100030437
4. Kingman, J.F.C. (1964) On doubly stochastic Poisson process. *Proceedings of the Cambridge Philosophical Society*. 60(4). pp. 923–930. DOI: 10.1017/S030500410003838X
5. Basharin, G.P., Kokotushkin, V.A. & Naumov, V.A. (1979) O metode ekvivalentnykh zamen rascheta fragmentov setey svyazi. Ch. 1 [On the equivalent substitutions method for computing fragments of communication networks. Part 1]. *Izvestiya AN SSSR. Tekhnicheskaya kibernetika – Proc. of USSR Academy of Sciences. Technical Cybernetics*. 6. pp. 92–99.
6. Basharin, G.P., Kokotushkin, V.A. & Naumov, V.A. (1980) O metode ekvivalentnykh zamen rascheta fragmentov setey svyazi. Ch. 2 [On the equivalent substitutions method for computing fragments of communication networks. Part 2]. *Izvestiya AN SSSR. Tekhnicheskaya kibernetika – Proc. of USSR Academy of Sciences. Technical Cybernetics*. 1. pp. 55–61.
7. Neuts, M.F. (1979) A versatile Markovian point process. *Journal of Applied Probability*. 16. pp. 764–779. DOI: 10.2307/3213143



8. Lucantoni, D.M. (1991) New results on the single server queue with a batch markovian arrival process. *Communications in Statistics Stochastic Models*. 7. pp. 1–46. DOI: 10.1080/15326349108807174
9. Gortsev, A.M. & Zuevich, V.L. (2010) Optimal state estimation of an asynchronous doubly stochastic flow of events with arbitrary number of the states. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 2. pp. 44–65. (In Russian).
10. Gortsev, A.M. & Leonova, M.A. (2010) Optimal state estimation of a generalized asynchronous doubly stochastic flow of events. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 1. pp. 33–47. (In Russian).
11. Bakholdina, M.A. & Gortsev, A.M. (2015) Optimal estimation of the states of modulated semisynchronous integrated flow of events in condition of its incomplete observability. *Applied Mathematical Sciences*. 9(29–32). pp. 1433–1451. DOI: 10.12988/ams.2015.5135
12. Gortsev, A.M. & Nezhelskaya, L.A. (1996) Estimation of parameters of synchronously alternating Poisson stream of events by the moment method. *Telecommunications and Radio Engineering*. 50(1). pp. 56–63.
13. Gortsev, A.M., Kalyagin, A.A & Nezhelskaya, L.A. (2014) Joint probability density of the duration of intervals of semisynchronous generalized flow of events with unextendable dead time. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 2(27). pp. 19–29. (In Russian).
14. Gortsev, A.M. & Nezhelskaya, L.A. (2003) Dead time period and parameter estimation of synchronous alternating flow of events. *Vestnik Tomskogo gosudarstvennogo universiteta – Tomsk State University Journal*. 284. pp. 232–239. (In Russian).
15. Gortsev, A.M., Leonova, M.A. & Nezhelskaya, L.A. (2013) Comparison of the ML- and MM-estimates of the dead time period in asynchronous generalized flow of events. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 4(25). pp. 32–42. (In Russian).
16. Nezhelskaya, L.A. & Sidorova, E.F. (2017) Algoritm optimal'nogo otsenivaniya sostoyaniy obobshchennogo sinkhronnogo potoka sobytiy vtorogo poryadka [Optimal estimation algorithm for the states of synchronous generalized flow of events of the second order]. *Trudy Tomskogo gosudarstvennogo universiteta. Seriya fiziko-matematicheskaya*. 301. pp. 105–113.
17. Nezhelskaya, L.A. & Sidorova, E.F. (2018) Optimal estimation of the states of synchronous generalized flow of events of the second order under its complete observability. *Communications in Computer and Information Science*. 912. pp. 157–171. DOI: 10.1007/978-3-319-97595-5\_13
18. Apanasovich, V.V., Kolyada, A.A. & Chernyavskii, A.F. (1988) *Statistical Analysis of Stochastic Flows in Physical Experiment*. Minsk: University Press.
19. Normey-Rico, J.E. (2007) *Control of Dead-Time Process*. London: Springer-Verlag.
20. Khazen, E.M. (1968) *Metody optimal'nykh statisticheskikh resheniy i zadachi optimal'nogo upravleniya* [Methods of optimal statistical decisions and problems of optimal control]. Moscow: Sovetskoe radio.
21. Levin, B.R. (1968) *Teoreticheskie osnovy statisticheskoy radiotekhniki* [Theoretical Foundations of Statistical Radio Engineering]. Moscow: Sovetskoe radio.
22. Kamke, E. (1976) *Spravochnik po obyknovennym differentsial'nym uravneniyam* [Handbook of Ordinary Differential Equations]. Translated from German by S. Fomin. Moscow: Nauka.
23. Sobol, I.M. (1973) *Chislennyye metody Monte-Karlo* [Numerical Monte-Carlo Methods]. Moscow: Nauka.

УДК 519.217.2+004.742.2  
DOI: 10.17223/19988605/45/5

Г.Ш. Цициашвили, М.А. Осипова, К.Е. Самуйлов, Ю.В. Гайдамака

# РАСПРЕДЕЛЕНИЕ РЕСУРСОВ В МНОГОКАНАЛЬНОЙ СИСТЕМЕ МАССОВОГО ОБСЛУЖИВАНИЯ С БЛОКИРОВКОЙ НА ОСНОВЕ СИНЕРГЕТИЧЕСКИХ ЭФФЕКТОВ

Строится оценка скорости сходимости к нулю вероятности отказа в многоканальной системе обслуживания, моделирующей телекоммуникационную сеть, при стремлении к бесконечности количества серверов и нагрузки. С ее помощью решается задача разделения ресурсов между различными пользователями телекоммуникационной сети.

**Ключевые слова:** многоканальная система массового обслуживания с отказами; телекоммуникационная сеть; модели телетрафика.

Теория массового обслуживания как современная область прикладной теории вероятностей разработана в рамках исследования операций и стала одним из главных инструментов для анализа систем радиосвязи. В настоящее время теория очередей играет важную роль в анализе эффективности будущих поколений телекоммуникационных сетей. Например, в пятом поколении сети, несмотря на ее высокую пропускную способность, до 10 Гбит/с, возникает необходимость совместного использования ограниченного количества ресурсов между различными приложениями и пользователями. Это приводит к формулированию и решению проблем распределения ресурсов с учетом ненадежной среды.

В настоящее время теория массового обслуживания развивается именно в этом направлении, имея в качестве своей основы работы таких специалистов, как А. Боровков, Б. Гнеденко, Л. Афанасьева, Е. Булинская, В. Королев, Е. Яровая [1–6]. При этом важную роль играют результаты, полученные в работах по математической теории телетрафика Ф. Келли, К. Росса, Г. Башарина, В. Вишневого [7–10].

В работе рассматривается  $n$ -канальная система массового обслуживания с отказами в предположении, что интенсивность входного потока пропорциональна  $n$ . Исследуется сходимость вероятности блокировки в этой системе к нулю при  $n \rightarrow \infty$ . Решается задача о построении совокупности многоканальных систем с отказами и одинаковой (или близкой) асимптотикой сходимости к нулю вероятности отказа. Подобная задача возникает при конструировании современных систем передачи данных [9, Гл. 2]. Особенностью асимптотических результатов является их приближенный характер, позволяющий получить достаточно удобные формулы для постановки и решения задач оптимизации систем передачи данных.

## 1. Асимптотические соотношения

Рассмотрим систему обслуживания  $A_n = M | M | n | 0$  с интенсивностью входного пуассоновского потока  $n\lambda$  и интенсивностями обслуживания  $\mu$  на всех  $n$  приборах,  $\rho = \lambda / \mu$ . Система  $A_n$  может рассматриваться как объединение  $n$  систем вида  $A_1 = M | M | 1 | 0$  с пуассоновскими входными потоками интенсивности  $\lambda$ . Количество заявок в системе  $A_n$  описывается процессом гибели и рождения  $x_n(t)$  с интенсивностями рождения и гибели  $\lambda_n(k) = n\lambda$ ,  $0 \leq k < n$ ,  $\mu_n(k) = k\mu$ ,  $0 < k \leq n$ .

Обозначим  $P_n(\rho)$  стационарную вероятность отказа в системе  $A_n$  при заданном  $\rho$ . Пусть  $a_n, b_n, n \geq 1$ , – две вещественные последовательности. При  $n \rightarrow \infty$  полагаем, что  $a_n \preceq b_n$ , если  $\limsup_{n \rightarrow \infty} \frac{a_n}{b_n} \leq 1$ , и что  $a_n \sim b_n$ , если  $b_n \preceq a_n \preceq b_n$ .

**Теорема 1.** Справедливо следующее предельное соотношение:  $P_n(1) \sim \sqrt{\frac{2}{\pi n}}, n \rightarrow \infty$ .

**Доказательство.** Пусть  $\varepsilon > 0$ , рассмотрим функцию  $f(x) = 1 - x - \exp(-(1 + \varepsilon)x)$ . Функция  $f(x)$  удовлетворяет следующим соотношениям:

$$f(0) = 0, f'(x) > 0, 0 < x < \frac{\ln(1 + \varepsilon)}{1 + \varepsilon}, f'(x) < 0, \frac{\ln(1 + \varepsilon)}{1 + \varepsilon} < x \leq 1, f(1) < 0.$$

Поэтому на отрезке  $[0, 1]$  существует единственное  $x(\varepsilon)$ , удовлетворяющее равенству  $f(x(\varepsilon)) = 0$ , и значит неравенствам  $1 - x \geq \exp(-(1 + \varepsilon)x), 0 \leq x \leq x(\varepsilon) < 1$ . Пусть  $p_n(k) = \lim_{t \rightarrow \infty} P(x_n(t) = k), 0 \leq k \leq n$ , тогда в силу [11. Гл. 2, § 1]

$$p_n(n-1) = p_n(n) \frac{\mu}{\lambda} \frac{n}{n}, p_n(n-2) = p_n(n) \left( \frac{\mu}{\lambda} \right)^2 \frac{n(n-1)}{n^2}, \dots$$

Следовательно, стационарная вероятность отказа в силу интегральной теоремы восстановления и закона больших чисел для процесса восстановления [12. Гл. 9, § 4, 5] удовлетворяет равенству

$$P_n(\rho) = p_n(n) = \left( \sum_{k=0}^n \rho^{-k} \prod_{j=0}^{k-1} \left( 1 - \frac{j}{n} \right) \right)^{-1}, \text{ и значит}$$

$$P_n(1) = \left( \sum_{k=0}^n \prod_{j=0}^{k-1} \left( 1 - \frac{j}{n} \right) \right)^{-1}, \quad (1)$$

где  $\prod_{j=0}^{-1}$  полагается равным 1. Из формулы (1) следует неравенство

$$P_n^{-1}(1) \geq \sum_{0 \leq k \leq nx(\varepsilon)} \prod_{j=0}^{k-1} \left( 1 - \frac{j}{n} \right) \geq \sum_{0 \leq k \leq nx(\varepsilon)} \prod_{j=0}^{k-1} \exp(-(1 + \varepsilon)j/n) \geq \sum_{1 \leq k \leq nx(\varepsilon)} \exp(-(1 + \varepsilon)k^2 / 2n).$$

Отсюда получаем, что

$$P_n^{-1}(1) \geq \int_1^{nx(\varepsilon)} \exp(-(1 + \varepsilon)x^2 / 2n) dx = \sqrt{\frac{n}{1 + \varepsilon}} \int_{\sqrt{\frac{1 + \varepsilon}{n}}}^{x(\varepsilon)\sqrt{\frac{n(1 + \varepsilon)}{1 + \varepsilon}}} \exp(-y^2 / 2) dy,$$

следовательно,

$$P_n(1) \sqrt{n} \leq (1 + \varepsilon) \left( \int_{\sqrt{\frac{1 + \varepsilon}{n}}}^{x(\varepsilon)\sqrt{\frac{n(1 + \varepsilon)}{1 + \varepsilon}}} \exp(-y^2 / 2) dy \right)^{-1} \rightarrow (1 + \varepsilon) \sqrt{\frac{2}{\pi}}, n \rightarrow \infty,$$

и значит  $\limsup_{n \rightarrow \infty} P_n(1) \sqrt{\frac{\pi n}{2}} \leq 1 + \varepsilon$ .

Используя формулу (1) и неравенство  $1 - x \leq \exp(-x), 0 \leq x \leq 1$ , получаем

$$P_n^{-1}(1) \leq \sum_{1 \leq k \leq n} \exp(-k(k-1) / 2n) \leq \sum_{1 \leq k \leq n} \exp(-(k-1)^2 / 2n) \leq \int_0^\infty \exp(-x^2 / 2n) dx,$$

откуда следует, что  $1 \leq \liminf_{n \rightarrow \infty} P_n(1) \sqrt{\frac{\pi n}{2}}$ . Полученные выше неравенства для верхнего и нижнего пределов приводят к утверждению теоремы 1.

**Теорема 2** При  $\rho < 1$  справедливы соотношения

$$\exp\left(-\frac{n \ln^2 \rho}{2}\right) \sqrt{\frac{2}{\pi n}} \sqrt{\frac{\rho}{8}} \preceq P_n(\rho) \preceq \exp\left(-\frac{n \ln^2 \rho}{2} \cdot \frac{\rho - 1}{\ln \rho}\right) \sqrt{\frac{2}{\pi n}} \sqrt{\frac{\ln \rho}{\rho - 1}}. \quad (2)$$

**Доказательство.** Пусть величина  $\varepsilon$  удовлетворяет равенству  $x(\varepsilon) = 1 - \rho > 0$ , значит  $\varepsilon = -\ln(1 - x(\varepsilon))/x(\varepsilon) - 1 = -\ln \rho / (1 - \rho) - 1 > 0$ . Из формулы (1) следует неравенство  $P_n^{-1}(\rho) \geq \psi_n(\rho)$ , где

$$\begin{aligned} \psi_n(\rho) &= \sum_{1 \leq k \leq nx(\varepsilon)} \rho^{-k} \exp(-(1+\varepsilon)k^2 / 2n) = \sum_{1 \leq k \leq nx(\varepsilon)} \exp\left(-\frac{(1+\varepsilon)k^2 + 2kn \ln \rho}{2n}\right) \geq \\ &\geq \exp\left(\frac{n \ln^2 \rho}{2(1+\varepsilon)}\right) \sum_{1 \leq k \leq nx(\varepsilon)-1} \exp\left(-\frac{1}{2}\left(\sqrt{\frac{1+\varepsilon}{n}}k + \sqrt{\frac{n}{1+\varepsilon}} \ln \rho\right)^2\right) \geq \\ &\geq \exp\left(\frac{n \ln^2 \rho}{2(1+\varepsilon)}\right) \int_1^{nx(\varepsilon)-1} \exp\left(-\frac{1}{2}\left(\sqrt{\frac{1+\varepsilon}{n}}x + \sqrt{\frac{n}{1+\varepsilon}} \ln \rho\right)^2\right) dx = \sqrt{\frac{n}{1+\varepsilon}} \theta_n(\rho), \\ \theta_n(\rho) &= \int_{\sqrt{\frac{1+\varepsilon}{n}} + \sqrt{\frac{n}{1+\varepsilon}} \ln \rho}^{\sqrt{\frac{1+\varepsilon}{n}}(nx(\varepsilon)-1) + \sqrt{\frac{n}{1+\varepsilon}} \ln \rho} \exp\left(-\frac{y^2}{2}\right) dy \rightarrow \int_{-\infty}^0 \exp\left(-\frac{y^2}{2}\right) dy = \sqrt{\frac{\pi}{2}}, n \rightarrow \infty. \end{aligned}$$

Таким образом, доказано неравенство

$$\limsup_{n \rightarrow \infty} P_n(\rho) \exp\left(\frac{n \ln^2 \rho}{2(1+\varepsilon)}\right) \sqrt{\frac{\pi n}{2(1+\varepsilon)}} \leq 1. \quad (3)$$

Перейдем теперь к построению нижней оценки для  $P_n(\rho)$ . Используя формулу (1) и неравенство  $1 - x \leq \exp(-x)$ ,  $0 \leq x \leq 1$ , получаем

$$\begin{aligned} P_n^{-1}(\rho) &\leq \sum_{k=0}^n \exp\left(-\frac{k^2 - k + 2nk \ln \rho}{2n}\right) = \exp\left(\frac{1}{2n}\left(n \ln \rho - \frac{1}{2}\right)^2\right) \chi_n(\rho), \\ \chi_n(\rho) &= \sum_{k=0}^n \exp\left(-\frac{(k + n \ln \rho - 1/2)^2}{2n}\right) \leq \sum_{k=-\infty}^{\infty} \exp\left(-\frac{1}{2}\left(\frac{k}{\sqrt{n}} + \sqrt{n} \ln \rho - \frac{1}{2\sqrt{n}}\right)^2\right) \leq \\ &\leq 2 + \sqrt{n} \int_{-\infty}^{\infty} \exp(-y^2 / 2) dy = 2 + \sqrt{2\pi n}. \end{aligned}$$

Таким образом, доказано неравенство

$$1 \leq \liminf_{n \rightarrow \infty} P_n(\rho) \exp\left(\frac{1}{2n}\left(n \ln \rho - \frac{1}{2}\right)^2\right) (2 + \sqrt{2\pi n}) = \liminf_{n \rightarrow \infty} P_n(\rho) \exp\left(\frac{n \ln^2 \rho}{2}\right) \sqrt{\frac{\pi n}{2}} \sqrt{\frac{8}{\rho}}. \quad (4)$$

Из соотношений (3), (4) следует утверждение теоремы 2.

**Замечание.** При  $\rho = 1 - \gamma$ ,  $\gamma \rightarrow 0$ , верхняя и нижняя оценки для вероятности отказа  $P_n(\rho)$  сближаются, поскольку множитель  $\frac{\rho - 1}{\ln \rho} \rightarrow 1$ . Причем множитель  $n \ln^2 \rho$  наиболее сильно влияет на вероятность отказа  $P_n(\rho)$ .

**Следствие.** Пусть  $\rho = 1 - n^{-\gamma}$ ,  $\gamma > 0$ , тогда соотношения (2) переписутся в виде:

$$\frac{1}{2} \sqrt{\frac{1}{\pi n}} \leq P_n(\rho) \leq \sqrt{\frac{2}{\pi n}}, \gamma \geq \frac{1}{2}, \quad \frac{1}{2} \sqrt{\frac{1}{\pi n}} \leq P_n(\rho) \exp\left(\frac{n^{1-2\gamma}}{2}\right) \leq \sqrt{\frac{2}{\pi n}}, \quad \gamma < \frac{1}{2}. \quad (5)$$

Доказательство следствия 1 основано на неравенствах (3), (4), справедливых при  $\rho = 1 - n^{-\gamma}$ . Такая зависимость коэффициента загрузки  $\rho = \lambda / \mu$  от числа каналов  $n$  позволяет выявить изменение вероятности отказа  $P_n(\rho)$  при больших  $n$  от параметра  $\gamma$ , который определяет скорость сходимости коэффициента загрузки к единице.

**Теорема 3.** При  $\rho > 1$  справедливо предельное соотношение  $P_n(\rho) \rightarrow 1 - \mu / \lambda$ ,  $n \rightarrow \infty$ .

**Доказательство.** Из формулы (1) следует, что

$$P_{(n)}(\rho) \geq \left(\sum_{k=0}^n \rho^{-k}\right)^{-1} = \frac{1 - \rho^{-1}}{1 - \rho^{-(n+1)}} \rightarrow 1 - \frac{\mu}{\lambda}, n \rightarrow \infty. \quad (6)$$

Зафиксируем  $\varepsilon > 0$ ,  $0 < \gamma < 1/2$  и определим  $n(\varepsilon)$  такое, чтобы при  $n > n(\varepsilon)$  выполнялось неравенство  $n^{1/2-\gamma} < n\varepsilon$ , тогда

$$\begin{aligned} P_{(n)}(\rho) &\leq \left( \sum_{0 \leq k \leq n^{1/2-\gamma}} \left( \frac{\mu}{\lambda} \right)^k \exp \left( -\frac{(1+\varepsilon)k^2}{2n} \right) \right)^{-1} \leq \\ &\leq \frac{\exp((1+\varepsilon)n^{-\gamma}/2)(1-\mu/\lambda)}{1-(\mu/\lambda)^{n^{1/2-\gamma}}} \rightarrow 1 - \frac{\mu}{\lambda}, n \rightarrow \infty. \end{aligned} \quad (7)$$

Формулы (6), (7) приводят к утверждению теоремы 3.

## 2. Объединение многоканальных систем с блокировкой

Предположим, что у нас имеется  $m$  независимых пуассоновских потоков заявок с интенсивностями  $\lambda = \lambda_1 = \dots = \lambda_m$  и параллельно работающими приборами с интенсивностью обслуживания на каждом из них, равной  $\mu$ . Полагаем, что заявка  $k$ -го потока одновременно обслуживается  $c_k$  приборами,  $1 \leq k \leq m$ . Такие системы естественным образом возникают, если по каналу связи передаются сообщения различного типа, например текст, устная речь и т.д. Требуется так распределить приборы между потоками различных сообщений, чтобы вероятности отказа  $P_n^{(k)}(1)$  для каждого из потоков  $k = 1, \dots, m$  были приблизительно одинаковыми. Подобная задача возникает при конструировании современных систем связи.

Пусть число приборов в  $k$ -й подсистеме равно  $n n_k$ , исходя из теоремы 1 следует потребовать, чтобы выполнялись равенства

$$\frac{n_1}{c_1} = \dots = \frac{n_m}{c_m}. \quad (8)$$

Перепишем эти равенства в виде:

$$n_2 = n_1 \frac{c_2}{c_1}, \dots, n_m = n_1 \frac{c_m}{c_1}.$$

Предположим, что числа  $\frac{c_2}{c_1}, \dots, \frac{c_m}{c_1}$  являются рациональными и перепишем их в виде:

$$\frac{c_2}{c_1} = \frac{p_2}{q_2}, \dots, \frac{c_m}{c_1} = \frac{p_m}{q_m},$$

где пары натуральных чисел  $(p_2, q_2), \dots, (p_m, q_m)$  состоят из взаимно простых чисел. Тогда для того, чтобы числа  $n_2, \dots, n_m$  были целыми, требуется, чтобы число  $n_1$  было кратно числам  $q_2, \dots, q_m$ . Следовательно, число  $n_1$  должно делиться на наименьшее общее кратное  $L$  чисел  $q_2, \dots, q_m$ . Таким образом, всевозможные значения чисел  $n_1, \dots, n_m$ , удовлетворяющие равенствам (8), выглядят следующим образом:  $n_1 = nL, n_2 = \frac{np_2L}{q_2}, \dots, n_m = \frac{np_mL}{q_m}, n = 1, 2, \dots$

Рассмотрим теперь случай, когда интенсивности входных потоков  $\lambda_1, \dots, \lambda_m$  различаются, и обозначим  $\rho_k = \frac{\lambda_k}{\mu}, k = 1, \dots, m$ . В этом случае естественно заменить равенства (8) на равенства

$$n_1 \frac{\ln^2 \rho_1}{c_1} = \dots = n_m \frac{\ln^2 \rho_m}{c_m}$$

и провести аналогичные рассуждения.

## Заключение

Полученные в настоящей работе результаты позволяют распределять ресурсы канала связи между различными типами сигналов так, чтобы вероятность потери передаваемых сигналов была достаточно малой. Решение этой задачи основано, с одной стороны, на синергетических эффектах в многоканальных системах массового обслуживания с отказами, а с другой стороны, на разложении натуральных чисел на простые множители и нахождении наименьшего общего кратного.

## ЛИТЕРАТУРА

1. Боровков А.А. Вероятностные процессы в теории массового обслуживания. М. : Наука, 1972.
2. Боровков А.А. Асимптотические методы в теории массового обслуживания. М. : Наука, 1980.
3. Gnedenko B.V., Korolev V.Yu. Random Summation: Limit Theorems and Applications. Boca Raton : CRC Press, 1996.
4. Afanasyeva L.G., Bulinskaya E.V. Certain Asymptotic Results for Random Walks in a Strip // Theory of Probability and its Applications. 1985. V. 29, is. 4. P. 677–693.
5. Afanasyeva L.G., Bashтова E.E., Bulinskaya E.V. Limit Theorems for Semi Markov Queues and Their Applications // Communications in Statistics. Part B: Simulation and Computation. 2012. V. 41, is. 6. P. 688–709.
6. Yarovaya E.B. Branching Random Walks with Several Sources // Mathematical Population Studies. 2012. V. 20. P. 14–26.
7. Kelly F. Blocking Probabilities in Large Circuit-Switched Networks // Advances in Applied Probability. 1986. V. 18. P. 473–505.
8. Ross K. Multiservice Loss Models for Broadband Telecommunication Networks. London : Springer, 1995.
9. Башарин Г.П., Гайдамака Ю.В., Самуйлов К.Е. Математическая теория телетрафика и ее приложения к анализу мультисервисных сетей связи следующих поколений // Автоматика и вычислительная техника. 2013. № 2. С. 11–21.
10. Vishnevsky V.M., Semenova O.V. Polling Systems: Theory and Applications for Broadband Wireless Networks. London : Academic Publishing, 2012.
11. Ивченко Г.И., Каштанов В.А., Коваленко И.Н. Теория массового обслуживания. М. : Высшая школа, 1982.
12. Боровков А.А. Теория вероятностей. М. : Наука, 1986.

Поступила в редакцию 8 апреля 2018 г.

Tsitsashvili G.Sh., Osipova M.A., Samouylov K.E., Gaidamaka Yu.V. (2018) THE ALLOCATION OF RESOURCES IN MULTI-CHANNEL LOSS QUEUEING SYSTEM BASED ON SYNERGISTIC EFFECTS. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitel'naya tekhnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 45. pp. 42–47

DOI: 10.17223/19988605/45/5

In this paper, we consider  $n$  – server loss system under the assumption that the intensity of the input flow is proportional to  $n$ . We investigate the convergence of the blocking probability in this system to zero at  $n \rightarrow \infty$ . A similar problem arises in the design of modern data transmission systems. A specific of suggested asymptotic results is that we did not obtain accuracy formulas or solutions of optimization problems for the transmission systems.

Consider queueing system  $A_n = M | M | n | 0$  with intensity of input flow  $n\lambda$  and intensities of service at all  $n$  servers,  $\rho = \lambda/\mu$ . Denote  $P_n(\rho)$  the stationary blocking probability in the system  $A_n$  at a given  $\rho$ . Let  $a_n, b_n, n \geq 1$ , be two real sequences. For  $n \rightarrow \infty$ .

we assume that  $a_n \preceq b_n$  if  $\limsup_{n \rightarrow \infty} \frac{a_n}{b_n} \leq 1$ . Let us say  $a_n \sim b_n$ , if  $b_n \preceq a_n \preceq b_n$ .

**Теорема 1.** The following limit ratio is true:  $P_n(1) \sim \sqrt{\frac{2}{\pi n}}$ ,  $n \rightarrow \infty$ .

**Теорема 2.** At  $\rho < 1$  following relations are valid

$$\exp\left(-\frac{n \ln^2 \rho}{2}\right) \sqrt{\frac{2}{\pi n}} \sqrt{\frac{\rho}{8}} \preceq P_n(\rho) \preceq \exp\left(-\frac{n \ln^2 \rho}{2} \cdot \frac{\rho-1}{\ln \rho}\right) \sqrt{\frac{2}{\pi n}} \sqrt{\frac{\ln \rho}{\rho-1}}.$$

Suppose that we have  $m$  independent Poisson flows of customers with intensities  $\lambda = \lambda_1 = \dots = \lambda_m$  and parallel servers with the intensity of service at each of them equal to  $\mu$ . We assume that the service of the  $k$ -th flow customer is realized on  $c_k$  servers,  $1 \leq k \leq m$ . We should like to distribute the servers between the flows so that the blocking probabilities  $P_n^{(k)}(1)$  for each of the flow  $k = 1, \dots, m$  are about the same.

