

ВЕСТНИК
ТОМСКОГО
ГОСУДАРСТВЕННОГО
УНИВЕРСИТЕТА

УПРАВЛЕНИЕ,
ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА
И ИНФОРМАТИКА

TOMSK STATE UNIVERSITY
JOURNAL OF CONTROL AND COMPUTER SCIENCE

Научный журнал

2017

№ 38

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»

Founder – Tomsk State University

EDITORIAL BOARD

Alexander Gortsev – Editor-in-Chief, Doctor of Sc., Prof., Head of the Operation Research Department, Dean of the Faculty of Applied Mathematics and Cybernetics Tomsk State University. Tel: +72822529599

Valery Smagin – Deputy Editor-in-Chief, Doctor of Sc., Prof. of the Applied Mathematics Department Tomsk State University. Tel: +72822529599

Lyudmila Nezhelskaya – Executive Editor, Cand. of Sc., Docent of the Operation Research Department.
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 Semenko – 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, Dean of the Faculty of Informatics Tomsk State University

Mais Farkhadov – Doctor of Sc., Head of the laboratory Institute of Control Sciences of Russian Academy of Sciences (Moscow, Russia).

Gurami Tsitsiashvili – Doctor of Sc., Prof., Chief researcher Institute for Applied Mathematics Far Eastern Branch of RAS, Prof. Far Eastern Federal University (Vladivostok, Russia)

Editorial address:

Faculty of applied mathematics and cybernetics
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

Enco Orzinger
PhD, Prof.
University of Rome
Italy

Paolo Prinetto
Prof.
Politechnic Institute
Torino, Italy

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

Nina Yevtushenko
Doctor of Sc., Prof.
Tomsk State University
Russia

Yervant Zorian
PhD, Fellow & Chief Architect, Synopsys, Mountain View, CA, USA

JORNAL 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, декан факультета прикладной математики и кибернетики ТГУ. Тел. +72822529599

Смагин Валерий Иванович – зам. гл. редактора, проф., д-р техн. наук, проф. кафедры прикладной математики ТГУ. Тел. +72822529599

Нежельская Людмила Алексеевна – ответственный секретарь, доц., канд. техн. наук, доц. кафедры исследования операций ТГУ.
E-mail: vestnik_uvti@mail.tsu.ru

Воробейчиков Сергей Эрикович – проф., 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. кафедрой 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.т.н., проф. каф. системного анализа и исследования операций, СГАУ им. акад. М.Ф. Решетнева (г. Красноярск)

Сусенко Сергей Петрович – проф., д-р техн. наук, зав. кафедрой прикладной информатики, декан факультета информатики ТГУ

Фархадов Манс Паша Оглы – д-р техн. наук, зав. лаборатории 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, факультет прикладной математики и кибернетики

Телефон / факс: +72822529599

E-mail: vestnik_uvti@mail.tsu.ru

РЕДАКЦИОННЫЙ СОВЕТ

Ана Роза Ковали
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

Енцо Орзингер
д-р философии, проф.
Римский университет,
Италия

Паоло Принетто
проф.
Политехнический институт,
Турин, Италия

Жильберт Сапорта

д-р философии, проф.
Университет им. Пьера и Марии,
Кюри, Париж, Франция

Раймонд Убар
проф.
Технологический университет,
Таллинн, Эстония

Рейндерт Нобель
д-р, доцент
Свободный университет,
Амстердам, Нидерланды

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

Гараева Э.А., Мансимов К.Б. Необходимое условие оптимальности в задаче управления с дискретным временем при недифференцируемом критерии качества	4
--	---

ОБРАБОТКА ИНФОРМАЦИИ

Бураков Д.П., Гарина М.И. Исследование структуры предпочтений ЛПР с использованием типовых обобщающих функций	11
Dogadova T.V., Vasiliev V.A. Adaptive prediction of stochastic differential equations with unknown parameters	17
Медведев Г.А. Об одной квадратичной модели временной структуры доходности	24
Rouban A.I. The sensitivity functionals in the bolts's problem for multivariate dynamic systems described by ordinary integral equations	30
Трофимов С.П., Иванов А.В. Разрыв двойственности в полубесконечном линейном программировании и анализ качества ограничений геометрических объектов	37

ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

Горохов М.М., Переведенцев Д.А. Проектирование базы знаний экспертной системы управления научными проектами	47
Пазников А.А. Оптимизация делегирования выполнения критических секций на выделенных процессорных ядрах	52
Сарапулов А.В., Уманский А.Б. Реконфигурирование бортовой вычислительной машины для повышения отказоустойчивости	59

ДИСКРЕТНЫЕ ФУНКЦИИ И АВТОМАТЫ

Бурдонов И.Б., Косачев А.С. Тестирование системы автоматов	63
--	----

ПРОЕКТИРОВАНИЕ И ДИАГНОСТИКА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Буторина Н.Б. О свойствах программируемых логических блоков, реализующих кодовые слова равновесных кодов	72
--	----

МЕМОАРЫ, ПАМЯТНЫЕ ДАТЫ, ПЕРСОНАЛИИ

85 лет профессору Ф.П. Тарасенко	80
Сведения об авторах	83

CONTENTS

CONTROL OF DYNAMICAL SYSTEMS

Garaeva E.A., Mansimov K.B. Necessary optimality condition in one discrete control problem from nondifferentiable control cost	4
--	---

DATA PROCESSING

Burakov D.P., Garina M.I. Analysis of decision maker's preferences structure with standard aggregative functions	11
Dogadova T.V., Vasiliev V.A. Adaptive prediction of stochastic differential equations with unknown parameters	17
Medvedev G.A. On a quadratic model of yield term structure	24
Rouban A.I. The sensitivity functionals in the bolts's problem for multivariate dynamic systems described by ordinary integral equations	30
Trofimov S.P., Ivanov A.V. The duality gap in semi-infinite linear programming and the quality analysis of geometrical objects' constraints	37

INFORMATICS AND PROGRAMMING

Gorokhov M.M., Perevedentsev D.A. Design of the knowledge base of expert system for management scientific projects	47
Paznikov A.A. Optimization method of remote core locking	52
Sarapulov A.V., Umansky A.B. Reconfiguring the on-board computer to increase stability in case of failure	59

DISCRETE FUNCTION AND AUTOMATONS

Burdonov I.B., Kossatchev F.C. Testing of automata system	63
---	----

DESIGNING AND DIAGNOSTICS OF COMPUTER SYSTEMS

Butorina N.B. The properties of programmable logic blocks realizing code words of (m, n) -code	72
--	----

MEMOIRS, MEMORABLE DATE, PERSONALITIS

85 years to prof. F.P. Tarasenko	80
Brief information about the authors	83

УПРАВЛЕНИЕ ДИНАМИЧЕСКИМИ СИСТЕМАМИ

УДК 517.977

DOI: 10.17223/19988605/38/1

Э.А. Гараева, К.Б. Мансимов

НЕОБХОДИМОЕ УСЛОВИЕ ОПТИМАЛЬНОСТИ В ЗАДАЧЕ УПРАВЛЕНИЯ С ДИСКРЕТНЫМ ВРЕМЕНЕМ ПРИ НЕДИФФЕРЕНЦИРУЕМОМ КРИТЕРИИ КАЧЕСТВА

Рассмотрена одна дискретная задача оптимального управления с негладким критерием качества. Установлено необходимое условие оптимальности в терминах производных по направлениям.

Ключевые слова: дискретная задача оптимального управления; приращение критерия качества; необходимое условие оптимальности; производная по направлениям.

В работах [1–3] изучена задача оптимального управления, представляющая собой дискретный аналог одной из задач, рассмотренной в работах А.И. Москаленко (см.: [4, 5]) и занимающей промежуточное положение между задачами оптимального управления с сосредоточенными и распределенными параметрами. Установлен аналог дискретного принципа максимума, уравнения Эйлера, выведен аналог линеаризованного условия максимума.

В предлагаемой статье рассматривается задача, аналогичная задаче из [1–3], но при предположении недифференцируемости функционала качества. Выведено необходимое условие оптимальности в терминах производных по направлениям.

1. Постановка задачи

Пусть управляемый объект описывается системой разностных уравнений

$$z(t+1, x) = f(t, x, z(t, x), u(t)), \quad t = t_0, t_0 + 1, \dots, t_1 - 1; \quad x = x_0, x_0 + 1, \dots, x_1, \quad (1)$$

с начальным условием

$$z(t_0, x) = y(x), \quad x = x_0, x_0 + 1, \dots, x_1, \quad (2)$$

где $y(x)$ – n -мерная вектор-функция, являющаяся решением задачи

$$\begin{aligned} y(x+1) &= g(x, y(x), v(x)), \quad x = x_0, x_0 + 1, \dots, x_1 - 1, \\ y(x_0) &= y_0. \end{aligned} \quad (3)$$

Здесь $f(t, x, z, u)$ ($g(x, y, v)$) – заданная n -мерная вектор-функция, непрерывная по совокупности переменных вместе с частными производными по z (y); y_0 – заданный постоянный вектор; t_0, t_1, x_0, x_1 – заданные числа, причем разности $t_1 - t_0$ и $x_1 - x_0$ есть натуральные числа; $u(t)$ ($v(x)$) – r (q)-мерный вектор управляющих воздействий со значениями из заданного непустого, ограниченного множества U (V), т.е.

$$\begin{aligned} u(t) &\in U \subset R^r, \quad t \in T = \{t_0, t_0 + 1, \dots, t_1 - 1\}, \\ v(x) &\in V \subset R^q, \quad x \in X = \{x_0, x_0 + 1, \dots, x_1 - 1\}. \end{aligned} \quad (4)$$

Пару $(u(t), v(x))$ с вышеперечисленными свойствами назовем допустимым управлением.

Рассмотрим задачу о минимуме функционала

$$S(u, v) = \varphi_1(y(x_1)) + \sum_{x=x_0}^{x_1-1} \varphi_2(x, z(t, x)) \quad (5)$$

при ограничениях (1)–(4). Здесь $\varphi_1(y)$ ($\varphi_2(x, z)$) – заданная скалярная функция, удовлетворяющая условию Липшица по y (z) и имеющая производные по y (z) по любому направлению.

Допустимое управление $(u^\circ(t), v^\circ(x))$, доставляющее минимум функционалу (5) при ограничениях (1)–(4), назовем оптимальным управлением, а соответствующий процесс $(u^\circ(t), v^\circ(x), z^\circ(t, x), y^\circ(x))$ – оптимальным процессом.

2. Вспомогательные факты

Пусть $(u^\circ(t), v^\circ(x), z^\circ(t, x), y^\circ(x))$ – фиксированный допустимый процесс. Через $(\bar{u}(t) = u^\circ(t) + \Delta u(t), \bar{v}(x) = v^\circ(x) + \Delta v(x), \bar{z}(t, x) = z^\circ(t, x) + \Delta z(t, x), \bar{y}(x) = y^\circ(x) + \Delta y(x))$ обозначим произвольный допустимый процесс.

Тогда приращение $(\Delta z(t, x), \Delta y(x))$ состояния $(z^\circ(t, x), y^\circ(x))$ будет решением задачи

$$\Delta z(t+1, x) = f(t, x, \bar{z}(t, x), \bar{u}(t)) - f(t, x, z^\circ(t, x), u^\circ(t)), \quad (6)$$

$$\Delta z(t_0, x) = \Delta y(x), \quad (7)$$

$$\Delta y(x+1) = g(x, \bar{y}(x), \bar{v}(x)) - g(x, y^\circ(x), v^\circ(x)), \quad (8)$$

$$\Delta y(x_0) = 0. \quad (9)$$

Используя формулу Тейлора из (6)–(9), получаем, что приращение $(\Delta z(t, x), \Delta y(x))$ состояния $(z^\circ(t, x), y^\circ(x))$ является решением линеаризованной задачи

$$\Delta z(t+1, x) = f_z(t, x) \Delta z(t, x) + \Delta_{\bar{u}(t)} f(t, x) + \eta_1(t, x; \Delta u), \quad (10)$$

$$\Delta z(t_0, x) = \Delta y(x), \quad (11)$$

$$\Delta y(x+1) = g_z(x) \Delta y(x) + \Delta_{\bar{v}(x)} g(x) + \eta_2(x; \Delta v(x)), \quad (12)$$

$$\Delta y(x_0) = 0, \quad (13)$$

где по определению

$$\eta_1(t, x; \Delta u) = \Delta_{\bar{u}(t)} f_z(t, x) \Delta z(t, x) + o_1(\|\Delta z(t, x)\|),$$

$$\eta_2(x; \Delta v(x)) = \Delta_{\bar{v}(x)} g_y(x) \Delta y(x) + o_2(\|\Delta y(x)\|).$$

Здесь и в дальнейшем для простоты изложений используется обозначения типа

$$f_z(t, x) \equiv f(t, x, z^\circ(t, x), u^\circ(t)),$$

$$g_y(x) = g(x, y^\circ(x), v^\circ(x)),$$

$$\Delta_{\bar{u}(t)} f(t, x) \equiv f(t, x, z^\circ(t, x), \bar{u}(t)) - f(t, x, z^\circ(t, x), u^\circ(t)),$$

$$\Delta_{\bar{v}(x)} g(x) = g(x, y^\circ(x), \bar{v}(x)) - g(x, y^\circ(x), v^\circ(x)).$$

Уравнение (10), (12) можно интерпретировать как линейное неоднородное разностное уравнение относительно $\Delta z(t, x)$ ($\Delta y(x)$).

Поэтому на основе формулы о представлении решений линейных неоднородных разностных уравнений (см. например, [6, 7]) решения задач (10), (11) и (12), (13) соответственно можно представить в виде

$$\Delta z(t, x) = F(t, t_0 - 1; x) \Delta z(t_0, x) + \sum_{\tau=t_0}^{t-1} F(t, \tau; x) \Delta_{\bar{u}(\tau)} f(\tau, x) + \eta_3(t, x; \Delta u(t)), \quad (14)$$

$$\Delta y(x) = \sum_{s=x_0}^{x-1} \Phi(x, s) \Delta_{\bar{v}(s)} g(s) + \eta_4(x; \Delta v(x)), \quad (15)$$

где по определению

$$\eta_3(t, x; \Delta u(t)) = \sum_{\tau=t_0}^{t-1} F(t, \tau; x) \eta_1(\tau, x; \Delta u(\tau)), \quad \eta_4(x; \Delta v(x)) = \sum_{s=x_0}^{x-1} \Phi(x, s) \eta_2(s; \Delta v(s)).$$

Здесь $F(t, \tau; x)$, $\Phi(x, s)$ – $(n \times n)$ матричные функции, являющиеся решениями уравнений

$$F(t, \tau-1, x) = F(t, \tau; x) f_z(\tau, x), \quad F(t, t-1, x) = E, \quad \Phi(x, s-1) = \Phi(x, s) g_y(s), \\ \Phi(x, x-1) = E, \quad (E - (n \times n) \text{ единичная матрица}).$$

Поскольку $\Delta z(t_0, x) = \Delta y(x)$, то, учитывая (15), представление (14) решения $\Delta z(t, x)$ задачи (10), (11) записывается в виде

$$\Delta z(t, x) = \sum_{s=x_0}^{x-1} F(t, t_0-1; x) \Phi(x, s) \Delta_{\bar{v}(s)} g(s) + \sum_{\tau=t_0}^{t-1} F(t, \tau; x) \Delta_{\bar{u}(\tau)} f(\tau, x) + \eta_5(t, x; \Delta u(t), \Delta v(x)), \quad (16)$$

где по определению

$$\eta_5(t, x; \Delta u(t), \Delta v(x)) = \eta_3(t, x; \Delta u(t)) + F(t, t_0-1; x) \eta_4(x; \Delta v(x)).$$

В дальнейшем нам понадобится оценка нормы приращения траектории.

Ясно, что

$$\Delta z(t+1, x) = \sum_{\tau=t_0}^t (\Delta z(\tau+1, x) - \Delta z(\tau, x)) + \Delta z(t_0, x), \quad \Delta y(x+1) = \sum_{s=x_0}^x (\Delta y(s+1) - \Delta y(s)).$$

Отсюда с учетом задач (10)–(13) будем иметь

$$\Delta z(t+1, x) = \sum_{\tau=t_0}^t \left[f(\tau, x, \bar{z}(\tau, x), \bar{u}(\tau)) - f(\tau, x, z^\circ(\tau, x), u^\circ(\tau)) \right] + \Delta y(x), \quad (17)$$

$$\Delta y(x+1) = \sum_{s=x_0}^{x-1} \left[g(s, \bar{y}(s), \bar{v}(s)) - g(s, y^\circ(s), v^\circ(s)) \right]. \quad (18)$$

Переходя к норме в обеих частях этих соотношений и используя условия Липшица, после некоторых преобразований получим

$$\|\Delta z(t+1, x)\| \leq \sum_{\tau=t_0}^t \|\Delta_{\bar{u}(\tau)} f(\tau, x)\| + L_1 \sum_{\tau=t_0}^t \|\Delta z(\tau, x)\| + \|\Delta y(x)\|, \quad (19)$$

$$\|\Delta y(x+1)\| \leq \sum_{s=x_0}^x \|\Delta_{\bar{v}(s)} g(s)\| + L_2 \sum_{s=x_0}^x \|\Delta y(s)\|. \quad (20)$$

Здесь $L_i = \text{const} > 0$, $i = 1, 2$, – некоторые постоянные.

Применяя дискретный аналог леммы Гронуолла–Беллмана (см. например, [7, 8]) к неравенству (20), получим

$$\|\Delta y(x)\| \leq L_3 \sum_{s=x_0}^x \|\Delta_{\bar{v}(s)} g(s)\| \quad (L_3 = \text{const} > 0). \quad (21)$$

Далее, учитывая оценку (21) в (20), а затем применяя лемму Гронуолла–Беллмана приходим к оценке

$$\|\Delta z(t, x)\| \leq L_4 \left[\sum_{\tau=t_0}^{t-1} \|\Delta_{\bar{u}(\tau)} f(\tau, x)\| + \sum_{s=x_0}^{x-1} \|\Delta_{\bar{v}(s)} g(s)\| \right], \quad (22)$$

где $L_4 = \text{const} > 0$.