Let the number of servers in the  $k$ -th subsystem be  $n n_k$ , from Theorem 1 we obtain that the basic equations

$$\frac{n_1}{c_1} = \dots = \frac{n_m}{c_m} \quad (1)$$

are fulfilled. We rewrite these equations in the form  $n_2 = n_1 \frac{c_2}{c_1}, \dots, n_m = n_1 \frac{c_m}{c_1}$ . Assume that the numbers  $\frac{c_2}{c_1}, \dots, \frac{c_m}{c_1}$  are rational and

rewrite them as  $\frac{c_2}{c_1} = \frac{p_2}{q_2}, \dots, \frac{c_m}{c_1} = \frac{p_m}{q_m}$ , where pairs of positive integers  $(p_2, q_2), \dots, (p_m, q_m)$  consist of mutually prime numbers.

Then, for the numbers  $n_2, \dots, n_m$  to be integers, it requires that number  $n_1$  is a multiple of  $q_2, \dots, q_m$ . Therefore the number  $n_1$  should be divided by the smallest common multiple  $L$  of the numbers  $q_2, \dots, q_m$ . Thus, all possible values of the numbers  $n_1, \dots, n_m$ , satisfying the basic equality (1), look like these

$$n_1 = nL, n_2 = \frac{np_2L}{q_2}, \dots, n_m = \frac{np_mL}{q_m}, n = 1, 2, \dots$$

Keywords: multiserver queuing system with blocking; telecommunication network; models of teletrack.

*TSITSIAHVILI Gurami Shalvovich* (Doctor of Physics and Mathematics, Professor, Main Researcher of Institute for Applied Mathematics, Far Eastern Branch of Russian Academy Sciences, Vladivostok, Russian Federation).

E-mail: guram@iam.dvo.ru

*OSIPOVA Marina Anatolievna* (Candidate of Physics and Mathematics, Associate Professor, Researcher of Institute for Applied Mathematics, Far Eastern Branch of Russian Academy Sciences, Vladivostok, Russian Federation).

E-mail: mao1975@list.ru

*SAMOUYLOV Konstantin Evgenievich* (Doctor of Technical Sciences, Head of the Department of Applied Informatics and Probability Theory, Faculty of Physical and Mathematical Sciences of Russian Friendship University of Peoples (RUDN), Moscow, Russian Federation).

E-mail: samuylov\_ke@rudn.university

*GAYDAMAKA Yulia Vasilievna* (Doctor of Physics and Mathematics, Associate Professor of the Department of Applied Informatics and Probability Theory, Faculty of Physical and Mathematical Sciences of Russian Friendship University of Peoples (RUDN), Moscow, Russian Federation).

E-mail: gaydamaka\_yuv@rudn.university

## REFERENCES

1. Borovkov, A.A. (1972) *Veroyatnostnye protsessy v teorii massovogo obsluzhivaniya* [Stochastic processes in queuing theory]. Moscow: Nauka.
2. Borovkov, A.A. (1980) *Asimptoticheskie metody v teorii massovogo obsluzhivaniya* [Asymptotic methods in queuing theory]. M.: Nauka.
3. Gnedenko, B.V. & Korolev, V.Yu. (1996) *Random Summation: Limit Theorems and Applications*. Boca Raton: CRC Press.
4. Afanasyeva, L.G. & Bulinskaya, E.V. (1985) Certain asymptotic results for random walks in a strip. *Theory of Probability and Its Applications*. 29(4). pp. 677–693. DOI: 10.1137/1129094
4. Afanasyeva, L.G. & Bashtova, E.E. & Bulinskaya, E.V. (2012) Limit theorems for semi Markov queues and their applications. *Communications in Statistics. Part B: Simulation and Computation*. 41(6). pp. 688–709. DOI: 10.1080/03610918.2012.625255
5. Yarovaya, E.B. (2012) Branching Random Walks with Several Sources. *Mathematical Population Studies*. 20. pp. 14–26. DOI: 10.1080/08898480.2013.748571
6. Kelly, F. (1986) Blocking probabilities in large circuit-switched networks. *Advances in Applied Probability*. 18. pp. 473–505. DOI: 10.2307/1427309
8. Ross, K. (1995) *Multiservice loss models for broadband telecommunication networks*. London: Springer.
9. Basharin, G.P., Gaidamaka, Yu.V. & Samouylov, K.E. (2013) Mathematical theory of teletraffic and its applications to analysis of multiservice communication networks of next generations. *Avtomatika i vychislitel'naya tekhnika – Automatics and Computer Engineering*. 2. pp. 11–21. (In Russian).
10. Vishnevsky, V.M. & Semenova, O.V. (2012) *Polling systems: Theory and applications for broadband wireless networks*. London: Academic Publishing.
11. Ivchenko, G.I., Kashtanov, V.A. & Kovalenko, I.N. (1982) *Teoriya massovogo obsluzhivaniya* [Queuing theory]. Moscow: Vysshaya shkola.
12. Borovkov, A.A. (1986) *Teoriya veroyatnostey* [Probability theory]. Moscow: Nauka.

М.А. Шкленник, А.Н. Моисеев

# ИССЛЕДОВАНИЕ ПОТОКОВ ЗАЯВОК В ДВУХФАЗНОЙ СИСТЕМЕ МАССОВОГО ОБСЛУЖИВАНИЯ С НЕОГРАНИЧЕННЫМ ЧИСЛОМ ПРИБОРОВ И ПОВТОРНЫМИ ОБРАЩЕНИЯМИ

Рассматривается двухфазная система массового обслуживания с неограниченным числом приборов на каждой фазе и возможностью повторного обращения заявок к каждой из фаз. Исследуется многомерный случайный процесс, описывающий потоки заявок, поступающие на фазы системы. Получено аналитическое выражение для производящей функции исследуемого процесса в нестационарном режиме, позволяющее определить его вероятностные характеристики.

**Ключевые слова:** двухфазная система массового обслуживания с неограниченным числом приборов; повторное обслуживание; производящая функция; произвольное время обслуживания; метод предельной декомпозиции.

В теории массового обслуживания решение значительного числа задач проводится в предположении, что входящий поток заявок является стационарным и имеет распределение Пуассона. В большинстве случаев это предположение подтверждается статистическим анализом реальных потоков [1]. Также пуассоновские процессы могут быть использованы для аппроксимации процессов в случае наблюдения достаточно большого числа независимых потоков [2–6].

Большинство задач теории массового обслуживания для моделей с простейшим входящим потоком и экспоненциальным распределением времени обслуживания были решены еще в XX в. В 1956 г. Б.А. Севастьянов решил задачу Эрланга для систем с произвольной функцией распределения времени обслуживания  $M/G/N$  [7, 8] и показал, что, что при  $N \rightarrow \infty$  распределение числа занятых приборов сходится к пуассоновскому. В 1969 г. Л. Такач [9] показал, что количество клиентов в системе  $M/G/\infty$  имеет распределение Пуассона в стационарном режиме, которое зависит от средней скорости поступления заявок и среднего времени обслуживания вызовов. В работах [10, 11] доказано, что число занятых приборов в системе  $M/M/\infty$  имеет распределение Пуассона. В работе [12] проведено исследование суммарного потока заявок в системе  $M/M/\infty$  с повторным обслуживанием. Исследование процессов числа заявок в многофазных системах и сетях массового обслуживания с рекуррентным обслуживанием представлено в работах [13–15].

Однако исследованию потоков в системах массового обслуживания (СМО) с неограниченным числом обслуживающих приборов с произвольным временем обслуживания, а также многофазных СМО с возможностью повторного обращения заявки в систему посвящено не так уж много работ. В [16] проведено исследование суммарного потока заявок в однофазной системе с входящим ММРР-потоком и повторными обращениями. В работе [17] для исследования системы  $M/GI/\infty$  предложен метод предельной декомпозиции. В работе [18] методом предельной декомпозиции исследованы суммарный и двумерный потоки обращений к системе  $M/GI/\infty$ . В работах [19–21] исследованы потоки обращений в двухфазных СМО с неограниченным числом приборов на каждой фазе и повторными обращениями. В настоящей работе предлагается метод предельной декомпозиции для анализа различных потоков в двухфазной СМО с неограниченным числом обслуживающих приборов и возможностью повторного обращения заявки к любой фазе системы.

## 1. Математическая модель

Рассмотрим двухфазную систему массового обслуживания с неограниченным числом обслуживающих устройств на каждой фазе (рис. 1). На вход системы поступает простейший поток заявок



с параметром  $\lambda$ . Время обслуживания заявок на первой фазе независимо и имеет одинаковое распределение для каждого прибора с произвольной функцией распределения  $B_1(x)$ . По завершении обслуживания на первой фазе заявка с вероятностью  $r_{11}$  может обратиться к первой фазе для повторного обслуживания, либо с вероятностью  $r_{12}$  может перейти на вторую фазу, либо с вероятностью  $(1 - r_{11} - r_{12})$  может покинуть систему. На второй фазе время обслуживания также имеет одинаковое распределение для каждого прибора с произвольной функцией распределения  $B_2(x)$ . По завершении обслуживания на второй фазе заявка с вероятностью  $r_{21}$  может обратиться к первой фазе для повторного обслуживания, либо с вероятностью  $r_{22}$  может повторно обратиться ко второй фазе, либо с вероятностью  $(1 - r_{21} - r_{22})$  может покинуть систему.

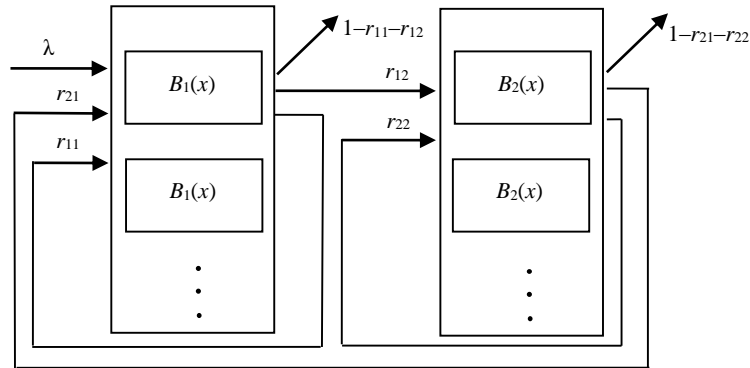


Рис. 1. Двухфазная система массового обслуживания с обратной связью

Введем следующие обозначения:

$l(t)$  – число обращений к системе из внешнего источника (первичные заявки), реализованных к моменту времени  $t$ ;

$n_{12}(t)$  – число обращений ко второй фазе системы после обслуживания на первой фазе;

$n_{11}(t)$  – число повторных обращений к первой фазе системы после обслуживания на первой фазе;

$n_{21}(t)$  – число повторных обращений к первой фазе системы после обслуживания на второй фазе;

$n_{22}(t)$  – число повторных обращений ко второй фазе системы после обслуживания на второй фазе.

Пусть в начальный момент времени система была пуста. Ставится задача исследования многомерного случайного процесса  $\{l(t, N), n_{11}(t, N), n_{12}(t, N), n_{21}(t, N), n_{22}(t, N)\}$ , описывающего потоки заявок в системе.

## 2. Метод предельной декомпозиции

Входящий поток заявок, интенсивность которого равна  $\lambda$ , разделим на  $N$  независимых потоков по полиномиальной схеме с равными вероятностями. Интенсивность каждого такого потока будет равна  $\lambda/N$ . Определим для заявок каждого из полученных потоков единственную однолинейную двухфазную линию обслуживания с отказами (рис. 2). В этой системе заявки, поступающие в период времени, когда хотя бы одна фаза линии занята, не обслуживаются (теряются).

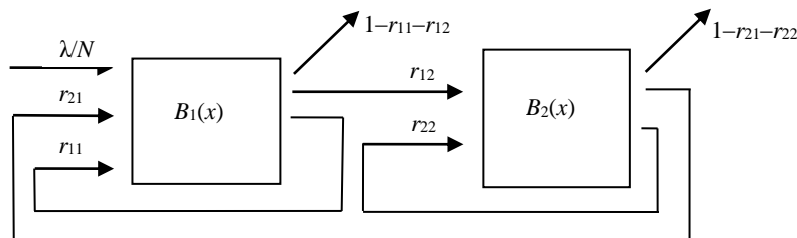


Рис. 2. Однолинейная двухфазная СМО с отказами

Нетрудно доказать, что при  $N \rightarrow \infty$  вероятностью потерь заявок можно пренебречь, и тогда характеристики исходной системы будут сходиться к суммарным характеристикам совокупности  $N$  однолинейных двухфазных СМО. Итак, задача нахождения распределения вероятностей числа обращений в двухфазной бесконечнолинейной СМО сводится к решению задачи нахождения распределения вероятностей числа обращений в однолинейной СМО с отказами.

### 3. Исследование однолинейной двухфазной СМО с отказами

В описанной выше однолинейной системе будем рассматривать многомерный случайный процесс  $\{l(t, N), n_{11}(t, N), n_{12}(t, N), n_{21}(t, N), n_{22}(t, N)\}$ , описывающий число обращений, реализованных за время  $t$  в систему извне, с первой фазы на первую фазу, с первой на вторую фазу, со второй на первую и со второй на вторую фазу соответственно. Введем производящую функцию данного многомерного процесса в виде:

$$g(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N) = \sum_{l=0}^{\infty} \sum_{n_{11}=0}^{\infty} \sum_{n_{12}=0}^{\infty} \sum_{n_{21}=0}^{\infty} \sum_{n_{22}=0}^{\infty} x^l y_{11}^{n_{11}} y_{12}^{n_{12}} y_{21}^{n_{21}} y_{22}^{n_{22}} \cdot P(l, n_{11}, n_{12}, n_{21}, n_{22}, t, N).$$

Докажем следующее утверждение.

**Теорема.** Пусть функции  $f_k(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$ ,  $k = 1, 2$  определяются следующими преобразованиями Фурье–Стилтьеса:

$$\begin{aligned} \phi_1(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22}) &= \int_0^{\infty} e^{j\alpha t} d_t f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, t) = \\ &= \frac{\lambda B_1^*(\alpha)}{(1 - r_{11} y_{11} B_1^*(\alpha))(1 - r_{22} y_{22} B_2^*(\alpha)) - r_{12} y_{12} r_{21} y_{21} B_1^*(\alpha) B_2^*(\alpha)} \times \\ &\times \left\{ (1 - r_{22} y_{22} B_2^*(\alpha)) \cdot \left( x - \frac{1 - r_{22}}{r} (1 - r_{11} y_{11}) + \frac{r_{12}}{r} r_{21} y_{21} \right) + \right. \\ &\left. + B_2^*(\alpha) r_{21} y_{21} \left( \frac{1 - r_{22}}{r} r_{12} y_{12} - \frac{r_{12}}{r} (1 - r_{22} y_{22}) \right) \right\}, \end{aligned} \quad (1)$$

$$\begin{aligned} \phi_2(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22}) &= \int_0^{\infty} e^{j\alpha t} d_t f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, t) = \\ &= \frac{\lambda B_2^*(\alpha)}{(1 - r_{11} y_{11} B_1^*(\alpha))(1 - r_{22} y_{22} B_2^*(\alpha)) - r_{12} y_{12} r_{21} y_{21} B_1^*(\alpha) B_2^*(\alpha)} \times \\ &\times \left\{ (1 - r_{11} y_{11} B_1^*(\alpha)) \cdot \left( \frac{1 - r_{22}}{r} r_{12} y_{12} - \frac{r_{12}}{r} (1 - r_{22} y_{22}) \right) + \right. \\ &\left. + B_1^*(\alpha) r_{12} y_{12} \left( x - \frac{1 - r_{22}}{r} (1 - r_{11} y_{11}) + \frac{r_{12}}{r} r_{21} y_{21} \right) \right\}, \end{aligned} \quad (2)$$

где  $B_k^*(\alpha) = \int_0^{\infty} e^{j\alpha t} dB_k(t)$  – преобразование Фурье–Стилтьеса от функции распределения  $B_k(t)$  времени обслуживания на фазах системы,  $k = 1, 2$ ;  $r = (1 - r_{11})(1 - r_{22}) - r_{12}r_{21}$ . Тогда производящая функция  $g(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N)$  многомерного случайного процесса  $\{l(t, N), n_{11}(t, N), n_{12}(t, N), n_{21}(t, N), n_{22}(t, N)\}$  в двухфазной однолинейной системе с повторными обращениями к фазам имеет вид:

$$\begin{aligned} g(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N) &= 1 + \frac{1}{N} \cdot \left\{ (x - 1) \lambda t + [r_{11}(y_{11} - 1) + r_{12}(y_{12} - 1)] \int_0^t f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds + \right. \\ &\left. + [r_{21}(y_{21} - 1) + r_{22}(y_{22} - 1)] \int_0^t f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds \right\} + o(N^{-2}). \end{aligned} \quad (3)$$

**Доказательство.** Поскольку процесс  $\{l(t, N), n_{11}(t, N), n_{12}(t, N), n_{21}(t, N), n_{22}(t, N)\}$  не является марковским, введем в рассмотрение дополнительные переменные: процесс  $k(t)$  – состояние линии обслуживания, т.е. если  $k$ -я фаза линии занята, то  $k(t) = k$ ,  $k = 1, 2$ , а если линия свободна, то  $k(t) = 0$ ;  $z(t)$  – длина интервала времени от момента  $t$  до момента окончания текущего обслуживания, если линия занята.

Процесс  $\{k(t), l(t, N), n_{11}(t, N), n_{12}(t, N), n_{21}(t, N), n_{22}(t, N), z(t)\}$  является марковским. Распределение вероятностей этого процесса обозначим следующим образом:

–  $P_0(l, n_{11}, n_{12}, n_{21}, n_{22}, t, N) = P\{k(t) = 0, l(t, N) = l, n_{11}(t, N) = n_{11}, n_{12}(t, N) = n_{12}, n_{21}(t, N) = n_{21}, n_{22}(t, N) = n_{22}\}$  – вероятность того, что в момент времени  $t$  линия свободна и к этому моменту к системе обратилось  $l$  заявок из внешнего источника,  $n_{11}$  заявок обратилось для повторного обслуживания с первой фазы на первую,  $n_{12}$  заявок обратилось с первой фазы на вторую,  $n_{21}$  заявок обратилось повторно к первой фазе после обслуживания на второй фазе,  $n_{22}$  заявок повторно обратилось ко второй фазе системы после обслуживания на второй фазе системы;

–  $P_k(l, n_{11}, n_{12}, n_{21}, n_{22}, z, t, N) = P\{k(t) = k, l(t, N) = l, n_{11}(t, N) = n_{11}, n_{12}(t, N) = n_{12}, n_{21}(t, N) = n_{21}, n_{22}(t, N) = n_{22}, z(t) < z\}$ ,  $k = 1, 2$ , – вероятность того, что в момент времени  $t$  занята  $k$ -я фаза системы, к этому моменту к системе обратилось  $l$  заявок из внешнего источника,  $n_{11}$  заявок обратилось для повторного обслуживания с первой фазы на первую,  $n_{12}$  заявок обратилось с первой фазы на вторую,  $n_{21}$  заявок обратилось повторно к первой фазе после обслуживания на второй фазе,  $n_{22}$  заявок повторно обратилось ко второй фазе системы после обслуживания на второй фазе системы, и остаточное время обслуживания заявки, находящейся в системе, меньше  $z$ .

Для распределения вероятностей многомерного процесса  $\{k(t), l(t, N), n_{11}(t, N), n_{12}(t, N), n_{21}(t, N), n_{22}(t, N), z(t)\}$  составим прямую систему дифференциальных уравнений Колмогорова.

$$\begin{aligned} \frac{\partial P_0(l, n_{11}, n_{12}, n_{21}, n_{22}, t, N)}{\partial t} &= (1 - r_{11} - r_{12}) \frac{\partial P_1(l, n_{11}, n_{12}, n_{21}, n_{22}, 0, t, N)}{\partial z} + \\ &+ (1 - r_{21} - r_{22}) \frac{\partial P_2(l, n_{11}, n_{12}, n_{21}, n_{22}, 0, t, N)}{\partial z} - \frac{\lambda}{N} P_0(l, n_{11}, n_{12}, n_{21}, n_{22}, t, N), \end{aligned} \quad (4)$$

$$\begin{aligned} \frac{\partial P_1(l, n_{11}, n_{12}, n_{21}, n_{22}, z, t, N)}{\partial t} &= \frac{\partial P_1(l, n_{11}, n_{12}, n_{21}, n_{22}, z, t, N)}{\partial z} - \\ &- \frac{\partial P_1(l, n_{11}, n_{12}, n_{21}, n_{22}, 0, t, N)}{\partial z} + \frac{\lambda}{N} P_0(l - 1, n_{11}, n_{12}, n_{21}, n_{22}, t, N) B_1(z) + \\ &+ r_{11} B_1(z) \frac{\partial P_1(l, n_{11} - 1, n_{12}, n_{21}, n_{22}, 0, t, N)}{\partial z} + r_{21} B_1(z) \frac{\partial P_2(l, n_{11}, n_{12}, n_{21} - 1, n_{22}, 0, t, N)}{\partial z}, \end{aligned} \quad (5)$$

$$\begin{aligned} \frac{\partial P_2(l, n_{11}, n_{12}, n_{21}, n_{22}, z, t, N)}{\partial t} &= \frac{\partial P_2(l, n_{11}, n_{12}, n_{21}, n_{22}, z, t, N)}{\partial z} - \\ &- \frac{\partial P_2(l, n_{11}, n_{12}, n_{21}, n_{22}, 0, t, N)}{\partial z} + r_{12} B_2(z) \frac{\partial P_1(l, n_{11}, n_{12} - 1, n_{21}, n_{22}, 0, t, N)}{\partial z} + \\ &+ r_{22} B_2(z) \frac{\partial P_2(l, n_{11}, n_{12}, n_{21}, n_{22} - 1, 0, t, N)}{\partial z}. \end{aligned} \quad (6)$$

Начальные условия для системы (4)–(6) запишем в виде:

$$\begin{aligned} P_0(l, n_{11}, n_{12}, n_{21}, n_{22}, 0, N) &= \begin{cases} 0, & \text{если } l > 0, n_{11} > 0, n_{12} > 0, n_{21} > 0, n_{22} > 0, \\ R_0(N), & \text{если } l = n_{11} = n_{12} = n_{21} = n_{22} = 0, \end{cases} \\ P_k(l, n_{11}, n_{12}, n_{21}, n_{22}, z, 0, N) &= \begin{cases} 0, & \text{если } l > 0, n_{11} > 0, n_{12} > 0, n_{21} > 0, n_{22} > 0, \\ R_k(z, N), & \text{если } l = n_{11} = n_{12} = n_{21} = n_{22} = 0, \quad k = 1, 2, \end{cases} \end{aligned}$$

где  $R_0(N)$ ,  $R_1(z, N)$ ,  $R_2(z, N)$  – стационарное распределение вероятностей занятости фаз в однолинейной СМО, удовлетворяющее условию нормировки:

$$R_0(N) + R_1(\infty, N) + R_2(\infty, N) = 1.$$

Введем следующие функции:

$$g_0(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N) = \sum_{l=0}^{\infty} \sum_{n_{11}=0}^{\infty} \sum_{n_{12}=0}^{\infty} \sum_{n_{21}=0}^{\infty} \sum_{n_{22}=0}^{\infty} x^l y_{11}^{n_{11}} y_{12}^{n_{12}} y_{21}^{n_{21}} y_{22}^{n_{22}} \cdot P_0(l, n_{11}, n_{12}, n_{21}, n_{22}, t, N),$$

$$g_k(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t, N) = \sum_{l=0}^{\infty} \sum_{n_{11}=0}^{\infty} \sum_{n_{12}=0}^{\infty} \sum_{n_{21}=0}^{\infty} \sum_{n_{22}=0}^{\infty} x^l y_{11}^{n_{11}} y_{12}^{n_{12}} y_{21}^{n_{21}} y_{22}^{n_{22}} \cdot P_k(l, n_{11}, n_{12}, n_{21}, n_{22}, z, t, N), \quad k = 1, 2.$$

Тогда, из уравнений (4)-(6) следует, что эти функции удовлетворяют уравнениям

$$\begin{aligned} \frac{\partial g_0(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N)}{\partial t} &= (1 - r_{11} - r_{12}) \frac{\partial g_1(x, y_{11}, y_{12}, y_{21}, y_{22}, 0, t, N)}{\partial z} + \\ &+ (1 - r_{21} - r_{22}) \frac{\partial g_2(x, y_{11}, y_{12}, y_{21}, y_{22}, 0, t, N)}{\partial z} - \frac{\lambda}{N} g_0(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N), \end{aligned} \quad (7)$$

$$\begin{aligned} \frac{\partial g_1(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t, N)}{\partial t} &= \frac{\partial g_1(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t, N)}{\partial z} - \\ &- \frac{\partial g_1(x, y_{11}, y_{12}, y_{21}, y_{22}, 0, t, N)}{\partial z} + \frac{\lambda}{N} x B_1(z) g_0(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N) + \\ &+ r_{11} y_{11} B_1(z) \frac{\partial g_1(x, y_{11}, y_{12}, y_{21}, y_{22}, 0, t, N)}{\partial z} + r_{21} y_{21} B_1(z) \frac{\partial g_2(x, y_{11}, y_{12}, y_{21}, y_{22}, 0, t, N)}{\partial z}, \end{aligned} \quad (8)$$

$$\begin{aligned} \frac{\partial g_2(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t, N)}{\partial t} &= \frac{\partial g_2(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t, N)}{\partial z} - \\ &- \frac{\partial g_2(x, y_{11}, y_{12}, y_{21}, y_{22}, 0, t, N)}{\partial z} + r_{12} y_{12} B_2(z) \frac{\partial g_1(x, y_{11}, y_{12}, y_{21}, y_{22}, 0, t, N)}{\partial z} + \\ &+ r_{22} y_{22} B_2(z) \frac{\partial g_2(x, y_{11}, y_{12}, y_{21}, y_{22}, 0, t, N)}{\partial z}. \end{aligned} \quad (9)$$

Начальные условия для системы (7)-(9) имеют вид:

$$g_0(x, y_{11}, y_{12}, y_{21}, y_{22}, 0, N) = R_0(N),$$

$$g_k(x, y_{11}, y_{12}, y_{21}, y_{22}, z, 0, N) = R_k(z, N), \quad k = 1, 2.$$

В работе [22] показано, что

$$R_1(z, N) = \frac{\lambda}{N} \cdot \frac{1 - r_{22}}{r} \int_0^z (1 - B_1(u)) du,$$

$$R_2(z, N) = \frac{\lambda}{N} \cdot \frac{r_{12}}{r} \int_0^z (1 - B_2(v)) dv,$$

где  $r = (1 - r_{11})(1 - r_{22}) - r_{12}r_{21}$ .