3. Необходимое условие оптимальности

Предположим, что множества

$$f(t, x, z^\circ(t, x), U) = \{ \alpha : \alpha = f(t, x, z^\circ(t, x), u), \quad u \in U \}, \quad (23)$$

$$g(x, y^o(x), V) = \{\beta: \beta = g(x, y^o(x), V), v \in V\} \quad (24)$$

выпуклы.

Перейдем к выводу необходимого условия оптимальности в рассматриваемой задаче. Считая $(u^o(t), v^o(x))$ оптимальным управлением, его специальное приращение определим по формуле

$$\begin{cases} \Delta u_\varepsilon(t) \equiv u(t; \varepsilon) - u^o(t), & t \in T, \\ \Delta v_\varepsilon(x) \equiv 0, & x \in X, \end{cases} \quad (25)$$

где $\varepsilon \in [0, 1]$ – произвольное число, а $u(t; \varepsilon) \in U$, $t \in T$, – произвольная допустимая управляющая функция, такая, что

$$\Delta_{u(t; \varepsilon)} f(t, x) = \varepsilon \Delta_{u(t)} f(t, x).$$

Здесь $u(t)$ – произвольная допустимая управляющая функция, соответствующая $u(t; \varepsilon)$. Это возможно в силу выпуклости множества (23). Через $(\Delta z_\varepsilon(t, x), \Delta y_\varepsilon(x))$ обозначим специальное приращение траектории $(z^o(t, x), y^o(x))$, отвечающее приращению (25) управления $(u^o(t), v^o(x))$. Из оценок (21), (22) сразу следует, что

$$\|\Delta z_\varepsilon(t, x)\| \sim \varepsilon, \quad \|\Delta y_\varepsilon(x)\| = 0.$$

Поэтому из (14) следует, что

$$\Delta z_\varepsilon(t_1, x) = \varepsilon \sum_{t=t_0}^{t_1-1} F(t_1, t) \Delta_{u(t)} f(t, x) + o(\varepsilon).$$

Полагая

$$\ell(u) = \varepsilon \sum_{t=t_0}^{t_1-1} F(t_1, t) \Delta_{u(t)} f(t, x),$$

эта формула записывается в виде

$$\Delta z_\varepsilon(t_1, x) = \varepsilon \ell(u) + o(\varepsilon). \quad (26)$$

Вычислим специальное приращение критерия качества (5), соответствующее приращению (25) управления $(u^o(t), v^o(x))$. Имеем

$$\begin{aligned} \Delta S_\varepsilon(u^o, v^o) &= S(u^o + \Delta u_\varepsilon, v^o) - S(u^o, v^o) = \sum_{x=x_0}^{x_1-1} [\varphi_2(x, z(t_1, x) + \Delta z_\varepsilon(t_1, x)) - \varphi_2(x, z(t_1, x))] = \\ &= \sum_{x=x_0}^{x_1-1} [\varphi_2(x, z(t_1, x) + \varepsilon \ell(u) + o(\varepsilon)) - \varphi_2(x, z(t_1, x))] = \\ &= \sum_{x=x_0}^{x_1-1} [\varphi_2(x, z(t_1, x) + \varepsilon \ell(u) + o(\varepsilon)) - \varphi_2(x, z(t_1, x) + \varepsilon \ell(u)) + \\ &\quad + (\varphi_2(x, z(t_1, x) + \varepsilon \ell(u)) - \varphi_2(x, z(t_1, x)))]. \end{aligned}$$

Поскольку $\varphi_2(x, z)$ удовлетворяет условию Липшица по z и имеет производные по направлениям, то из последнего соотношения получаем, что вдоль оптимального процесса $(u^o(t), v^o(x))$

$$\varepsilon \sum_{x=x_0}^{x_1-1} \frac{\partial \varphi_2(x, z(t_1, x))}{\partial \ell(u)} + o(\varepsilon) \geq 0.$$

Отсюда при достаточно малых $\varepsilon \in [0, 1]$ следует, что

$$\sum_{x=x_0}^{x_1-1} \frac{\partial \varphi_2(x, z(t_1, x))}{\partial \ell(u)} \geq 0. \quad (27)$$

Теперь специальное приращение оптимального управления $(u^\circ(t), v^\circ(x))$ определим по формуле

$$\Delta u_\mu(t) = 0, \quad t \in T, \quad \Delta v_\mu(x) = v(x, \mu) - v^\circ(x), \quad x \in X. \quad (28)$$

Здесь $\mu \in [0, 1]$ – произвольное число, а $v(\mu, x)$ – произвольная допустимая управляющая функция, такая, что

$$\Delta_{v(x, \mu)} g(x) = \mu \Delta_{v(x)} g(x),$$

где $v(x)$ – соответствующая $v(x, \mu)$ произвольная управляющая функция.

Через $(\Delta z_\mu(t, x), \Delta y_\mu(x))$ обозначим специальное приращение оптимальной траектории, соответствующее приращению (28) управления $(u^\circ(t), v^\circ(x))$.

Из оценок (21), (22) следует, что

$$\|\Delta z_\mu(t, x)\| \sim \mu, \quad \|\Delta y_\mu(x)\| \sim \mu. \quad (29)$$

С учетом (29) из представлений (15), (16) следует, что

$$\Delta z_\mu(t_1, x) = \mu \sum_{s=x_0}^{x_1-1} F(t_1, t_0 - 1; x) \Phi(x, s) \Delta_{v(s)} g(s) + o(\mu), \quad (30)$$

$$\Delta y_\mu(x_1) = \mu \sum_{x=x_0}^{x_1-1} \Phi(x_1, x) \Delta_{v(x)} g(x) + o(\mu). \quad (31)$$

Полагая

$$q_1(v, x) = \sum_{s=x_0}^{x-1} F(t_1, t_0 - 1, x) \Phi(x, s) \Delta_{v(s)} g(s), \quad q_2(v) = \sum_{x=x_0}^{x_1-1} \Phi(x_1, x) \Delta_{v(x)} g(x),$$

представление (30), (31) запишем соответственно в виде

$$\Delta y_\mu(x_1) = \varepsilon q_1(v) + o(\mu), \quad (32)$$

$$\Delta z_\mu(t_1, x) = \varepsilon q_2(v, x) + o(\mu). \quad (33)$$

С помощью (32), (33) вычислим специальное приращение функционала качества, соответствующее приращению (28) оптимального управления $(u^\circ(t), v^\circ(x))$.

Имеем

$$\begin{aligned} \Delta S_\mu(u^\circ, v^\circ) &= S(u^\circ, v^\circ + \Delta v_\mu) - S(u^\circ, v^\circ) = [\varphi_1(y^\circ(x_1) + y_\mu(x_1)) - \varphi_1(y^\circ(x_1))] + \\ &\quad + \sum_{x=x_0}^{x_1-1} [\varphi_2(x, z(t_1, x) + \Delta z_\mu(t_1, x)) - \varphi_2(x, z(t_1, x))] = \\ &= [\varphi_1(y^\circ(x_1) + \mu q_1(v) + o(\mu)) - \varphi_1(y^\circ(x_1) + \mu q_1(v))] + [\varphi_1(y^\circ(x_1) + \mu q_1(v)) - \varphi_1(y^\circ(x_1))] - \\ &\quad - \sum_{x=x_0}^{x_1-1} [\varphi_2(x, z^\circ(t_1, x) + \mu q_2(v, x) + o(\mu)) - \varphi_2(x, z^\circ(t_1, x) + \mu q_2(v, x))] + \\ &\quad + [\varphi_2(x, z^\circ(t_1, x) + \mu q_2(v, x)) - \varphi_2(x, z^\circ(t_1, x))]. \end{aligned}$$

Следовательно, имеет место разложение

$$\Delta S_\mu(u^\circ, v^\circ) = \mu \left[\frac{\partial \varphi_1(y^\circ(x_1))}{\partial q_1(v)} + \sum_{x=x_0}^{x_1-1} \frac{\partial \varphi_2(x, z^\circ(t_1, x))}{\partial q_2(x, v)} \right] + o(\mu). \quad (34)$$

Из разложения (34) в силу произвольности $\mu \in [0, 1]$ следует, что

$$\frac{\partial \varphi_1(y^\circ(x_1))}{\partial q_1(v)} + \sum_{x=x_0}^{x_1-1} \frac{\partial \varphi_2(x, z^\circ(t_1, x))}{\partial q_2(x, v)} \leq 0. \quad (35)$$

Сформулируем полученный результат.

Теорема. Пусть множества (23), (24) выпуклы. Тогда для оптимальности допустимого управления $(u^\circ(t), v^\circ(x))$ необходимо, чтобы неравенства (27) и (35) выполнялись соответственно для всех $u(t) \in U, t \in T, v(x) \in V, x \in X$.

Неравенства (27), (35) являются необходимыми условиями оптимальности первого порядка в терминах, производных по направлениям (см., например, [9–11]).

Замечание. Используя необходимые условия оптимальности (27), (35), можно получить необходимые условия оптимальности в задаче на минимакс (см., например, [10, 11]). Из них следует также аналог дискретного условия максимума.

Заключение

Для одной специфической негладкой задачи оптимального управления дискретными системами при помощи аналога метода явной линеаризации получено необходимое условие оптимальности первого порядка в терминах производных по направлениям. Полученный результат может быть применен для исследования задачи на минимакс для рассматриваемой системы.

ЛИТЕРАТУРА

1. Гараева Э.А., Мансимов К.Б. Об одной дискретной задаче оптимального управления // Вестник БГУ. Сер. физ.-мат. наук. 2014. № 1. С. 35–41.
2. Гараева Э.А., Мансимов К.Б. Необходимые условия оптимальности в одной дискретной задаче оптимального управления // Материалы Международной конференции, посвященной 55-летию ИМ и М НАН Азербайджана. Баку, 2014. С. 236–238.
3. Гараева Э.А., Мансимов К.Б. Об одной дискретной задаче оптимального управления // Вестник БГУ. Сер. физ.-мат. наук. 2014. № 1. С. 15–21.
4. Москаленко А.И. Об одном классе задач оптимального регулирования // Журнал вычислительной математики и мат.-физики. 1969. № 1. С. 69–95.
5. Москаленко А.И. Некоторые вопросы теории оптимального управления : автореф. дис. ... канд. физ.-мат. наук. Томск, 1971. 20 с.
6. Габасов Р., Кириллова Ф.М. Оптимизация линейных систем. Минск : Изд-во БГУ, 1973.
7. Мансимов К.Б. Дискретные системы. Баку : Изд-во Бакин. гос. ун-та, 2013. 131 с.
8. Васильев Ф.П. Методы решения экстремальных задач. М. : Наука, 1981. 400 с.
9. Демьянов В.Ф. Минимакс: Дифференцируемость по направлениям. Л. : Изд-во ЛГУ, 1974. 112 с.
10. Демьянов В.Ф., Виноградова Т.К. и др. Задачи теории оптимизации и управления. Л. : Изд-во ЛГУ, 1982. 324 с.
11. Демьянов В.Ф., Рубинов А.М. Основы негладкого анализа и квази-дифференциальное исчисление. М. : Наука, 1990. 482 с.

Гараева Эсмירה Акиф кызы. E-mail: esmira.qarayeva@mail.ru

Институт систем управления НАН Азербайджана (г. Баку)

Мансимов Камил Байрамали оглы, д-р физ.-мат. наук, профессор. E-mail: mansimovbkamil@gmail.com

Бакинский государственный университет

Поступила в редакцию 24 сентября 2016 г.

Garaeva Esmira A. (Institute of Control Systems of NAS Azerbaijan, Baku, Azerbaijan).

Mansimov Kamil B. (Baku State University, Baku, Azerbaijan).

Necessary optimality condition in one discrete control problem from nondifferentiable control cost.

Keywords: discrete optimal control problem; increment the cost functional; necessary optimality conditions; directional derivative.

DOI: 10.17223/19988605/38/1

In this article we consider the problem of necessary optimality condition in one discrete control problem but assuming nondifferentiability quality functional. We derive a necessary condition for optimality in terms of directional derivatives.

Let managed object described by a system of difference equations

$$z(t+1, x) = f(t, x, z(t, x), u(t)), \quad t = t_0, t_0 + 1, \dots, t_1 - 1; \quad x = x_0, x_0 + 1, \dots, x_1, \quad (1)$$

with the initial condition

$$z(t_0, x) = y(x), \quad x = x_0, x_0 + 1, \dots, x_1, \quad (2)$$

where n – dimensional vector function $y(x)$ is a solution of

$$y(x+1) = g(x, y(x), v(x)), \quad x = x_0, x_0 + 1, \dots, x_1 - 1, \quad y(x_0) = y_0. \quad (3)$$

Here $f(t, x, z, u)$ ($g(x, y, v)$) is the given n – dimensional vector-function continuous with respect to all variables together with the partial derivatives with respect to z (y), y_0 is a given constant vector, t_0, t_1, x_0, x_1 are given numbers, the differences $t_1 - t_0$ and $x_1 - x_0$ are natural numbers, $u(t)$ ($v(x)$) is $r(q)$ -dimensional vector of control actions with values from a specified non-empty, bounded set

$$\begin{aligned} u(t) &\in U \subset R^r, \quad t \in T = \{t_0, t_0 + 1, \dots, t_1 - 1\}, \\ v(x) &\in V \subset R^q, \quad x \in X = \{x_0, x_0 + 1, \dots, x_1 - 1\}. \end{aligned} \quad (4)$$

Our goal is to minimize the functional

$$S(u, v) = \varphi_1(y(x_1)) + \sum_{x=x_0}^{x_1-1} \varphi_2(x, z(t_1, x))$$

under the constraints (1)–(4).

Here $\varphi_1(y)$ ($\varphi_2(x, z)$) is the given scalar function satisfying the Lipschitz condition with respect to y (z) and having derivatives with respect to y (z) in any direction.

REFERENCES

1. Garayeva, E.A. & Mansimov, K.B. (2014) On a discrete optimal control problem. *Vestnik BGU. Ser. fiz.-mat. Nauk – Baku University News Journal. Ser. Physics and Mathematics*. 1. pp. 35–41. (In Russian).
2. Garayeva, E.A. & Mansimov, K.B. (2014) [Necessary conditions for optimality in a discrete optimal control problem]. *Materials of the International Conf. Dedicated. to the 55th Anniversary of MI and M National Academy of Sciences of Azerbaijan*. Baku. MI and M the Azerbaijan National Academy of Sciences. pp. 236–238. (In Russian).
3. Garayeva, E.A. & Mansimov, K.B. (2014) On a discrete optimal control problem. *Vestnik BGU. Ser. fiz.-mat. Nauk – Baku University News Journal. Ser. Physics and Mathematics*. 1. pp. 15–21. (In Russian).
4. Moskalenko, A.I. (1969) Ob odnom klasse zadach optimal'nogo regulirovaniya [On a class of optimal control problems]. *Zhurnal vychislitel'noy matematiki i mat.-fiziki – Computational mathematics and Mathematical Physics*. 1. pp. 69–95.
5. Moskalenko, A.I. (1971) *Nekotorye voprosy teorii optimal'nogo upravleniya* [Some questions in the optimal control theory]. Abstract of Physics and Mathematics Cand. Diss. Tomsk.
6. Gabasov, R. & Kirillova, F.M. (1973) *Optimizatsiya lineynykh sistem* [Optimization of linear systems]. Minsk: Belarusian State University.
7. Mansimov, K.B. (2013) *Diskretnye sistemy* [Discrete Systems]. Baku: Baku State University.
8. Vasiliev, F.P. (1981) *Metody resheniya ekstremal'nykh zadach* [Methods for solving extreme problems]. Moscow: Nauka.
9. Demianov, V.F. (1974) *Minimaks: Differentsiruemost' po napravleniyam* [Minimax: differentiable in the direction]. Leningrad: Leningrad State University.
10. Demianov, V.F. et al. (1982) *Zadachi teorii optimizatsii i upravleniya* [The tasks of the theory of optimization and management]. Leningrad: Leningrad State University.
11. Demianov, V.F. & Rubinov, A.M. (1990) *Osnovy negladkogo analiza i kvazi-differentsial'noe ischislenie* [Fundamentals of non-smooth analysis and quasi-differential calculus]. Moscow: Nauka.

ОБРАБОТКА ИНФОРМАЦИИ

УДК 519.8

DOI: 10.17223/19988605/38/2

Д.П. Бураков, М.И. Гарина

ИССЛЕДОВАНИЕ СТРУКТУРЫ ПРЕДПОЧТЕНИЙ ЛПР С ИСПОЛЬЗОВАНИЕМ ТИПОВЫХ ОБОБЩАЮЩИХ ФУНКЦИЙ

Предлагаются алгоритм тестирования предпочтений ЛПР на соответствие ожидаемым им результатам выбора при использовании обобщающих функций и методы определения типа обобщающей функции и вектора её весовых коэффициентов, способных сделать выбранную ЛПР точку наилучшей. Рассматриваются аддитивная, мультипликативная и максиминная обобщающие функции.

Ключевые слова: обобщающая функция; многокритериальная теория полезности; предпочтения; Парето-доминирование.

Известно, что если структура предпочтений лица, принимающего решение (ЛПР), отвечает требованиям «рациональности», т.е. соответствует аксиоматике Эджворта–Парето, то альтернатива, выбираемая в задаче определения оптимальной на основании учета многих критериев, должна находиться в множестве Парето X_P , т.е. недоминируемых по всем критериям альтернатив [1]. Таким образом, если для альтернативы $a \in X$ имеется доминирующая ее по Парето-альтернатива $b \in X$ (т.е. выполняется $b \succ_p a$), то при любом способе упорядочения альтернатива b должна занять более высокое место, чем a .

Этап формулирования предпочтений ЛПР предшествует выбору оптимальной альтернативы через выбор соответствующего ей недоминируемого вектора $y(x)$. Поэтому полученные результаты выбора могут не соответствовать интуитивным ожиданиям, которыми ЛПР руководствовался при формулировании предпочтений. Предпочтения ЛПР, представленные отношением векторного доминирования, могут быть описаны одной из трех обобщающих функций F_O [3]: аддитивной АОФ, мультипликативной МОФ или максиминной функцией Гермейера мОФ [2], которые монотонны относительно этого отношения. Указанные функции используют векторный параметр $w > 0$, компоненты которого выражают относительную важность критериев с точки зрения ЛПР.

Поставим следующую задачу. Существует ли для векторной оценки $y(x)$, выбранной в множестве $Y = F(X)$, $Y \subset \mathbb{R}^n$, такая функция F_O и такой вектор весов w ($w > 0$, $\langle w, e \rangle = 1$, $w, e, 0 \in \mathbb{R}^n$, $e = (1, \dots, 1)$, $0 = (0, \dots, 0)$), которые доставили бы указанной точке наибольшее значение скалярной обобщающей функции F_O ? И если да, то какая это функция и каковы компоненты $w = (w_1, w_2, \dots, w_n)$.

1. Модель оценивания

В многокритериальных задачах рассматриваются альтернативы, характеризующиеся набором из n признаков, выражающихся функциями $f_1(x), \dots, f_n(x)$ так, что альтернативе x в соответствие ставится векторная оценка $y(x) = (f_1(x), f_2(x), \dots, f_n(x))$, а множеству альтернатив X – множество их векторных оценок Y (если функции $f_j(x)$ числовые, то $Y \subset \mathbb{R}^n$) [2]. На наборе признаков ЛПР определяет набор критериев вида $f_j(x) \rightarrow \text{extr}$, $j = 1, \dots, n$ (этого всегда можно добиться, введя соответствующие замены на множестве признаков и требований ЛПР). После этого отношение доминирования Парето индуцируется отношением покомпонентного векторного доминирования на Y так, что $\forall a, b \in X : y(a) \geq y(b) \Leftrightarrow a \succ_p b$, а множество недоминируемых векторов $Y_P \subseteq Y$ образует образ $X_P \subseteq X$.

Назовем *Парето-оболочкой* множества Y $\text{conv}_P(Y)$ множество векторных оценок из Y , которые образуют объединение множеств Парето, получаемых на Y при всех сочетаниях направлений оптимизации критериев. Точки, входящие в Парето-оболочку, принадлежат поверхности тела (необязательно выпуклого) минимального объема, заключающего в себе множество Y целиком. В простейшем случае (рис. 1) это тело может представлять собой невыпуклый многогранник, вершинами которого являются точки $\text{conv}_P(Y)$.

Ясно, что для любой внутренней точки $y \in Y \setminus \text{conv}_P(Y)$ при любом сочетании направлений оптимизации критериев найдется такая точка $y^* \in \text{conv}_P(Y)$, что $y^* \geq y$. Следовательно, точка y ни при каких условиях и способах упорядочения на данном наборе критериев не может получить первое место.

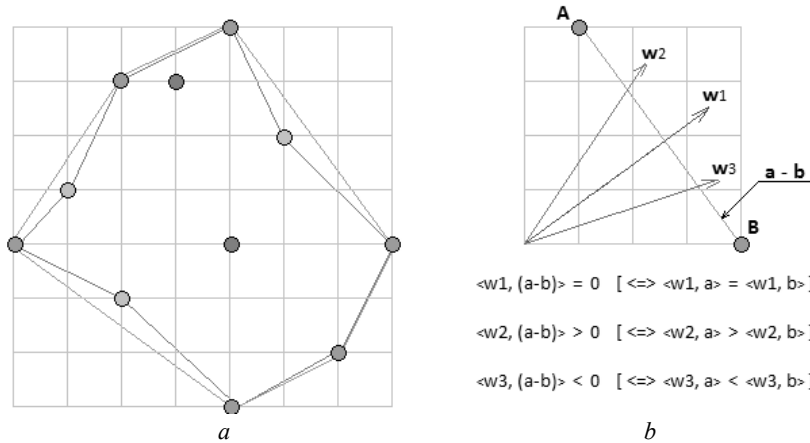


Рис. 1. Парето-оболочки: a – Парето-оболочка и линейная оболочка Y ; b – двумерный случай использования аддитивной ОФ

Предлагается алгоритм тестирования предпочтений ЛПР на соответствие ожидаемым им результатам выбора. Для этого ЛПР указывает произвольную альтернативу $x \in X$. Первый тест заключается в определении потенциальной оптимальности указанной альтернативы. Чтобы альтернатива была потенциально-оптимальной, она должна принадлежать $\text{conv}_P(Y)$ – *Парето-оболочке* множества Y , получаемой объединением 2^n множеств Парето Y_P для всех возможных сочетаний направлений оптимизации критериев. Если $y(x) \notin \text{conv}_P(Y)$, то ни при каком сочетании требований вида $f(x) \rightarrow \text{extr}$ альтернатива x не может являться оптимальной. Следовательно, если ЛПР предполагает, что эта альтернатива оптимальна, то ему следует добавить дополнительные признаки и критерии (так как $|Y_P|$ растет с увеличением n). Если $y(x) \in \text{conv}_P(Y)$, то имеется способ выбора указанной альтернативы x в качестве оптимальной, более того, среди указанных функций F_O найдется такая, для которой можно подобрать w , делающий $y(x)$ оптимальной.

2. Аддитивная обобщающая функция

В силу линейности аддитивной обобщающей функции (АОФ) ее поверхности безразличия (изокванты) представляют собой гиперплоскости в $\mathbf{R}^n$, подмножеством которого является Y . Поэтому АОФ может доставить наибольшее значение (при некотором конкретном векторе весов w) только тем точкам Y , которые входят в его *выпуклую линейную оболочку*, $\text{conv}_L(Y)$, т.е. являются вершинами выпуклого многогранника минимального объема, полностью включающего в себя Y (см. рис. 1, a). Для любой конфигурации множества Y выполняется $\text{conv}_L(Y) \subseteq \text{conv}_P(Y) \subseteq Y \subseteq \mathbf{R}^n$.

В этом смысле АОФ схожа с целевой функцией стандартной задачи линейного программирования, которая всегда в качестве оптимального решения выбирает одну из вершин выпуклого многогранника или в случае перпендикулярности одной из его граней – любую из точек, принадлежащих этой грани.

Рассмотрим двумерный случай. Пусть имеются две векторных оценки a и b и скалярная АОФ: $F_A(w; y) = \langle w, y \rangle$, где w – вектор весов, $w > 0$, $w \cdot e^T = 1$. Чтобы векторные оценки a и b имели одинако-

вую оценку АОФ, необходимо, чтобы градиент АОФ был перпендикулярен прямой, проходящей через точки $\mathbf{a}$ и $\mathbf{b}$. Так как градиент АОФ – вектор $\mathbf{w}$, то это равносильно условию $\langle \mathbf{w}, (\mathbf{a} - \mathbf{b}) \rangle = 0$ – скалярное произведение равно нулю. Это условие равносильно $\langle \mathbf{w}, \mathbf{a} \rangle = \langle \mathbf{w}, \mathbf{b} \rangle$. Заодно заметим, что по правилам матричного умножения $F_A(\mathbf{w}; \mathbf{a}) = \mathbf{w}^T \cdot \mathbf{a}$, а $F_A(\mathbf{w}; \mathbf{b}) = \mathbf{w}^T \cdot \mathbf{b}$. Следовательно, если требуется найти такой вектор $\mathbf{w}$, чтобы выполнялось $F_A(\mathbf{w}; \mathbf{a}) > F_A(\mathbf{w}; \mathbf{b})$, он должен удовлетворять системе

$$\begin{cases} \langle \mathbf{w}, \mathbf{a} \rangle > \langle \mathbf{w}, \mathbf{b} \rangle \\ \mathbf{w} > \mathbf{0} \\ \langle \mathbf{w}, \mathbf{e} \rangle = 1 \end{cases}, \text{ или, что равносильно: } \begin{cases} \langle \mathbf{w}, (\mathbf{a} - \mathbf{b}) \rangle > 0 \\ \mathbf{w} > \mathbf{0} \\ \langle \mathbf{w}, \mathbf{e} \rangle = 1. \end{cases}$$

Нарушение требования неотрицательности $\mathbf{w}$ сигнализирует, что для достижения векторной оценкой наибольшего значения АОФ соответствующий критерий должен минимизироваться (в исходной постановке задача неявно требует максимизации всех критериев). Иллюстрация для двумерного случая приведена на рис. 1, б.

В n -мерном случае любые n векторных оценок $\mathbf{a}_1, \dots, \mathbf{a}_n$ образуют в $\mathbf{R}^n$ гиперплоскость. Чтобы они имели одинаковые оценки АОФ $F_A(\mathbf{w}; \mathbf{a}_i) = \mathbf{w}^T \cdot \mathbf{a}_i$, требуется выполнение условия

$$\langle \mathbf{w}, (\mathbf{a}_i - \mathbf{a}_j) \rangle = 0, \forall j = 1, \dots, N, i \neq j.$$

Следовательно, чтобы некоторая оценка $\mathbf{a}_i$ в данной гиперплоскости имела наибольшую величину АОФ, необходимо найти вектор $\mathbf{w}$, удовлетворяющий условию

$$\langle \mathbf{w}, (\mathbf{a}_i - \mathbf{a}_j) \rangle > 0, \forall j = 1, \dots, N, i \neq j.$$

В матричной форме совокупность строк $\langle \mathbf{w}, (\mathbf{a}_i - \mathbf{a}_j) \rangle = 0$ имеет вид $\mathbf{w} \cdot \Delta \mathbf{A}^T = \mathbf{0}$, где $\Delta \mathbf{A}$ – матрица, строками которой являются векторы $(\mathbf{a}_i - \mathbf{a}_j)$. Если точка $\mathbf{a}$ – вершина многогранника, т.е. лежит на пересечении k граней, то для ее оптимальности достаточно взять в качестве $\mathbf{w}$ любой вектор, являющийся выпуклой комбинацией векторов $\mathbf{w}_{\perp 1}, \dots, \mathbf{w}_{\perp k}$ нормалей соответствующих граней.

3. Мультипликативная обобщающая функция

Если точка лежит на грани, но не является вершиной многогранника, АОФ сможет сделать альтернативу, соответствующую этой точке, оптимальной (при $\mathbf{w}$, являющемся нормалью к этой грани), но не сможет различить ее с другими точками этой грани. Для различения точек можно привлечь мультипликативную обобщающую функцию (МОФ). Например, на рис. 2, а, АОФ $F_A(\mathbf{w}; \mathbf{y})$ с вектором весов $\mathbf{w} = (1/3, 2/3)$ присвоит одинаковые оценки точкам A, B, C и D, но любое изменение весов сделает наилучшей по АОФ точку A или D; не удастся подобрать вектор $\mathbf{w}$, выбирающий B или C.

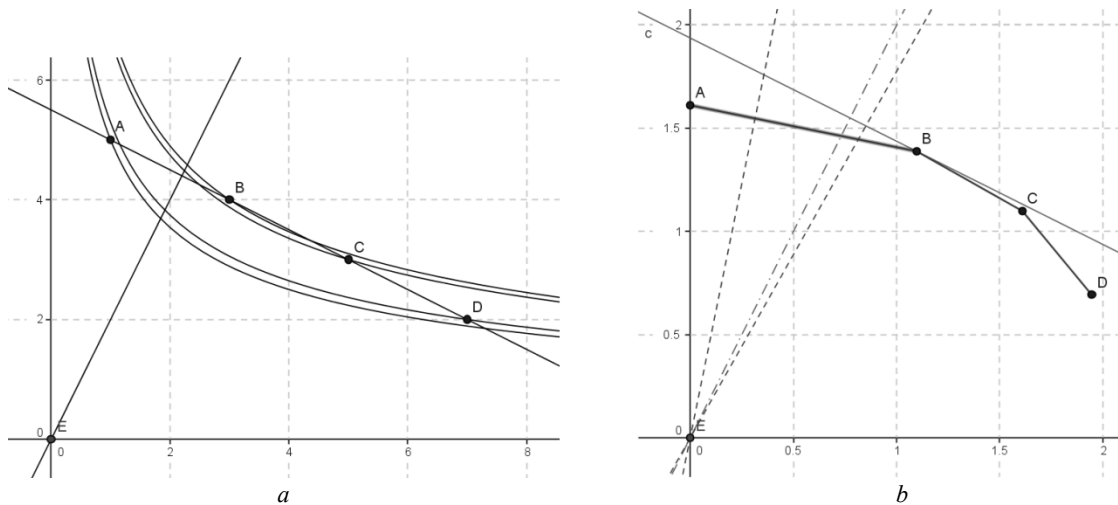


Рис. 2. Изокванта МОФ: а – различающие свойства МОФ; б – логарифмическое преобразование МОФ в АОФ

МОФ $F_M(\mathbf{w}; \mathbf{y}) = \prod_{j=1}^n y_j^{w_j}$ может различить точки внутри грани. Например, для того же вектора весов $\mathbf{w} = (1/3, 2/3)$ МОФ упорядочивает точки грани следующим образом: $B \succ C \succ D \succ A$. На рис. 2 в виде гипербол показаны изокванты МОФ для случая $\mathbf{R}^2$.

Задача поиска вектора $\mathbf{w}$, такого, чтобы МОФ $F_M(\mathbf{w}; \mathbf{y})$ в качестве наилучшей выбрала бы указанную точку $\mathbf{y}$ грани, сводится к уже рассмотренной выше задаче на поиск вектора $\mathbf{w}$, доставляющего наибольшее значение АОФ для указанной точки логарифмированием $F_M(\mathbf{w}; \mathbf{y})$:

$$\ln(F_M(\mathbf{w}; \mathbf{y})) = \ln\left(\prod_{j=1}^n y_j^{w_j}\right) = \sum_{j=1}^n w_j \cdot \ln(y_j) = \langle \mathbf{w}, \ln(\mathbf{y}) \rangle.$$

В множестве логарифмированных оценок $\ln(Y)$ точки A, B, C и D перестанут лежать на одной грани, а образуют выпуклую поверхность, как показано на рис. 2, *b*. Следовательно, можно решить задачу о подборе вектора весов, находящегося внутри конуса, образованного нормальными гранями, пересекающихся в указанной вершине (например, вектор $\mathbf{w} = (1/3, 2/3)$, доставляющий наибольшее на грани значение МОФ для точки B и изображенный штрих-пунктирной линией, находится внутри конуса, образованного нормальными для граней AB и BC , пересекающихся в точке B).

4. Максиминная обобщающая функция (mОФ)

МОФ не может ни при каком векторе $\mathbf{w}$ сделать лучшей точку, лежащую в области $\text{conv}_R(Y) \setminus \text{conv}_L(Y)$ ниже изокванты МОФ, проходящей через вершины соответствующей грани линейной оболочки $\text{conv}_L(Y)$. Однако это может сделать максиминная функция $F_m(\mathbf{w}, \mathbf{y}) = \min_{j=1, \dots, n} \{y_j \cdot w_j\}$, способная в качестве лучшего выбрать любой недоминируемый вектор – как находящийся выше, так и ниже максимальных на Y изоквант F_A и F_M [4]. Это связано с тем, что у этой ОФ самая нелинейная форма изокванты, представляющая собой поверхность прямоугольного конуса, направленного противоположно конусу доминирования точки. Наилучшее место получает точка, расположенная на наиболее удаленной от начала координат линии уровня.

Для определения вектора $\mathbf{w}$ можно провести вектор $\mathbf{w}'$ из начала координат непосредственно в интересующую точку. Он всегда будет внутри конуса предпочтения этой точки по mОФ. Назовем его вектором «квази»-весов, потому что для предпочтения выбранной точки по mОФ надо взять вектор $\mathbf{w}$, являющийся одной из перестановок $\mathbf{w}'$. Для двумерного случая $\mathbf{w}$ будет вектором $\mathbf{w}'$, отраженным относительно главной диагонали, а в случае размерности пространства $n > 2$ таких векторов потенциально оказывается $(n! - 1)$.