Тогда

$$R_0(N) = 1 - R_1(\infty, N) - R_2(\infty, N) = 1 - \frac{\lambda}{N} \cdot \frac{1}{r} [(1 - r_{22})b_1 + r_{12}b_2],$$

где  $b_1$  и  $b_2$  – математические ожидания времени обслуживания на первой и второй фазах соответственно.

Решение системы уравнений (7)–(9) будем искать в виде:

$$g_0(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N) = 1 - \frac{1}{N} F_0(x, y_{11}, y_{12}, y_{21}, y_{22}, t) + o(N^{-2}), \quad (10)$$

$$g_k(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t, N) = \frac{1}{N} F_k(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t) + o(N^{-2}), \quad k = 1, 2, \quad (11)$$

где  $F_0(\cdot)$ ,  $F_1(\cdot)$  и  $F_2(\cdot)$  – некоторые функции, не зависящие от  $N$ .

Подставив выражения (10), (11) в систему (7)–(9), получим систему уравнений для функций  $F_0(\cdot)$  и  $F_k(\cdot)$ ,  $k = 1, 2$ :

$$\frac{\partial F_0(x, y_{11}, y_{12}, y_{21}, y_{22}, t)}{\partial t} = \lambda - (1 - r_{11} - r_{12})f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, t) - (1 - r_{21} - r_{22})f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, t), \quad (12)$$

$$\frac{\partial F_1(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t)}{\partial t} = \frac{\partial F_1(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t)}{\partial z} + \lambda x B_1(z) + (r_{11} y_{11} B_1(z) - 1)f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, t) + r_{21} y_{21} B_1(z) f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, t), \quad (13)$$

$$\frac{\partial f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t)}{\partial t} = \frac{\partial f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t)}{\partial z} + (r_{22} y_{22} B_2(z) - 1)f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, t) + r_{12} y_{12} B_2(z) f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, t), \quad (14)$$

где  $f_k(x, y_{11}, y_{12}, y_{21}, y_{22}, t) = \frac{\partial F_k(x, y_{11}, y_{12}, y_{21}, y_{22}, 0, t)}{\partial z}$ ,  $k = 1, 2$ ,

с начальными условиями:

$$F_0(x, y_{11}, y_{12}, y_{21}, y_{22}, 0) = N \cdot (1 - R_0(N)), \quad (15)$$

$$F_k(x, y_{11}, y_{12}, y_{21}, y_{22}, 0) = N \cdot R_k(z, N), \quad k = 1, 2. \quad (16)$$

Решая дифференциальные уравнения (13) и (14) в частных производных первого порядка, найдем вид функций  $F_1(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t)$  и  $F_2(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t)$ :

$$F_1(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t) = \frac{\lambda}{r} (1 - r_{22}) \int_0^{z+t} (1 - B_1(u)) du + \int_0^t [\lambda x B_1(z + t - s) + (r_{11} y_{11} B_1(z + t - s) - 1)f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) + r_{21} y_{21} B_1(z + t - s) f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s)] ds, \quad (17)$$

$$F_2(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t) = \frac{\lambda}{r} r_{12} \int_0^{z+t} (1 - B_2(v)) dv + \int_0^t [r_{12} y_{12} B_2(z + t - s) f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) + (r_{22} y_{22} B_2(z + t - s) - 1)f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s)] ds. \quad (18)$$

Из уравнения (12) несложно найти вид функции  $F_0(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$ :

$$F_0(x, y_{11}, y_{12}, y_{21}, y_{22}, t) = \lambda \left[ \frac{1 - r_{22}}{r} b_1 + \frac{r_{12}}{r} b_2 + t \right] + (1 - r_{11} - r_{12}) \int_0^t f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds - (1 - r_{21} - r_{22}) \int_0^t f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds. \quad (19)$$

Тогда, учитывая (10), (11), получаем

$$g(x, y_{11}, y_{12}, y_{21}, y_{22}, z, t, N) = 1 - \frac{1}{N} F_0(y_{11}, y_{12}, y_{21}, y_{22}, t) + \frac{1}{N} F_1(y_{11}, y_{12}, y_{21}, y_{22}, z, t) + \frac{1}{N} F_2(y_{11}, y_{12}, y_{21}, y_{22}, z, t) + o(N^{-2}). \quad (20)$$

При  $z \rightarrow \infty$  выражения (17), (18) примут вид:

$$F_1(x, y_{11}, y_{12}, y_{21}, y_{22}, t) = \frac{\lambda}{r} (1 - r_{22}) b_1 + \lambda x t + (r_{11} y_{11} - 1) \int_0^t f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds + r_{21} y_{21} \int_0^t f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds, \\ F_2(x, y_{11}, y_{12}, y_{21}, y_{22}, t) = \frac{\lambda}{r} r_{12} b_2 + r_{12} y_{12} \int_0^t f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds + (r_{22} y_{22} - 1) \int_0^t f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds.$$

Тогда, учитывая (20), производящая функция  $g(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N)$  многомерного случайного процесса  $\{l(t, N), n_{11}(t, N), n_{12}(t, N), n_{21}(t, N), n_{22}(t, N)\}$  в двухфазной однолинейной системе с повторными обращениями к фазам имеет вид:

$$\begin{aligned} g(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N) = \\ = 1 + \frac{1}{N} \cdot \left\{ (x-1)\lambda t + [r_{11}(y_{11}-1) + r_{12}(y_{12}-1)] \int_0^t f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds + \right. \\ \left. + [r_{21}(y_{21}-1) + r_{22}(y_{22}-1)] \int_0^t f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds \right\} + o(N^{-2}), \end{aligned}$$

что совпадает с выражением (3) теоремы.

Для нахождения неизвестных функций  $f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$  и  $f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$  продифференцируем равенства (17) и (18) и рассмотрим их при  $z = 0$ . Получим систему интегральных уравнений для функций  $f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$  и  $f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$ :

$$\begin{aligned} f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, t) = \lambda \left[ \frac{(1-r_{22})}{r} (1-B_1(t)) + x B_1(t) \right] + \\ + r_{11} y_{11} \int_0^t b_1(t-s) f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds + r_{21} y_{21} \int_0^t b_1(t-s) f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds, \end{aligned} \quad (21)$$

$$\begin{aligned} f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, t) = \lambda \frac{r_{12}}{r} (1-B_2(t)) + \\ + r_{12} y_{12} \int_0^t b_2(t-s) f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds + r_{22} y_{22} \int_0^t b_2(t-s) f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds, \end{aligned} \quad (22)$$

где  $b_1(t)$  и  $b_2(t)$  – плотность распределения времени обслуживания на первой и второй фазах соответственно.

Решение системы интегральных уравнений (21)–(22) относительно неизвестных функций  $f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$  и  $f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$  можно найти, используя преобразование Фурье–Стилтьеса вида:

$$\begin{aligned} \Phi_k(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22}) = \int_0^\infty \frac{\partial f_k(x, y_{11}, y_{12}, y_{21}, y_{22}, t)}{\partial t} e^{j\alpha t} dt = \\ = \int_0^\infty e^{j\alpha t} d_t f_k(x, y_{11}, y_{12}, y_{21}, y_{22}, t), \quad k = 1, 2. \end{aligned}$$

Тогда из системы уравнений (21)–(22) можно получить систему для функций  $\Phi_1(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22})$  и  $\Phi_2(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22})$ :

$$\begin{aligned} (r_{11} y_{11} B_1^*(\alpha) - 1) \Phi_1(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22}) + r_{21} y_{21} B_1^*(\alpha) \Phi_2(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22}) = \\ = -\frac{\lambda}{r} (x r - (1-r_{22})(r_{11} y_{11} - 1) + r_{12} r_{21} y_{21}) B_1^*(\alpha), \end{aligned} \quad (23)$$

$$\begin{aligned} r_{12} y_{12} B_2^*(\alpha) \Phi_1(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22}) + (r_{22} y_{22} B_2^*(\alpha) - 1) \Phi_2(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22}) = \\ = -\frac{\lambda}{r} (r_{12} (1-r_{22}) y_{12} + r_{12} (r_{22} y_{22} - 1)) B_2^*(\alpha), \end{aligned} \quad (24)$$

где  $B_k^*(\alpha) = \int_0^\infty \frac{\partial B_k(t)}{\partial t} e^{j\alpha t} dt = \int_0^\infty e^{j\alpha t} b_k(t) dt, \quad k = 1, 2.$

Решая систему (23)–(24), получим выражения (1) и (2) для функций  $\Phi_1(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22})$  и  $\Phi_2(\alpha, x, y_{11}, y_{12}, y_{21}, y_{22})$ , определяющие вид функций  $f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$  и  $f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$ .

Теорема доказана.

#### 4. Исследование двухфазной СМО с неограниченным числом приборов и повторными обращениями на каждой фазе

Определим совместную производящую функцию многомерного процесса  $\{l(t), n_{11}(t), n_{12}(t), n_{21}(t), n_{22}(t)\}$ , описывающего потоки в двухфазной системе с неограниченным числом линий и повторными обращениями на каждой фазе, в виде:

$$G(x, y_{11}, y_{12}, y_{21}, y_{22}, t) = \sum_{l=0}^{\infty} \sum_{n_{11}=0}^{\infty} \sum_{n_{12}=0}^{\infty} \sum_{n_{21}=0}^{\infty} \sum_{n_{22}=0}^{\infty} x^l y_{11}^{n_{11}} y_{12}^{n_{12}} y_{21}^{n_{21}} y_{22}^{n_{22}} \cdot P(l, n_{11}, n_{12}, n_{21}, n_{22}, t).$$

Тогда, в силу независимости однолинейных систем, рассмотренных выше, производящая функция  $G(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$  определяется выражением

$$\begin{aligned} G(x, y_{11}, y_{12}, y_{21}, y_{22}, t) &= \lim_{N \rightarrow \infty} g(x, y_{11}, y_{12}, y_{21}, y_{22}, t, N)^N = \\ &= \lim_{N \rightarrow \infty} \left( 1 + \frac{1}{N} \cdot \left\{ (x-1)\lambda t + [r_{11}(y_{11}-1) + r_{12}(y_{12}-1)] \int_0^t f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds + \right. \right. \\ &\quad \left. \left. + [r_{21}(y_{21}-1) + r_{22}(y_{22}-1)] \int_0^t f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds \right\} + o(N^{-2}) \right)^N = \\ &= \exp \left\{ (x-1)\lambda t + [r_{11}(y_{11}-1) + r_{12}(y_{12}-1)] \int_0^t f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds + \right. \\ &\quad \left. + [r_{21}(y_{21}-1) + r_{22}(y_{22}-1)] \int_0^t f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds \right\}. \end{aligned}$$

Так как совместная производящая функция многомерного распределения вероятностей процесса  $\{l(t), n_{11}(t), n_{12}(t), n_{21}(t), n_{22}(t)\}$  не равна произведению производящих функций одномерных распределений, то очевидно, что потоки не являются независимыми, поэтому анализ таких потоков требует их совместного рассмотрения.

#### Заключение

В настоящей работе рассмотрена математическая модель обслуживания заявок в неоднородной двухфазной системе массового обслуживания  $M/GI/\infty \rightarrow GI/\infty$  с возможностью повторного обращения заявки к любой фазе системы. Методом предельной декомпозиции получено аналитическое выражение для производящей функции многомерного потока обращений заявок к фазам системы. Полученное выражение может быть использовано для определения вероятностных характеристик компонент рассматриваемого многомерного потока.

Полученные результаты могут быть использованы для анализа потоков в различных социально-экономических системах, где имеет место повторное обращение клиентов к системе при различных условиях, например в торговых или страховых компаниях, а также в технических системах распределенной обработки больших данных и облачных сервисах.

#### ЛИТЕРАТУРА

1. Brown L., Gans N., Mandelbaum A., Sakov A. Statistical Analysis of a Telephone Call Center // A Queueing-Science Perspective. Journal of the American Statistical Association. 2005. V. 100. P. 36–50.
2. Neuts M.F. The infinite-server queue with Poisson arrivals and semi Markovian services // Operations Research. 1972. V. 20, No. 2. P. 425–433.

3. Хинчин А.Я. Работы по математической теории массового обслуживания. М. : Физматлит, 1963. 236 с.
4. Ососков Г.А. Одна предельная теорема для потоков однородных событий // Теория вероятностей и ее применения. 1956. Т. 1, № 2. С. 274–282.
5. Григелионис Б.И. Об асимптотическом разложении остаточного члена в случае сходимости к закону Пуассона // Литовский математический сборник. 1962. Т. 2, № 1. С. 135–143.
6. Григелионис Б.И. О сходимости сумм ступенчатых случайных процессов к пуассоновскому // Теория вероятностей и ее применения. 1963. Т. 8, № 2. С. 189–194.
7. Севастьянов Б.А. Эргодическая теорема для марковских процессов и ее приложение к телефонным системам с отказами // Теория вероятностей и ее применения. 1957. Т. 2, № 1. С. 106–116.
8. Севастьянов Б.А. Формулы Эрланга в телефонии при произвольном законе распределения длительности разговора // Труды Третьего Всесоюз. матем. съезда. 1956. М. : АН СССР, 1959. Т. 4. С. 68–70.
9. Takács L. On Erlang's formula // Annals of Mathematical Statistics. 1969. V. 40. P. 71–78.
10. Kleinrock L. Queueing systems. New York : Wiley Interscience, 1975. V. 1: Theory. 417 p.
11. Кёнинг Д., Рыков В.В., Штоян Д. Теория массового обслуживания. М. : Моск. ин-т нефтехим. и газовой промышленности, 1979. 112 с.
12. Моисеева С.П., Морозова А.С., Назаров А.А. Исследование суммарного потока обращений в бесконечно линейной СМО с повторным обслуживанием // Вестник Томского государственного университета. 2006. № 290. С. 173–175.
13. Назаров А.А., Моисеев А.Н. Исследование открытой немарковской сети массового обслуживания  $GI-(GI)^\infty$  с высокоинтенсивным рекуррентным входящим потоком // Проблемы передачи информации. 2013. Т. 49, вып. 2. С. 78–91.
14. Моисеев А.Н., Назаров А.А. Асимптотический анализ многофазной системы массового обслуживания с высокоинтенсивным рекуррентным входящим потоком // Автотметрия. 2014. Т. 50, № 2. С. 67–76.
15. Моисеев А.Н., Назаров А.А. Бесконечнолинейные системы и сети массового обслуживания. Томск : Изд-во НТЛ. 2015. 240 с.
16. Melikov A., Zadiranova L., Moiseev A. Two Asymptotic Conditions in Queue with MMPP Arrivals and Feedback // Communications in Computer and Information Science. 2016. V. 678. P. 231–240.
17. Назаров А.А., Моисеева С.П., Морозова А.С. Исследование СМО с повторным обращением и неограниченным числом обслуживающих приборов методом предельной декомпозиции // Вычислительные технологии. 2008. Т. 13, № 55. С. 88–92.
18. Моисеева С.П., Ананина И.А., Назаров А.А. Исследование потоков в системе  $M/GI/\infty$  с повторными обращениями // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2009. № 3 (8). С. 56–66.
19. Ананина И.А. Исследование суммарного потока обращений в двухфазной бесконечнолинейной СМО с повторными обращениями // Научное творчество молодежи : материалы XIV Всерос. науч.-практ. конф. Томск : Изд-во Том. ун-та, 2010. Ч. 1. С. 3–5.
20. Моисеева С.П., Захорольная И.А. Математическая модель параллельного обслуживания кратных заявок с повторными обращениями // Автотметрия. 2011. Т. 47. № 6. С. 51–58.
21. Ананина И.А. Математическая модель процесса изменения дохода торговой компании, расширяющей свое присутствие на рынке // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2011. № 2 (15). С. 5–14.
22. Shklennik M., Moiseeva S., Moiseev A. Analysis of Queueing Tandem with Feedback by the Method of Limiting Decomposition // CCIS. 2017. V. 800. P. 147–157.

Поступила в редакцию 12 февраля 2018 г.

Shklennik M.A., Moiseev A.N. (2018) ANALYSIS OF CUSTOMERS FLOWS IN THE INFINITE-SERVER QUEUEING TANDEM WITH FEEDBACK. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitel'naja tehnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 45. pp. 48–58

DOI: 10.17223/19988605/45/6

The paper presents the study of a two-stage infinite-server queueing system with feedback. The arrival process is a stationary Poisson process with the rate equals to  $\lambda$ . Service times at the first stage are independent and identically distributed (i.i.d.) with an arbitrary distribution function  $B_1(x)$ . After a completion of the service at the first stage, the customer may return back to the first stage for a new service with the probability  $r_{11}$  or it may move to the second stage with the probability  $r_{12}$  or it may leave the system with the probability  $(1 - r_{11} - r_{12})$ . Service times at the second stage are i.i.d. with an arbitrary distribution function  $B_2(x)$ . When the service at the second stage is completed, the customer may return to the first stage with the probability  $r_{21}$  or it may get a new service at the second stage with the probability  $r_{22}$  or it may leave the system with the probability  $(1 - r_{21} - r_{22})$ .

The problem is to study multidimensional stochastic process describing the flows of requests in the system.

The method of limiting decomposition is used for the study. We divide the arrival process in the considered tandem into  $N$  independent processes according to a polynomial scheme with identical probabilities. As the rate of the original arrivals was equal to  $\lambda$ ,



then the intensity of each generated Poisson process will be equal to  $\lambda/N$ . After that we construct a single-line tandem for each of these arrival processes to serve their customers. The considered single-line two-stage queueing tandem is a system with losses, that is, the customers arrived during a period of any stage busyness are not servicing (they are lost). The total probability characteristics of the independent one-line systems constructed in this way coincide with the corresponding characteristics of the original infinite-server system if  $N \rightarrow \infty$ .

It is shown that the generating function of multidimensional stochastic process  $\{l(t), n_{11}(t), n_{12}(t), n_{21}(t), n_{22}(t)\}$  is as follows:

$$G(x, y_{11}, y_{12}, y_{21}, y_{22}, t) = \exp \left\{ (x-1)\lambda t + [r_{11}(y_{11}-1) + r_{12}(y_{12}-1)] \int_0^t f_1(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds + \right. \\ \left. [r_{21}(y_{21}-1) + r_{22}(y_{22}-1)] \int_0^t f_2(x, y_{11}, y_{12}, y_{21}, y_{22}, s) ds \right\},$$

where for functions  $f_k(x, y_{11}, y_{12}, y_{21}, y_{22}, t)$ ,  $k = 1, 2$  analytic expressions of Fourier-Stieltjes transformations are obtained.

The obtained results can be used for the analysis of flows in different social and economic systems, where there is a repeated circulation of customers to the system under different conditions, for example, in trading or insurance companies, as well as in technical systems for distributed processing of big data and cloud services.

Keywords: infinite-server queueing tandem; feedback; generation function; method of limiting decomposition.

SHKLENNIK Mariya Alexandrovna (National Research Tomsk State University, Tomsk, Russian Federation).

E-mail: Shklennikm@yandex.ru

MOISEEV Alexander Nikolaevich (Doctor of Physical and Mathematical Sciences, Associate Professor, National Research Tomsk State University, Tomsk, Russian Federation).

E-mail: moiseev.tsu@gmail.com

## REFERENCES

1. Brown, L., Gans, N., Mandelbaum, A. & Sakov, A. (2005) Statistical analysis of a telephone call center. *A Queueing-Science Perspective. Journal of the American Statistical Association*. 100. pp. 36–50. DOI: 10.1198/016214504000001808
2. Neuts, M.F. (1972) The infinite-server queue with Poisson arrivals and semi Markovian services. *Operations Research*. 20(2). pp. 425–433. DOI: 10.1287/opre.20.2.425
3. Khinchin, A.Ya. (1963) *Raboty po matematicheskoy teorii massovogo obsluzhivaniya* [Works on the mathematical theory of queueing]. Moscow: Fizmatlit.
4. Ososkov, G.A. (1956) A limit theorem for flows of similar events. *Theory of Probability and Its Applications*. 1(2). pp. 248–255. DOI: 10.1137/1101020
5. Grigelionis, B.I. (1962) Ob asimptoticheskom razlozhenii ostatnochnogo chlena v sluchaye skhodimosti k zakonu Puassona [Accuracy of approximation of a superposition of renewal process by a Poisson process]. *Litovskii Matematicheskii Sbornik*. II(2). pp. 135–143.
6. Grigelionis, B.I. (1961) On the convergence of sums of random step processes to a Poisson process. *Theory of Probability and Its Applications*. 8. pp. 177–182. DOI: 10.1137/1108017
7. Sevastyanov, B.A. (1957) An ergodic theorem for Markov processes and its application to telephone systems with refusals. *Theory of Probability and Its Applications*. 2. pp. 104–112. DOI: 10.1137/1102005
8. Sevastyanov, B.A. (1956) Formuly Erlanga v telefonii pri proizvol'nom zakone raspredeleniya dlitel'nosti razgovora [Erlang Formulas in Telephony under an Arbitrary Law of the Distribution of the Duration of a Conversation]. In: Abramov, A.A. & Nikolskiy, S.M. (eds) *Trudy Tret'ego Vsesoyuznogo matematicheskogo s'ezda* [Proceedings of the Third All-Russian Mathematics Congress]. Vol. 4. Moscow: USSR AS. pp. 68–70.
9. Takács, L. (1969) On Erlang's formula. *Annals of Mathematical Statistics*. 40. pp. 71–78. DOI: 10.1214/aoms/1177697805
10. Kleinrock L. (1975) *Queueing systems*. Vol. 1. New York: Wiley.
11. König, D., Rykov, V., & Shtoyan, D. (1979) *Teoriya massovogo obsluzhivaniya* [Queueing models (Markov models and Markovization methods)]. Moscow: Institute of Petrochemical and Gas Industry.
12. Moiseeva, S.P., Morozova, A.S. & Nazarov, A.A. (2006) Investigation of the flow of appeals in infinite lines queueing system with repeated service. *Vestnik Tomskogo gosudarstvennogo universiteta – Tomsk State University Journal*. 290. pp.173–175. (In Russian)
13. Nazarov, A.A. & Moiseev, A.N. (2013) Analysis of an open non-Markovian GI–(GI| $\infty$ )<sup>K</sup> queueing network with high-rate renewal arrival process. *Problemy Peredachi Informatsii – Problems of Information Transmission*. 49(2). pp. 167–178. (In Russian).
14. Moiseev, A.N. & Nazarov, A.A. (2014) Asymptotic analysis of a multistage queueing system with a high-rate renewal arrival process. *Optoelectronics, Instrumentation and Data Processing*. 50(2). pp. 163–171.
15. Moiseev, A.N. & Nazarov, A.A. (2015) *Beskonechnolinyne sistemy i seti massovogo obsluzhivaniya* [Infinite-server queueing systems and networks]. Tomsk: NTL.

16. Melikov, A., Zadiranova, L. & Moiseev, A. (2016) Two Asymptotic Conditions in Queue with MMPP Arrivals and Feedback. *CCIS*. 678. pp. 231–240.
17. Nazarov, A.A., Moiseeva, S.P. & Morozova, A.S. (2008) Analysis of queueing system with unlimited number of servers and feedback by the method of limiting decomposition. *Vychislitel'nye tekhnologii – Computation Technologies*. 13(55). pp. 88–92. (In Russian).
18. Moiseeva, S.P., Ananina, I.A. & Nazarov, A.A. (2009) Research of streams in system  $M|GI|_{\infty}$  with repeated references the method of limiting decomposition. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 3(8). pp. 56–66. (In Russian).
20. Ananina, I.A. (2010) [Analysis of total flow of customers in queueing tandem system with unlimited number of servers and feedback by the method of limiting decomposition]. *Nauchnoe tvorchestvo molodezhi* [Scientific creativity of youth]. Proc. of the 14th All-Russian Conference. Tomsk, April 15–16, 2010. Tomsk. pp. 3–5. (In Russian).
20. Moiseeva, S.P. & Zakhornaya, I.A. (2011) Mathematical model of parallel retrial queueing of multiple requests. *Optoelectronics, Instrumentation and Data Processing*. 47(6). pp. 567–572. DOI: 10.3103/S8756699011060276
21. Ananina, I.A. (2011) Mathematical model of the income change process of the trading company expanding the presence in the market. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 2(15). pp. 5–14. (In Russian).
22. Shklennik, M., Moiseeva, S. & Moiseev, A. (2017) Analysis of queueing tandem with feedback by the method of limiting decomposition. *CCIS*. 800. pp. 147–157.

**ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ**

УДК 004.43

DOI: 10.17223/19988605/45/7

**С.Б. Арыков****РЕШЕНИЕ ПРИКЛАДНЫХ ЗАДАЧ В СИСТЕМЕ  
ПАРАЛЛЕЛЬНОГО ПРОГРАММИРОВАНИЯ АСПЕКТ**

Экспериментальная система параллельного программирования Аспект предназначена для исследования новых форм представления алгоритмов и методов задания управления в параллельных программах. В статье кратко рассмотрены архитектура системы Аспект, особенности языка программирования Аспект, результаты применения системы Аспект для решения задач разложения матрицы, статистического моделирования методом Монте-Карло и моделирования методом «частицы-в-ячейках».

**Ключевые слова:** система параллельного программирования Аспект; фрагментированное программирование; представление алгоритмов с высокой степенью неопределенности.

Современные суперкомпьютеры имеют сложную архитектуру. Они состоят из огромного количества вычислительных узлов, на каждом из которых есть несколько процессоров, каждый процессор содержит несколько ядер. Активно внедряются гибридные системы, где в дополнение к основным процессорам используются графические ускорители, специализированные вычислители типа Cell B.E., Matrix2000 или Intel Xeon Phi, вычислители на базе ASIC и PLD. Стремительный прогресс в использовании параллелизма аппаратной частью привел к большому разрыву между предоставляемыми ресурсами и возможностями по их эффективному использованию с помощью имеющихся средств разработки.

Для преодоления этого разрыва в России и за рубежом ведутся активные исследования по созданию средств параллельного программирования высокого уровня. К наиболее известным российским проектам можно отнести систему OpenTS [1] (Институт программных систем им. А.К. Айламазяна РАН), системы DVM [2] и НОРМА [3] (Институт прикладной математики им. М.В. Келдыша), язык mpC [4] (Институт системного программирования им. В.П. Иванникова), систему SFP [5] (Институт систем информатики им. А.П. Ершова), язык Пифагор [6] (Сибирский федеральный университет), систему ДВОР [7] (Южный федеральный университет), комплекс для параллельного программирования Graphplus templet [8] (Самарский государственный аэрокосмический университет). Результаты этих исследований показывают, что создание универсальной системы параллельного программирования высокого уровня трудно достижимо – слишком высока специфика отдельных предметных областей. Поэтому требуется дальнейшее изучение специализированных способов представления алгоритмов.

В статье представлены результаты исследований решения ряда прикладных задач в системе параллельного программирования Аспект. Аспект – это экспериментальная система, разрабатываемая в Институте вычислительной математики и математической геофизики СО РАН и предназначенная для исследования новых форм представления алгоритмов и методов задания управления в параллельной программе. В статье кратко рассмотрены архитектура системы Аспект, особенности языка программирования Аспект, результаты применения системы Аспект для решения задачи LU-разложения матриц, задачи оценки математических ожиданий аддитивных функционалов от траекторий диффузионных процессов и задачи о взаимодействии короткого лазерного импульса с плазмой методом «частицы-в-ячейках».

## 1. Система параллельного программирования Аспект

В основе системы параллельного программирования Аспект [9] лежат асинхронная модель вычислений с управлением на основе строгого частичного порядка [10] и фрагментированное программирование [11] как способ разработки параллельных программ. Архитектурно система Аспект состоит из транслятора с языка Аспект и исполнительной подсистемы.