Заключение

Таким образом показано, что тип используемой для выбора оптимального решения функции определяется местоположением $\mathbf{y}(x)$ в оболочке Y : если $\mathbf{y}(x) \in \text{conv}_R(Y) \cap \text{conv}_L(Y)$, то можно использовать АОФ, если $\ln(\mathbf{y}(x)) \in \text{conv}_R(\ln(Y)) \cap \text{conv}_L(\ln(Y))$, то можно использовать МОФ, а в противном случае – максиминную функцию. При $\mathbf{y}(x) \in \text{conv}_R(Y)$ можно определить пару $\langle F_O, \mathbf{w} \rangle$, позволяющую сделать указанную ЛПР альтернативу x оптимальной. При этом ЛПР имеет возможность провести тест предпочтений с указанием своего вектора $\mathbf{w}$ или с указанным им порядком предпочтения на множестве критериев, порождающем семейство векторов $\{\mathbf{w}\}$, компоненты которых удовлетворяют заданному отношению порядка. В этом случае проверяется, могут ли указанные предпочтения быть описаны одной из трех типовых F_O . Если такое описание возможно, считается, что структура предпочтений ЛПР представима типовой обобщающей функцией. Если же подобное описание противоречит предпочтениям ЛПР, то он имеет возможность как скорректировать свои предпочтения, заменив $\mathbf{w}$ на автоматически определенный, так и использовать другой способ решения. В предельном случае с согласия ЛПР осуществляется автоматическая настройка предпочтений под требуемый результат, т.е. в результате указания оптимизируемой альтернативы x ЛПР получает автоматическую настройку задачи выбора, гарантирующую оптимальность указанной альтернативы. Предлагается следующий алгоритм исследования предпочтений ЛПР:

1. Поиск оболочки Парето и линейной выпуклой оболочки $\text{conv}_P(Y)$ и $\text{conv}_L(Y)$ для облака векторных оценок Y .
2. Проверка возможности сделать выбранную ЛПР точку наилучшей и при утвердительном ответе – поиск подходящей ОФ и вектора весовых коэффициентов:
 - а. Если точка входит в $Y \setminus \text{conv}_P(Y)$, она не может стать наилучшей.
 - б. Если точка входит в $\text{conv}_P(Y) \setminus \text{conv}_L(Y)$, она может стать наилучшей при использовании мОФ.
 - в. Если точка входит в $\text{conv}_L(Y)$ и является вершиной, она может стать наилучшей, причем имеется аналитический способ определения направлений оптимизации и вектора $\mathbf{w}$ для АОФ.
 - г. Если точка входит в $\text{conv}_L(Y)$, но не является вершиной, то сделать ее первой среди вершин ее грани сможет МОФ, причем имеется аналитический способ определения направлений оптимизации и вектора $\mathbf{w}$.

ЛИТЕРАТУРА

1. Ногин В.Д. Принятие решений в многокритериальной среде: количественный подход. М. : Физматлит, 2005. 176 с.
2. Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач. М. : Физматлит, 2007. 256 с.
3. Микони С.В. Теория принятия управленческих решений. СПб. : Лань, 2015. 448 с.
4. Ногин В.Д. Границы применимости распространенных методов скаляризации при решении задач многокритериального выбора // Методы возмущений в гомологической алгебре и динамика систем : межвуз. сб. науч. тр. Саранск : Изд-во Мордов. ун-та, 2004. С. 59–68.

Бураков Дмитрий Петрович, канд. техн. наук. E-mail: bds@yandex.ru

Гарина Марина Игоревна, канд. техн. наук. E-mail: migarina@gmail.com

Петербургский государственный университет путей сообщения Императора Александра I (г. Санкт-Петербург)

Поступила в редакцию 14 марта 2016 г.

Burakov Dmitry P., Garina Marina I. (Saint-Petersburg State Transport University, Russian Federation).

Analysis of decision maker's preferences structure with standard aggregative functions.

Keywords: aggregative function; multicriteria utility theory; preferences; Pareto dominance.

DOI: 10.17223/19988605/38/2

If decision maker preferences structure is under the conditions of rationality (axioms of Edgeworth-Pareto), then any object determined as optimal with any multicriteria aggregative function must be in Pareto set X_P , e.g. set of non-dominated. All objects in multicriteria tasks are characterized by n features, which are evaluated by functions $f_1(x), \dots, f_n(x)$. E.g. an object x has an assigned vector $\mathbf{y}(x) = (f_1(x), f_2(x), \dots, f_n(x))$, and a set of objects X has an image – a set of vectors Y (if all functions $f_i(x)$ are numerical, then $Y \subset \mathbf{R}^n$). The decision maker forms for every feature a criteria $f_j(x) \rightarrow \text{extr}, j = 1, \dots, n$. At that, the component-wise vector domination relation in Y induces the Pareto dominance relation, and the set of non-dominated vectors $Y_P \subseteq Y$ determines $X_P \subseteq X$. Because the step of formalization of preferences precedes the step of choosing of an optimal object as one of non-dominated vectors $\mathbf{y}(x)$, the choice results may differ from decision maker's intuitive expectations.

We consider an algorithm of testing the decision maker preferences for compliance with his expectation. The decision maker chooses any object $x \in X$. The first test is about if this object can be optimal. The object can be optimal if it belongs to $\text{conv}_P(Y)$ – Pareto's envelope of the set Y . This envelope is a union of 2^n Pareto sets Y_P for all possible combinations of criteria directions. If $\mathbf{y}(x) \notin \text{conv}_P(Y)$, then object cannot be optimal with any combination of conditions $f_j(x) \rightarrow \text{extr}$. Therefore, if the decision maker wants it to be optimal, he must add some more features and form the addition criteria for them (because $|Y_P|$ increases with n).

If $\mathbf{y}(x) \in \text{conv}_P(Y)$, the conditions with which the object x can be optimal, are exist. Any of three aggregative functions F_O can describe the decision maker's preferences expressed as the vector dominance relation: additive function, multiplicative function or maxmin Germeyer function. All of them are monotone in respect of this relation. They use a vector parameter $\mathbf{w} > \mathbf{0}$, which components fix the relative criteria importance for the decision maker. If $\mathbf{y}(x) \in \text{conv}_P(Y)$, then one of these functions can be used with some vector $\mathbf{w}$ to choose $\mathbf{y}(x)$ as optimal, and we can calculate that vector $\mathbf{w}$. The type of function is determined by the $\mathbf{y}(x)$ location in the envelope of Y : if $\mathbf{y}(x) \in \text{conv}_P(Y) \cap \text{conv}_L(Y)$, then we can use the additive function, if $\ln(\mathbf{y}(x)) \in \text{conv}_P(\ln(Y)) \cap \text{conv}_L(\ln(Y))$, then we can use the multiplicative function, and otherwise – maxmin function. Here $\text{conv}_L(Y)$ is the linear convex envelope of Y . So, if $\mathbf{y}(x) \in \text{conv}_P(Y)$, our second test makes for decision maker a pair $\langle F_O, \mathbf{w} \rangle$ to choose the specified object x as optimal.

The decision maker may perform the preference test for concrete vector $\mathbf{w}$, or for concrete order of preference of criteria. There is the large set of $\{\mathbf{w}\}$, components of which satisfy a given order. In this case, we must test if there is a standard aggregative function F_O that can describe the given preferences. If yes, we declare that the decision maker's preference structure can be represented by this standard function. Otherwise, the decision maker can change his preferences by using the calculated vector $\mathbf{w}$ instead his own or another decision method. If the decision maker agrees, then the automatic preferences configuration, which will provide the desired result, is possible.

REFERENCES

1. Nogin, V.D. (2005) *Prinyatie resheniy v mnogokriterial'noy srede: kolichestvennyy podkhod* [The multicriteria decision making: quantitative approach]. Moscow: Fizmatlit.
2. Podinovskiy, V.V. & Nogin, V.D. (2007) *Pareto-optimal'nye resheniya mnogokriterial'nykh zadach* [Pareto-optimal decisions of multicriteria tasks]. Moscow: Fizmatlit.
3. Mikoni, S.V. (2015) *Teoriya prinyatiya upravlencheskikh resheniy* [The managerial decision making theory]. St. Petersburg: Lan'.
4. Nogin, V.D. (2004) Granitsy primenimosti rasprostranennykh metodov skalyarizatsii pri reshenii zadach mnogokriterial'nogo vybora [The scope of applicability of common methods of scalarization in solving problems of multicriteria choice]. In: Shchennikov, V.N. (ed.) *Metody vozmushcheniy v gomologicheskoy algebre i dinamika sistem* [Methods of disturbance in homological algebra and systems dynamics]. Saransk: Ogarev Mordovia State University. pp. 59–68.

УДК 519.2

DOI: 10.17223/19988605/38/3

T.V. Dogadova, V.A. Vasiliev

ADAPTIVE PREDICTION OF STOCHASTIC DIFFERENTIAL EQUATIONS WITH UNKNOWN PARAMETERS

This study (research grant N8.1.55.2015) was supported by The Tomsk State University Academic D.I. Mendeleev Fund Program in 2015-2016 and by RFBR Grant 16-01-00121 A.

This paper proposes adaptive predictors of continuous-time dynamic systems with unknown parameters. Predictors are based on the truncated parameter estimators. In particular, there are considered the Ornstein-Uhlenbeck process and one-parameter stochastic delay differential equation. In this paper the truncated estimation method is first applied to continuous-time systems. Asymptotic and non-asymptotic properties of the predictors are investigated. There is also found the rate of convergence of the second moment of a prediction error to its minimum value.

Keywords: truncated estimation; continuous-time dynamic systems; prediction; delay differential equations; Ornstein-Uhlenbeck process.

Prediction is a momentous problem in modern world. Distinctive fields of science, for instance, economics, financial mathematics, engineering, biology etc. appeal to mathematical tools to get predictors of the real dynamic processes they explore. If a model allows making predictions of high statistical quality it is considered to be beneficial. Since models of dynamic systems often have unknown parameters, we have to deal with estimation problem in order to construct adaptive predictors. The quality of adaptive prediction significantly depends on a choice of estimators of the model parameters. Adaptive prediction problem for discrete-time systems was solved in [1] on the basis of truncated estimators proposed in [2]. In this paper we present analytically investigated predictors of continuous-time systems constructed on the basis of truncated estimators of dynamic parameters.

1. Prediction of Ornstein-Uhlenbeck process

Assume the model

$$dx_t = ax_t dt + dw_t, \quad t \geq 0, \quad (1)$$

with an unknown parameter a , where x_0 is zero mean random variable having finite moments of all order, w_t is the standard Wiener process, x_0 and w_t are mutually independent. Suppose that the process (1) is stable, i.e. the parameter $a < 0$. Note that in this case for every $m \geq 1$

$$\sup_{t \geq 0} E x_t^{2m} < \infty. \quad (2)$$

The purpose is to construct a predictor for x_t by observations $x^{t-u} = (x_s)_{0 \leq s \leq t-u}$ which is optimal in a sense of the risk function introduced below. Here $u > 0$ is a fixed time delay.

Using the solution of the process x_t for given $u > 0$ we have the representation

$$x_t = \lambda x_{t-u} + \xi_{t,t-u}, \quad t \geq u,$$

where $\lambda = e^{au}$, $\xi_{t,t-u} = \int_{t-u}^t e^{a(t-s)} dw_s$, $E \xi_{t,t-u} = 0$ and $\sigma^2 := E \xi_{t,t-u}^2 = \frac{1}{2a} [e^{2au} - 1]$.

Optimal in the mean square sense predictor is the conditional mathematical expectation

$$x_t^0 = \lambda x_{t-u}, \quad t > u.$$

As in practice the parameter a and, as follows, λ are unknown, it is impossible to construct the optimal predictor for real processes. In order to solve the problem of prediction we define an adaptive predictor that is constructed by estimator a_t of unknown parameter a .

Define adaptive predictor as

$$\hat{x}_t(t-u) = \hat{\lambda}_{t-u} x_{t-u}, \quad t > u, \quad (3)$$

where $\hat{\lambda}_{t-u} = e^{\hat{a}_{t-u}u}$, $t > u$; $\hat{a}_t = \text{proj}_{(-\infty, 0]} a_t$, a_t is the truncated estimator of the parameter a constructed similar to discrete-time case [2] on the basis of the least squares estimator

$$a_t = \frac{\int_0^t x_v dx_v}{\int_0^t x_v^2 dv} \chi \left(\int_0^t x_v^2 dv \geq t \log^{-1} t \right). \quad (4)$$

Lemma. Assume the model (1). Then for $t-u > s_0 := \exp(2|a|)$ and some numbers C_p estimators $\hat{a}_t$ and $\hat{\lambda}_t$ have the properties:

$$E(\hat{a}_t - a)^{2p} \leq \frac{C_p}{t^p} \quad (5)$$

and

$$E(\hat{\lambda}_t - \lambda)^{2p} \leq \frac{C_p}{t^p}, \quad p \geq 1. \quad (6)$$

Proof of Lemma. We prove the property (5) similar to [2]. By the definition (4) of the estimator a_t and using (1) let us find representation for the deviation of the estimator

$$a_t - a = \frac{\int_0^t x_v dw_v}{\int_0^t x_v^2 dv} \chi \left(\int_0^t x_v^2 dv \geq t \log^{-1} t \right) - a \cdot \chi \left(\int_0^t x_v^2 dv < t \log^{-1} t \right).$$

Define

$$g_t = \frac{1}{t} \int_0^t x_v^2 dv, \quad g = -\frac{1}{2a}, \quad f_t = \frac{1}{t} \int_0^t x_v dw_v.$$

Then

$$E(a_t - a)^{2p} = E \left[\frac{f_t}{g_t} \right]^{2p} \cdot \chi[g_t \geq \log^{-1} t] + a^{2p} \cdot P[g_t < \log^{-1} t] =: I_1 + I_2. \quad (7)$$

In what follows, C will denote a generic non-negative constant whose value is not critical (and not always the same).

Using the Cauchy-Schwarz-Bunyakovsky inequality for the first summand we get:

$$\begin{aligned} I_1 &= E \left[\frac{f_t}{g} + f_t \frac{g - g_t}{gg_t} \right]^{2p} \cdot \chi[g_t \geq \log^{-1} t] = \frac{E f_t^{2p}}{g^{2p}} \cdot \left(1 + \sum_{k=1}^{2p} C_{2p}^k \cdot \frac{(g - g_t)^k}{g_t^k} \right) \cdot \chi[g_t \geq \log^{-1} t] \leq \\ &\leq \frac{C}{t^p} + C \cdot \log t \cdot E f_t^{2p} \cdot |g - g_t| \leq \frac{C}{t^p} + C \cdot \log t \cdot (E f_t^{4p} \cdot E(g - g_t)^2)^{\frac{1}{2}} \leq \\ &\leq \frac{C}{t^p} + C \cdot \log t \cdot \frac{1}{t^p} \cdot (E(g - g_t)^2)^{\frac{1}{2}}. \end{aligned}$$

By the Ito formula for x_t^2 it is true that

$$dx_t^2 = 2ax_t^2 dt + 2x_t dw_t + dt$$

and then

$$g_t - g = \frac{1}{2a} \frac{x_t^2}{t} - \frac{1}{at} \int_0^t x_t dw_t.$$

Thus, by making use of the properties of the Ito integral and (2) for every $m \geq 1$ it holds

$$E(g_t - g)^{2m} \leq \frac{C}{t^m}. \quad (8)$$

Then

$$I_1 \leq \frac{C}{t^p} + C \frac{\log t}{t^{p+1/2}}, \quad t > u, \quad (9)$$

and applying (8), as well as the Chebyshev inequality for $t > s_0$ we have

$$I_2 \leq a^{2p} \cdot P(|g_t - g| \geq g - \log^{-1} t) \leq \frac{a^{2p}}{(g - \log^{-1} t)^{4p}} \cdot E(g_t - g)^{4p} \leq \frac{C}{t^{2p}}. \quad (10)$$

From (7), (9), (10) and definition of $\hat{a}_t$ the property (5) follows.

The assertion (6) follows from the obvious inequality $|\hat{\lambda}_t - \lambda| \leq u|\hat{a}_t - a|$, which can be obtained by the Taylor expansion for the exponent $\exp((\hat{a}_t - a)u)$. Lemma is proven.

Now we are ready to investigate the statistical properties of the adaptive predictor (3). The prediction error has the form

$$e_t(t-u) := x_t - \hat{x}_t(t-u) = (\lambda - \hat{\lambda}_{t-u})x_{t-u} + \xi_{t,t-u}.$$

Thus, for some C

$$\lim_{t \rightarrow \infty} t \cdot (Ee_t^2(t-u) - \sigma^2) \leq C$$

and if there is a priori information that $|a| \leq L$ then

$$Ee_t^2(t-u) - \sigma^2 \leq \frac{C}{t}, \quad t > u + \exp(2L).$$

2. Delay differential equation

Assume $w = (w_t, t \geq 0)$ is a real-valued standard Wiener process, a and b are real numbers and $x = (x_t, t \geq -r)$ is a solution of the stochastic delay differential equation

$$dx_t = ax_t dt + bx_{t-r} dt + dw_t, \quad t \geq 0, \quad (11)$$

with some fixed initial condition $x_t = X_0(t), t \in [-r, 0]$, where $X_0(\cdot)$ is a continuous stochastic process independent of $w(\cdot)$.

The solution x of (11) exists, it is pathwise uniquely determined and can be represented as (see, e.g., [3, 4]):

$$x_t = x_0(t)X_0(0) + b \int_{-r}^0 x_0(t-s-r)X_0(s)ds + \int_0^t x_0(t-s)dw_s, \quad t \geq 0.$$

Obviously, it has continuous paths for $t \geq 0$ with probability one and, conditionally on $X_0(\cdot)$, x is a Gaussian zero mean process. Here $(x_0(t), t \geq -r)$ denotes the so-called fundamental solution of the deterministic equation

$$\dot{x}(t) = ax_0(t) + bx_0(t-r), \quad x_0(0) = 1 \text{ and } x_0(t) = 0, t \in [-r, 0].$$

Assume in the next section that the solution x of (11) is being observed, the parameter $a = 0$ and the parameter b is unknown and has to be estimated.

Estimation problems for stochastic differential equations with time delay have been considered in a few papers up to now – see, e.g., [3; 5–7] and the references therein.

2.1. One-parameter delay differential equation

In this section we consider the differential equation with time-delay of a special structure. Assume the model

$$dx_t = bx_{t-r}dt + dw_t, \quad t \geq 0, \quad (12)$$

with an unknown parameter b such that the process (12) is stable. Note it is stable when the parameter $b \in (-\pi/2, 0)$, see [3]. Now we construct optimal and adaptive predictors for the process (12). Optimal in the mean square sense predictor is the conditional mathematical expectation

$$z_t^{(k)}(t-u) = E(x_t | x_{t-u})$$

that satisfies the following equation

$$\begin{aligned} z_t^{(k)}(t-u) = & x_{t-u} + b \int_{t-u}^{[t-(u-r)] \wedge t} x_{v-r} dv + b \int_{[t-(u-kr)] \wedge t}^t z_{v-kr}^{(0)}(t-u) dv + \\ & + b \sum_{i=1}^{k-1} \int_{t-r}^t z_{v-(k-i)r}^{(i)}(t-u) dv, \quad kr < u \leq (k+1)r, \quad t > u. \end{aligned} \quad (13)$$

Here $\alpha \wedge \beta$ means the minimum between α and β .

Since the parameter b in the definition of the optimal predictors $z_t^{(k)}(t-u)$ is unknown, we define the adaptive predictor by formula (13) replacing the unknown b with $\hat{b}_{t-u}$, where $\hat{b}_{t-u}$ is the projection

$$\hat{b}_{t-u} = \text{proj}_{[-\pi/2, 0]} b_{t-u} \quad (14)$$

of the truncated estimator of the parameter b which is defined as follows

$$b_t = \frac{\int_r^t x_{v-r} dx_v}{\int_r^t x_{v-r}^2 dv} \chi \left(\int_r^t x_{v-r}^2 dv \geq t \log^{-1} t \right), \quad t > \max(u, r).$$

Define the numbers $\sigma_0^2 = \int_0^\infty x_0^2(v) dv$ and $s_0 = \max \{ r, \exp(\sigma_0^{-2}) \}$, where $x_0(\cdot)$ is the solution of the characteristic equation $\dot{x}_0(v) = bx_0(v-r)$, $v \geq 0$ of the process (12) with $x_0(0) = 1$ and $x_0(v) = 0$, when $v \in [-r, 0)$.

For $t > u + s_0$ estimators (14) have the properties

$$E(\hat{b}_t - b)^{2p} \leq \frac{C_p}{t^p}, \quad p \geq 1,$$

which can be proven similarly to Lemma.

Denote the adaptive prediction error and rewrite it in the form

$$e_t^{(k)}(t-u) := x_t - \hat{z}_t^{(k)}(t-u) = e_t^0(t-u) + \hat{e}_t^{(k)}(t-u),$$

where $e_t^0(t-u) = x_t - E(x_t | x_{t-u})$ and $\hat{e}_t^{(k)}(t-u) = z_t^{(k)}(t-u) - \hat{z}_t^{(k)}(t-u)$. Then for every fixed $k \geq 0$ the following limit inequalities hold

$$\overline{\lim}_{t \rightarrow \infty} t \cdot \left(E(e_t^{(k)}(t-u))^2 - \sigma_0^2 \right) \leq C.$$

If it is known that $b \in [b_0, b_1]$, $-\pi/2 < b_0 < b_1 < 0$, then for $t-u > s_1 = \max \{ r, \exp(\sigma_1^{-2}) \}$, where $\sigma_1^2 = \inf_{b \in [b_0, b_1]} \sigma_0^2$ the non-asymptotic property is fulfilled

$$E(e_t^{(k)}(t-u))^2 - \sigma_0^2 \leq \frac{C}{t}.$$

3. Simulations

To confirm the convergence of the truncated estimators (4) and the properties of predictors (3) constructed on the basis of these estimators we made the simulations.

For this purpose we used the software package MATLAB. In Tables I, II the average

$$a_T = \frac{1}{100} \cdot \sum_{k=1}^{100} a_T(k)$$

of estimators $a_T(k)$ constructed according to (4) by the k -th realization $x^{(k)} = (x_t^{(k)})$, $k = 1, \dots, 100$ of the process (1) and its quality characteristics (empirical mean square error):

$$S_a^2(T) = \frac{1}{100} \sum_{k=1}^{100} (a_T(k) - a)^2$$

for different durations of observations T are given. Define the discretization step h and the number of discrete observations $N = h^{-1}T$.

Table 1

Estimators a_T and their quality characteristics $S_a^2(T)$ for $h = 0,1$

A	$N = 1000, T = 100$		$N = 2000, T = 200$		$N = 5000, T = 500$	
	a_T	$S_a^2(T)$	a_T	$S_a^2(T)$	a_T	$S_a^2(T)$
-0,3	-0,3135	0,0065	-0,3051	0,0030	-0,3070	0,0011
-0,5	-0,5113	0,0082	-0,5180	0,0059	-0,4969	0,0017
-0,8	-0,8283	0,0154	-0,8034	0,0074	-0,7949	0,0033
-1	-1,0126	0,0186	-0,9979	0,0111	-1,0017	0,0051

Table 2

Estimators a_T and their quality characteristics $S_a^2(T)$ for $h = 0,15$

A	$N = 1000, T = 150$		$N = 2000, T = 300$		$N = 5000, T = 750$	
	a_T	$S_a^2(T)$	a_T	$S_a^2(T)$	a_T	$S_a^2(T)$
-0,3	-0,3054	0,0050	-0,3098	0,0019	-0,3042	0,0001
-0,5	-0,5107	0,0053	-0,5081	0,0036	-0,5050	0,0014
-0,8	-0,8263	0,0124	-0,8114	0,0042	-0,8041	0,0020
-1	-1,0146	0,0111	-1,0165	0,0061	-1,0122	0,0028

According to the simulation results given in tables 1, 2 we can say that the empirical mean square error (MSE) $S_a^2(T)$ of the estimators a_T becomes less with growth of the sample size (see fig. 1 as well). It means that the estimator's value becomes closer to the true value of the parameter. This fact proves the effectiveness of the presented estimation procedure.

There were also constructed predictors of the observed process by the obtained estimators of $a = -0,5$ with $u = 1, h = 0,1$:

$$\hat{x}_t(t-1) = e^{\hat{a}_{t-1}} x_{t-1}, \quad t > 1.$$

The results are presented on the graphs below. The solid one identifies the real values of the observed process, the stipple one shows the predictor's value (see fig. 2).

For the fig. 3 there was computed the empirical mean square prediction error that is equal to

$$S_x^2(l) = \frac{1}{100} \sum_{k=1}^{100} \left(x_{lh}^{(k)} - \hat{x}_{lh}^{(k)}(lh-1) \right)^2, \quad l = \overline{10, N}$$

with $T = 100, N = 1000, u = 1, h = 0,1$. It shows that the prediction error converges to the value $\sigma^2 = 1 - e^{-1} \approx 0,6321$ which corresponds to the obtained theoretical results.

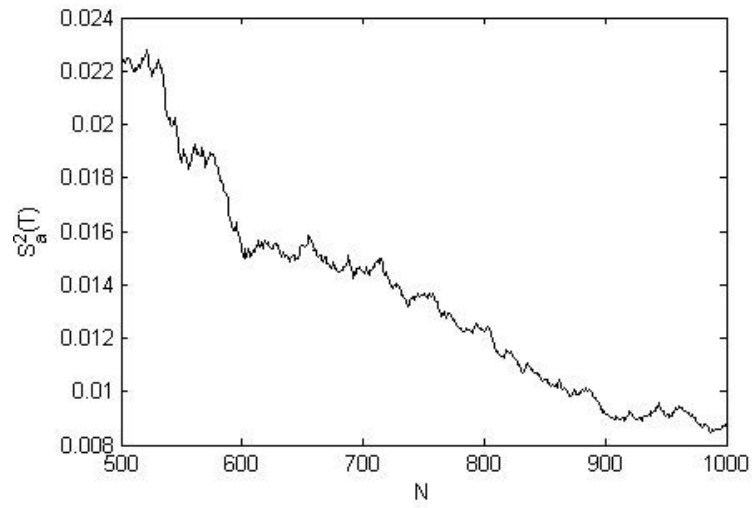


Fig. 1. Empirical MSE $S_a^2(T)$ for $a = -0,5; T = 100; h = 0,1; N = 1000$

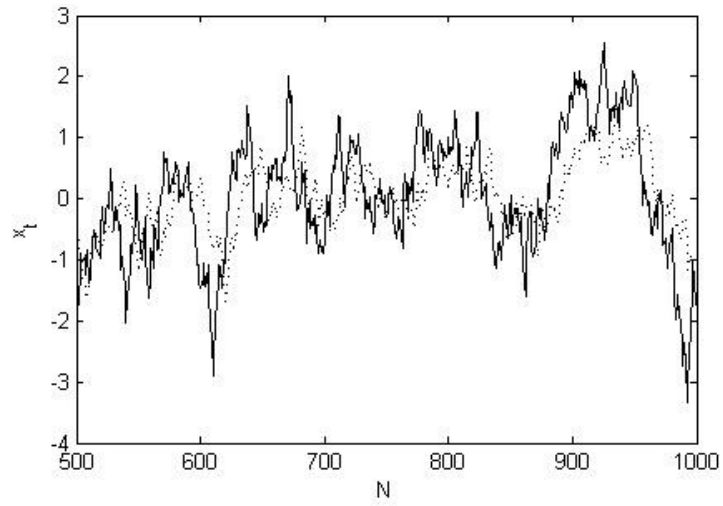


Fig. 2. Process x_t (—) and adaptive predictor $\hat{x}_t(t-1)$ (· · ·)

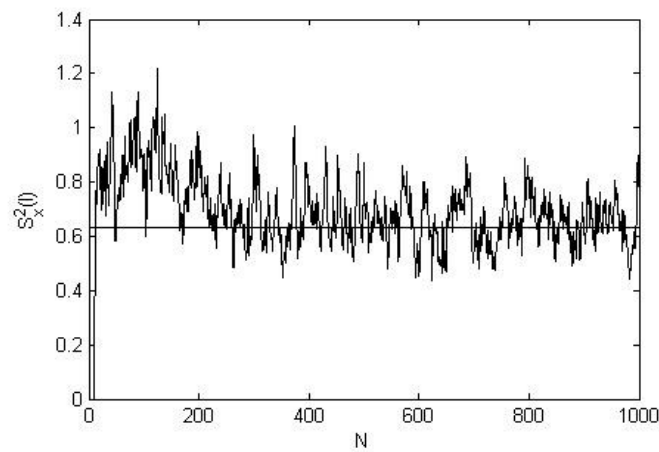


Fig. 3. Empirical MSE $S_x^2(t)$

Conclusion

Adaptive prediction problem of the Ornstein-Uhlenbeck process and one-parameter linear stochastic delay differential equation is solved. Non-asymptotic upper boundary of the prediction error is found. It is shown

that this boundary is inverse proportional to the duration of observations. Non-asymptotic properties of adaptive predictors are obtained due to the usage of the truncated estimators [2] of the unknown parameters constructed by samples of fixed size. This method can be applied to various problems of parametric and non-parametric statistics. In this paper it is first employed to continuous-time systems. The obtained results can be used for the optimization problem of proposed predictors in the sense of special risk functions similar to discrete-time models [1].

REFERENCES

1. Kusainov, M.I. & Vasiliev, V.A. (2015) On optimal adaptive prediction of multivariate autoregression. *Sequential Analysis: Design Methods and Applications*. 34(2). pp. 211–234. DOI: 10.1080/07474946.2015.1030977
2. Vasiliev, V.A. (2014) A Truncated Estimation Method with Guaranteed Accuracy. *Annals of Institute of Statistical Mathematics*. 66. pp. 141–163. DOI: 10.1007/s10463-013-0409-x
3. Guschin, A.A. & Kuchler, U. (1999) Asymptotic Inference for a Linear Stochastic Differential Equation with Time Delay. *Bernoulli*. 5(6). pp. 1059–1098.
4. Myschkis, A.D. (1972) Linear Differential Equations with Delayed Argument. Moscow: Nauka. (In Russian).
5. Kuchler, U. & Vasiliev, V.A. (2001) On Sequential Parameter Estimation for Some Linear Stochastic Differential Equations with Time Delay. *Sequential Analysis*. 20. pp. 117–146. DOI: 10.1081/SQA-100106052
6. Kuchler, U. & Vasiliev, V.A. (2010) On guaranteed parameter estimation of a multiparameter linear regression process. *Automatica, Journal of IFAC, Elsevier*. 46(4). pp. 637–646. DOI: 10.1016/j.automatica.2010.01.003
7. Kuchler, U. & Kutoyants, Yu. A. (2000) Delay estimation for some stationary diffusion-type processes. *Scandinavian Journal of Statistics*. 27(3). pp. 405–414. DOI: 10.1111/1467-9469.00197

Dogadova Tatiana Valerievna. E-mail: aurora1900@mail.ru

Vasiliev Vyacheslav Arturovich, Dr. Science, prof. E-mail: vas@mail.tsu.ru

National Research Tomsk State University, Russian Federation

Поступила в редакцию 3 августа 2016 г.

Догадова Татьяна Валерьевна, Васильев Вячеслав Артурович (Национальный исследовательский Томский государственный университет, Российская Федерация).

Адаптивное прогнозирование стохастических систем с непрерывным временем.

Ключевые слова: адаптивные прогнозы; усеченное оценивание; системы с непрерывным временем, дифференциальные уравнения с запаздыванием; процесс Орнштейна–Уленбека.

DOI: 10.17223/19988605/38/3

В данной работе представлены адаптивные прогнозы для динамических систем с непрерывным временем и неизвестными параметрами динамики. Прогнозы строятся на основе усеченных оценок параметров. В частности, рассматриваются процесс Орнштейна–Уленбека и однопараметрическое стохастическое дифференциальное уравнение с запаздыванием. В статье усеченный метод оценивания впервые применяется в системах с непрерывным временем. Исследованы асимптотические и неасимптотические свойства прогнозов. Найдена скорость сходимости второго момента ошибки прогноза к ее минимальному значению.

УДК 336:51

DOI: 10.17223/19988605/38/4

Г.А. Медведев

ОБ ОДНОЙ КВАДРАТИЧНОЙ МОДЕЛИ ВРЕМЕННОЙ СТРУКТУРЫ ДОХОДНОСТИ

Рассмотрена квадратичная модель временной структуры процентных ставок доходности бескупонной облигации, когда краткосрочная ставка имеет квадратичную зависимость от переменных состояния. Анализ проводится для случая, когда переменные состояния образуют вектор с независимыми компонентами, а временная структура определяется при нейтральной к риску вероятностной мере. Показано, что в этом случае процесс краткосрочной ставки имеет распределение гамма, такое же как для аффинной модели Даффи–Кана. Проводится сравнение временных структур доходности этой аффинной модели с квадратичной моделью доходности. Показано, что предельные (краткосрочные и долгосрочные) доходности этих моделей полностью совпадают, хотя сами временные структуры различаются. Показано, что на форму рассмотренной квадратичной временной структуры значения компонент вектора переменных состояний не влияют, она зависит только от величины стартовой процентной ставки. Сравнительные свойства аффинной модели Даффи–Кана и квадратичной модели доходности иллюстрируются численным примером.

Ключевые слова: временные структуры доходности; квадратичные модели; нейтральная к риску вероятностная мера; аффинная модель Даффи–Кана.

В рамках теории диффузионных процессов существуют разнообразные версии изменения краткосрочных процентных ставок доходности. Тем не менее до сих пор не появилось такой модели, которая смогла бы быть подходящей основой для построения временной структуры доходности, близкой к существующей на реальном финансовом рынке. Наиболее известны модели процентных ставок, приводящие к аффинным временным структурам доходности, поскольку они просты и подразумевают решение в аналитическом виде. Однако воспроизведения реальных временных структур с помощью аффинных моделей неточны. В последнее время развитие моделей идет в двух направлениях: увеличение размерности моделей и отказ от аффинных свойств. В качестве представителей такого развития наиболее популярны сейчас так называемые квадратичные модели процессов процентных ставок [1], в которых процесс процентной ставки $r(t)$ задается уравнениями

$$\begin{aligned} dX(t) &= \xi(X(t)) dt + \sigma(X(t)) dW(t), \quad t > t_0, \quad X(t_0) = X_0, \\ r(t) &= \alpha + X(t)^T \Psi X(t), \quad X(t) \in \mathbb{R}^n, \quad \alpha \in \mathbb{R}^1, \quad \Psi \in \mathbb{R}^{n \times n}. \end{aligned}$$

Обычно $\alpha \geq 0$, Ψ – симметрическая положительно определенная матрица. Когда вектор $\xi(X)$ линейно зависит от X , а матрица $\sigma(X)$ не зависит от X , процесс $X(t)$ является гауссовым и в стационарном режиме имеет, скажем, математическое ожидание μ и матрицу ковариации V .

Утверждение. При принятых условиях производящая функция моментов процесса процентной ставки $r(t)$ определяется соотношением

$$M(z) = e^{\alpha z - \mu^T V^{-1} \mu / 2 + \mu^T (V - 2z V \Psi V)^{-1} \mu / 2} \frac{|V|^{1/2}}{|V - 2z V \Psi V|^{1/2}},$$

где $|V| = \det V$, z – вещественная переменная с областью определения, задаваемой неравенством

$$|V - 2z V \Psi V| > 0.$$

Следствие 1. Стационарное математическое ожидание процесса $r(t)$

$$E[r] = \alpha + \mu^T \Psi \mu + V \circ \Psi,$$

где $\circ$ – сумма произведений соответствующих элементов матриц, $V \circ \Psi = \sum_{i=1}^n \sum_{j=1}^n V_{ij} \Psi_{ij}$.

Следствие 2. Пусть Ψ и V – диагональные матрицы $\Psi = \psi I$, $V = \nu I$. В этом случае

$$M(z) = e^{\alpha z - \mu^T \mu z \psi / (1 - 2z \nu \psi)} \frac{1}{(1 - 2z \nu \psi)^{n/2}},$$

а основные стационарные моменты процесса $r(t)$ вычисляются по формулам

$$E[r] = \alpha + n \nu \psi + \mu^T \mu \psi, \text{ Var}[r] = 2\nu (n \nu + 2\mu) \psi^2, \\ E[(r - E[r])^3] = 8\nu^2 (n \nu + 3\mu) \psi^3.$$

Следствие 3. Если условия следствия 2 дополнить равенством $\mu = 0$, то маргинальным распределением процесса $r(t)$ будет сдвинутое распределение гамма с параметром сдвига α , параметром масштаба $1/2\nu\psi$ и параметром формы $n/2$.