Суть фрагментированного программирования заключается в представлении алгоритма решения задачи в виде множества *фрагментов данных* и множества *фрагментов кода*. Фрагмент кода получает на вход набор входных фрагментов данных, на основе которых вычисляет набор выходных фрагментов данных. Подстановка фрагментов данных в качестве параметров фрагмента кода называется *применением* фрагмента кода к фрагментам данных (один и тот же фрагмент кода может применяться к различным фрагментам данных). Совокупность фрагмента кода и его входных и выходных фрагментов данных называется *фрагментом вычислений*. На множестве фрагментов вычислений задается частичный порядок, т.е. множество ограничений на порядок выполнения фрагментов (управление).

Для реализации фрагментированного программирования на практике разработан язык управления вычислениями Аспект [12]. В нем фрагменты кода и фрагменты данных записываются на существующем процедурном языке программирования (в настоящее время поддерживается только C++), а фрагменты вычислений и частичный порядок их исполнения задаются средствами языка Аспект. Фрагменты вычислений, для которых зависимости явно не заданы, рассматриваются как независимые. Подробная информация о способах задания управления в языке Аспект приведена в [13].

Фрагментированная программа преобразуется транслятором языка Аспект в код на языке C++ и включается в одноименный класс, который предоставляет интерфейс для добавления фрагментов вычислений в очередь исполнения, а также осуществляет учет необходимых параметров (количество исполняемых фрагментов, количество готовых к исполнению фрагментов и др.). Для каждого фрагмента данных создается отдельный тип, с использованием которого объявляются данные задачи (переменные класса). Для каждого фрагмента кода генерируется метод класса, а также две переменных: признак завершения и переменная синхронизации.

Исполнительная подсистема Аспект состоит из набора функциональных модулей: менеджер потоков (управляет распределением работы между процессорами (ядрами); обращается за очередной порцией работы к планировщику), менеджер памяти (управляет распределением памяти в системе, осуществляет ее выделение / освобождение для очередей и фрагментированных программ), планировщик (управляет набором приоритизированных очередей, реализованных в виде мониторов; имеет сложность планирования  $O(1)$ ).

## 2. Разложение матриц

На примере задачи разложение квадратной матрицы  $A$  в произведение матриц  $L$  и  $U$  (LU-разложение) рассмотрим применение фрагментированного программирования для решения численных задач.

**Постановка задачи.** Пусть имеется матрица  $A = (a_{ij})$ ,  $i = 1 \dots N$ ,  $j = 1 \dots N$ . Ее разложение в виде  $A = LU$ , где  $L = (l_{ij})$  – нижняя треугольная матрица, а  $U = (u_{ij})$  – верхняя треугольная матрица с единицами на главной диагонали, может быть вычислено по формулам [14]:

$$l_{ij} = a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{kj} \quad (i \geq j), \quad (1)$$

$$u_{ij} = \frac{1}{l_{ii}} \left( a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj} \right), \quad (i < j). \quad (2)$$

**Фрагментация алгоритма.** Матрица  $A$  собирается из фрагментов данных (подматриц)  $A[i][j]$ . Выделяются четыре фрагмента кода F1, F2, F3 и F4. Первый фрагмент кода обрабатывает фрагменты данных, расположенные на главной диагонали матрицы  $A$ , по формулам (1) и (2); второй – фрагменты данных справа от диагональных элементов по формуле (1); третий – фрагменты данных под диагональными элементами по формуле (2); четвертый пересчитывает внутреннюю матрицу по формуле (3):

$$x_{ij} = a_{ij} - \sum_k l_{ik} u_{kj}. \quad (3)$$

Вычисления организуются по шагам. Первый шаг показан на рис. 1. Квадраты отображают фрагменты данных. Внутри каждого квадрата указано, какой экземпляр фрагмента вычислений его обрабатывает. Стрелками отображены зависимости между экземплярами фрагментов вычислений, а цифра в центре показывает порядок исполнения экземпляров.

Первым запускается на исполнение экземпляр фрагмента вычислений G1[0] (применение фрагмента кода F1 к фрагменту данных A[0][0]), после чего параллельно могут исполняться экземпляры G2[0][1], G2[0][2], G3[1][0] и G3[2][0]. Любой из экземпляров с порядком исполнения 3 может начать исполнение, как только завершатся оба экземпляра, от которых он зависит. Например, G4[0][1][2] может начать исполнение после завершения G3[1][0] и G2[0][2].

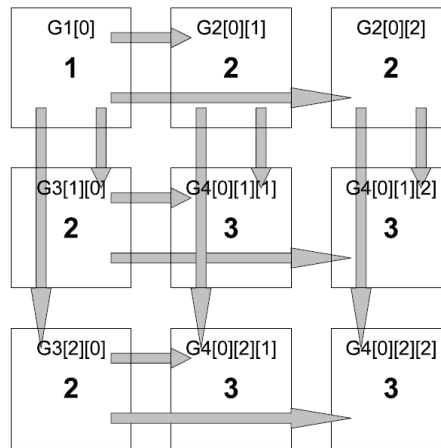


Рис. 1. Первый шаг вычисления LU-разложения матрицы A

Более подробно решение задачи LU-разложения матрицы разбирается в [11]. Текст программы LU-разложения матрицы на языке программирования Аспект можно найти в [13].

**Результаты эксперимента.** В качестве примера рассмотрим решение задачи о LU-разложении вещественной матрицы размера  $5\,040 \times 5\,040$  при размере фрагмента данных  $90 \times 90$  на системе HP ProLiant DL580 G5 (4 процессора Intel Xeon X7350, 16 ядер, 256 Гбайт RAM). Результаты представлены в табл. 1.

Таблица 1

Результаты измерений производительности программы решения задачи о LU-разложении

Характеристика	Количество ядер				
	1	2	4	8	16
Время счета, с	31,01	15,6	7,93	4,13	2,35
Ускорение	1	1,99	3,91	7,51	13,2

Аспект-программа демонстрирует практически линейное ускорение, что объясняется фрагментированным представлением алгоритма, за счет которого достигается более эффективное использование кэш-памяти. По этой же причине фрагментированный код лучше масштабируется.

### 3. Метод Монте-Карло

Статистическое моделирование является одним из численных методов решения математических задач, при котором искомые величины представляются вероятностными характеристиками какого-либо явления и приближенно определяются путем наблюдения за поведением модели этого явления [15]. Задачи статистического моделирования, как правило, требуют больших вычислительных ресурсов, поскольку в среднем выборочное значение может долго вычисляться и / или из соображений точности необходимо моделировать большое число выборочных реализаций.

Рассмотрим задачу оценки математических ожиданий аддитивных функционалов от траекторий диффузионных процессов, представляющих собой интегралы по некоторым малым областям расширенного фазового пространства от концентрации траекторий<sup>1</sup> [16]. На ее примере можно проследить, как в системе Аспект реализуются алгоритмы, которые можно фрагментировать на большое число независимых фрагментов вычислений.

**Постановка задачи.** Пусть трехмерный диффузионный процесс  $y(\cdot)$  задан системой стохастических дифференциальных уравнений в смысле Ито:

$$dy(t) = a(y(t))dt + b(y(t))dw(t), \quad 0 \leq t \leq T, \quad y(0) = y_0, \quad (4)$$

где  $a(y)$  – трехмерная векторная функция,  $b(y)$  – матричная функция размерности  $3 \times 3$ ,  $w(t)$  – трехмерный винеровский процесс,  $y_0$  – случайный вектор.

Моделирование траекторий системы (4) выполняется с помощью метода Эйлера:

$$y_{i+1} = y_i + \Delta t a(y_i) + \sqrt{\Delta t} b(y_i) \xi_i, \quad i = 0, 1, \dots, n,$$

где  $y_i$  – численное решение системы (4) в узле  $t_i$  равномерной сетки с постоянным шагом  $\Delta t = T/n$ , а  $\{\xi_i\}$  – последовательность независимых между собой нормальных случайных векторов с независимыми в совокупности компонентами, имеющими стандартное нормальное распределение.

Пусть необходимо вычислить полную концентрацию диффузионных траекторий  $\Phi(T, y_c)$  во временном интервале  $[0, T]$  в некоторой удаленной от источника точке  $y_c$ . При переходе к приближенной оценке точные диффузионные траектории заменим приближенными траекториями Эйлера, а точку  $y_c$  – шаром  $\Omega_c$  малого радиуса  $r_c$  с центром в точке  $y_c$ . Тогда будем иметь следующую статистическую оценку:

$$\Phi(T, y_c) \approx E \zeta_D = E \sum_{i=0}^n \omega(y_i) \Delta t,$$

где  $\omega(y) = \chi \Omega_c(y)/V_c$ ,  $V_c$  – объем шара  $\Omega_c$ .

**Фрагментация алгоритма.** Разобьем всё пространство моделируемых траекторий на  $m$  областей. Тогда входные фрагменты данных – это начальные значения генератора случайных чисел в каждой области, выходные фрагменты данных – значение концентрации в каждой области.

Фрагментов кода необходимо два: первый вычисляет концентрацию в интервале  $[j \cdot m, (j+1) \cdot m - 1]$  (здесь  $j$  – номер части пространства моделирования), а второй осуществляет сложение всех концентраций, полученных на каждой области пространства моделирования, и вычисление окончательного результата.

Фрагменты вычислений получаются применением фрагмента кода первого типа к каждому входному фрагменту данных. Так как моделируемые реализации не зависят друг от друга, потоковое управление между этими фрагментами вычислений отсутствует. Все фрагменты вычислений первого типа должны выполняться до начала исполнения фрагмента вычисления второго типа.

**Результаты эксперимента.** В качестве примера рассмотрим систему (4) на временном интервале  $[0, 10]$  с постоянным вектором сноса  $a \equiv (0, 0, 5)^T$ , единичной матрицей диффузии  $b \equiv I$ , шаром  $\Omega_c$  радиуса  $r_c = 0,1$  с центром в точке  $y_c = (8, 0, 0)$  и нулевым начальным условием  $y_0 \equiv 0$  [17]. Для метода Эйлера положим  $\Delta t = 10^{-2}$ .

Таблица 2

Результаты измерений производительности программы оценки математических ожиданий аддитивных функционалов от траекторий диффузных процессов методом Монте-Карло

Характеристика	Количество ядер				
	1	2	4	8	16
Время счета, с	23 805,50	12 051,00	5 939,00	2 990,00	1 510,68
Ускорение	1	1,97	4	7,96	15,75

<sup>1</sup> Текст последовательной программы был предоставлен М.А. Марченко, ИВМиМГ СО РАН.

Результаты измерений на системе HP ProLiant DL580 G5 для 64 000 000 смоделированных траекторий приведены в табл. 2. Видно, что ускорение практически линейное, что достигается за счет независимого расчета концентрации в каждом интервале, а также хорошей локальности данных, используемых в ходе расчета.

Эта задача, тривиальная с точки зрения распараллеливания, тем не менее показывает, что сама система программирования Аспект не вносит существенных задержек и не требует существенных накладных расходов на исполнение фрагментов.

#### 4. Метод частицы-в-ячейках

Методы частиц, как и конечно-разностные методы, образуют особую группу вычислительных алгоритмов, объединенных способом дискретизации исходной математической задачи [18]. Их применение широко распространено для решения сложных физических задач (например, задач физики бесстолкновительной плазмы). Вычислительная сложность метода объясняется огромным количеством частиц, необходимых для получения результатов с хорошей точностью.

Рассмотрим решение задачи о взаимодействии короткого лазерного импульса с плазмой методом «частицы-в-ячейках»<sup>2</sup> [19]. Она позволяет оценить применение системы Аспект для решения достаточно сложной задачи численного моделирования, включающей в себя несколько различных алгоритмов.

**Постановка задачи.** В области, имеющей форму параллелепипеда, находится фольга, представляемая тонким слоем плазмы. Плазма состоит из электронов и ионов одного типа. Лазерный импульс (в виде пакета электромагнитных волн различной амплитуды и поляризации) входит через поверхность параллелепипеда, взаимодействует с фольгой, частично отражаясь и проходя через нее.

Описываемый физический процесс представляется уравнениями Власова для ионов и электронов:

$$\frac{\partial f_{i,e}}{\partial t} + \vec{v} \frac{\partial f_{i,e}}{\partial \vec{r}} + \vec{F}_{i,e} \frac{\partial f_{i,e}}{\partial \vec{p}} = 0, \quad \vec{F}_{i,e} = q_{i,e} \left( \vec{E} + \frac{1}{c} [\vec{v}, \vec{B}] \right),$$

которые дополняются уравнениями Максвелла для электрического и магнитного полей:

$$\text{rot } \vec{B} = \frac{4\pi}{c} \vec{j} + \frac{1}{c} \frac{\partial \vec{E}}{\partial t},$$

$$\text{rot } \vec{E} = -\frac{1}{c} \frac{\partial \vec{B}}{\partial t},$$

$$\text{div } \vec{E} = 4\pi \rho,$$

$$\text{div } \vec{B} = 0.$$

В свою очередь, плотность заряда  $\rho$  и плотность тока  $\vec{j}$  определяются из функций распределения ионов и электронов:

$$\vec{j} = \sum_{i,e} q_{i,e} \int f_{i,e} \vec{v} d\vec{v},$$

$$\rho = \sum_{i,e} q_{i,e} \int f_{i,e} d\vec{v}.$$

Здесь индексы  $i$  и  $e$  обозначают тип частиц (ионы и электроны);  $q_{i,e}$  – заряд;  $f_{i,e}$  – функция распределения частиц.

В начальный момент времени частицы располагаются в области  $x_0 < x < x_0 + \delta$  и образуют тонкую фольгу ( $\delta \approx \lambda$ , где  $\lambda$  – длина волны лазерного импульса). Известны линейные размеры и амплитуда волн лазерного импульса. Задана плотность плазмы. Скорости частиц, компоненты напряженностей электрического и магнитного полей равны нулю.

Методы и алгоритмы решения, а также дискретизация задачи детально описаны в работе [19].

<sup>2</sup> Текст последовательной программы был предоставлен В.А. Вшивковым, ИВМиМГ СО РАН.

**Фрагментация алгоритма.** Значения компонент напряженности электрического и магнитного полей, а также плотности заряда и тока дискретизованы на одинаковых (возможно, сдвинутых во времени и пространстве) сетках, каждая из которых собирается из фрагментов данных размера  $f_i \times f_i \times f_k$ . Информация обо всех частицах хранится в массиве фрагментов данных по  $f_j$  частиц в каждом фрагменте.

Один шаг моделирования выполняется с помощью пяти А-программ, которые содержат по несколько фрагментов кода и вычисляют магнитное поле (две А-программы), ток, электрическое поле и передвижение частиц. Вычисление электромагнитных полей и тока по каждому измерению выполняется отдельными фрагментами кода. Отдельные фрагменты кода обрабатывают вхождение импульса в пространство моделирования и границы этого пространства. Исполнение А-программ происходит последовательно, весь параллелизм сосредоточен внутри.

Фрагменты вычислений получаются применением фрагментов кода к соответствующим фрагментам данных сетки (электромагнитные поля и ток) либо к фрагментам данных массива частиц.

**Результаты эксперимента.** В качестве примера рассмотрим решение задачи со следующими параметрами. Фольга расположена на расстоянии  $16$  длин волны лазерного импульса  $\lambda$  от левого края области. Амплитуда лазерного импульса равна  $20$ , длина и ширина импульса –  $30\lambda$  и  $10\lambda$  соответственно. Временной шаг  $\tau$  составляет  $0,01$ , число частиц одного заряда в ячейке –  $32$ , общее число частиц –  $10\,485\,760$ . Область разделена на  $40 \times 40 \times 1$  ячеек, размер пространства –  $5 \times 5 \times 0,125$ .

Результаты измерений на сервере лаборатории синтеза параллельных программ ИВМиМГ СО РАН для 5 шагов моделирования приведены в табл. 3.

Таблица 3

**Результаты измерений производительности программы моделирования взаимодействия короткого лазерного импульса с плазмой методом частицы-в-ячейках**

Характеристика	Количество ядер				
	1	2	4	8	16
Время счета, без НТ	21,92	11,03	6,14	3,48	н/д
Ускорение, без НТ	1	1,99	3,57	6,30	н/д
Время счета, с НТ	21,92	18,68	10,0	5,1	3,57

Особенность задачи состоит в том, что расчет электромагнитных полей распараллеливается хуже (из-за интенсивного доступа к памяти), чем расчет движения частиц. Поэтому чем больше частиц используется при моделировании, тем лучше показатель ускорения. В то же время большее количество частиц приводит к лучшей точности.

### Заключение

В статье рассмотрено применение системы параллельного программирования Аспект для решения ряда практических задач численного моделирования. Показано, что если задача допускает хорошее распараллеливание, она может быть эффективно реализована в системе Аспект без существенных накладных расходов на организацию исполнения фрагментов вычислений.

Результаты измерения производительности фрагментированных программ подтвердили гипотезу исследования: по высокоуровневому фрагментированному представлению алгоритма для выбранной предметной области возможна автоматическая генерация достаточно эффективных параллельных программ.

### ЛИТЕРАТУРА

1. Абрамов С.М., Кузнецов А.А., Роганов В.А. Кроссплатформенная версия Т-системы с открытой архитектурой // Вычислительные методы и программирование. 2007. Т. 8, № 1. С. 175–180.
2. Бахтин В.А., Клинов М.С., Крюков В.А., Поддериюгина Н.В., Притула М.Н., Сазанов Ю.Л. Расширение DVM-модели параллельного программирования для кластеров с гетерогенными узлами // Вестник Южно-Уральского государственного университета. Сер. Математическое моделирование и программирование. 2012. № 18, вып. 12. С. 82–92.



3. Андрианов А.Н., Бугеря А.Б., Ефимкин К.Н., Колударов П.И. Модульная архитектура компилятора языка Норма+ // Препринты ИПМ им. М. В. Келдыша. 2011. № 064. 16 с.
4. Lastovetsky A. Adaptive parallel computing on heterogeneous networks with mpC // Parallel Computing. 2002. V. 28, No. 10. P. 1369–1407.
5. Kasyanov V.N., Stasenko A.P. A functional programming system SFP: Sisal 3.1 language structures decomposition // LNCS. 2007. V. 4671. P. 62–73.
6. Легалов А.И. Функциональный язык для создания архитектурно-независимых параллельных программ // Вычислительные технологии. 2005. Т. 10, № 1. С. 71–89.
7. Штейнберг Б.Я., Абрамов А.А., Алымова Е.В., Баглий А.П., Гуда С.А., Дубров Д.В., Кравченко Е.Н., Морылев Р.И., Нис З.Я., Петренко В.В., Полуян С.В., Скиба И.С., Шаповалов В.Н., Штейнберг О.Б., Штейнберг Р.Б., Юрушкин М. Диалоговый высокоуровневый автоматический распараллеливатель (ДВОР) // Труды Всероссийской суперкомпьютерной конференции «Научный сервис в сети Интернет» (Новороссийск, 20–26 сентября 2010 г.). М. : Изд-во МГУ, 2010. С. 71–75.
8. Востокин С.В., Литвинов В.Г., Хайрутдинов А.Р. Применение комплекса параллельного программирования Graphplus template в моделировании // Программные продукты и системы. 2012. № 3. С. 14–18.
9. Arykov S.B., Malyshev V.E. Asynchronous Language and System of Numerical Algorithms Fragmented Programming // LNCS. 2009. Vol. 5698. P. 1–7.
10. Арыков С.Б. Асинхронная модель вычислений над общей памятью // Вестник УГАТУ. 2016. Т. 20. № 3. С. 114–121.
11. Арыков С.Б. Фрагментированное программирование численных задач // Сборник научных статей VII Международной научно-практической конференции «Многоядерные процессоры, параллельное программирование, ПЛИС, системы обработки сигналов» (Барнаул, 10–11 марта 2017 г.). Барнаул : Изд-во Алт. ун-та, 2017. С. 20–24.
12. Арыков С.Б. Асинхронное программирование численных задач // Параллельные вычислительные технологии (ПаВТ'2010) : труды международной научной конференции (Уфа, 29 марта – 2 апреля 2010 г.). Челябинск : Изд. центр ЮУрГУ, 2010. С. 28–39.
13. Arykov S. Defining Order of Execution in Aspect Programming Language // LNCS. 2017. V. 10421. P. 265–271.
14. Фаддеев Д.К., Фаддеева В.Н. Вычислительные методы линейной алгебры. 2-е изд. М. : Наука, 1963. 656 с.
15. Михайлов Г.А., Войтишек А.В. Численное статистическое моделирование. Методы Монте-Карло. М. : Академия, 2006. 368 с.
16. Марченко М.А., Михайлов Г.А. Весовые алгоритмы статистического моделирования диффузионных процессов // Журнал вычислительной математики и математической физики. 2003. Т. 43, № 4. С. 571–584.
17. Марченко М.А. Комплекс программ MONC для распределенных вычислений методом Монте-Карло // Сибирский журнал вычислительной математики. 2004. Т. 7, № 1. С. 43–55.
18. Григорьев Ю.Н., Вишников В.А., Федорук М.П. Численное моделирование методами частиц-в-ячейках. Новосибирск : Изд-во СО РАН, 2004. 360 с.
19. Вишников В.А., Вишников К.В., Дудникова Г.И. Алгоритмы решения задачи взаимодействия лазерного импульса с плазмой // Вычислительные технологии. 2001. Т. 6, № 2. С. 47–63.

Поступила в редакцию 18 апреля 2018 г.

Arykov S.B. (2018) SOLUTION OF APPLIED PROBLEMS IN THE PARALLEL PROGRAMMING SYSTEM ASPECT. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitel'naya tekhnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 45. pp. 59–67

DOI: 10.17223/19988605/45/7

Modern supercomputers have a complex heterogeneous architecture, which can include in various combinations CPU, GPU, DSP, ASIC, PLD, etc. It is difficult to efficiently use all these hardware resources with the available software tools for developing parallel programs, therefore further development of automated high-level parallel programming systems is an actual task.

Programming system Aspect is an experimental system of parallel programming developed at the Institute of Computational Mathematics and Mathematical Geophysics of the Siberian Branch of the Russian Academy of Sciences and designed to study new forms of representation of algorithms and methods for specifying control in a parallel programs. It is based on an asynchronous model of computation controlled by strict partial order and fragmented programming as a way of developing parallel programs. System Aspect consists of a translator from the Aspect language and the executive subsystem.

The essence of the fragmented programming is to represent an algorithm and its implementing program as a set of data fragments and code fragments. In the course of execution, the fragmented structure of a program is kept.

Each code fragment is supplied with a set of input data fragments (formal parameters) used to compute output data fragments. The substitution of data fragments as parameters into a code fragment is referred to as applying a code fragment to data fragments (the same code fragment may be applied to different data fragments). The code fragment with its input and output data fragments constitutes

a computation fragment. On the set of computation fragments, a partial order (control scheme) is defined. The resulting program is created from such computation fragments.

Aspect language for computations control was developed to implement fragmented programming on practice. In that language, code fragments and data fragments are defined in existing procedural programming language (currently only C++ is supported), while special constructs are used to define computation fragments and dependencies between them. The construction of control scheme is based on two key features: strict partial order and explicit support of massive control schemes. Computation fragments that do not have explicitly specified dependencies are considered independent and can be executed simultaneously.

Fragmented algorithms for solving LU-matrix decomposition problem, estimating the mathematical expectations of additive functionals from the trajectories of diffusion processes problem, and the interaction of a short laser impulse with plasma by the “particle-in-cell” method were developed and implemented in the Aspect system.

The results of experiments on the HP ProLiant DL580 G5 shared memory system (4 Intel Xeon X7350 processors, 16 cores, 256 GB RAM) showed that due to the locality of the data fragments and the efficient use of cache memory, fragmented programs allow achieving acceleration, close to linear, and provide good scalability. Experiments also showed that Aspect system does not introduce significant delays and does not require significant overhead for the execution of fragmented program.

Fragmentation of the algorithm may require its significant modification, and, therefore, cannot be performed automatically. This is a price for the good properties of fragmented programs.

**Keywords:** parallel programming system Aspect; technology of fragmented programming; representation of algorithms with high degree of nonproceduralism.

**ARYKOV Sergey Borisovich** (Candidate of Physics and Mathematics, Associate Professor, Institute of Computational Mathematics and Mathematical Geophysics of Siberian Branch of Russian Academy of Science, Novosibirsk State Technical University, Novosibirsk, Russian Federation).

E-mail: arykov@sscc.ru

## REFERENCES

1. Abramov, S.M., Kuznetsov, A.A. & Roganov, V.A. (2007) Cross-platform version of T-system with open architecture. *Vychislitelnye metody i programmirovaniye – Numerical Methods and Programming*. 8(1). pp. 175–180. (In Russian).
2. Bakhtin, V.A., Klinov, M.S., Kryukov, V.A., Podderyugina, N.V., Pritula, M.N. & Sazanov, Yu.L. (2012) Extension of DVM parallel programming model for clusters with heterogeneous nodes. *Vestnik Yuzhno-Uralskogo gosudarstvennogo universiteta: seriya “Matematicheskoe modelirovaniye i programmirovaniye” – South Ural State University Bulletin. Mathematical Modelling, Programming & Computer Software*. 18(12). pp. 82–92. (In Russian).
3. Andrianov, A.N., Bugerya, A.B., Efimkin, K.N. & Koludarov, P.I. (2011) Modular architecture of Norma+ language compiler. *Preprinty IPM im. M.V. Keldysha – Keldysh Institute Preprints*. 64. (In Russian).
4. Lastovetsky, A. (2002) Adaptive parallel computing on heterogeneous networks with mpC. *Parallel Computing*. 28(10). pp. 1369–1407. DOI: 10.1016/S0167-8191(02)00159-X
5. Kasyanov, V.N. & Stasenko, A.P. (2007) A functional programming system SFP: Sisal 3.1 language structures decomposition. *LNCS*. 4671. pp. 62–73.
6. Legalov, A.I. (2005) Functional language for creating architecturally independent parallel programs. *Vychislitelnye tekhnologii – Computational Technologies*. 10(1). pp. 71–89. (In Russian).
7. Steinberg, B.Ya., Abramov, A.A., Alimova, E.V., Bagliy, A.P., Guda, S.A., Dubrov, D.V., Kravchenko, E.N., Morylev, R.I., Nis Z.Ya., Petrenko, V.V., Poluyan, S.V., Skiba, I.S., Shapovalov, V.N., Steinberg, O.B., Steinberg, R.B. & Yurushkin, M. (2010) [Dialog high-level automatic parallelizer (DHAP)]. *Nauchnyy servis v seti Internet* [Scientific Service on the Internet]. Proc. of the All-Russian Supercomputer Conference. Novorossiysk, September 20–26, 2010. Moscow: Moscow State University. pp. 71–75. (In Russian).
8. Vostokin, S.V., Litvinov, V.G. & Khayrutdinov, A.R. (2012) Application of the parallel programming complex Graphplus templet in modeling. *Programmye produkty i sistemy – Software and Systems*. 3. pp. 14–18. (In Russian).
9. Arykov, S.B. & Malyshev, V.E. (2009) Asynchronous Language and System of Numerical Algorithms Fragmented Programming. *LNCS*. 5698. pp. 1–7. DOI: 10.1007/978-3-642-03275-2\_1
10. Arykov, S.B. (2016) Asynchronous model of computation for systems with shared memory. *Vestnik UGATU*. 20(3). pp. 114–121. (In Russian).
11. Arykov, S.B. (2017) [Fragmented programming of numerical problems]. *Mnogoyadernye protsessory, parallel'noye programmirovaniye, PLIS, sistemy obrabotki signalov* [Multi-core processors, parallel programming, FPGA, signal processing systems]. Proc. of the Seventh International Conference. Barnaul, March 10–11, 2017. Barnaul: Altai State University. pp. 20–24. (In Russian).
12. Arykov, S.B. (2010) [Asynchronous programming of numerical tasks]. *Parallel'nye vychislitel'nye tekhnologii (PaVT'2010)* [Parallel computing technologies (PaVT'2010)]. Proc. of the International Conference. Ufa, March 29 – April 2, 2010. Chelyabinsk: South Ural State University. pp. 28–39. (In Russian).