Заметим, что сдвинутое распределение гамма характеризует также краткосрочную процентную ставку в модели Даффи–Кана, где процесс $r(t)$ следует стохастическому дифференциальному уравнению [2]

$$dr(t) = k(\theta - r(t))dt + \sqrt{2kD \frac{r(t) - \alpha}{\theta - \alpha}} dW(t). \quad (1)$$

Таким образом, модель Даффи–Кана (1) и квадратичная модель

$$dX(t) = -K X(t) dt + \Sigma dW(t), t > t_0, X(t_0) = X_0, \\ r(t) = \alpha + X(t)^T \Psi X(t), X(t) \in \mathbb{R}^n, \alpha \in \mathbb{R}^1, \Psi \in \mathbb{R}^{n \times n}, \quad (2)$$

порождают случайные процессы $r(t)$ с одним и тем же распределением, когда условия следствий 2 и 3 выполняются, т.е. $K = kI$, Σ – диагональная матрица с элементами $\sqrt{2k\nu}$, $\Psi = \psi I$. При этом параметры уравнения (1) должны быть согласованы с параметрами модели (2) следующими равенствами:

$$\theta = \alpha + n \nu \psi, D = 2 n \nu^2 \psi^2.$$

Представляет интерес выяснить, насколько разными (или близкими) будут временные структуры процентных ставок доходности для этих двух моделей при данных условиях. Рассмотрим этот вопрос в нейтральной к риску постановке, когда рыночные цены риска равны нулю.

Основываясь на известных результатах [2], временную структуру процентных ставок доходности $y(\tau)$ (кривую доходности) и форвардных процентных ставок $f(\tau)$ (форвардную кривую) для модели (1) можно записать в виде

$$y_1(\tau) = \alpha + (r - \alpha) \frac{B(\tau)}{\tau} + \frac{kn\nu\psi}{G} \left(1 - \frac{\ln(1 + gB(\tau))}{\tau g} \right), \quad (3)$$

$$f_1(\tau) = r + n\nu\psi k B(\tau) - k B(\tau) (r - \alpha) - 2k\nu\psi B(\tau)^2 (r - \alpha), \quad (4)$$

где $B(\tau) = \left(\frac{\varepsilon}{e^{\varepsilon\tau} - 1} + G \right)^{-1}$, $\varepsilon = \sqrt{k^2 + 4k\nu\psi}$, $g = (\varepsilon - k) / 2$, $G = (\varepsilon + k) / 2$.

Заметим, что обе эти кривые имеют одни и те же предельные значения как при краткосрочных, так и при долгосрочных доходностях:

$$\lim_{\tau \rightarrow 0} y_1(\tau) = \lim_{\tau \rightarrow 0} f_1(\tau) = r; \lim_{\tau \rightarrow \infty} y_1(\tau) = \lim_{\tau \rightarrow \infty} f_1(\tau) \equiv y(\infty) = \alpha + \frac{ng}{2} < \theta. \quad (5)$$

Для модели (2) эти кривые определяются с помощью гиперболических функций соотношениями [3]

$$y_2(\tau) = \alpha + \sum_{i=1}^n \frac{1}{2\tau} \left(\frac{2\psi_i X_i^2}{\varepsilon_i \text{cth}(\varepsilon_i \tau) + k_i} + \ln \left(\text{ch}(\varepsilon_i \tau) + \frac{k_i}{\varepsilon_i} \text{sh}(\varepsilon_i \tau) \right) - k_i \tau \right), \quad (6)$$

$$f_2(\tau) = \alpha + \sum_{i=1}^n \left(\frac{\varepsilon_i^2 \psi_i X_i^2}{(\varepsilon_i \text{ch}(\varepsilon_i \tau) + k_i \text{sh}(\varepsilon_i \tau))^2} + \frac{1}{2} \frac{\varepsilon_i^2 - k_i^2}{\varepsilon_i \text{cth}(\varepsilon_i \tau) + k_i} \right), \quad (7)$$

где $\varepsilon_i \equiv \sqrt{k_i^2 + 4k_i \nu_i \psi_i} \geq k_i$, $1 \leq i \leq n$. Согласно (2) имеем $r = \alpha + \sum_{i=1}^n \psi_i X_i^2$. Для нашего случая, когда по

принятым предположениям параметры k , ν и ψ не зависят от индекса, формулы (6) и (7) упрощаются к виду

$$y_2(\tau) = \alpha + \frac{r - \alpha}{\tau \varepsilon \operatorname{cth}(\varepsilon \tau) + \tau k} + \frac{n}{2\tau} \ln \left(\operatorname{ch}(\varepsilon \tau) + \frac{k}{\varepsilon} \operatorname{sh}(\varepsilon \tau) \right) - \frac{nk}{2}, \quad (8)$$

$$f_2(\tau) = \alpha + \frac{\varepsilon^2 (r - \alpha)}{(\varepsilon \operatorname{ch}(\varepsilon \tau) + k \operatorname{sh}(\varepsilon \tau))^2} + \frac{n}{2} \frac{\varepsilon^2 - k^2}{\varepsilon \operatorname{cth}(\varepsilon \tau) + k}, \quad (9)$$

который не предусматривает явной зависимости от конкретных значений латентных переменных состояния X , а параметры k , ε и ν те же, что и в формулах (3) и (4). Анализ показывает, что кривые (8) и (9) имеют одинаковые предельные значения как при краткосрочных, так и при долгосрочных доходностях, точно такие же, как у кривых (3) и (4), приведенных в равенствах (5).

Из представлений (8) и (9) видно, что при предположениях следствий 2 и 3 в квадратичной модели (2) при фиксированной процентной ставке r в отличие от ожидаемого семейства кривых, как в [3], существуют единственные кривые $y_2(\tau)$ и $f_2(\tau)$. Это объясняется тем, что эти кривые зависят не от конкретных значений X_i , а от суммы квадратов всех переменных состояний X_i , которая как раз задает фиксированную процентную ставку r .

Заметим, что функции (3)–(4) и (8)–(9) для малых τ имеют следующие производные:

$$\left. \frac{dy_1(\tau)}{d\tau} \right|_{\tau=0} = \left. \frac{1}{2} \frac{df_1(\tau)}{d\tau} \right|_{\tau=0} = \left. \frac{1}{2} \frac{dy_2(\tau)}{d\tau} \right|_{\tau=0} = \left. \frac{1}{4} \frac{df_2(\tau)}{d\tau} \right|_{\tau=0} = \frac{k(\theta - r)}{2},$$

где $\theta = \alpha + n \nu \psi$ – стационарное математическое ожидание процесса процентной ставки $r(t)$. Это говорит, в частности, о том, что при малых τ , во-первых, форвардная кривая изменяется в два раза быстрее, чем кривая доходности для обеих моделей, во-вторых, вектор латентных переменных X не влияет на знак производных, и, в-третьих, кривые квадратичной модели при малых τ изменяются в два раза быстрее, чем кривые аффинной модели Даффи–Кана.

Как известно, в практике обычно встречаются кривые доходности (и форвардные кривые) трех типов: нормальные (монотонно возрастающие), инверсные (монотонно убывающие) и кривые с максимумом («горбатые»). Все эти три типа могут порождаться рассматриваемыми здесь моделями в зависимости от значения стартовой процентной ставки r . Когда $r < y(\infty)$, получается нормальная кривая. Когда $r > \theta$, кривые доходности инверсные. Если $y(\infty) < r < \theta$, кривая доходности имеет максимум. На рис. 1–3 для иллюстрации эти случаи показаны. Чтобы представить эти кривые «целиком» для всего интервала значений сроков до погашения $\tau \in (0, \infty)$, использовано нелинейное преобразование сроков до погашения $u = 1 - e^{-\rho \tau}$, которое отображает положительную полуось $(0, \infty)$ в единичный интервал $(0, 1)$. Принятое при расчетах численное значение $\rho = \ln 10/30 = 0,07675$ соответствует тому, что сроки до погашения от 0 до 30 отображаются в интервал $(0; 0,9)$. Так что $y(\tau)$ отображает $Y(u)$, а $f(\tau)$ отображает $F(u)$, где $\tau = -\ln(1 - u) / \rho$.

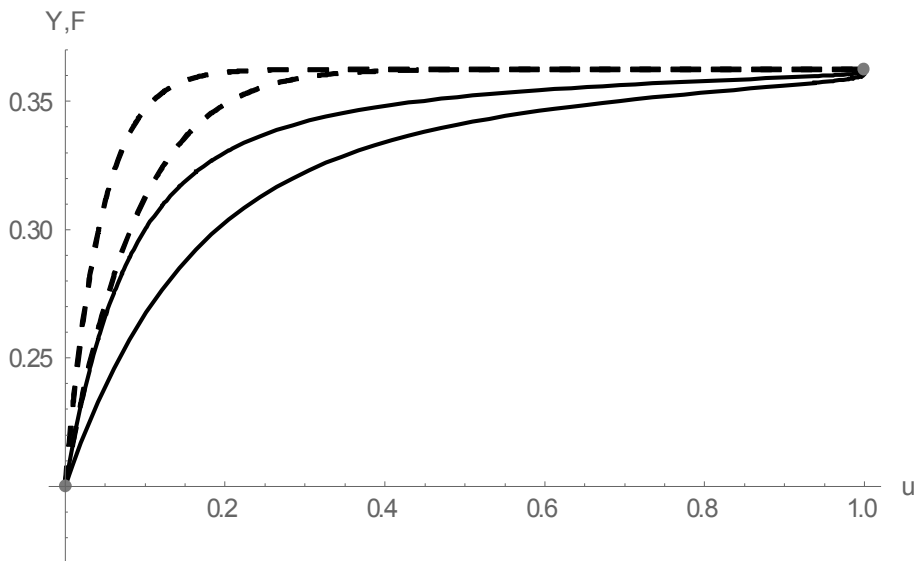


Рис. 1. Нормальные типы кривых $Y(u)$ и $F(u)$; $r = 0,2 < y(\infty) = 0,3625$

Кривые доходности $Y(u)$ показаны сплошными линиями, а форвардные кривые $F(u)$ – пунктирными. Круглыми маркерами показаны предельные значения доходностей при $\tau \rightarrow 0$ ($u \rightarrow 0$) и при $\tau \rightarrow \infty$ ($u \rightarrow 1$). Кривые доходности для модели Даффи–Кана располагаются ниже соответствующих кривых квадратичной модели для нормального типа (рис. 1) и выше – для инверсного типа (рис. 2). В качестве параметров моделей были выбраны следующие: $\alpha = 0$, $n = 5$, $v = 0,08$, $\psi = 1$, $k = 0,7$ ($k = 0,5$ для рис. 3), $\theta = 0,4$, $y(\infty) = 0,3625$ ($y(\infty) = 0,3508$ для рис. 3).

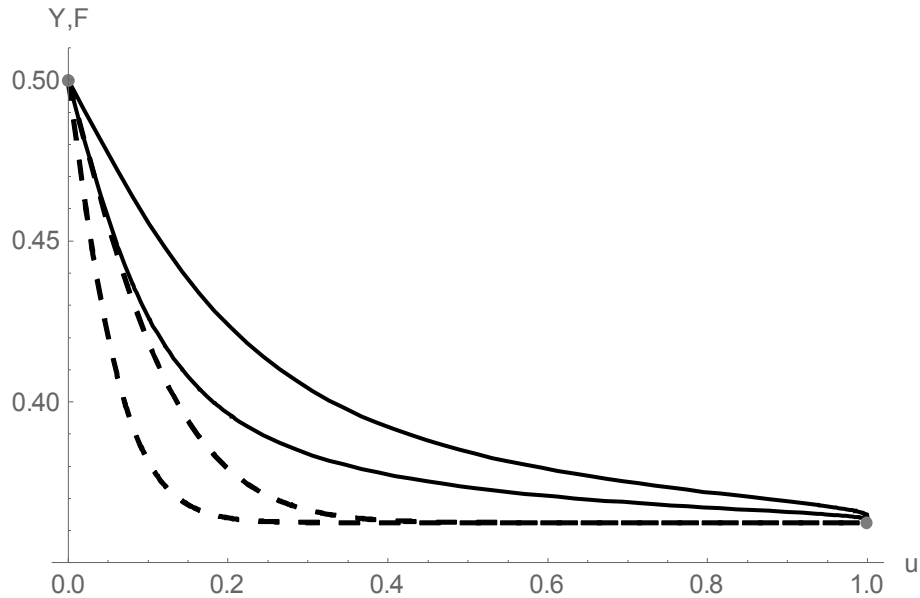


Рис. 2. Инверсные типы кривых; $r = 0,5 > \theta = 0,4$

Таким образом, когда в квадратичной модели любой размерности n латентные переменные состояния X независимы и одинаково распределены по нормальному закону с нулевым средним, то временная структура процентных ставок доходности не зависит от конкретных значений переменных X , а определяется только стартовым значением r текущей краткосрочной процентной ставки точно так же, как в аффинных моделях. При этом долгосрочные предельные ставки получаются такими же, как в модели Даффи–Кана. Причем распределение вероятностей процесса краткосрочной процентной ставки в этих моделях оказывается идентичным – сдвинутым гамма-распределением.

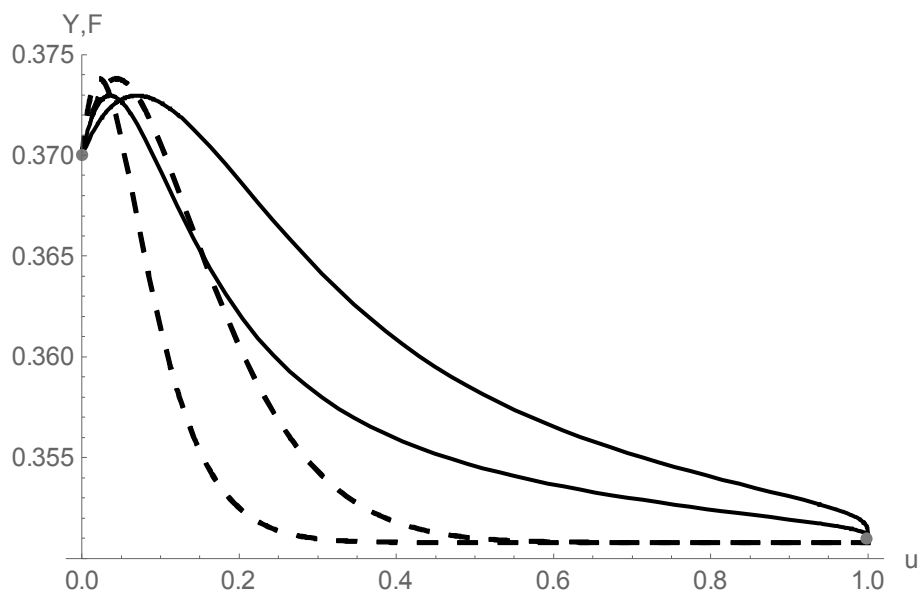


Рис. 3. Кривые с максимумом; $y(\infty) = 0,3508 < r = 0,37 < \theta = 0,4$

Особенностью квадратичной модели является более быстрая, чем в модели Даффи–Кана, сходимость к предельному значению долгосрочной доходности, которая иллюстрируется на рис. 4 с помощью отношений

$$\delta Y = \frac{Y_2(u) - y(\infty)}{Y_1(u) - y(\infty)}, \quad \delta F = \frac{F_2(u) - y(\infty)}{F_1(u) - y(\infty)}.$$

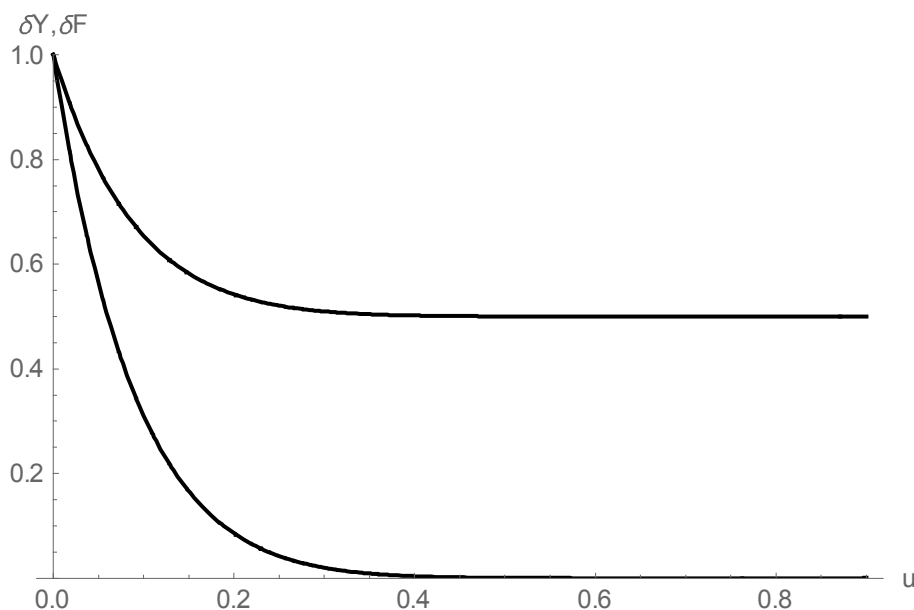


Рис. 4. Отношения δY (верхняя кривая) и δF (нижняя кривая), характеризующие сравнительную скорость сходимости кривых доходности рис. 1 к предельному значению $y(\infty)$

Из рисунка следует, что для данных рис. 1 практически для всех сроков погашения кривая $y_2(\tau)$ в два раза ближе к предельному значению $y(\infty)$, чем кривая $y_1(\tau)$. Что касается форвардной кривой $f_2(\tau)$, то она практически уже достигает своего предельного значения для $\tau > 9$ лет ($u > 0,5$).