13. Arykov, S.B. (2017) Defining Order of Execution in Aspect Programming Language. *LNCS*. 10421. pp. 265–271. DOI: 10.1007/978-3-319-62932-2\_25
14. Faddeev, D.K. & Faddeeva V.N. (1963) *Vychislitel'nye metody lineynoy algebrы* [Computational Methods of Linear Algebra]. 2nd ed. Moscow: Nauka.
15. Mikhailov, G.A. & Voishtek, A.V. (2006) *Chislennoe statisticheskoe modelirovanie* [Numerical statistical modeling. Monte-Carlo methods]. Moscow: Akademiya.
16. Marchenko, M.A. & Mikhailov, G.A. (2003) Weighted algorithms for statistical modeling of diffusion processes. *Zhurnal vychislitel'noy matematiki i matematicheskoy fiziki*. 43(4). pp. 571–584. (In Russian).
17. Marchenko, M.A. (2004) Kompleks programm MONC dlya raspredelennykh vychisleniy metodom Monte-Karlo [A set of programs MONC for distributed computing by Monte Carlo method]. *Sibirskiy zhurnal vychislitel'noy matematiki – Siberian Journal of Computational Mathematics*. 7(1). pp. 43–55.
18. Grigoriev, Yu.N., Vshivkov, V.A. & Fedoruk, M.P. (2004). *Chislennoe modelirovanie metodami chastits-v-yacheykakh* [Numerical simulation by particle-in-cell methods]. Novosibirsk: SB RAS.
19. Vshivkov, V.A., Vshivkov, K.V. & Dudnikova, G.I. (2001) Algorithms for solving the problem of interaction of a laser impulse with a plasma. *Vychislitelnye tekhnologii – Computational Technologies*. 6(2). pp. 47–63. (In Russian).

УДК 004.8, 638.14

DOI: 10.17223/19988605/45/8

**V.A. Kulyukin, S. Mukherjee, Yu.B. Burkatovskaya****CLASSIFICATION OF AUDIO SAMPLES BY CONVOLUTIONAL NETWORKS IN AUDIOBEEHIVE MONITORING**

In the investigation, we consider the problem of classification of audio samples resulting from the audio beehive monitoring. Audio beehive monitoring is a key component of electronic beehive monitoring (EBM) that can potentially automate the identification of various stressors for honeybee colonies. We propose to use convolutional neural networks (ConvNets) and compare developed ConvNets in classifying audio samples from electronic beehive monitors deployed in live beehives. As a result, samples are placed in one of the three non-overlapping categories: bee buzzing (B), cricket chirping (C), and ambient noise (N). We show that ConvNets trained to classify raw audio samples perform slightly better than ConvNets trained to classify spectrogram images of audio samples. We demonstrate that ConvNets can successfully operate in situ on low voltage devices such as the credit card size raspberry pi computer.

**Keywords:** deep learning; machine learning; convolutional neural networks; audio classification; audio processing; electronic beehive monitoring.

Beekeepers ascertain the state of their honey bee colonies by listening to them, because bee buzzing carries information on colony behavior and phenology. While experienced beekeepers can tell audio changes in sounds produced by stressed colonies, they may not always be able to determine the exact causes of the changes without hive inspections. Unfortunately, hive inspections can be quite invasive in that they disrupt the life cycle of bee colonies and put additional stress on the bees.

Audio beehive monitoring is a key component of EBM that attracts considerable research and development effort, because honey bees generate specific sounds when exposed to stressors such as failing queens, predatory mites and airborne toxicants. Bromenschenk et al. [1] designed a system for profiling acoustic signatures of honeybee colonies, analyzing these signatures, and identifying various stressors for the colonies from this analysis. A foam insulated microphone probe assembly is inserted through a hole drilled in the back of the upper super of a Langstroth hive that consists of two supers. The probe's microphone is connected via a microphone amplifier to a computer with an audio card and audio processing software that performs the Fast Fourier Transform (FFT) of captured wav files. The frequency spectrum is processed into a running average and an overall average to associate sound spectra with acoustic variations. In an experiment, bee colonies treated with naphtha, ammonia, or toluene produced statistically different acoustic signatures.

Ferrari et al. [2] assembled a system for monitoring swarm sounds in beehives. The system consists of a microphone, a temperature sensor, and a humidity sensor placed in a beehive and connected to a computer in a nearby barn via underground cables. In a field experiment, the sounds were recorded at a sample rate of 2 kHz and analyzed with Matlab. The researchers monitored three beehives for 270 hours and observed that swarming was indicated by an increase of the buzzing frequency at about 110 Hz with a peak at 300 Hz when the swarm left the hive. Another finding was that a swarming period correlated with a rise in temperature from 33 °C to 35 °C with a temperature drop to 32 °C at the actual time of swarming.

Mezquida and Martinez [3] developed a distributed audio monitoring system for apiaries. The system consists of nodes and wireless sensor units with one node per apiary and one sensor unit per hive. A node is an embedded solar-powered computer. A sensor unit consists of an omnidirectional microphone and a temperature sensor. Sensor units are placed at the bottom of each hive protected by a grid to prevent propolization. The system was tested in an apiary of 15 Langstroth hives where up to 10 hives were monitored continuously by taking 8 second audio samples every hour with a sampling rate of 6,250 Hz. The timestamped frequency spectra obtained with Fourier transform and temperature data were logged in a SQL database from May 2008

to April 2009. The researchers reported that sound volume and sound intensity at medium and low frequencies showed no identifiable daily patterns in the winter but exhibited distinguishable daily patterns in the spring.

Kulyukin et al. [4] designed an algorithm for digitizing bee buzzing signals with harmonic intervals into A440 piano note sequences to obtain symbolic representations of audio signals over specific time periods. When viewed as a time series, such sequences correlate with other timestamped data such as estimates of bee traffic levels or temperatures [5]. The note range detected by the proposed algorithm on 3421.52MB of 30-second wav files contained the first four octaves, with the lowest note being A0 and the highest note being F#4. In a 24-hour period, the peaks in the frequency counts started in the first octave (D1), shifted higher to C3 and C#3 in the third octave, and returned back to the first octave at the end of the selected time period. Several notes in the fourth octave, e.g., F#4 and C#4, were also detected but their frequency counts were substantially lower than those of the peaks in the first three octaves.

Ramsey et al. [6] presented a method to analyze and identify honeybee swarming events through a computational analysis of acoustic vibrations captured with accelerometers placed on outside walls of hives. The researchers placed two accelerometers on the outer walls of two Langstroth hives with *Apis mellifera* honey bees, one accelerometer per hive. A cavity was drilled in the center of the back wall of each hive. Both hives were located in close proximity to each other and approximately 10 meters away from a house with a PC running a Linux distribution. The accelerometers were placed in the cavities and connected to a dual channel conditioner placed between the hives and encased in a waterproof acrylic box. The conditioner was coupled to the indoor PC with two coaxial cables through which the captured vibration samples were saved on the PC's hard disk. An ad hoc roof was placed above both hives to control for vibrations caused by rain drops. The vibration samples, each one hour long, were logged from April to June 2009. Averaged frequency spectra were computed for a frequency resolution of 20 Hz and an averaging time of 510 seconds. The spectra were combined into one day long spectrograms which were analyzed with the Principal Component Analysis (PCA) for feature extraction. The witnessed swarming events were juxtaposed with the corresponding spectrograms and were found to exhibit a unique set of features.

In this article, we contribute to the body of research on audio beehive monitoring by designing and evaluating ConvNets that classify audio samples from microphones located above landing pads of Langstroth beehives. The proposed audio processing methods are applicable to any audio samples obtained from microphones placed inside or outside beehives. To ensure the replicability of our findings reported in this article we have made public our source code [7] and our curated dataset of 9110 audio samples [8].

## **1. Materials and methods**

**Hardware.** All audio data were captured by BeePi, a multi-sensor electronic beehive monitoring system we designed and built in 2014. A BeePi monitor consists of a raspberry pi computer, a miniature camera, a microphone splitter, 3 or 4 microphones connected to the splitter, a solar panel, a temperature sensor, a battery, and a hardware clock. The solar version of BeePi is shown in Fig 1. All hardware components, except for solar panels, are placed in a Langstroth super (see Fig 2). The solar panel is placed either on top of or next to a beehive, as shown in Fig 1. We used the Pi Model B+ 512MB RAM models, Pi T-Cobblers, half-size breadboards, waterproof DS18B20 digital temperature sensors, and Pi cameras. For solar harvesting, we used the Renogy 50 watts 12 Volts monocrystalline solar panels, Renogy 10 Amp PWM solar charge controllers, and Renogy 10ft 10AWG solar adaptor kits. For power storage, we used two types of rechargeable batteries: the UPG 12V 12Ah F2 lead acid AGM deep cycle battery and the Anker Astro E7 26800mAh battery. Each BeePi unit is equipped with four Neewer 3.5mm mini lapel microphones that are placed either above the landing pad or embedded in beehive walls. In this study, the microphones were placed 10cm above the landing pad with two microphones on each side, as shown in Fig 3.

BeePi monitors have so far had four field deployments. The first deployment was in Logan, UT (Sept. 2014) when one BeePi unit was placed into an empty hive and ran exclusively on solar power for two weeks. The second deployment was in Garland, UT (Dec. 2014 – Jan. 2015) when one BeePi unit was placed in a hive with overwintering bees and successfully operated for nine out of the fourteen days of deployment

exclusively on solar power and collected about 200 MB of data. The third deployment was in North Logan, UT (Apr. 2016 – Nov. 2016) where four BeePi units were placed into four beehives at two small apiaries and collected 20GB of data. The fourth deployment was in both Logan and North Logan, UT (Apr. 2017 – Sept. 2017) when four BeePi units again placed into four beehives at two small apiaries to collect 220GB of audio, video, and temperature data.



Fig. 1. Deployed BeePi monitor

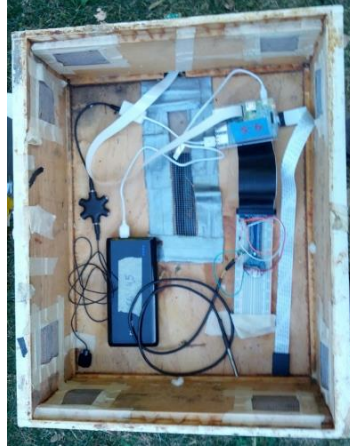


Fig. 2. BeePi hardware components



Fig. 3. Four microphones above a landing pad

**Audio Data.** The training and testing audio data were captured by two of our four BeePi monitors deployed in Logan, UT and North Logan, UT from April 2017 to September 2017 in four Langstroth beehives with Italian honey bee colonies. Each monitor saved a 30-second audio wav file every 15 minutes recorded with the four USB microphones placed 10cm above the beehives' landing pads and plugged into a raspberry pi computer. Each 30-second audio sample was segmented into 2-second wav samples with a 1-second overlap, resulting in 28 2-second wav samples per one 30-second file.

We obtained the ground truth classification by manually labeling 9110 2-second audio samples by listening to each sample and placing it into one of the three non-overlapping categories: bee buzzing (B), cricket chirping (C), and ambient noise (N). The B category consisted of the samples where we could hear the buzzing of bees. The C category consisted of the files where we could hear the chirping of crickets. The N category included all samples where we could not clearly hear either bee buzzing or cricket chirping. The final labeled dataset included 3,000 B samples, 3,110 C samples, and 3,000 N samples.

**Convolutional Neural Networks.** Representation learning is a branch of AI that investigates automatic acquisition of representations for detection and classification from raw signals. Standard machine learning techniques are limited in their ability to acquire representations from raw data, because they require considerable feature engineering to develop robust feature extractors that convert raw signals into feature vectors used in classification methods [9]. ConvNets are deep networks designed to acquire multi-layer representations from raw data. Starting from raw input, each layer is a transformation function that transforms its input to the one acceptable for the next layer. Many features of these layers are learned automatically by a general purpose procedure known as backpropagation [10]. Compositions of these transformations emulate complex classification functions. ConvNets have been successfully applied to image classification [11, 12], speech recognition [13], and bioinformatics [14].

In ConvNets, filters of various sizes are convolved with a raw input signal to obtain a stack of filtered signals. This stack is referred to as the convolutional layer. The convolved signals are normalized. A standard choice for normalization is the rectified linear units (ReLUs) that convert all negative values to 0's [15]. This layer is called the normalization layer. The size of each signal from the convolutional layer or the normalization layer can be reduced in a pooling layer where a window and a stride are used to shift the window in stride steps across the input signal retaining the maximum value from each window. The fourth type of layer in a ConvNet is the fully connected layer (FC) where the stack of processed signals is construed as a 1D vector where each value is connected via a synapse (a weighted connection that transmits a value from one unit to another)

to each node in the next layer. In addition to standard layers, ConvNets can have custom layers that implement arbitrary transformation functions. The architectural features of a ConvNet that cannot be dynamically changed through backpropagation are called hyperparameters and include the number and size of features in convolutional layers, the window size and stride in maxpooling layers, and the number of neurons in FC layers.

A fundamental research objective of this study is to compare ConvNets trained on spectrograms obtained through Fourier analysis with ConvNets trained on raw audio data. Toward that end, two ConvNet architectures were designed and trained to handle two types of input signal. The first architecture (SpectConvNet) was designed to classify spectrogram images. The second architecture (RawConvNet) was designed to classify raw audio waveforms.

SpectConvNet designed to classify spectrogram images is shown in Fig 4. The input signal is a  $100 \times 100 \times 32$  RGB image of the spectrogram of a 2-second wav audio sample. The signal goes through a  $100 \times 100 \times 100$  convolutional (Conv-2D) layer with 100 filters and then through a  $50 \times 50 \times 100$  maxpooling (Maxpool-2D) layer with a kernel size of 2. The first Maxpool-2D layer is followed by a stack of two Conv-2D layers with 100 and 200 filters, respectively. The output of the second Conv-2D layer is processed by a  $25 \times 25 \times 200$  Maxpool-2D layer with a kernel size of 2 and then by a stack of two fully connected (FC) layers.

The first FC layer has fifty neurons whereas the second FC layer has three neurons corresponding to the B, C, and N classes. The first FC layer (FC-50) uses the rectified linear unit (ReLU) activation function. To avoid overfitting, random dropout with a keep probability of 0.5 is applied to the output of the first FC layer. The second FC layer (FC-3) has the softmax activation function [16]. The softmax function is used to represent a probability distribution over  $k$  different possible outcomes by mapping a  $k$ -dimensional vector of arbitrary real values into another  $k$ -dimensional vector whose values are in the range (0,1) that add up to 1. The softmax function emphasizes the largest values and de-emphasizes the smallest values. Specifically, if  $X = [x_1, \dots, x_k]$  is a  $k$ -dimensional vector, then the softmax function maps each element  $x_i$  of  $X$  to the real value specified in Eq. 1. The softmax function is used to transform a  $k$ -dimensional vector  $X$  to a  $k$ -dimensional vector  $Z$ , as shown in Eq. 2.

$$S(x_i) = e^{x_i} / \sum_{j=1}^k e^{x_j}, \quad \text{where } 1 \leq i \leq k; \quad (1)$$

$$Z = [S(x_1), \dots, S(x_k)]. \quad (2)$$

In multiclass classification, given a sample vector  $X$  and its score vector  $Z$ , the probability of  $X$  belonging to class  $j$  is given in Eq.3, where  $Z_i$  is the  $i$ -th element of  $Z$ .

$$P(X = i | Z) = e^{Z_i} / \sum_{j=1}^k e^{Z_j}. \quad (3)$$

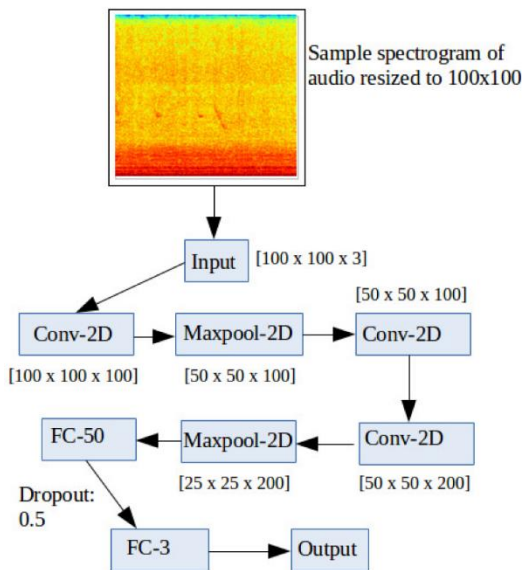


Fig. 4. SpectConvNet architecture

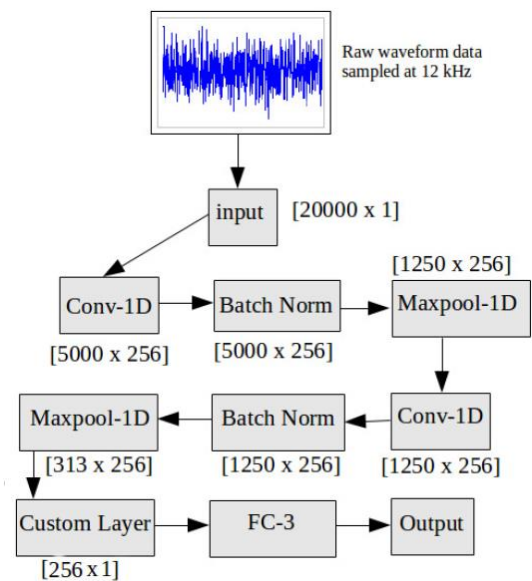


Fig. 5. RawConvNet architecture



RawConvNet designed to classify raw audio samples is shown in Fig 5. The input signal to RawConvNet is a raw audio wav file downsampled to 12 245 kHz and normalized to have a mean of 0 and a variance of 1. The input signal goes to a  $5000 \times 256$  Conv-1D layer with 256 filters with a filter size  $n$ , where  $n \in \{3, 10, 30, 80, 100\}$ , and a stride of 4. To discover the optimal sizes of receptive fields for the convolutional layers, the following receptive field sizes for both layers were used during training and evaluation:  $3 \times 1$ ,  $10 \times 1$ ,  $30 \times 1$ ,  $80 \times 1$ , and  $100 \times 1$ . The impact of the receptive field's size on the ConvNet's performance is described in the section about the experiments. The output of the Conv-1D layer goes into a  $5000 \times 256$  batch normalization layer to reduce internal covariate shift [17]. The normalized output is given to a  $1250 \times 256$  Maxpool-1D layer with a kernel size of 4. After the first Maxpool-1D layer the signal goes through a  $1250 \times 256$  Conv-1D layer with a stride of 1 the output of which is normalized with another  $1250 \times 256$  batch normalization layer and subsequently maxpooled with a  $313 \times 256$  Maxpool-1D layer. The output of this Maxpool-1D layer is processed by a  $256 \times 1$  custom layer that calculates the mean of the tensor along the first axis. A tensor is an  $n$ -dimensional array. For example, a 1D tensor is a vector, a 2D tensor is a matrix, a 3D tensor is a cube, and a 4D tensor is a vector of cubes, a 5D tensor is a matrix of cubes and so on. The custom layer is designed to calculate the global average over each feature map, thus reducing each feature map tensor into a single float value.

The output of the custom layer goes into a FC layer with 3 neurons (FC-3) that correspond to the B, C, and N classes. Both convolutional layers use ReLUs. Each convolutional layer has  $L_2$  regularization and a weight decay of 0.001. Batch normalization is applied to the output of each convolutional layer to reduce the computational costs and accelerate the learning process. Batch normalization forces the activations in feature maps to have a standard deviation of 1 and a mean of 0. The output of each convolutional layer is a tensor of rank 4, i.e.,  $[B, H, W, D]$ , where  $B$  is a batch size,  $H \times W$  is a feature map size, and  $D$  is the number of channels or dimensions. The output of the batch normalization layer is controlled by two parameters that are learned to best represent the activation of the feature maps.

## 2. Experiments

Both ConvNet architectures were trained on a Xeon(R) CPU W3565 computer with 5.8GiB of RAM and 64-bit Ubuntu 14.04 LTS. Both architectures were trained with the Adam optimizer and categorical cross-entropy [18]. The Adam optimizer is a variant of stochastic gradient descent that can handle sparse gradients on noisy problems. Both architectures were trained for 100 epochs with a batch size of 128 and a 70/30 train/test split of the 9110 audio samples described in the section on audio data, which resulted in 6377 audio samples used for training (70% of 9110 audio samples) and 2733 audio samples used for testing (30% of 9110 audio samples). Table 1 gives the performance results comparing SpectConvNet with RawConvNet with different sizes of the receptive field. The results in the table show that when the validation accuracy is lowest (98.87%) when the size of the receptive field is 3 and highest (99.93%) when the size of the receptive field is 80. The runtime per epoch is slowest when the receptive field's size is 3. Figures 5 and 6 show the validation loss and accuracy curves for RawConvNet for different receptive field sizes, respectively. As the number of training epochs increases the validation losses decrease while the validation accuracies increase for each receptive field size.

Table 1

Performance Summary of Two ConvNet Architectures

Model Name	Training Loss	Training Acc	Validation Loss	Validation Acc	Runtime per Epoch
Spectrogram ConvNet	0,00619	99,04%	0,00702	99,13%	690 secs
RawConvNet ( $n = 3$ )	0,02759	99,24%	0,03046	98,87%	460 secs
RawConvNet ( $n = 10$ )	0,01369	99,74%	0,01429	99,60%	462 secs
RawConvNet ( $n = 30$ )	0,00827	99,91%	0,00679	99,71%	465 secs
RawConvNet ( $n = 80$ )	0,00692	99,85%	0,00432	99,93%	462 secs
RawConvNet ( $n = 100$ )	0,00456	99,97%	0,00785	99,74%	505 secs



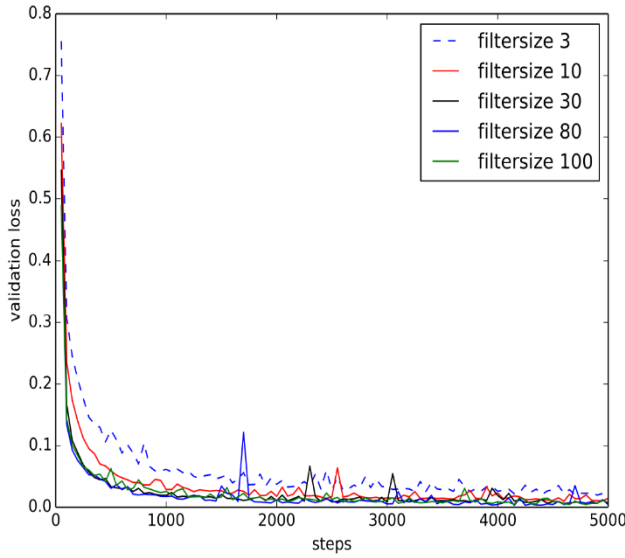


Fig. 6. Validation Loss Curves for RawConvNet

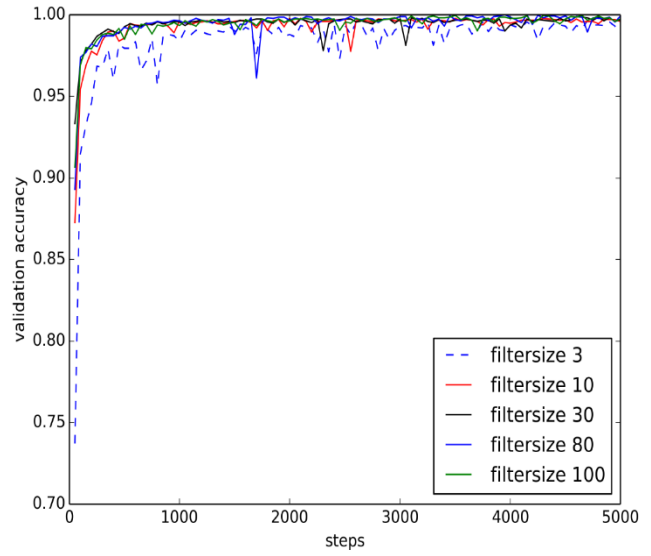


Fig. 7. Validation Accuracy Curves for RawConvNet

To estimate the contribution of the custom layer, three deeper models were designed where the custom layer was replaced with various combinations of FC and convolutional layers. In the first ConvNet (ConvNet 1), the first three layers and the fifth layer are identical to RawConvNet and the fourth layer is replaced with an FC layer with 256 neurons. In the second ConvNet (ConvNet 2), the first three layers are identical to RawConvNet. In layer 4, batch normalization is performed on the output of layer 3 and is given to layer 5 consisting of an FC layer with 256 neurons with a dropout of 0.5. Layer 6 is identical to the last layer in RawConvNet and ConvNet 1 and consists of an FC layer with 3 neurons that correspond to the B, C, and N classes with the softmax activation function. In the third ConvNet (ConvNet 3), the first three layers are identical as the three layers of ConvNets 1 and 2. In layer 4, the output of layer 3 is convolved with 256 filters with a filter size of 3 and a stride of 1. In layer 5, batch normalization is performed on the output of layer 4 before passing it to layer 6 that consists of an FC layer with 256 neurons with a dropout of 0.5. Layer 7 is the same as in ConvNets 1 and 2. Table 2 gives the results of comparing RawConvNet with ConvNet 1, ConvNet 2, and ConvNet 3. All four networks were trained for 100 epochs. The results indicate that the validation accuracies of ConvNets 1, 2, and 3 are slightly lower than the validation accuracy of the custom ConvNet. However, ConvNets 1, 2, and 3 have substantially higher validation losses. The results indicate that increasing the number of layers and adding more depth may not necessarily result in higher classification performance.

Table 2

Comparing RawConvNet with ConvNets 1, 2, and 3 after 100 Epochs

Model Name	Training Loss	Training Acc	Validation Loss	Validation Acc	Runtime per Epoch
RawConvNet ( $n = 80$ )	0,00619	99,04%	0,00702	99,13%	690 secs
ConvNet 1	0,02759	99,24%	0,03046	98,87%	460 secs
ConvNet 2	0,01369	99,74%	0,01429	99,60%	462 secs
ConvNet 3	0,00827	99,91%	0,00679	99,71%	465 secs

RawConvNet was persisted in a Python pickle file on the Linux computer where it was trained and then saved on the sdcard of a raspberry pi 3 model B v1.2. The raspberry pi was powered with a fully charged Anker Astro E7 26800mAh portable battery.

Two experiments were then performed on the raspberry pi to estimate how feasible it is to do in situ audio classification with the ConvNet on the raspberry pi. Two hundred thirty second raw audio samples from the audio data set were placed in a local folder on the pi computer. In the first experiment, a script was implemented in Python 2.7 to run every 15 minutes. The script would load RawConvNet from the pickle file, load

an audio sample from the local folder, split it into non-overlapping two second segments, and classify each two second segment with RawConvNet. In the first experiment, the fully charged battery supported audio classification for 40 hours during which 162 30-second samples were processed. It took the algorithm, on average, 13.66 seconds to process one 30-second audio sample.

In the second experiment, the script was modified to process four 30-second audio files once every 60 minutes to estimate whether a batch approach to in situ audio classification would result in better power efficiency, because the persisted ConvNet would be loaded into memory only once per every four thirty second audio samples. In the second experiment, the fully charged Anker battery supported audio classification for 43 hours during which 172 thirty second audio samples were processed. It took the algorithm 37.68 seconds, on average, to classify a batch of four 30-second audio sample, which results in 9.42 seconds per a 30-second audio sample.

### 3. Conclusion

Our experiments indicate that that it is possible to build ConvNets that classify raw audio samples as well as ConvNets that classify audio samples by converting them into spectrograms. On our data set of 9110 audio samples, RawConvNet, our ConvNet trained to classify raw audio samples, achieved a validation accuracy of 99.93%, which is slightly higher than the validation accuracy of 99.13% achieved by SpectConvNet, our ConvNet trained to classify spectrograms. While it may be possible to experiment with different ConvNet architectures to improve the validation accuracy of SpectConvNet, the validation accuracy of 99.93% achieved by RawConvNet does not leave much room for improvement.

We investigated whether making ConvNets deeper by adding more layers results in a higher classification performance than a single custom layer. Our experiments suggest that adding more layers to ConvNets does not necessarily improve classification performance. Thus, deeper is not necessarily better. Specifically, our experimental results show that, on our data set, RawConvNet, a shallower ConvNet with a single custom layer, achieved a higher validation accuracy (99.93%) than each of the validation accuracies of the three deeper ConvNets without a custom layer (i.e., 97.59%, 98.21%, 98.02%). Our experiments show that RawConvNet also achieved a validation loss (0.00432), which is smaller than each of the validation losses of three ConvNets without a custom layer (i.e., 0.57427, 0.59022, 0.59429). This empirical evidence suggests that ConvNets trained to classify raw audio samples perform slightly better than ConvNets trained to classify spectrogram images of audio samples.

The deeper ConvNets required more time and consequently more power for training. Thus, RawConvNet took, on average, 462 seconds per epoch for training whereas ConvNets 1, 2, and 3 took, on average, 545 seconds, 532 seconds, and 610 seconds, respectively.

### REFERENCES

1. Bromenschen, J.J., Henderson, C.B., Seccomb, R.A., Rice, S.D. & Etter, R.T. (2007) Honey bee acoustic recording and analysis system for monitoring hive health. *U.S. Patent 2007/0224914 A1*. Sep. 27.
2. Ferrari, S., Silva, M., Guarino, M. & Berckmans, D. (2008) Monitoring of swarming sounds in bee hives for early detection of the swarming period. *Computers and Electronics in Agriculture*. 64(1). pp. 72–77. DOI: 10.1016/j.compag.2008.05.010
3. Atauri, M.D. & Llorente, M.J. (2009) Platform for beehives monitoring based on sound analysis: A perpetual warehouse for swarm's daily activity. *Spanish Journal of Agricultural Research*. 7(4). pp. 824–828. DOI: 10.5424/sjar/2009074-1109
4. Kulyukin, V.A., Putnam, M. & Reka, S.K. (2016) Digitizing buzzing signals into A440 piano note sequences and estimating forage traffic levels from images in solar-powered, electronic beehive monitoring. *Lecture Notes in Engineering and Computer Science: Proc. of the International MultiConference of Engineers and Computer Scientists*. 1. pp. 82–87.
5. Kulyukin, V.A. & Reka, S.K. (2016) Toward sustainable electronic beehive monitoring: algorithms for omnidirectional bee counting from images and harmonic analysis of buzzing signals. *Engineering Letters*. 24(3). pp. 317–327.
6. Ramsey, M., Bencsik, M. & Newton, M.I. (2017) Long-term trends in the honeybee 'whooping signal' revealed by automated detection. *PLoS One*. 12(2). DOI: 10.1371/journal.pone.0171162

7. Mukherjee, S. & Kulyukin, V.A. (2017) *Python source code for audio classification experiments with convolutional neural networks*. [Online] Available from: [https://github.com/sarba-jit/EBM\\_Audio\\_Classification](https://github.com/sarba-jit/EBM_Audio_Classification).
8. Kulyukin, V.A., Mukherjee, S. & Amlathe, P. (2017) *BUZZ1: A database of audio samples from live beehives captured with BeePi monitors*. [Online] Available from: <https://usu.app.box.com/v/BeePiAudioData>.
9. Bengio, Y., Courville, A. & Vincent, P. (2013) Representation learning: a review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 35(8). pp. 1798–1828. DOI: 10.1109/TPAMI.2013.50
10. Mitchell, T.M. (1997). *Machine learning*. McGraw-Hill.
11. Krizhevsky, A., Sutskever, I. & Hinton, G.E. (2012) ImageNet classification with deep convolutional neural networks. *Proc. of the 25th International Conference on Neural Information Processing Systems*. 1. pp. 1097–1105.
12. Farabet, C., Couprie, C., Najman, L. & LeCun, Y. (2013) Learning hierarchical features for scene labeling. *IEEE Trans Pattern Anal Mach Intell*. 35(8). pp. 1915–1929. DOI: 10.1109/TPAMI.2012.231
13. Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A.R., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. & Kingsbury, B. (2012) Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. *Signal Processing Magazine*. 29(6). pp. 82–97. DOI: 10.1109/MSP.2012.2205597
14. Leung, M.K.K., Xiong, H.Y., Lee, L.J. & Frey, B.J. (2014). Deep learning of the tissue-regulated splicing code. *Bioinformatics*. 30(12). pp. 121–129. DOI: 10.1093/bioinformatics/btu277
15. Nair, V. & Hinton, G.E. (2010) Rectified linear units improve restricted boltzmann machines. *Proc. of the 27th International Conference on Machine Learning*. pp. 807–814.
16. Sutton, R.S. & Barto, A.G. (1998). *Introduction to Reinforcement Learning*. MIT Press.
17. Ioffe, S. & Szegedy, C. (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. *JMLR Workshop and Conference*. 37. pp. 448–456.
18. Kingma, D. & Ba, J. (2015) Adam: A Method for Stochastic Optimization. *Proc. of the 3rd International Conference on Learning Representations (ICLR)*.

Received: May 2, 2018

Kulyukin V.A., Mukherjee S., Burkatovskaya Yu.B. CLASSIFICATION OF AUDIO SAMPLES BY CONVOLUTIONAL NETWORKS IN AUDIO BEEHIVE MONITORING. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitel'naya tekhnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 45. pp. 68–75

DOI: 10.17223/19988605/45/8

Кулюкин В.А., Мукерджи С., Буркатовская Ю.Б. КЛАССИФИКАЦИЯ АУДИОФАЙЛОВ С ПОМОЩЬЮ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ В ЭЛЕКТРОННОМ АУДИОМОНИТОРИНГЕ УЛЬЕВ. *Вестник Томского государственного университета. Управление, вычислительная техника и информатика*. 2018. № 45. С. 68–75

В работе решается задача классификации аудиофайлов, полученных в результате аудиомониторинга пчелиных ульев. Аудиомониторинг улья является ключевым компонентом электронного мониторинга ульев, с помощью которого потенциально можно автоматизировать идентификацию различных стрессовых факторов для пчелиных семей. Предлагается использовать сверточные нейронные сети, которые сравниваются по качеству классификации образцов звука от систем электронного мониторинга ульев, развернутых в живых ульях. В результате файлы относятся к одной из трех непересекающихся категорий: жужжание пчел (В), стрекот сверчков (С) и шум (N). Показано, что сети, обученные для классификации необработанных аудиофайлов, лучше, чем сети, обученные для классификации спектрограмм. Продемонстрирована успешная работа сетей in situ на низковольтных устройствах, таких как компьютер raspberry pi.

Ключевые слова: глубокое обучение; сверточные нейронные сети; классификация аудио; обработка аудио; электронный мониторинг пчелиных ульев.

*KULYUKIN Vladimir Alekseevich* (Candidate of Technical Sciences, Associate Professor of the Department of computer science Utah State University, Logan, USA).

E-mail: [vladimir.kulyukin@usu.edu](mailto:vladimir.kulyukin@usu.edu)

*MUKHERJEE Sarbajit* (PhD Student of the Department of computer science Utah State University, Logan, USA).

E-mail: [sarbajit.mukherjee@aggiemail.usu.edu](mailto:sarbajit.mukherjee@aggiemail.usu.edu)

*BURKATOVSKAYA Yulia Borisovna* (Candidate of Physical and Mathematical Sciences, Associate Professor, National Research Tomsk Polytechnic University, Russian Federation).

E-mail: [tracey@tpu.ru](mailto:tracey@tpu.ru)

УДК 004.457

DOI: 10.17223/19988605/45/9

**М.С. Сорокин, Г.Б. Захарова****ИНТЕЛЛЕКТУАЛЬНЫЙ ПРОГРАММНЫЙ ИНСТРУМЕНТАРИЙ  
ДЛЯ ПОДГОТОВКИ НОВОСТНЫХ ТЕЛЕВИЗИОННЫХ ПРОГРАММ**

На основе структуризации бизнес-процессов телерадиокомпании ГТРК «Урал» разработаны подход и средства автоматизации, которые позволили повысить эффективность производства новостных телевизионных программ почти на 70%. Решена проблема необходимости повторного ввода данных за счет конструктора интерфейсов, автоматизирован процесс наложения графики на видео, создан обработчик роликов для приведения звука и видео к единым стандартам. Для архивации видео и эффективного поиска в архиве разработана система, которая выполняет автоматическое распознавание видеоизображений с помощью нейронных сетей на основе машинного обучения.

**Ключевые слова:** TV-технологии; автоматизация бизнес-процессов; конструктор интерфейсов; наложение графики на видео; нейронные сети; распознавание видеоизображений.

Подготовка программ в современной телевизионной компании отличается большой сложностью, трудоемкостью, динамичностью и слабой степенью автоматизации. При разработке медийной продукции задействуется большое количество сотрудников – режиссеры, редакторы, корреспонденты, художники-графики, монтажеры. Все они взаимодействуют в сложной информационной среде, многие действия не формализованы, человеческий фактор влияет на время и качество производства. Только структуризация и автоматизация бизнес-процессов может повысить эффективность работы и улучшить общее качество выпускаемого контента.

Универсальных решений по автоматизации работ на телевидении, в частности для производства новостных программ, на сегодняшний день не существует. Более того, процессы являются достаточно закрытыми, узконаправленными, требуют больших трудозатрат. В качестве аналогов автоматизации можно назвать систему производства новостей NewsNet [1], в которой реализованы средства работы с лентами в едином пространстве для обмена информацией и получения доступа к медиа архиву телеканала. Открытая универсальная система Cinegy News [2] для автоматизированного производства и выпуска новостей содержит интегрированные между собой модули записи входящих новостных потоков и поступающих видеосюжетов, верстки сюжетов и выпусков новостей. Высокоавтоматизированным является также новостное производство в среде Media Backbone Hive от Sony [3]. Однако интеллектуальные функции в названных ресурсах не реализованы.

В данной работе на примере одной из ведущих телевизионных компаний Свердловской области – ГТРК «Урал» – показано, как многие стратегически важные бизнес-процессы по подготовке новостных программ были сокращены по времени в десятки раз с улучшением качества выполнения задач. Работа выполнялась на основе тщательного анализа всех выполняемых работ по производству медиа продукции по методике [4]. В результате была построена подробная схема бизнес-процессов аналогично тому, как это сделано в работе [5]. Все процессы были классифицированы на допускающие полную или частичную автоматизацию, а также выделены неавтоматизируемые функции. Схема не приводится в силу большого объема, были выделены, реализованы и описаны далее автоматизированные процессы.

Выполненная автоматизация бизнес-процессов в TV-компании построена на принципах модульности, интерактивности, масштабируемости, что существенно сократило число ошибок при подготовке новостных программ и значительно повысило эффективность рабочего процесса.

## 1. Конструктор интерфейсов для наложения текстов на видеоролики

Компания ГТРК «Урал» производит новостные телевизионные программы для двух телеканалов: «Россия 1» – 12 выпусков в день и «Россия 24» – 2 выпуска в день, базовая графика на каналах имеет различный формат. Для сюжетов приходится готовить множество информационных роликов, ролики для данных телеканалов также имеют ряд существенных отличий, для их создания используются разные проекты и, соответственно, разные шаблоны. Имеется порядка 10 видов информационных роликов, к примеру тексты с фото и без них, трансляция звука с фото и без фото, анонсы, промо и пр. Таким образом, объем и разнообразие производимой ежедневно медийной продукции огромны, на ее производство уходит в среднем до пяти человеко-часов в день, многие операции носят рутинный характер и делаются вручную.

Для автоматизации данного вида работ, в частности ввода текста в многообразные шаблоны роликов, был разработан конструктор интерфейсов, позволивший унифицировать действия пользователя. Инструмент позволяет для ввода данных создавать интерфейсы из типовых групп и панелей, содержащих все элементы графического пользовательского интерфейса. Это чекбокс для управления параметром с двумя состояниями, радиокнопка для выбора одной опции из predetermined набора, выпадающий список для выбора одного из нескольких заранее определенных значений параметра, надпись (static text), поле ввода (dynamic text) для ввода небольшого объема текста без переноса строк, вкладка (tabs), которая позволяет в окне приложения переключаться между несколькими формами, кнопка, «нажатие» на которую приводит к некоторому действию из заготовленных шаблонов или выполнению пользовательского скрипта.

При использовании данных интерфейсов пользователь получает возможность многократно изменять единообразные удобные формы для ввода данных в проект (ранее ввод не был унифицирован), а также импортировать и экспортировать их через XML.

Технология реализации данного инструмента опирается на возможности программы Adobe After Effects, предназначенной для редактирования видео и динамических изображений, применения цифровых видеоэффектов и пр. Данная программа является де факто промышленным стандартом для производства медийной продукции на телевидении.

Конструктор реализован в среде Adobe ExtendScript Toolkit CC, которая обеспечивает доступ к многофункциональным средствам разработки и выполнения скриптов JavaScript для работы напрямую с приложениями Adobe, поддерживающими сценарии. Сценарий – это последовательность команд, предписывающих приложению выполнить последовательность операций. Сценарии можно использовать для автоматического выполнения повторяющихся задач, сложных вычислений, а также для выполнения некоторых функций, непосредственно не представленных в графическом пользовательском интерфейсе. Сценарий, который использует Adobe, называется ExtendScript (JSX), он является диалектом JavaScript или, точнее, EcmaScript.

Конструктор представляет собой JavaScript файл, который вызывается напрямую из After Effects. Для этого необходимо открыть проект, затем меню File, Scripts, Form\_constructor.jsx. Конструктор работает в два этапа: сначала пользователь выбирает поля ввода и элементы управления, а далее производится связка формы и проекта After Effects через управляющий слой. Из конструктора формы сохраняются в файлы JSX для дальнейшего использования. Вызвать форму можно из программы After Effects через меню File. Формы поддерживают возможность выгрузки и загрузки данных в файл XML, что позволяет использовать одни данные без повторного ввода в нескольких проектах Adobe After Effects, значительно повышая производительность и понижая риск возникновения ошибки. Производительность работ составила порядка трех человеко-часов вместо пяти.

## 2. Модуль подключения текстовых и графических данных к видеоформату

Существует ряд задач, где автоматизация ввода данных в текстовые поля недостаточна. Необходимо обеспечить сложную связь между полями в форме, содержащей данные, с возможностью

регулировать продолжительность отображения наложенных на видео графики и текста, а также задавать ключевые кадры в ролике. С этой целью разработан инструмент, достаточно сложный в настройке и требующий в использовании понимания базовых принципов программирования и логики.

Одна из решаемых таким образом задач – подготовка роликов метеопрогноза. Каждый выпуск новостей содержит метеопрогноз, над созданием которого работают художники компьютерной графики. На создание качественного шаблонного проекта погоды уходит до трех месяцев, сюда входят съемка, отбор, обработка видеоматериала, разработка компьютерной графики и анимации. Прогноз погоды представляет собой текстовые и графические элементы, наложенные поверх фонового видеоролика. Каждый день создается четыре вида роликов: утренняя погода, погода сегодня, погода завтра и студийная погода.

Несмотря на кажущуюся простоту, эта повседневная работа требует большой концентрации и трудозатрат. Соответствующие бизнес-процессы с некоторыми отличиями в каждом из четырех видов прогноза содержали следующие действия: получение документа от гидрометцентра в формате Word, корректировка и приведение этого документа к единому виду, открытие проекта Adobe After Effects с шаблоном погоды, ввод данных из Word в Adobe After Effects, отправка на рендер, рендеринг и перекодирование полученного видеоролика с использованием кодека QuickTimeb в файловой транскодинг-системе Rhozet ProMedia Carbon, далее отправка в сетевое хранилище. На подготовку прогноза ежедневно уходило порядка четырех часов.

Для повышения эффективности работы был реализован модуль подготовки видеороликов метеопрогноза GeneratorXML на основе единого файла данных о погоде. Файл в формате XML, удобный для машинного чтения, содержит все необходимые данные для подготовки информационных заставок погоды в течение дня. Через эргономичный интерфейс данные вводятся однократно, без дублирования, с возможностью редактирования имеющихся, в том числе архивных, данных. Цепочка бизнес-процессов свелась при этом до двух действий: получение документа Word от гидрометцентра и ввод этих данных в разработанный модуль подготовки погоды. Затрачиваемое время – 20 минут.

Далее для изготовления ролика необходимо создать множество ключевых кадров, которые отражают любое изменение состояния объектов на видеоизображении. Для подключения анимации без необходимости создания множества ключевых кадров вручную был использован такой инструмент в Adobe After Effects, как выражения. Язык выражений основан на стандартном языке JavaScript. Выражение используется для оценки одиночного свойства слоя в конкретный момент времени и сообщает приложению о характеристике свойства. С помощью выражений можно создавать связи между свойствами слоя и использовать ключевые кадры конкретного свойства для создания динамической анимации других слоев.

Для автоматизации данного вида графики был добавлен шаблон формы на три дня, на неделю, формирование суточного прогноза по городу, реализована возможность связки полей друг с другом, возможность загружать данные из Интернета.

В результате автоматизации трудозатраты по подготовке видео с наложением данных от гидрометцентра сократились почти в четыре раза и составили немногим более часа.

### **3. Автоматизация обработки подготовленных видеороликов под оборудование**

По стандартам телекомпании видеофайлы должны быть строго определенного формата с конкретными параметрами и соответствовать спецификации оборудования. Поскольку видео поступает в различных форматах, был реализован обработчик готовых роликов для перекодировки видеофайлов, приведения звука и видео к общепринятым стандартам.

Обработчик видео и звука представляет из себя конвертор на базе языка Python и библиотеки FFmpeg для работы с медиафайлами. Конвертор принимает на вход файлы в любом поддерживаемом формате и с любыми параметрами через заранее заданную папку. Эта папка становится специальной смотрящей (Watch Folder) папкой; любой файл, который оказывается в ней, попадает в обработку.

При попадании на конвертор видеофайл сначала сжимается до заданного разрешения, а также приводит форму пикселя к стандарту, в случае с компанией ГТРК «Урал» – PAL DV (720 × 576, Aspect ratio – 1,07). Далее идет работа с чересстрочностью, устанавливается нижний полукадр. Затем обрабатывается звук: устанавливается частота дискретизации Sampling rate 48,0 kHz, общий уровень производимого шума приводится к 23 LUFS (Loudness Unit relative to Full Scale – единица громкости относительно полной шкалы) в соответствии со стандартом R-128 Европейского вещательного союза (EBU).

На выходе конвертор возвращает файлы, соответствующие требованиям, с пометкой о соответствии. Настройка параметров обработки производится через простой графический интерфейс.

Время обработки файлов с применением конвертора сократилось почти вдвое: 17 минут вместо получаса.

#### 4. Автоматическое формирование архива и распознавание изображений в видеофайлах

ТВ-компании в процессе деятельности создают огромное количество видео контента от видеосъемок до компьютерной графики, во многих случаях эти проекты подразумевают повторное использование. Организуются архивы с соответствующим штатом сотрудников, которые каталогизируют материалы и по запросам предоставляют доступ. В целях автоматизации данного вида деятельности разработана система, позволяющая структурировать и архивировать видеоматериал и выполнять его эффективный поиск на основе распознавания изображений в видеофайлах. Процесс распознавания выполняется посредством сверточной нейронной сети, в данном случае применялась открытая библиотека Keras [6], настроенная на работу с сетями глубокого обучения. В отличие от программы, описанной в работе [7], такой подход позволил распознавать более сложные сцены, но в рамках заданных классов. Реализация выполнена на языке Python [8].

В данной системе автоматизации поиска видео можно выделить три части:

- 1) поддержка видеоархива: фоновое приложение, которое при внесении в первичный архив нового видео выполняет его обработку по выделению ключевых кадров и оптимизацию;
- 2) распознавание изображений на видео с помощью нейронной сети;
- 3) собственно поиск, определяющий по запросу пользователя искомый объект на видео и возвращающий пути к найденным релевантным видео в архиве.

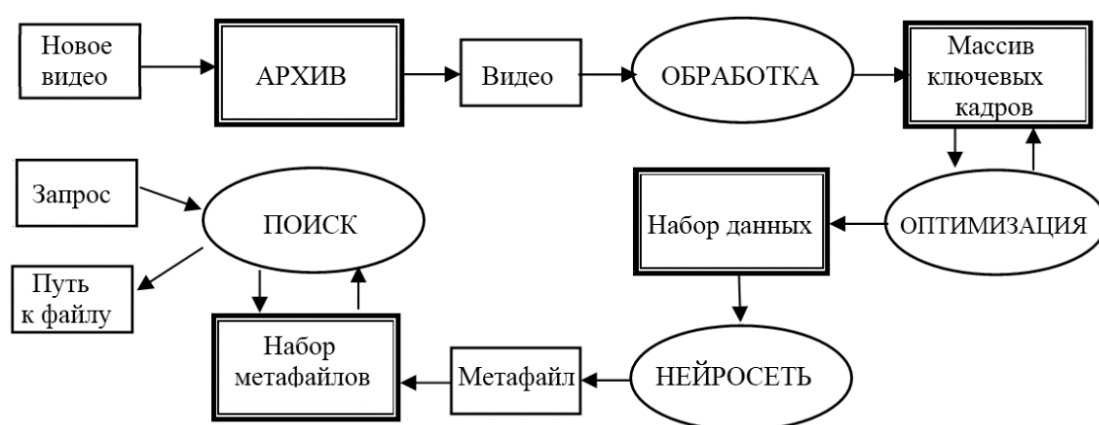


Рис. 1. Структура системы поддержки видеоархива и поиска видео

##### 4.1. Поддержка видеоархива

Фоновое приложение получает на вход папку отслеживания. На фазе инициализации программа проверяет соответствие каждого видеофайла в папке с уже имеющимися метаданными к этим файлам.

При выявлении видеофайлов без метаданных, а также при появлении новых файлов, в папке создается поток для построения соответствующих данных. В потоке формируется дочерний процесс, который позволяет получить из видеофайла массив ключевых кадров. Набор ключевых кадров хранится во временной папке. Есть несколько способов отделения видео от шума, выделения релевантной информации из кадров: 1) взять все кадры – кадр за кадром; 2) брать каждый  $n$ -й кадр; 3) отсеивать пустые, повторяющиеся кадры, оставляя наиболее информативные. В данной системе используется третий подход. Массив изображений сокращается за счет удаления схожих между собой и прочих не содержащих полезной информации кадров.

#### 4.2. Распознавание изображений на видео

На основе массива изображений готовится набор данных (batch) для анализа нейронной сетью. Изображения классифицируются тремя моделями: 1) выделение объекта на изображении; 2) определение места внутри или вне помещения; 3) распознавание времени года. Модель (model), основная структура данных Keras, – это способ организации слоев. Простейшим типом модели является последовательная модель (sequential model), линейная совокупность слоев, которая и используется в данном случае. Когда сеть содержит несколько скрытых слоев, она приобретает способность самообучаться и выделять ключевые признаки, которые наилучшим образом описывают входные данные. Отличительное свойство Keras, которое выделяет его из других фреймворков (например, TensorFlow), – это автоматический расчет размеров слоев: достаточно указать только размерность входного слоя, а Keras автоматически проинициализирует все остальные слои.

Для обучения каждой модели нейронной сети использовалось по три набора изображений: набор данных для обучения (порядка 20 000 изображений); набор данных для проверки (оценка качества обучения сети в процессе обучения – 10 000 изображений); набор данных для тестирования (оценка качества после завершения обучения – 5 000 изображений). Для загрузки изображений в Keras используются генераторы изображений (ImageDataGenerator), которые позволяют автоматически загружать изображения с диска в память компьютера, преобразовывать их в вид, необходимый Keras, и передавать в модель.

Первая модель сети отвечает за определение класса, к которому относится изображение по виду объектов на нем: человек, автомобиль, компьютер и т.д. (Label Detection). Keras позволяет не только обучать глубокие нейронные сети, но и содержит несколько готовых, предварительно обученных сетей, которые можно использовать, не тратя время и ресурсы на обучение, так называемый перенос обучения (transfer learning). Большая часть сетей предназначена для распознавания объектов на изображениях из краудсорсинговой базы данных ImageNet [9], содержащей более 14 млн изображений, привязанных к семантическому тезаурусу английских существительных WordNet. Для распознавания применялась VGG16 – сверточная сеть из 16 слоев. Сеть состоит из двух частей: первая часть выделяет характерные признаки в изображении, а вторая отвечает за классификацию объекта по этим признакам. На выходе сеть выдает вероятности того (в формате one hot encoding), что на изображении тот или иной класс объекта.

Вторая модель сети представляет собой бинарный классификатор, который определяет место съемки внутри или снаружи помещения (Indoor / Outdoor) и содержит 10 слоев (8 для выделения признаков и 2 для классификации). Принимает на вход изображение размером  $224 \times 224$  пиксела с тремя каналами RGB. Использует оптимизатор Adam (adaptive moment estimation), сочетает в себе идею накопления движения и идею более слабого обновления весов для типичных признаков.

Третья модель сети различает время года съемки (Season). Модель состоит из 13 слоев выделения признаков сети VGG16 плюс 3 слоя классификации. Характеристики входного изображения как в предыдущем случае.

Данные по кадрам усредняются. Выявляются доминирующие и наиболее вероятные образы. Из полученных сведений формируется метафайл, который содержит: хэш файла, общую продолжительность видеофайла, количество релевантных кадров, снаружи / внутри помещения, время года, 10–200 наиболее вероятных объектов.



### 4.3. Поиск

Приложение поиска предоставляет интерфейс с возможностью фильтрации по ключевым характеристикам, полученным в результате работы второй и третьей моделей (зима / лето, улица / помещение), и поиск по меткам в метафайле. Программа объединяет сведения о файлах в заданной папке, после чего производит поиск по этим данным. Далее выводит превью соответствующих запросу видеофайлов в порядке релевантности с описанием других имеющихся образов в этих видеофайлах.

Данная система позволила существенно сократить время поиска в видеоархиве и оперативно получать доступ к видеозаписям независимо от количества файлов в архиве.

### Заключение

Представленная интеллектуальная система автоматизации подготовки новостных телевизионных программ внесла множественные изменения в бизнес-процессы TV-компании ГТРК «Урал». Наряду с описанными четырьмя инструментами был разработан ряд других программ: для подготовки промо-роликов, изготовления инфографики и др. Программный код составил порядка 20 тыс. строк, были разработаны словари баз данных суммарным объемом более 16 тыс. строк.