К сожалению, хотя квадратичные модели доходности активно обсуждаются в литературе, природа квадратичных моделей оказывается загадочной, поскольку они являются математическими абстракциями, и ни один из авторов, описывающих свойства этих моделей, не привел объяснения, каким образом пользователь этих моделей может наблюдать значения латентных переменных состояния X и вообще существуют ли они, как они определяются и каков их экономический смысл.

ЛИТЕРАТУРА

1. Ahn D.-H., Dittmar R.F., Gallant A.R. Quadratic term structure models: Theory and evidence // Review of Financial Studies. 2001. V. 15. P. 243–288.
2. Медведев Г.А. О временной структуре доходности. 3. Однофакторная модель Даффи–Кана // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2012. № 3 (20). С. 71–80.
3. Медведев Г.А., Павлив Д.А. О квадратичных моделях доходности в риск-нейтральной среде // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2016. № 4 (37). С. 44–56.

Медведев Геннадий Алексеевич, д-р физ.-мат. наук, профессор. E-mail: MedvedevGA@bsu.by
Белорусский государственный университет (г. Минск, Республика Беларусь)

Поступила в редакцию 22 августа 2016 г.

Medvedev Gennady A. (Belarusian State University, Belarus, Minsk).

On a quadratic model of yield term structure.

Keywords: time yield structure; quadratic model; risk neutral probability measure; Duffie–Kan affine model.

DOI: 10.17223/19988605/38/4

Within framework of the theory of diffusion processes there are various versions of an evolution of short-term yield interest rates. Nevertheless till now there was no such model which could be a suitable basis for construction of term structure of yield close to existing one on real financial market. The models of interest rates leading to affine term structures of yield are simple, most known and imply a solution in an analytical form. However, reproduction of real term structures by means of affine models are inexact. Recently development of models goes in two directions: increase dimension of models and refusal of affine properties. As representatives of such development are most popular now so-called quadratic models of interest rate processes in which interest rate process $r(t)$ is set by the equations

$$\begin{aligned} dX(t) &= \xi(X(t)) dt + \sigma(X(t)) dW(t), \quad t > t_0, \quad X(t_0) = X_0, \\ r(t) &= \alpha + X(t)^T \Psi X(t), \quad X(t) \in \mathbb{R}^n, \quad \alpha \in \mathbb{R}^1, \quad \Psi \in \mathbb{R}^{n \times n}. \end{aligned}$$

Usually $\alpha \geq 0$, Ψ is a symmetrical positive definite matrix. When the vector $\xi(X)$ linearly depends on X , and the matrix $\sigma(X)$ does not depend on X , process $X(t)$ is Gaussian and in a stationary conditions has, say, expectation μ and a matrix of a covariance V . If Ψ and V are diagonal matrixes $\Psi = \psi I$, $V = \nu I$, and $\mu = 0$ the shifted gamma distribution with shift parameter α , scale parameter $1/2\nu\psi$ and form parameter $n/2$ will be marginal distribution of process $r(t)$. The shifted gamma distribution characterizes also the short-term interest rate in affine model of Duffie–Kan. Thus, the Duffie–Kan model and quadratic model generate the stochastic processes $r(t)$ with identical distribution. In the paper, the explicit expressions for the term structure of zero-coupon yield to maturity and forward interest rate curve for both models are obtained. Also, the differences between the yield term structures of the models considered in the risk-neutral setting are discussed, when the market price of risk is zero. It is shown that if in quadratic model of any dimension n latent state variable X are independent and identical distributed under the normal law with a zero expectation the term structure of interest rates of yield does not depend on concrete values of variables X , and is depended only on starting value r of the current short-term interest rate in the same way as in affine models. Thus long-term limiting rates turn out the same, as in model of Duffie–Kan. Comparative properties of affine Duffie–Kan model and quadratic model of yield are illustrated by a numerical example.

REFERENCES

1. Ahn, D.-H., Dittmar, R.F. & Gallant, A.R. (2001) Quadratic term structure models: Theory and evidence. *Review of Financial Studies*. 15. pp. 243–288.
2. Medvedev, G.A. (2012) On term structure of yield rates. 3. The Duffie–Kan one-factor model. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 3(20). pp. 71–80. (In Russian).
3. Medvedev, G.A. & Pavliv, D.A. (2016) On quadratic models of yield in risk-neutral world. *Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 4(37). pp. 44–56. (In Russian).

A.I. Rouban

THE SENSITIVITY FUNCTIONALS IN THE BOLTS'S PROBLEM FOR MULTIVARIATE DYNAMIC SYSTEMS DESCRIBED BY ORDINARY INTEGRAL EQUATIONS

The variation method is applied to calculation sensitivity functionals, which connect the first variation of quality functionals of systems operate (the Bolts's problem) with variations of variables and constant parameters, for the multidimensional nonlinear dynamic systems described by the generalized ordinary Volterra's second-kind integral equations.

Keywords: variation method; sensitivity functional; sensitivity coefficient; integral equation; conjugate equation.

The sensitivity functional (SF) connect the first variation of quality functional with variations of variable and constant parameters. Coefficients before variations of constant parameters name the sensitivity coefficients (SC). They are components of vector gradient from quality functional according to constant parameters.

The problem of calculation of SF and SC of dynamic systems is principal in the analysis and syntheses of control laws, identification, optimization [1–7]. The first-order sensitivity characteristics are mostly used. Later on we shall examine only SC and SF of the first-order.

Consider a vector output $y(t)$ of dynamic object model under continuous time $t \in [t_0, t^1]$, implicitly depending on vectors parameters $\tilde{\alpha}(t), \bar{\alpha}$ and functional I constructed on $y(t)$ under $t \in [t_0, t^1]$. The first variation δI of functional I and variations $\delta \tilde{\alpha}(t)$ are connected with each other with the help of a single-line functional – SF with respect to variable parameters $\tilde{\alpha}(t)$: $\delta_{\tilde{\alpha}(t)} I = \int_{t_0}^{t^1} V(t) \delta \tilde{\alpha}(t) dt$. SC with respect to constant parameters $\bar{\alpha}$ are called a gradient of I on $\bar{\alpha}$: $(dI/d\bar{\alpha})^T \equiv \nabla_{\bar{\alpha}} I$. SC are a coefficients of single-line relationship between the first variation of functional δI and the variations $\delta \bar{\alpha}$ of constant parameters $\bar{\alpha}$:

$$\delta_{\bar{\alpha}} I = (\nabla_{\bar{\alpha}} I)^T \delta \bar{\alpha} = (dI/d\bar{\alpha}) \delta \bar{\alpha} \equiv \sum_{j=1}^m \frac{\partial I}{\partial \bar{\alpha}_j} \delta \bar{\alpha}_j.$$

The direct method of SC calculation (by means of the differentiation of quality functional with respect to constant parameters) inevitably requires a solution of cumbersome sensitivity equations to sensitivity functions $W(t)$. $W(t)$ is the matrix of single-line relationship of the first variation of dynamic model output with parameter variations $\delta y(t) = W(t) \delta \bar{\alpha}$. For instance, for functional $I = \int_{t_0}^{t^1} f_0(y(t), \bar{\alpha}, t) dt$ we have following SC vector

$$(\text{row vector}): dI/d\bar{\alpha} = \int_{t_0}^{t^1} [(\partial f_0 / \partial y) W(t) + \partial f_0 / \partial \bar{\alpha}] dt.$$

For obtaining the matrix $W(t)$ it is necessary to decide bulky system equations – sensitivity equations. The j -th column of matrix $W(t)$ is made of the sensitivity functions $dy(t)/d\bar{\alpha}_j$ with respect to component $\bar{\alpha}_j$ of vector $\bar{\alpha}$. They satisfy a vector equation (if y is a vector) resulting from dynamic model (for y) by derivation [1–3] on a parameter $\bar{\alpha}_j$.

To variable parameters such a method is inapplicable because the sensitivity functions exist with respect to constant parameters.

For relatively simply classes of dynamic systems it is shown that in the SC calculation it is possible to get rid of deciding the bulky sensitivity equations due to the passage of deciding the conjugate equations – conjugate with respect to dynamic equations of object. Method of receipt of conjugate equations (it was offered in 1962) is cumbersome, because it is based on the analysis of sensitivity equations, and it does not get its development.

Variational method [4], ascending to Lagrange's, Hamilton's, Euler's memoirs, makes possible to simplify the process of determination of conjugate equations and formulas of account of SF and SC. On the basis of this method it is an extension of quality functional by means of inclusion into it object dynamic equations by means of Lagrange's multipliers and obtaining the first variation of extended functional on phase coordinates of object and on interesting parameters. Dynamic equations for Lagrange's multipliers are obtained due to set equal to a zero (in the first variation of extended functional) the functions before the first variations of phase coordinates. Given simplification first variation of extended functional brings at presence in the right part only parameter variations, i.e. it is got the SF. If all parameters are constant that the parameters variations are carried out from corresponding integrals and at the final result in obtained functional variation the coefficients before parameters variations are the required SC. Given method was used in [7–9] for dynamic systems described by ordinary continuous Volterra's second-kind integral and integro-differential equations. In this article the variational method of account of SC and of SF develops more general (on a comparison with papers [8, 9]) continuous many-dimensional non-linear dynamic systems circumscribed by the vectorial non-linear continuous ordinary Volterra's second-kind integral equations with variable and constant parameters. The more common quality functional (the Bolts's Problem) is used also.

1. Problem statement

We suppose that the dynamic object is described by system of non-linear continuous Volterra's ordinary integral equations (IE) of the second genus (more general than in the monography [7. P. 74]):

$$y(t) = r(\tilde{\alpha}(t), \bar{\alpha}, t_0, t) + \int_{t_0}^t K(t, y(s), \tilde{\alpha}(s), \bar{\alpha}, s) ds, \quad t_0 \leq t \leq t^1, \quad t_0 = t_0(\bar{\alpha}), \quad t^1 = t^1(\bar{\alpha}). \quad (1)$$

Here: initial t_0 and final t^1 instants are known functions of constant parameters $\bar{\alpha}$. $\tilde{\alpha}(t)$, $\bar{\alpha}$ are a vector-columns of interesting variable and constant parameters; y is a vector-column of phase coordinates; $r(\cdot)$, $K(\cdot)$ are known continuously differentiated limited vector-functions.

Variables $\eta(t)$ at each current moment of time t are connected with phase coordinates $y(t)$ by known transformation

$$\eta(t) = \eta(y(t), \tilde{\alpha}(t), \bar{\alpha}, t), \quad t \in [t_0, t^1], \quad (2)$$

where $\eta(\cdot)$ – also continuous, continuously differentiable, limited (together with the first derivatives) vector-function. Equation (1.2) is often known as model of a measuring apparatus. The required parameters $\tilde{\alpha}(t)$, $\bar{\alpha}$ are inserted also in it. A dimensionalities of vectors y and η can be various.

The quality of functioning of system it is characterised of functional

$$I = \int_{t_0}^{t^1} f_0(\eta(t), \tilde{\alpha}(t), \bar{\alpha}, t) dt + I_1(\eta(t^1), \bar{\alpha}, t^1), \quad (3)$$

depending on $\tilde{\alpha}(t)$ and $\bar{\alpha}$. The conditions for function $f_0(\cdot)$, $I_1(\cdot)$ are the same as for $K(\cdot)$, $r(\cdot)$. With use of a functional (1.3) the optimization problem (in the theory of optimal control) are named as the Bolts's problem. From it as the individual variants follow: Lagrange's problem (when there is only integral component) and Mayer's problem (when there is only second component – function from phase coordinates at a finishing point).

With the purpose of simplification of appropriate deductions with preservation of a generality in all transformations (1.1) – (1.3) there are two vectors of parameters $\tilde{\alpha}(t)$, $\bar{\alpha}$. If in the equations (1.1)–(1.3) parame-

ters are different then it is possible formally to unit them in two vectors $\tilde{\alpha}(t), \bar{\alpha}$, to use obtained outcomes and then to make appropriate simplifications, taking into account a structure of a vectors $\tilde{\alpha}(t), \bar{\alpha}$.

By obtaining of results the obvious designations:

$$r(t) \equiv r(\tilde{\alpha}(t), \bar{\alpha}, t_0, t), K(t, s) \equiv K(t, y(s), \tilde{\alpha}(s), \bar{\alpha}, s), \eta(t) \equiv \eta(y(t), \tilde{\alpha}(t), \bar{\alpha}, t),$$

$$f_0(t) \equiv f_0(\eta(t), \tilde{\alpha}(t), \bar{\alpha}, t), I_1(t^1) \equiv I_1(\eta(t^1), \bar{\alpha}, t^1)$$

are used.

Is shown also that the variation method without basic modifications allows to receive SF

$$\delta I(\alpha) = \int_{t_0-\tau}^{t^1} V(t) \delta \tilde{\alpha}(t) dt + [dI(\alpha)/d\tilde{\alpha}(t^1)] \delta \tilde{\alpha}(t^1) + [dI(\alpha)/d\bar{\alpha}] \delta \bar{\alpha} \text{ in relation to variable and constant parameters.}$$

2. Variational method for models (1)–(3)

Complement a quality functional (2) by restrictions-equalities (1) by means of Lagrange's multipliers $\gamma(t)$, $t \in [t_0, t^1]$, (column vectors) and get the extended functional

$$I = I(\alpha) + \int_{t_0}^{t^1} \gamma^T(t) [r(t) + \int_{t_0}^t K(t, s) ds - y(t)] dt, \quad (4)$$

which complies with $I(\alpha)$ when (1.1) is fulfilled. Take into account the form of functional I , change an order of integrating in double integral inside of triangular area (see fig. 1):

$$\left(\text{i.e. } \int_{t_0}^{t^1} \int_{t_0}^t A(t, s) ds dt = \int_{t_0}^{t^1} \int_t^{t^1} A(s, t) ds dt \right),$$

$$\int_{t_0}^{t^1} \gamma^T(t) \int_{t_0}^t K(t, s) ds dt = \int_{t_0}^{t^1} \int_t^{t^1} \gamma^T(s) K(s, t) ds dt, \quad (5)$$

and then extended functional (4) accepts a form:

$$I = I_1(t^1) + \int_{t_0}^{t^1} \{ f_0(t) + \gamma^T(t) [r(t) - y(t)] + \int_t^{t^1} \gamma^T(s) K(s, t) ds \} dt. \quad (6)$$

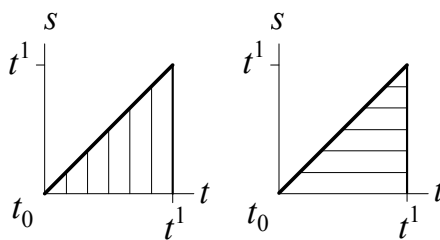


Fig. 1. Triangular area and order of an integration

Find the first variation for I with respect to $\delta y(t)$ and to $\delta \tilde{\alpha}(t)$ ($t \in [t_0, t^1]$), $\delta \tilde{\alpha}(t^1)$, $\delta \bar{\alpha}$ taking account:

- 1) dependence the right member of IE (1.1) on $y(t)$; 2) interconnection (3) between $\eta(t)$ and $y(t)$, $\tilde{\alpha}(t)$, $\bar{\alpha}$;
- 3) dependence t_0 , t^1 , $I_1(t^1)$ on $\bar{\alpha}$ [i.e. $t_0 = t_0(\bar{\alpha})$, $t^1 = t^1(\bar{\alpha})$, $I_1(t^1) \equiv I_1(\eta(t^1), \bar{\alpha}, t^1)$]:

$$\delta I = \Phi(t^1) \delta y(t^1) + \int_{t_0}^{t^1} \left[\frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial y(t)} + \int_t^{t^1} \gamma^T(s) \frac{\partial K(s, t)}{\partial y(t)} ds - \gamma^T(t) \right] \delta y(t) dt +$$

$$+ \int_{t_0}^{t^1} \left[\frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial \tilde{\alpha}(t)} + \frac{\partial f_0(t)}{\partial \tilde{\alpha}(t)} + \gamma^T(t) \frac{\partial r(t)}{\partial \tilde{\alpha}(t)} + \int_t^{t^1} \gamma^T(s) \frac{\partial K(s, t)}{\partial \tilde{\alpha}(t)} ds \right] \delta \tilde{\alpha}(t) dt +$$

$$\begin{aligned}
& + \frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial \tilde{\alpha}(t^1)} \delta \tilde{\alpha}(t^1) + \left\{ \frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial \bar{\alpha}} + \frac{\partial I_1(t^1)}{\partial \bar{\alpha}} + \right. \\
& + \int_{t_0}^{t^1} \left[\frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial \bar{\alpha}} + \frac{\partial f_0(t)}{\partial \bar{\alpha}} + \gamma^T(t) \frac{\partial r(t)}{\partial \bar{\alpha}} + \int_t^{t^1} \gamma^T(s) \frac{\partial K(s,t)}{\partial \bar{\alpha}} ds \right] dt + \\
& + \left[-f_0(t_0) + \int_{t_0}^{t^1} \gamma^T(t) \left(\frac{\partial r(t)}{\partial t_0} - K(t, t_0) \right) dt \right] \frac{dt_0}{d\bar{\alpha}} + \\
& + \left. \left[\frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial t^1} + \frac{\partial I_1(t^1)}{\partial t^1} + f_0(t^1) \right] \frac{dt^1}{d\bar{\alpha}} \right\} \delta \bar{\alpha}, \tag{7}
\end{aligned}$$

here

$$\frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial y(t^1)} \equiv \Phi(t^1).$$