После внедрения системы были проведены измерения времени, затраченного на производство медиапродукции до и после автоматизации по всем видам производства. Общая экономия времени в день составила 320 минут: вместо 557 минут стало уходить 236, т.е. 68% экономии. Улучшилось качество материалов, уменьшилось количество ошибок за счет удобных интерфейсов ввода данных и отсутствия дублирующих процессов.

### ЛИТЕРАТУРА

1. Система производства новостей News Net. 2016. 19 июля // Advview : сайт о телевидении, автоматизации, оборудовании, маркетинге и рекламе. URL: [https://advview.ru/cat\\_automation/nrcs/sistema-proizvodstva-novostej-newsnet/](https://advview.ru/cat_automation/nrcs/sistema-proizvodstva-novostej-newsnet/) (дата обращения: 14.08.2018).
2. Cinegy News TV Pack: как автоматизировать выпуск новостей на телеканале. 2017. 22 авг. // Advview : сайт о телевидении, автоматизации, оборудовании, маркетинге и рекламе. URL: [https://advview.ru/cat\\_automation/nrcs/cinegy-news-i-tv-pack-kak-avtomatizirovat-vypusk-novostej-na-telekanale/](https://advview.ru/cat_automation/nrcs/cinegy-news-i-tv-pack-kak-avtomatizirovat-vypusk-novostej-na-telekanale/) (дата обращения: 14.08.2018).
3. Новостное производство с Media Backbone Hive от Sony. 2016. 10 окт. // Advview : сайт о телевидении, автоматизации, оборудовании, маркетинге и рекламе. URL: [https://advview.ru/cat\\_automation/nrcs/novostnoe-proizvodstvo-s-media-backbone-hive-ot-sony/](https://advview.ru/cat_automation/nrcs/novostnoe-proizvodstvo-s-media-backbone-hive-ot-sony/) (дата обращения: 14.08.2018).
4. Волков О.Н. Стандарты и методологии моделирования бизнес-процессов // Связьинвест : корпоративное издание. 2005. № 7. С. 20–24.
5. Быков В.В. Информационные технологии и автоматизация ТВ-производства // T-Comm: телекоммуникации и транспорт. 2012. Т. 6, № 9. С. 46–49.
6. Keras code and weights files for popular deep learning models. URL: <https://github.com/fchollet/deep-learning-models> (accessed: 12.08.2018).
7. Голубинский А.Н., Толстых А.А. Распознавание объектов на телевизионных изображениях с использованием аппарата сверточных нейронных сетей // Вестник Воронежского института МВД России. 2017. № 1. С. 71–81.
8. Chollet F. Deep Learning with Python. Manning Publications? 2017. 384 p.
9. ImageNet. URL: <http://image-net.org/about-overview> (accessed: 12.08.2018).

Поступила в редакцию 22 июня 2018 г.

Sorokin M.S., Zakharova G.B. (2018) THE INTELLIGENT SOFTWARE TOOLKIT FOR PRODUCTION OF NEWS TV-PROGRAMS. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitel'naya tekhnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 45. pp.76–82

DOI: 10.17223/19988605/45/9

Based on the structuring of business processes of the TV and Radio Company "GTRK Ural", automation tools have been developed that have significantly increased the efficiency of the production of news programs.

The designer of interfaces was developed and introduced, due to what the problem of necessity of repeated data input is solved. The constructor is a JavaScript file that is called directly from Adobe After Effects and works in two stages: the user constructs input

fields and interface controls, then the Adobe After Effects form and project are linked through the control layer, saving to JSX files and further use.

The management tools of the already prepared media products have been implemented. So, for the development and filling of computer graphics templates in the preparation of weather forecast videos, automation technology was used based on the expressions and scenarios of the Adobe After Effects program and the toolkit for overlaying graphics on the video image was created. The weather forecast generator module generates a common weather data file in XML format based on hydrometeorological information. The file, convenient for machine reading, contains all the necessary data for the preparation of weather information posters during the day. Through the ergonomic interface, data are entered once, without duplication, with the ability to edit existing, including archived, data. At the same time, the chain of business processes instead of six actions before automation was reduced to two actions: obtaining a Word document from the hydrometeorological center and entering this data into the developed module, which takes about 20 minutes a day instead of the previous 4 hours.

The converter for ready-made clips has been created to bring sound and video to unified standards, working through a special interface written in Python.

With a view to efficiently structuring and archiving video material and subsequent search in the archive, it was developed the system that allows to automate processes of recognizing the contents of video files using neural networks based on machine learning. The recognition process is performed by means of a convolutional neural network, an open Keras library was used, tuned to work with deep learning networks. Images are classified according to three models: Label Detection, a VGG16 convolutional network of 16 layers was used for recognition; the definition of a place inside or outside the room (Indoor / Outdoor); Season recognition, the model consists of 13 layers of VGG16 network feature highlighting plus 3 layers for classification.

After the introduction of the system in the working process, the time spent on the production of media products before and after automation was measured for all types of production. The total time savings per day was 320 minutes: instead of 557 minutes 236 were spent, i.e., 68% time savings. The quality of materials was improved, the number of errors was reduced due to convenient data entry interfaces and the absence of duplicate processes.

Keywords: TV-technology; overlay graphics on video; neural networks; recognition of video images.

*SOROKIN Mikhail Sergeevich* (Ural Federal University, Ekaterinburg, Russian Federation).

E-mail: sjfh@yandex.ru

*ZAKHAROVA Galina Borisovna* (Candidate of Technical Sciences, Associate Professor, Ural State University of Architecture and Art, Ural Federal University, Ekaterinburg, Russian Federation).

E-mail: zgb555@gmail.com

## REFERENCES

1. Advview.ru. (2016a) *News Net news production system*. 19th July. [Online] Available from: [https://advview.ru/cat\\_automation/nrcs/sistema-proizvodstva-novostej-newsnet/](https://advview.ru/cat_automation/nrcs/sistema-proizvodstva-novostej-newsnet/). (Accessed: 14th August 2018). (In Russian).
2. Advview.ru. (2017) *Cinegy News TV Pack: kak avtomatizirovat' vypusk novostey na telekanale* [Cinegy News TV Pack: how to automate the news release on the channel]. 22nd August. [Online] Available from: [https://advview.ru/cat\\_automation/nrcs/cinegy-news-i-tv-pack-kak-avtomatizirovat-vypusk-novostej-na-telekanale/](https://advview.ru/cat_automation/nrcs/cinegy-news-i-tv-pack-kak-avtomatizirovat-vypusk-novostej-na-telekanale/). (Accessed: 14th August 2018).
3. Advview.ru. (2016) *Novostnoye proizvodstvo s Media Backbone Hive ot Sony* [News production with Media Backbone Hive from Sony]. 10th October. [Online] Available from: [https://advview.ru/cat\\_automation/nrcs/novostnoe-proizvodstvo-s-media-backbone-hive-ot-sony/](https://advview.ru/cat_automation/nrcs/novostnoe-proizvodstvo-s-media-backbone-hive-ot-sony/). (Accessed: 14th August 2018).
4. Volkov, O.N. (2005) Standarty i metodologii modelirovaniya biznes-protsessov [Standards and methodologies for modeling business processes]. *Svyaz'invest*. 7. pp. 20–24.
5. Bykov, V.V. (2012) Information technologies and automation of TV production. *T-Comm: telekommunikatsii i transport – T-Comm: Telecommunications and Transport*. 6(9). pp. 46–49. (In Russian).
6. GitHub. (n.d.) *Keras code and weights files for popular deep learning models*. [Online] Available from: <https://github.com/fchollet/deep-learning-models>. (Accessed: 12th August 2018).
7. Golubinsky, A.N. & Tolstykh, A.A. (2017) Recognition of objects on television images using the apparatus of convolutional neural networks. *Vestnik Voronezhskogo instituta MVD Rossii – Bulletin of the Voronezh Institute of the Ministry of Internal Affairs of Russia*. 1. pp. 71–81. (In Russian).
8. Chollet, F. (2017) *Deep Learning with Python*. Manning Publications.
9. *ImageNet*. (n.d.) [Online] Available from: <http://image-net.org/about-overview> (Accessed: 14th August 2018). (In Russian).

## ДИСКРЕТНЫЕ ФУНКЦИИ И АВТОМАТЫ

УДК 519.7

DOI: 10.17223/19988605/45/10

Ю.В. Поттосин

### ПАРАЛЛЕЛЬНАЯ ДЕКОМПОЗИЦИЯ СИСТЕМЫ ЧАСТИЧНЫХ БУЛЕВЫХ ФУНКЦИЙ

Предлагается подход к решению задачи многоблочной параллельной декомпозиции системы частичных булевых функций. Подход предполагает интервальное представление заданной системы в виде пары троичных матриц и последующее рассмотрение графов ортогональности строк указанных матриц и сводит данную задачу к нахождению кратчайшего покрытия множества ребер одного графа полными двудольными подграфами (бикликами) другого графа. Описано два метода решения рассматриваемой задачи, использующих предлагаемый подход.

**Ключевые слова:** система частичных булевых функций; троичная матрица; полный двудольный подграф.

Под декомпозицией системы булевых функций понимается ее представление в виде суперпозиции двух или более систем функций, каждая из которых в некотором смысле проще исходной системы. Задача декомпозиции булевых функций является одной из важных и сложных задач из области логического проектирования, ее успешное решение непосредственно влияет на качество и стоимость проектируемых цифровых устройств. Декомпозиция системы булевых функций, описывающей поведение некоторого дискретного устройства, ведет к разбиению его на отдельные блоки, что облегчает дальнейшую процедуру логического синтеза. Как показано в работах [1, 2], данной задаче посвящено значительное количество статей, однако вопрос еще требует исследований [3]. В настоящей статье рассматривается задача декомпозиции системы булевых функций в следующей постановке.

Задана система частичных (не полностью определенных) булевых функций в виде векторной функции  $f(x) = (f_1(x), f_2(x), \dots, f_m(x))$ , где компонентами вектора  $x = (x_1, x_2, \dots, x_n)$  являются булевы переменные, составляющие множество  $X$ . Требуется найти суперпозицию  $f(x) \leq \Phi(g_1(z_1), g_2(z_2), \dots, g_k(z_k))$ , где  $z_1, z_2, \dots, z_k$  – векторные переменные, компонентами которых служат соответственно переменные из подмножеств  $Z_1, Z_2, \dots, Z_k$  (возможно, пересекающихся) множества  $X = \{x_1, x_2, \dots, x_n\}$ , а символ  $\leq$  обозначает отношение реализации, т. е. значения компонент  $\Phi_1, \Phi_2, \dots, \Phi_m$  векторной функции  $\Phi$  совпадают со значениями компонент функции  $f$  везде, где эти значения определены. При этом мощность  $|Z_i|$  ( $i = 1, 2, \dots, k$ ) должна быть ограничена некоторой заданной величиной  $p$ , а число  $k$  должно быть минимальным и меньшим, чем  $n$ . Указанная декомпозиция определяет структуру логической схемы, показанную на рис. 1. Такой вид декомпозиции назван *многоблочной параллельной декомпозицией* [4]. Подобная задача при  $k = 2$  решалась в статье [5].

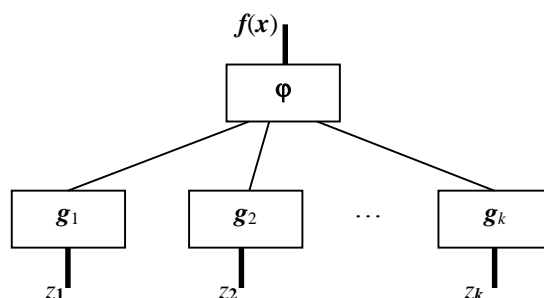


Рис. 1. Структура логической схемы

В подавляющем большинстве публикаций, рассматривающих задачу декомпозиции булевых функций, подмножества  $Z_1, Z_2, \dots, Z_k$  считаются заданными [2, 4, 6, 7]. Вопросу поиска таких подмножеств, при которых существует соответствующая декомпозиция, посвящено не так много публикаций. Среди работ, где рассматривается данный вопрос, можно назвать работы [8–13]. В данной статье предлагается подход, не требующий конкретного задания подмножеств  $Z_1, Z_2, \dots, Z_k$ . Его можно рассматривать как дальнейшее развитие подхода, представленного в работе [14].

## 1. Описание подхода

Предлагаемый подход к решению данной задачи требует интервального задания системы частичных булевых функций [3] – в виде пары троичных матриц  $X, F$  размерности  $l \times n$  и  $l \times m$  соответственно. Столбцы матрицы  $X$  соответствуют переменным  $x_1, x_2, \dots, x_n$ , а столбцы матрицы  $F$  – функциям  $f_1(x), f_2(x), \dots, f_m(x)$ . Строка матрицы  $X$  представляет интервал булева пространства, а соответствующая ей строка матрицы  $F$  – значения функций на этом интервале. Символ « $\rightarrow$ » в  $i$ -й строке и  $j$ -м столбце матрицы  $F$  означает, что  $i$ -й интервал не используется для задания функции  $f_j(x)$ . Строки матриц  $X$  и  $F$  имеют единую нумерацию.

Рассмотрим графы  $G_X = (V, E_X)$  и  $G_F = (V, E_F)$ , где множество вершин  $V$  является множеством общих номеров строк матриц  $X$  и  $F$ , а множества ребер  $E_X$  и  $E_F$  являются множествами пар номеров ортогональных строк матриц  $X$  и  $F$  соответственно. Две строки троичной матрицы ортогональны, если имеется столбец, у которого в одной из этих строк расположен ноль, а в другой – единица [3]. Система функций задана корректно с помощью матриц  $X$  и  $F$ , если  $E_F \subseteq E_X$ , т.е.  $G_F$  является остовным подграфом графа  $G_X$ .

**Замечание.** Любая пара матриц  $(X, F)$  указанного вида может рассматриваться как представление некоторой системы частичных булевых функций, если граф  $G_F$  является остовным подграфом графа  $G_X$ .

Каждому ребру из множества  $E_X$  приписано множество переменных из множества  $X = \{x_1, x_2, \dots, x_n\}$ , по которым соответствующие строки ортогональны. Полному двудольному подграфу, или *биклике*, графа  $G_X$  припишем множество переменных из  $X$ , взятых по одной из каждого ребра, принадлежащего данной биклике. Биклику назовем *допустимой*, если число приписанных ей переменных не превышает  $p$  и если она содержит хотя бы одно ребро из множества  $E_F$ .

Множество переменных, приписываемых биклике, определяется следующим образом. Пусть  $\{x_i, x_j, \dots, x_k\}$  – множество переменных, по которым ортогональны две строки, соответствующие ребру из множества  $E_X$ . Образует элементарную дизъюнкцию  $x_i \vee x_j \vee \dots \vee x_k$  из этих переменных. Получим конъюнктивную нормальную форму (КНФ), членами которой будут указанные дизъюнкции, взятые по всем ребрам, входящим в данную биклику. После удаления возможных поглощаемых элементарных дизъюнкций преобразуем полученную КНФ, раскрыв скобки, в дизъюнктивную нормальную форму (ДНФ). Множество переменных, приписанных биклике, составят переменные, входящие в элементарную конъюнкцию минимального ранга полученной ДНФ.

**Утверждение.** Для системы частичных булевых функций  $f(x)$ , заданной троичными матрицами  $X$  и  $F$ , существует реализующая ее суперпозиция  $\Phi(g_1(z_1), g_2(z_2), \dots, g_k(z_k))$ , если существует покрытие множества  $E_F$  допустимыми бикликами графа  $G_X$ , число которых  $k$ .

Пусть получено указанное покрытие бикликами  $B_1, B_2, \dots, B_k$ . Каждая биклика  $B_i$  может быть задана парой множеств вершин  $\langle V_i', V_i'' \rangle$ , поскольку каждая вершина из  $V_i'$  связана в биклике ребрами со всеми вершинами из  $V_i''$ . Каждая функция  $g_i(z_i)$  задается матрицами  $X_i$  и  $F_i$ . Матрица  $X_i$  является минором матрицы  $X$ , образованном столбцами, соответствующими переменным, приписанным биклике  $B_i$ . Матрица  $F_i$  состоит из одного столбца, где в строке с номером, соответствующим вершине из  $V_i'$ , находится 0, в строке с номером, соответствующим вершине из  $V_i''$ , находится 1 (или наоборот), а в строке, для которой нет соответствующих вершин ни в  $V_i'$ , ни в  $V_i''$ , находится символ « $\rightarrow$ ». Векторная

функция  $\Phi$  задается матрицами  $U$  и  $\Phi$ . Матрица  $U$  состоит из столбцов, представляющих матрицы  $F_1, F_2, \dots, F_k$ , а матрица  $\Phi$  совпадает с матрицей  $F$ . Действительно, согласно приведенному выше замечанию пара матриц  $(U, \Phi)$  может рассматриваться как представление системы частичных булевых функций. Нетрудно видеть, что для любого значения вектора  $x$ , произвольно взятого из области определения любой функции  $f_i$  заданной системы, значения функций  $\phi_i$  и  $f_i$  будут совпадать. Следовательно, пары матриц  $(X_1 F_1), (X_2 F_2), \dots, (X_k F_k)$  и  $(U, \Phi)$  представляют искомую суперпозицию. Это представление обладает избыточностью в виде повторяемых и поглощаемых строк в матрицах, а также знаков «—» в одностолбцовых матрицах  $F_i$ . Такую избыточность легко устранить, удалив упомянутые строки из матриц.

## 2. Точный метод

Точный метод, гарантирующий минимум числа функций в искомой суперпозиции, описан в статье [15] и заключается в выполнении процесса, состоящего из следующих этапов.

1. Нахождение всех максимальных допустимых биклик в графе  $G_X$ . Для этого можно использовать метод, представленный в работе [16].

2. Получение кратчайшего покрытия множества  $E_F$  найденными бикликами. Если число биклик, составляющих покрытие, не меньше  $n$ , то для заданной системы функций не существует нетривиальной декомпозиции указанного вида.

3. Определение булевых функций  $g_1(z_1), g_2(z_2), \dots, g_k(z_k)$  и векторной функции  $\Phi$ .

На этапе получения покрытия можно продолжить оптимизацию решения, уменьшая сумму чисел компонент векторов  $z_1, z_2, \dots, z_k$ . Тогда каждую биклику надо снабдить весом в виде числа приписанных ей переменных и решать задачу о взвешенном покрытии. При доопределении не полностью определенных булевых функций в процессе декомпозиции некоторые переменные могут оказаться несущественными аргументами. Тогда можно выбирать вариант с наименьшим числом существенных аргументов.

**Пример 1.** Пусть система частичных булевых функций  $f(x)$  задана следующими троичными матрицами:

$$X = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & - & - & 0 & 1 & - \\ 1 & - & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & - & 1 & - & 1 & 0 \\ 0 & 0 & - & 1 & 1 & - \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix}, \quad F = \begin{bmatrix} f_1 & f_2 & f_3 \\ 0 & 0 & 1 \\ - & 1 & 1 \\ 0 & - & 0 \\ 1 & 1 & - \\ - & 0 & 1 \\ 1 & 0 & - \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix}.$$

Требуется получить суперпозицию  $f(x) \leq \Phi(g_1(z_1), g_2(z_2), \dots, g_k(z_k))$  при минимальном  $k$  и числе  $p$  компонент каждого из векторов  $z_1, z_2, \dots, z_k$ , не превышающем 3.

Граф  $G_X = (V, E_X)$  с множеством вершин  $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$  представим в виде перечня ребер. В табл. 1 представлены эти ребра и приписанные им переменные. Граф  $G_F = (V, E_F)$  имеет то же множество вершин, а его множество ребер  $E_F$  отличается от  $E_X$  только тем, что в нем отсутствуют ребра  $v_2v_4$  и  $v_5v_6$ .

Таблица 1

Ребра графа  $G_X = (V, E_X)$  с приписанными им переменными

$v_1v_2$	$v_1v_3$	$v_1v_4$	$v_1v_6$	$v_2v_3$	$v_2v_4$	$v_2v_5$	$v_2v_6$	$v_3v_4$	$v_3v_5$	$v_3v_6$	$v_4v_5$	$v_4v_6$	$v_5v_6$
$x_1$	$x_4 x_5 x_6$	$x_1 x_2 x_3 x_4 x_6$	$x_1 x_4$	$x_1 x_4 x_5$	$x_4$	$x_1$	$x_4$	$x_1 x_3 x_5$	$x_5 x_6$	$x_1 x_5$	$x_1 x_3 x_6$	$x_2$	$x_1$

Для графа  $G_X$  получено 18 максимальных допустимых биклик. Искомое покрытие составляют следующие биклики с соответствующими КНФ и ДНФ:

$$\begin{aligned} \langle \{v_1, v_5\}, \{v_2, v_3, v_4, v_6\} \rangle - x_1(x_5 \vee x_6) &= x_1 x_5 \vee x_1 x_6, \\ \langle \{v_1, v_2, v_3, v_4, v_5\}, \{v_6\} \rangle - x_1 x_2 x_4, \\ \langle \{v_1, v_2, v_4, v_5, v_6\}, \{v_3\} \rangle - (x_5 \vee x_6)(x_1 \vee x_5) &= x_1 x_6 \vee x_5. \end{aligned}$$

Представленные ниже матрицы задают искомую суперпозицию.

$$\begin{aligned} X_1 = \begin{matrix} & x_1 & x_5 \\ \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} \end{matrix}, F_1 = \begin{matrix} & g_1 \\ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} \end{matrix}; \quad X_2 = \begin{matrix} & x_1 & x_2 & x_4 \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & - & 0 \\ 1 & - & 1 \\ 0 & 1 & 1 \\ 1 & - & - \\ 0 & 0 & 1 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} \end{matrix}, F_2 = \begin{matrix} & g_2 \\ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} \end{matrix}; \quad X_3 = \begin{matrix} & x_5 \\ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} \end{matrix}, F_3 = \begin{matrix} & g_3 \\ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} \end{matrix}; \\ \\ U = \begin{matrix} & g_1 & g_2 & g_3 \\ \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} \end{matrix}, \Phi = \begin{matrix} & \varphi_1 & \varphi_2 & \varphi_3 \\ \begin{bmatrix} 0 & 0 & 1 \\ - & 1 & 1 \\ 0 & - & 0 \\ 1 & 1 & - \\ - & 0 & 1 \\ 1 & 0 & - \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} \end{matrix}. \end{aligned}$$

После устранения избыточности получим следующие матрицы, представляющие искомые системы частичных булевых функций:

$$\begin{aligned} X_1 = \begin{matrix} & x_1 & x_5 \\ \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} \end{matrix}, F_1 = \begin{matrix} & g_1 \\ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} \end{matrix}; \quad X_2 = \begin{matrix} & x_1 & x_2 & x_4 \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & - & 0 \\ 0 & 1 & 1 \\ 1 & - & - \\ 0 & 0 & 1 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} \end{matrix}, F_2 = \begin{matrix} & g_2 \\ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} \end{matrix}; \quad X_3 = \begin{matrix} & x_5 \\ \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \begin{matrix} 1 \\ 2 \end{matrix} \end{matrix}, F_3 = \begin{matrix} & g_3 \\ \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \begin{matrix} 1 \\ 2 \end{matrix} \end{matrix}; \\ \\ U = \begin{matrix} & g_1 & g_2 & g_3 \\ \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} \end{matrix}, \Phi = \begin{matrix} & \varphi_1 & \varphi_2 & \varphi_3 \\ \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & - & 0 \\ 1 & 0 & - \end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} \end{matrix}. \end{aligned}$$

В результате совместной минимизации в классе ДНФ полученных систем булевых функций получим следующие матричные представления ДНФ:

$$X = \begin{matrix} & x_1 & x_2 & x_4 & x_5 \\ \begin{bmatrix} 1 & - & - & 1 \\ 0 & 0 & 1 & - \\ - & - & - & 1 \end{bmatrix} & \begin{matrix} g_1 & g_2 & g_3 \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{matrix} \end{matrix}; \quad U = \begin{matrix} & g_1 & g_2 & g_3 \\ \begin{bmatrix} 1 & - & 1 \\ 1 & 0 & - \\ - & - & 1 \end{bmatrix} & \begin{matrix} \varphi_1 & \varphi_2 & \varphi_3 \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{matrix} \end{matrix}.$$

Применение описанного метода не всегда приведет к решению задачи практической размерности за приемлемое время. Достижимой верхней границей числа всех максимальных биклик в графе является  $2^{n-1} - 1$ , где  $n$  – число вершин графа, и, кроме того, задача кратчайшего покрытия имеет не полиномиальную сложность. Этот метод следует считать основой для разработки эвристических методов решения данной задачи. Описание одного из них представлено далее.

### 3. Эвристический метод

Первым шагом на пути к решению является определение нижней границы числа биклик, составляющих искомое покрытие, и выделение из графа  $G_F$  такого числа изолированных ребер с наибольшим числом приписанных переменных. В полном графе  $K_n$  мощность покрытия бикликами ребер графа не меньше, чем  $\lceil \log_2 n \rceil$ , где  $\lceil a \rceil$  – целое число, ближайшее сверху к  $a$ . Если взять раскраску вершин графа  $G_F$  и заменить каждое множество одноцветных вершин одной вершиной, сохранив все ребра, то получим полный граф, скажем  $K_m$ . Таким образом, выделим, как указано выше,  $\lceil \log_2 m \rceil = k$  изолированных ребер и объявим их начальной совокупностью биклик  $B_1, B_2, \dots, B_k$ . (Поскольку метод не претендует на получение точного решения, можно воспользоваться последовательной раскраской графа, не гарантирующей минимума числа цветов.)

Дальнейший процесс решения задачи представляется как последовательность шагов, на каждом из которых выбирается пара  $(v_i, V_j^s)$ , где  $v_i \notin V_j^1, V_j^2$ , и вершина  $v_i$  связана ребрами из  $E_X$  со всеми вершинами из  $V_j^s$ . Вершина  $v_i$  вносится в множество  $V_j^t$  ( $t \neq s$ ), и таким образом вносятся новые ребра в биклику  $B_j = \langle V_j^1, V_j^2 \rangle$ . Естественно, такое действие имеет смысл, когда среди этих ребер имеется хотя бы одно из ребер графа  $G_F$ , не присутствующее ни в одной из имеющихся биклик. Внесение вершины  $v_i$  в множество  $V_j^t$  сопровождается добавлением в КНФ, соответствующую биклике  $B_j$ , элементарных дизъюнкций, связанных с вносимыми ребрами. При этом, естественно, учитывается закон поглощения  $(a \vee b) a = a$ , и надо, чтобы изменяемая биклика оставалась допустимой. То есть ДНФ, получаемая раскрытием скобок в КНФ, должна содержать хотя бы одну элементарную конъюнкцию ранга, не превышающего  $p$ .

Если такой пары не существует, в искомую совокупность вносится новая биклика в виде одного ребра из  $E_F$ , не принадлежащего ни одной из уже полученных биклик. Процесс заканчивается, когда каждое ребро из  $E_F$  окажется хотя бы в одной из биклик  $B_1, B_2, \dots, B_k$ .

Выбор пары  $(v_i, V_j^s)$  осуществляется по критериям, которые перечислены далее в порядке их применения.

1. Минимум ребер графа  $G_F$ , которые не сможет покрыть биклика  $B_j$  (такое ребро связывает пару вершин, присутствующей в той или другой доле биклики  $B_j$ ).
2. Максимум новых покрываемых ребер из  $E_F$ , вводимых в  $B_j$  вместе с вершиной  $v_i$ .
3. Наименьший минимальный ранг элементарной конъюнкции в соответствующей ДНФ.
4. Максимум числа элементарных конъюнкций минимального ранга в соответствующей ДНФ.

**Пример 2.** Пусть система частичных булевых функций  $f(x)$  задана матрицами из примера 1, и надо решить ту же задачу, что поставлена в этом примере.

Нижней границей мощности покрытия графа  $G_F$  бикликами является 2. Из табл. 1 выбираем два непересекающихся ребра с элементарными дизъюнкциями максимального ранга и получаем следующие биклики:

$$\begin{aligned} B_1 &= \langle \{v_1\}, \{v_4\} \rangle - (x_1 \vee x_2 \vee x_3 \vee x_4 \vee x_6), \\ B_2 &= \langle \{v_2\}, \{v_3\} \rangle - (x_1 \vee x_4 \vee x_5). \end{aligned}$$

Пара  $(v_5, \{v_4\})$  для варианта формирования биклики  $\langle \{v_1, v_5\}, \{v_4\} \rangle$  оценивается по приведенным критериям как одна из лучших. Действительно, ребро  $v_1v_5$  отсутствует в графе  $G_F$ . Число новых покрываемых ребер во всех случаях равно единице. Соответствующая КНФ имеет вид  $(x_1 \vee x_2 \vee x_3 \vee x_4 \vee x_6)(x_1 \vee x_3 \vee x_6)$  и, преобразованная по закону поглощения, совпадает с ДНФ  $x_1 \vee x_3 \vee x_6$ , имеющей три элементарных конъюнкции ранга 1. Таким образом, имеем теперь следующие биклики:

$$\begin{aligned} B_1 &= \langle \{v_1, v_5\}, \{v_4\} \rangle - (x_1 \vee x_3 \vee x_6), \\ B_2 &= \langle \{v_2\}, \{v_3\} \rangle - (x_1 \vee x_4 \vee x_5). \end{aligned}$$

На следующем шаге выбирается пара  $(v_2, \{v_1, v_5\})$ , и результатом следующего шага является совокупность биклик

$$B_1 = \langle \{v_1, v_5\}, \{v_2, v_4\} \rangle - x_1,$$

$$B_2 = \langle \{v_2\}, \{v_3\} \rangle - (x_1 \vee x_4 \vee x_5).$$

Такое последовательное внесение ребер в биклики приводит к совокупности биклик

$$B_1 = \langle \{v_1, v_5\}, \{v_2, v_3, v_4\} \rangle - x_1,$$

$$B_2 = \langle \{v_2, v_4\}, \{v_3, v_6\} \rangle - x_2 x_4 (x_1 \vee x_3 \vee x_5).$$

Ребро  $v_3v_6$  графа  $G_F$  не может быть покрыто ни одной из полученных биклик. У биклики  $B_2$  имеется доля, содержащая пару вершин  $v_3, v_6$ , а расширенная биклика  $B_1 = \langle \{v_1, v_5, v_6\}, \{v_2, v_3, v_4\} \rangle$  покрывает ребро  $v_3v_6$ , но не является допустимой, так как элементарные конъюнкции в соответствующей ДНФ имеют ранги, превышающие  $p = 3$ :  $x_1 x_2 x_4 (x_5 \vee x_6) = x_1 x_2 x_4 x_5 \vee x_1 x_2 x_4 x_6$ . Поэтому вводим биклику  $B_3 = \langle \{v_3\}, \{v_6\} \rangle$  с элементарной дизъюнкцией  $(x_1 \vee x_5)$  и добавляем к ней ребро  $v_1v_6$ . В результате получаем следующее покрытие:

$$B_1 = \langle \{v_1, v_5\}, \{v_2, v_3, v_4\} \rangle - x_1(x_5 \vee x_6) = x_1 x_5 \vee x_1 x_6,$$

$$B_2 = \langle \{v_2, v_4\}, \{v_3, v_6\} \rangle - x_2 x_4 (x_1 \vee x_3 \vee x_5) = x_1 x_2 x_4 \vee x_3 x_2 x_4 \vee x_2 x_4 x_5,$$

$$B_3 = \langle \{v_1, v_3\}, \{v_6\} \rangle - (x_1 \vee x_4) (x_1 \vee x_5) = x_1 \vee x_4 x_5.$$

Полученное покрытие определяет следующие матрицы:

$$\begin{aligned} X_1 = \begin{bmatrix} x_1 & x_5 \\ 1 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix}, F_1 = \begin{bmatrix} g_1 \\ 1 \\ 0 \\ 0 \\ 1 \\ - \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix}; \quad X_2 = \begin{bmatrix} x_1 & x_2 & x_4 \\ 1 & 0 & 0 \\ 0 & - & 0 \\ 1 & - & 1 \\ 0 & 1 & 1 \\ 1 & - & - \\ 0 & 0 & 1 \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix}, F_2 = \begin{bmatrix} g_2 \\ - \\ 0 \\ 1 \\ 0 \\ - \\ 1 \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix}; \quad X_3 = \begin{bmatrix} x_1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix}, F_3 = \begin{bmatrix} g_3 \\ 1 \\ - \\ 1 \\ - \\ 0 \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix}; \\ U = \begin{bmatrix} g_1 & g_2 & g_3 \\ 1 & - & 1 \\ 0 & 0 & - \\ 0 & 1 & 1 \\ 0 & 0 & - \\ 1 & - & - \\ - & 1 & 0 \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix}, \Phi = \begin{bmatrix} \varphi_1 & \varphi_2 & \varphi_3 \\ 0 & 0 & 1 \\ - & 1 & 1 \\ 0 & - & 0 \\ 1 & 1 & - \\ - & 0 & 1 \\ 1 & 0 & - \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} \end{aligned}$$

Устранение избыточности в этих матрицах дает следующее представление полученных частичных функций:

$$\begin{aligned} X_1 = \begin{bmatrix} x_1 & x_5 \\ 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}, F_1 = \begin{bmatrix} g_1 \\ 1 \\ 0 \\ 0 \end{bmatrix}; \quad X_2 = \begin{bmatrix} x_1 & x_2 & x_4 \\ 0 & - & 0 \\ 1 & - & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, F_2 = \begin{bmatrix} g_2 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}; \quad X_3 = \begin{bmatrix} x_1 \\ 1 \\ 0 \end{bmatrix}, F_3 = \begin{bmatrix} g_3 \\ 1 \\ 0 \end{bmatrix}; \\ U = \begin{bmatrix} g_1 & g_2 & g_3 \\ 1 & - & 1 \\ 0 & 0 & - \\ 0 & 1 & 1 \\ 1 & - & - \\ - & 1 & 0 \end{bmatrix}, \Phi = \begin{bmatrix} \varphi_1 & \varphi_2 & \varphi_3 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & - & 0 \\ - & 0 & 1 \\ 1 & 0 & - \end{bmatrix} \end{aligned}$$



Совместная минимизация полученных функций в классе ДНФ приводит к следующим матричным представлениям минимальных систем ДНФ:

$$X = \begin{matrix} & x_1 & x_2 & x_4 & x_5 \\ \begin{bmatrix} 1 & - & - & 1 \\ - & 0 & 1 & - \\ 1 & - & - & - \end{bmatrix} & , & G = \begin{matrix} & g_1 & g_2 & g_3 \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} & ; & U = \begin{matrix} & g_1 & g_2 & g_3 \\ \begin{bmatrix} - & 1 & 0 \\ 0 & 0 & - \\ 1 & - & - \end{bmatrix} & , & \Phi = \begin{matrix} & \varphi_1 & \varphi_2 & \varphi_3 \\ \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{matrix} \end{matrix}$$

### Заключение

Представлено два метода решения задачи многоблочной параллельной декомпозиции системы частичных булевых функций. Первый метод получает точное решение, т.е. гарантировано минимальное число блоков в структурной реализации заданной системы. Второй метод такой гарантии не дает, но позволяет решать задачу значительно быстрее, чем первый метод. Как было сказано, точный метод может служить основой для разработки других, эвристических методов, более пригодных для практических задач. К таким методам относится второй из описанных методов. Точный метод может быть использован как эталон для оценки качества решений, получаемых эвристическими методами. Под качеством решения следует понимать близость его к минимальному решению и простоту получаемых функций. Нетрудно заметить, что представленный эвристический метод получил решение для рассматриваемого примера по качеству, не отличающемуся от решения, полученного точным методом.

### ЛИТЕРАТУРА

1. Hassoun S., Sasao T. Logic Synthesis and Verification. The Springer International Series in Engineering and Computer Science. Kluwer Academic Publishers, 2001. 472 p.
2. Perkowski M.A., Grygiel S. A Survey of Literature on Functional Decomposition, Version IV (Technical report). Portland : Portland State University, Department of Electrical Engineering, 1995. 188 p.
3. An improved functional decomposition method based on FAST and the method of removal and operation / F. Yu et al. // International Conference on System Science and Engineering (ICSSE), Dalian, China, Jun. 2012. P. 487–492.
4. Закревский А.Д., Поттосин Ю.В., Черемисинова Л.Д. Логические основы проектирования дискретных устройств. М. : Физматлит, 2007. 592 с.
5. Закревский А.Д., Перишкин А.Е. Параллельная декомпозиция системы слабо определенных булевых функций // Логическое проектирование. Минск : Ин-т техн. кибернетики НАН Беларуси, 2000. Вып. 5. С. 59–66.
6. Поттосин Ю.В., Шестаков Е.А. Табличные методы декомпозиции систем полностью определенных булевых функций // Минск : Белорусская наука, 2006. 327 с.
7. Бибило П.Н. Декомпозиция булевых функций на основе решения логических уравнений. Минск : Беларус. навука, 2009. 211 с.
8. Files C.M., Perkowski M.A. New multivalued functional decomposition algorithms based on MDDs // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2000. V. 19, No. 9. P. 1081–1086.
9. Закревский А.Д. Комбинаторный поиск подходящих разбиений при декомпозиции булевых функций // Вестник Томского государственного университета. Приложение. 2006. № 18. С. 4–9.
10. Поттосин Ю.В., Шестаков Е.А. Применение аппарата покрытий троичных матриц для поиска разбиения множества аргументов при декомпозиции булевых функций // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2011. № 3 (16). С. 100–107.
11. Rawski M. Input variable partitioning method for decomposition-based logic synthesis targeted heterogeneous FPGAs // International Journal of Electronics and Telecommunications. 2012. V. 58, No. 1. P. 15–20.
12. Бибило П.Н. Применение диаграмм двоичного выбора при синтезе логических схем. Минск : Беларус. навука, 2014. 231 с.
13. Taghavi A.S., Pottosin Yu.V., Arasteh B. An input variable partitioning algorithm for functional decomposition of a system of Boolean functions based on the tabular method // Discrete Applied Mathematics. 2015. V. 185. P. 208–219.
14. Поттосин Ю.В., Шестаков Е.А. Декомпозиция системы частичных булевых функций с помощью покрытия графа полными двудольными подграфами // Новые информационные технологии в исследовании дискретных структур : доклады Второй всерос. конф. Екатеринбург : УрО РАН, 1998. С. 185–189.

15. Поттосин Ю.В. Метод многоблочной параллельной декомпозиции системы частичных булевых функций // Информатика. 2017. № 3 (55). С. 92–98.
16. Pottosina S., Pottosin Yu., Sedliak B. Finding maximal complete bipartite subgraphs in a graph // J. Applied Mathematics. 2008. V. 1, No. 1. P. 75–81.

Поступила в редакцию 22 апреля 2018 г.

Pottosin Yu.V. (2018) PARALLEL DECOMPOSITION OF A SYSTEM OF PARTIAL BOOLEAN FUNCTIONS. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitel'naja tehnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 45. pp. 83–91

DOI: 10.17223/19988605/45/10

The problem of Boolean functions decomposition is to represent a given system of Boolean functions as a superposition of simpler Boolean functions. In fact, the implementation of a system of Boolean functions by logical unites, or the synthesis of a combinational circuit, is reduced to decomposition, in which the obtained superposition includes functions implemented by logical unites.

The decomposition problem is considered in the following statement. Given a system of partial (incompletely specified) Boolean functions in the vector form,  $f(x) = (f_1(x), f_2(x), \dots, f_m(x))$ , where the components of the vector  $x = (x_1, x_2, \dots, x_n)$  are Boolean variables forming a set  $X$ . A superposition  $f(x) \leq \Phi(g_1(z_1), g_2(z_2), \dots, g_k(z_k))$  must be found, where the components of the vector  $z_i$ ,  $i = 1, 2, \dots, k$ , are the variable in the set  $Z_i \subset X$  and  $\leq$  denotes the realization relation, i.e., the values of the components of the vector function  $\Phi$  coincide with the values of the components of  $f$  everywhere they are specified. At that, the cardinality  $|Z_i|$ ,  $i = 1, 2, \dots, k$ , must be restricted by a given value  $p$ , and  $k$  must be minimum and less than  $n$ . An approach to solving the problem that does not demand the sets  $Z_1, Z_2, \dots, Z_k$  to be given is described.

The approach uses the interval representation of a system of partial Boolean functions, i.e., in the form of a pair of ternary matrices,  $X, F$ , of dimension  $l \times n$  and  $l \times m$ , respectively. The columns of the matrix  $X$  correspond to the values  $x_1, x_2, \dots, x_n$ , and the columns of the matrix  $F$  to the functions  $f_1(x), f_2(x), \dots, f_m(x)$ . A row of  $X$  gives an interval of Boolean space, and the corresponding row of  $F$  the function values at this interval. The symbol “–” in the  $i$ -th row and  $j$ -th column of  $F$  means that the  $i$ -th interval is not used to specify the function  $f_j(x)$ . The rows of  $X$  and  $F$  have common numeration.

The graphs  $G_X = (V, E_X)$  and  $G_F = (V, E_F)$  are considered, where  $V$  is the set of common numbers of rows of the matrices  $X$  and  $F$ , and  $E_X$  and  $E_F$  are the sets of pairs of orthogonal rows of the matrices  $X$  and  $F$ , respectively. A system of Boolean functions is given with matrices  $X$  and  $F$  correctly if  $E_F \subseteq E_X$ , i.e.,  $G_F$  is a spanned subgraph of  $G_X$ . Every edge in  $E_X$  is assigned with the variables from the set  $X = \{x_1, x_2, \dots, x_n\}$ , according to which the corresponding rows of  $X$  are orthogonal. A complete bipartite subgraph (*biclique*) of  $G_X$  is assigned with the set of variables in  $X$  taken one by one from each edge of the biclique. A biclique is called *admissible* if the number of variables assigned to it is at most  $p$ , and it contains at least one edge in  $E_F$ .

Let  $B_1, B_2, \dots, B_k$  be bicliques covering the set  $E_F$ . Any biclique  $B_i$  can be given by a pair of vertex sets  $\langle V_i', V_i'' \rangle$ . Every function  $g_i(z_i)$  of the required superposition is specified by matrices  $X_i$  and  $F_i$ . The matrix  $X_i$  is the minor of  $X$  formed by the columns corresponding to the variables assigned to the biclique  $B_i$ . The matrix  $F_i$  consists of one column, where the element with number corresponding to the vertex in  $V_i'$  is 0, and the element with number corresponding to the vertex in  $V_i''$  is 1 (or vice versa). The element that does not correspond to any vertex in  $V_i'$  or in  $V_i''$  is “–”. The vector function  $\Phi$  is given by matrices  $U$  and  $\Phi$ . The matrix  $U$  consists of the columns that are the one-column matrices  $F_1, F_2, \dots, F_k$ , and the matrix  $\Phi$  coincide with  $F$ .

Two methods for the considered problem are presented. The first method obtains an exact solution, i.e., the minimal number of blocks in a structural implementation of the given system of functions is ensured. This method is reduced to that first all maximal admissible bicliques are found in graph  $G_X$ , and then a shortest cover with them of the edges in  $E_F$  is obtained. The second method is heuristic one, it does not guarantee the minimal solution, but it allows solving the problem much faster in comparison with the first one. This method forms bicliques successively, that constitute a cover of the edges of graph  $G_F$  finally. The exact method can serve as a basis for developing other, heuristic methods, more applicable for practical tasks. The second method from the described ones is one of them. Moreover, the exact method can be used as an etalon for estimating the quality of solutions obtained by heuristic methods. The quality of the solution must be understood as its closeness to the optimal solution and simplicity of the obtained functions.

Keywords: system of partial Boolean functions; ternary matrix; complete bipartite subgraph.

POTTOSIN Yuri Vasilievich. (Candidate of Physics and Mathematics, Associate Professor, United Institute of Informatics Problems, NAS of Belarus, Minsk, Belarus).  
E-mail: pott@newman.bas-net.by

## REFERENCES

1. Hassoun, S. & Sasao, T. (2001) *Logic Synthesis and Verification*. The Springer International Series in Engineering and Computer Science. Kluwer Academic Publishers.

2. Perkowski, M.A. & Grygiel, S. (1995) *A Survey of Literature on Functional Decomposition, Version IV (Technical report)*. Portland, USA: Portland State University, Department of Electrical Engineering.
3. Yu, F., et al. (2012) An improved functional decomposition method based on FAST and the method of removal and operation. *International Conference on System Science and Engineering (ICSSE)*. Dalian, China. pp. 487–492. DOI: 10.1109/ICSSE.2012.6257233
4. Zakrevskiy, A.D., Pottosin, Yu.V. & Cheremisinova, L.D. (2007) *Logicheskie osnovy proektirovaniya diskretnykh ustroystv* [Logical Fundamentals for Design of Discrete Devices]. Moscow: Fizmatlit.
5. Zakrevskiy, A.D. & Peryshkin, A.E. (2000) Parallel'naya dekompozitsiya sistemy slabo opredelennykh bulevykh funktsiy [Parallel decomposition of a system of weakly specified Boolean functions]. *Logical Design*. 5. pp. 59–66.
6. Pottosin, Yu.V. & Shestakov, E.A. (2006) *Tablichnye metody dekompozitsii sistem polnost'yu opredelennykh bulevykh funktsiy* [Tabular Methods for Decomposition of Systems of Completely Specified Boolean Functions]. Minsk: Belorusskaya nauka.
7. Bibilo, P.N. (2009) *Dekompozitsiya bulevykh funktsiy na osnove resheniya logicheskikh uravneniy* [Decomposition of Boolean Functions Based on Solving Logical Equations]. Minsk: Belorusskaya nauka.
8. Files, C.M. & Perkowski, M.A. (2000) New multivalued functional decomposition algorithms based on MDDs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 19(9). pp. 1081–1086.
9. Zakrevskiy, A.D. (2006) Combinatorial search for appropriate partitions in decomposition of Boolean functions. *Vestnik Tomskogo gosudarstvennogo universiteta. Prilozheniye – Tomsk State University Journal. Appendix*. 18. pp. 4–9. (In Russian).
10. Pottosin, Yu.V. & Shestakov, E.A. (2011) Application of the ternary matrix cover technique to search for a partition of the set of arguments in decomposition of Boolean functions. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. no. 3(16). pp. 100–107. (In Russian).
11. Rawski, M. (2012) Input variable partitioning method for decomposition-based logic synthesis targeted heterogeneous FPGAs. *International Journal of Electronics and Telecommunications*. 58(1). pp. 15–20. DOI: 10.2478/v10177-012-0002-x
12. Bibilo, P.N. (2014) *Primenenie diagramm dvoichnogo vybora pri sinteze logicheskoy skhem* [Application of Binary Decision Diagrams to the Synthesis of Logic Circuits]. Minsk: Belorusskaya nauka.
13. Taghavi, A.S., Pottosin, Yu.V. & Arasteh, B. (2015) An input variable partitioning algorithm for functional decomposition of a system of Boolean functions based on the tabular method. *Discrete Applied Mathematics*. 185. pp. 208–219. DOI: 10.1016/j.dam.2014.12.013
14. Pottosin, Yu.V. & Shestakov, E.A. (1998) [Decomposition of a system of partial Boolean functions using a complete bipartite subgraph cover of a graph]. *Novye informatsionnye tekhnologii v issledovanii diskretnykh struktur* [New Information Technologies in Investigation of Discrete Structures]. Proc. of the Second All-Russian Conference. Ekaterinburg: Ural Branch of RAS. pp. 185–189. (In Russian).
15. Pottosin, Yu.V. (2017) A method for multi-block parallel decomposition of a system of partial Boolean functions. *Informatika*. 3(55). pp. 92–98. (In Russian).
16. Pottosina, S., Pottosin, Yu. & Sedliak, B. (2008) Finding maximal complete bipartite subgraphs in a graph. *Journal of Applied Mathematics*. 1(1). pp. 75–81. (In Russian).

### СВЕДЕНИЯ ОБ АВТОРАХ

**АРЫКОВ Сергей Борисович** – кандидат физико-математических наук, младший научный сотрудник лаборатории синтеза параллельных программ Института вычислительной математики и математической геофизики СО РАН; доцент кафедры параллельных вычислительных технологий Новосибирского государственного технического университета (г. Новосибирск). E-mail: arykov@sscc.ru

**БУРКАТОВСКАЯ Юлия Борисовна** – кандидат физико-математических наук, доцент отделения информационных технологий инженерной школы информационных технологий и робототехники Национального исследовательского Томского политехнического университета. E-mail: tracey@tpu.ru

**ГАЙДАМАКА Юлия Васильевна** – доктор физико-математических наук, доцент кафедры прикладной информатики и теории вероятностей факультета физико-математических и естественных наук Российского университета дружбы народов (г. Москва). E-mail: gaydamaka\_yuv@rudn.university

**ГАЛИЛЕЙСКАЯ Анастасия Александровна** – магистрант кафедры теории вероятностей и математической статистики Института прикладной математики и компьютерных наук Национального исследовательского Томского государственного университета. E-mail: lusta.nastya@mail.ru

**ДМИТРИЕВ Юрий Глебович** – доцент, доктор физико-математических наук, заведующий кафедрой теоретической кибернетики Института прикладной математики и компьютерных наук Национального исследовательского Томского государственного университета. E-mail: dmit70@mail.ru

**ДОМБРОВСКИЙ Владимир Валентинович** – профессор, доктор технических наук, заведующий кафедрой информационных технологий и бизнес аналитики Института экономики и менеджмента Национального исследовательского Томского государственного университета. E-mail: dombrovs@ef.tsu.ru

**ЗАХАРОВА Галина Борисовна** – доцент, кандидат технических наук, ведущий научный сотрудник Уральского государственного архитектурно-художественного университета; доцент кафедры моделирования управляемых систем Высшей школы экономики и менеджмента Уральского федерального университета им. первого Президента России Б.Н. Ельцина (г. Екатеринбург). E-mail: zgb555@gmail.com

**КОШКИН Геннадий Михайлович** – профессор, доктор физико-математических наук, профессор кафедры теоретической кибернетики Института прикладной математики и компьютерных наук Национального исследовательского Томского государственного университета. E-mail: kgm@mail.tsu.ru

**КУЛЮКИН Владимир Алексеевич** – кандидат технических наук, доцент кафедры компьютерных наук Университета штата Юта (г. Логан, США). E-mail: vladimir.kulyukin@usu.edu

**ЛИСОВСКАЯ Екатерина Юрьевна** – аспирант кафедры теории вероятностей и математической статистики Института прикладной математики и компьютерных наук Национального исследовательского Томского государственного университета. E-mail: ekaterina\_lisovs@mail.ru

**МОИСЕЕВ Александр Николаевич** – доцент, доктор физико-математических наук, профессор кафедры программной инженерии Института прикладной математики и компьютерных наук Национального исследовательского Томского государственного университета. E-mail: moiseev.tsu@gmail.com

**МУКЕРДЖИ Сарбаджит** – аспирант кафедры компьютерных наук Университета штата Юта (г. Логан, США). E-mail: sarbajit.mukherjee@aggiemail.usu.edu

**НЕЖЕЛЬСКАЯ Людмила Алексеевна** – доцент, доктор физико-математических наук, профессор кафедры исследования операций Института прикладной математики и компьютерных наук Национального исследовательского Томского государственного университета. E-mail: ludne@mail.ru

**ОСИПОВА Марина Анатольевна** – кандидат физико-математических наук, доцент кафедры алгебры, геометрии и анализа Дальневосточного федерального университета; научный сотрудник Института прикладной математики ДВО РАН (г. Владивосток). E-mail: mao1975@list.ru

**ПАШИНСКАЯ Татьяна Юрьевна** – кандидат физико-математических наук, доцент кафедры информационных технологий и бизнес аналитики Института экономики и менеджмента Национального исследовательского Томского государственного университета. E-mail: tatyana.obedko@mail.ru

**ПОТТОСИН Юрий Васильевич** – доцент, кандидат физико-математических наук, ведущий научный сотрудник лаборатории логического проектирования Объединенного института проблем информатики НАН Беларуси (г. Минск, Республика Беларусь). E-mail: pott@newman.bas-net.by

**САМУЙЛОВ Константин Евгеньевич** – профессор, доктор технических наук, заведующий кафедрой прикладной информатики и теории вероятностей факультета физико-математических и естественных наук Российского университета дружбы народов (г. Москва). E-mail: samuylov\_ke@pfur.ru

**СИДОРОВА Екатерина Филипповна** – магистрант кафедры исследования операций Института прикладной математики и компьютерных наук Национального исследовательского Томского государственного университета. E-mail: katusha\_sidorova@mail.ru

**СОРОКИН Михаил Сергеевич** – магистрант кафедры моделирования управляемых систем Высшей школы экономики и менеджмента Уральского федерального университета им. первого Президента России Б.Н. Ельцина (г. Екатеринбург). E-mail: sjfh@yandex.ru

**ЦИЦИАШВИЛИ Гурами Шалвович** – профессор, доктор физико-математических наук, профессор кафедры алгебры, геометрии и анализа Дальневосточного федерального университета, главный научный сотрудник Института прикладной математики ДВО РАН (г. Владивосток). E-mail: guram@iam.dvo.ru

**ШКЛЕННИК Мария Александровна** – аспирант кафедры теории вероятностей и математической статистики Института прикладной математики и компьютерных наук Национального исследовательского Томского государственного университета. E-mail: Shklennikm@yandex.ru

**Научный журнал**

**ВЕСТНИК  
ТОМСКОГО  
ГОСУДАРСТВЕННОГО  
УНИВЕРСИТЕТА**

**УПРАВЛЕНИЕ,  
ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА  
И ИНФОРМАТИКА**

**TOMSK STATE UNIVERSITY  
JOURNAL OF CONTROL AND COMPUTER SCIENCE**

**2018. № 45**

Редактор Е.Г. Шумская  
Оригинал-макет Е.Г. Шумской  
Редакторы-переводчики: Г.М. Кошкин; В.Н. Горенинцева  
Дизайн обложки Л.Д. Кривцовой

Подписано к печати 14.12.2018 г. Формат 60x84<sup>1/8</sup>.  
Гарнитура Times. Усл. печ. л. 11,2.  
Тираж 250 экз. Заказ № 3542. Цена свободная.

Дата выхода в свет 21.12.2018 г.

Журнал отпечатан на полиграфическом оборудовании  
Издательского Дома Томского государственного университета  
634050, г. Томск, Ленина, 36  
Тел. 8(382-2)–52-98-49; 8(382-2)–52-96-75  
Сайт: <http://publish.tsu.ru>; E-mail: [rio.tsu@mail.ru](mailto:rio.tsu@mail.ru)