Out of object equation (1) we calculate the first variation $\delta y(t^1)$ (variation, included in the first addend of (7))

$$\begin{aligned}
\delta y(t^1) &= \int_{t_0}^{t^1} \frac{\partial K(t^1, s)}{\partial y(s)} \delta y(s) ds + \int_{t_0}^{t^1} \frac{\partial K(t^1, s)}{\partial \tilde{\alpha}(s)} \delta \tilde{\alpha}(s) ds + \\
& + \frac{\partial r(t^1)}{\partial \tilde{\alpha}(t^1)} \delta \tilde{\alpha}(t^1) + \left\{ \frac{\partial r(t^1)}{\partial \bar{\alpha}} + \int_{t_0}^{t^1} \frac{\partial K(t^1, s)}{\partial \bar{\alpha}} ds + \left[\frac{\partial r(t^1)}{\partial t_0} - K(t^1, t_0) \right] \frac{dt_0}{d\bar{\alpha}} + \right. \\
& + \left. \left[\frac{\partial r(t^1)}{\partial t^1} + K(t^1, t^1) + \int_{t_0}^{t^1} \frac{\partial K(t^1, s)}{\partial t^1} ds \right] \frac{dt^1}{d\bar{\alpha}} \right\} \delta \bar{\alpha}. \tag{8}
\end{aligned}$$

Then the first variation (7) obtains the following form:

$$\delta I = \delta_{y(t)} I + \delta_{\tilde{\alpha}(t)} I + \delta_{\alpha(t^1)} I + \delta_{\bar{\alpha}} I, \tag{9}$$

$$\delta_{y(t)} I = \int_{t_0}^{t^1} \left[\Phi(t^1) \frac{\partial K(t^1, t)}{\partial y(t)} + \frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial y(t)} + \int_t^{t^1} \gamma^T(s) \frac{\partial K(s, t)}{\partial y(t)} ds - \gamma^T(t) \right] \delta y(t) dt, \tag{10}$$

$$\delta_{\tilde{\alpha}(t)} I = \int_{t_0}^{t^1} \left[\frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial \tilde{\alpha}(t)} + \frac{\partial f_0(t)}{\partial \tilde{\alpha}(t)} + \gamma^T(t) \frac{\partial r(t)}{\partial \tilde{\alpha}(t)} + \Phi(t^1) \frac{\partial K(t^1, t)}{\partial \tilde{\alpha}(t)} + \int_t^{t^1} \gamma^T(s) \frac{\partial K(s, t)}{\partial \tilde{\alpha}(t)} ds \right] \delta \tilde{\alpha}(t) dt, \tag{11}$$

$$\delta_{\alpha(t^1)} I = \left[\frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial \tilde{\alpha}(t^1)} + \Phi(t^1) \frac{\partial r(t^1)}{\partial \tilde{\alpha}(t^1)} \right] \delta \tilde{\alpha}(t^1), \tag{12}$$

$$\begin{aligned}
\delta_{\bar{\alpha}} I &= \left\{ \frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial \bar{\alpha}} + \frac{\partial I_1(t^1)}{\partial \bar{\alpha}} + \Phi(t^1) \left[\frac{\partial r(t^1)}{\partial \bar{\alpha}} + \int_{t_0}^{t^1} \frac{\partial K(t^1, s)}{\partial \bar{\alpha}} ds \right] + \right. \\
& + \int_{t_0}^{t^1} \left[\frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial \bar{\alpha}} + \frac{\partial f_0(t)}{\partial \bar{\alpha}} + \gamma^T(t) \frac{\partial r(t)}{\partial \bar{\alpha}} + \int_t^{t^1} \gamma^T(s) \frac{\partial K(s, t)}{\partial \bar{\alpha}} ds \right] dt + \\
& + \left[\Phi(t^1) \left[\frac{\partial r(t^1)}{\partial t_0} - K(t^1, t_0) \right] - f_0(t_0) + \int_{t_0}^{t^1} \gamma^T(t) \left[\frac{\partial r(t)}{\partial t_0} - K(t, t_0) \right] dt \right] \frac{dt_0}{d\bar{\alpha}} + \\
& + \left. \left[\Phi(t^1) \left[\frac{\partial r(t^1)}{\partial t^1} + K(t^1, t^1) + \int_{t_0}^{t^1} \frac{\partial K(t^1, s)}{\partial t^1} ds \right] + \right. \right.
\end{aligned}$$

$$+ \frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial t^1} + \frac{\partial I_1(t^1)}{\partial t^1} + f_0(t^1) \left] \frac{dt^1}{d\bar{\alpha}} \right\} \delta \bar{\alpha}. \quad (13)$$

In a variation (10) we equate with zero factors before variations of phase coordinates δy and discover: the conjugate equations for Lagrange's multipliers $\gamma(t)$

$$\gamma^T(t) = \Phi(t^1) \frac{\partial K(t^1, t)}{\partial y(t)} + \frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial y(t)} + \int_t^{t^1} \gamma^T(s) \frac{\partial K(s, t)}{\partial y(t)} ds, \quad t_0 \leq t \leq t^1. \quad (14)$$

These equations are decided in the opposite direction of time (from t^1).

In a result three components $\delta I = \delta_{\tilde{\alpha}(t)} I + \delta_{\tilde{\alpha}(t^1)} I + \delta_{\bar{\alpha}} I$ of the first variation of quality functional I in relation to variables $\tilde{\alpha}(t)$ and constant parameters $\tilde{\alpha}(t^1)$, $\bar{\alpha}$ are submitted accordingly by formulas (11), (12) and (13). This result is more common in relation to appropriate results of papers [7, 8]. Variables and constant parameters are present in integrated model of object, also at model of the measuring device and at generalized quality functional for system (the Bolts's Problem). An additional a dependence t_0, t^1 from $\bar{\alpha}$ are taken into account.

In a basis of calculation of sensitivity functionals the decision of the integrated equations of the object model in a forward direction of time and obtained integrated equations for Lagrange's multipliers in the opposite direction of time lays.

Example (The ordinary differential equations). Consider that the dynamic object is described by system of non-linear continuous differential equations with variable and constant parameters $\tilde{\alpha}(t)$, $\bar{\alpha}$:

$$\dot{y}(t) = f(y(t), \tilde{\alpha}(t), \bar{\alpha}, t), \quad t_0 \leq t \leq t^1, \quad y(t_0) = y_0(\bar{\alpha}, t_0). \quad (15)$$

We transform model (15) in Volterra's second-kind integral equation (1)

$$y(t) = y_0(\bar{\alpha}, t_0) + \int_{t_0}^t f(y(s), \tilde{\alpha}(s), \bar{\alpha}, s) ds, \quad t_0 \leq t \leq t^1. \quad (16)$$

Now

$$r(\cdot) = y_0(\bar{\alpha}, t_0), \quad K(t, s) = f(y(s), \tilde{\alpha}(s), \bar{\alpha}, s) \equiv f(s).$$

We write the conjugate equations (2.11) for Lagrange's multipliers

$$\gamma^T(t) = \frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial y(t)} + [\Phi(t^1) + \int_t^{t^1} \gamma^T(s) ds] \frac{\partial f(t)}{\partial y(t)}, \quad t_0 \leq t \leq t^1,$$

and SF (11), (12), (13)

$$\begin{aligned} \delta I &= \delta_{\tilde{\alpha}(t)} I + \delta_{\tilde{\alpha}(t^1)} I + \delta_{\bar{\alpha}} I, \\ \delta_{\tilde{\alpha}(t)} I &= \int_{t_0}^{t^1} \left[\frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial \tilde{\alpha}(t)} + \frac{\partial f_0(t)}{\partial \tilde{\alpha}(t)} + \Phi(t^1) \frac{\partial f(t)}{\partial \tilde{\alpha}(t)} + \int_t^{t^1} \gamma^T(s) \frac{\partial f(s)}{\partial \tilde{\alpha}(t)} ds \right] \delta \tilde{\alpha}(t) dt, \\ \delta_{\tilde{\alpha}(t^1)} I &= \frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial \tilde{\alpha}(t^1)} \delta \tilde{\alpha}(t^1), \\ \delta_{\bar{\alpha}} I &= \left\{ \frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial \bar{\alpha}} + \frac{\partial I_1(t^1)}{\partial \bar{\alpha}} + \Phi(t^1) \left[\frac{\partial y_0(\bar{\alpha}, t_0)}{\partial \bar{\alpha}} + \int_{t_0}^{t^1} \frac{\partial f(s)}{\partial \bar{\alpha}} ds \right] + \right. \\ &+ \int_{t_0}^{t^1} \left[\frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial \bar{\alpha}} + \frac{\partial f_0(t)}{\partial \bar{\alpha}} + \gamma^T(t) \frac{\partial y_0(\bar{\alpha}, t_0)}{\partial \bar{\alpha}} + \int_t^{t^1} \gamma^T(s) ds \right] \frac{\partial f(t)}{\partial \bar{\alpha}} dt + \\ &+ \int_{t_0-\tau}^{t_0} \gamma^T(t) dt \frac{\partial y_0(\bar{\alpha}, t_0)}{\partial \bar{\alpha}} + \left[\Phi(t^1) \left[\frac{\partial y_0(\bar{\alpha}, t_0)}{\partial t_0} - f(t_0) \right] - \right. \end{aligned}$$

$$-f_0(t_0) + \int_{t_0}^{t^1} \gamma^T(t) dt \left[\frac{\partial y_0(\bar{\alpha}, t_0)}{\partial t_0} - f(t_0) \right] \frac{dt_0}{d\bar{\alpha}} +$$

$$+ \left[\Phi(t^1) f(t^1) + \frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial t^1} + \frac{\partial I_1(t^1)}{\partial t^1} + f_0(t^1) \right] \frac{dt^1}{d\bar{\alpha}} \Big\} \delta \bar{\alpha}.$$

These results it is possible to represent in more customary (for differential equations) form. After change of variables:

$$\Phi(t^1) + \int_t^{t^1} \gamma^T(s) ds = \lambda^T(t), \quad t_0 \leq t \leq t^1; \quad \text{ore} \quad -\dot{\lambda}^T(t) = \gamma^T(t), \quad t_0 \leq t \leq t^1, \quad \lambda^T(t^1) = \Phi(t^1);$$

we obtain the conjugate equations in differential form

$$-\dot{\lambda}^T(t) = \frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial y(t)} + \lambda^T(t) \frac{\partial f(t)}{\partial y(t)}, \quad \lambda^T(t^1) = \Phi(t^1), \quad t_0 \leq t \leq t^1,$$

and than SF have the form

$$\delta I = \delta_{\bar{\alpha}(t)} I + \delta_{\alpha(t^1)} I + \delta_{\bar{\alpha}} I,$$

$$\delta_{\bar{\alpha}(t)} I = \int_{t_0}^{t^1} \left[\frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial \tilde{\alpha}(t)} + \frac{\partial f_0(t)}{\partial \tilde{\alpha}(t)} + \lambda^T(t) \frac{\partial f(t)}{\partial \tilde{\alpha}(t)} \right] \delta \tilde{\alpha}(t) dt,$$

$$\delta_{\alpha(t^1)} I = \frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial \tilde{\alpha}(t^1)} \delta \tilde{\alpha}(t^1),$$

$$\delta_{\bar{\alpha}} I = \left\{ \frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial \bar{\alpha}} + \frac{\partial I_1(t^1)}{\partial \bar{\alpha}} + \lambda^T(t_0) \frac{\partial y_0(\bar{\alpha}, t_0)}{\partial \bar{\alpha}} + \right.$$

$$+ \int_{t_0}^{t^1} \left[\frac{\partial f_0(t)}{\partial \eta(t)} \frac{\partial \eta(t)}{\partial \bar{\alpha}} + \frac{\partial f_0(t)}{\partial \bar{\alpha}} + \lambda^T(t) \frac{\partial f(t)}{\partial \bar{\alpha}} \right] dt +$$

$$+ \left[\lambda^T(t_0) \left[\frac{\partial y_0(\bar{\alpha}, t_0)}{\partial t_0} - f(t_0) \right] - f_0(t_0) \right] \frac{dt_0}{d\bar{\alpha}} +$$

$$\left. + \left[\Phi(t^1) f(t^1) + \frac{\partial I_1(t^1)}{\partial \eta(t^1)} \frac{\partial \eta(t^1)}{\partial t^1} + \frac{\partial I_1(t^1)}{\partial t^1} + f_0(t^1) \right] \frac{dt^1}{d\bar{\alpha}} \right\} \delta \bar{\alpha}.$$

Conclusion

The merit of variational method is applicability of its both for calculation of SF and SC. Besides the equations for Lagrange's multipliers remain without change.

Variables and constant parameters are present also at model of the measuring device and at generalized quality functional for system (the Bolts's Problem). In a basis of calculation of sensitivity functionals the decision of the integrated equations of model in a forward direction of time and obtained integrated equations for Lagrange's multipliers in the opposite direction of time lays.

Variation method of calculation of SF and SC allows a generalization on objects described by vectorial ordinary Volterra's second-kind integro-differential equations.

Integro-differential models structurally include separately integrated and differential models, and also 4 kinds of more simple integro-differential models which differ character of interaction of phase coordinates of integrated and differential parts. It is necessary to carry out transition from the integro-differential equation to corresponding integral equation, to use results of this paper and in them to execute return to initial variables. For the objects described by simpler integro-differential models enough in the received connected equations and in SF and SC to turn into a zero a corresponding components.

Variation method of calculation of SF and SC allows a generalization on objects described by vectorial dynamic equations with delay time and different classes of discontinuous dynamic equations.

Results are applicable at design of high-precision systems and devices.

This paper continues research in [7–9].

REFERENCES

1. Ostrovskiy, G.M. & Volin, Yu.M. (1967) *Methods of optimization of chemical reactors*. Moscow: Khimiya. (In Russian).
2. Speedy, C.B., Brown, R.F. & Goodwin, G.C. (1973) *Control theory: identification and optimal control*. Translated from English. Moscow: Mir. (In Russian).
3. Rosenvasser, E.N. & Yusupov, R.M. (eds) *Methods of sensitivity theory in automatic control*. Leningrad: Energiya. (In Russian)
4. Bryson, A.E. & Ho, Ju-Chi. (1972) *Applied optimal control*. Translated from English. Moscow: Mir. (In Russian).
5. Ruban, A.I. (1975) *Nonlinear dynamic object identification on the base of sensitivity algorithm*. Tomsk: Tomsk State University. (In Russian).
6. Rosenvasser, E.N. & Yusupov, R.M. (1981) *Sensitivity of control systems*. Moscow: Nauka. (In Russian).
7. Ruban, A.I. (1982) *Identification and sensitivity of complex systems*. Tomsk: Tomsk State University. (In Russian).
8. Ruban, A.I. (1996) Coefficients and functionals of sensitivity for multivariate systems described by integral equations. *Proceedings of Novosibirsk State Technical University*. 2(4). pp. 64–72. (In Russian).
9. Rouban, A.I. (1999) Coefficients and functionals of sensitivity for multivariate systems described by integral and integro-differential equations. *Advances in Modeling & Analysis: Series A. Mathematical Problems; General Mathematical Modeling. France: A.M.S.E.* 35(1). pp. 25–34.

Rouban Anatoly Ivanovich. Dr. Science, prof. E-mail: ai-rouban@mail.ru
Siberian Federal University, Krasnoyarsk, Russian Federation

Поступила в редакцию 9 октября 2016 г.

Рубан Анатолий Иванович (Сибирский федеральный университет, г. Красноярск, Российская Федерация).

Функционалы чувствительности в задаче Больца для многомерных динамических систем, описываемых обыкновенными интегральными уравнениями.

Ключевые слова: вариационный метод; функционал чувствительности; обыкновенное интегральное уравнение; функционал качества работы системы; задача Больца; сопряженное уравнение.

DOI: 10.17223/19988605/38/5

Вариационный метод применен для расчета функционалов чувствительности, которые связывают первую вариацию функционалов качества работы систем с вариациями переменных и постоянных параметров, для многомерных нелинейных динамических систем, описываемых обобщенными обыкновенными интегральными уравнениями Вольтерра второго рода.

УДК 519.2

DOI: 10.17223/19988605/38/6

С.П. Трофимов, А.В. Иванов

РАЗРЫВ ДВОЙСТВЕННОСТИ В ПОЛУБЕСКОНЕЧНОМ ЛИНЕЙНОМ ПРОГРАММИРОВАНИИ И АНАЛИЗ КАЧЕСТВА ОГРАНИЧЕНИЙ ГЕОМЕТРИЧЕСКИХ ОБЪЕКТОВ

Рассматривается пара двойственных задач полубесконечного линейного программирования. Предлагается геометрический способ анализа соотношений двойственности пары задач, основанный на использовании конической оболочки коэффициентов системы ограничений. Устанавливается связь наличия разрыва двойственности с незамкнутостью границы конической оболочки точек в многомерном пространстве. Построен нетривиальный пример задачи ЛП, для которой разрыв двойственности выполняется для неколлинеарных целевых векторов. На основе стандартных оптимизационных функций MATLAB разработана программа для анализа соотношений двойственности.

Ключевые слова: задача линейного программирования; разрыв двойственности; множество допустимых решений; коническая оболочка коэффициентов; система линейных неравенств; критерий соотношения двойственности; выпуклый незамкнутый конус.

Задача полубесконечного линейного программирования (ПбЛП) является важным объектом исследования в теории оптимизации [1]. Задача обычно изучается с двух точек зрения.

Численные, количественные методы предполагают задание бесконечного множества ограничений с помощью параметров. В дальнейшем бесконечная система аппроксимируется конечной подсистемой и решается симплексными методами.

Качественные методы используют свойства топологических пространств, а также строят выпуклые и конические оболочки точек, составленных из коэффициентов системы. При этом задача построения выпуклой оболочки не рассматривается.

Одной из важных целей изучения задачи ПбЛП является проверка соотношения двойственности, т.е. равенства оптимальных значений исходной и двойственной задач ЛП. С одной стороны, возникновение разрыва двойственности означает резкое ухудшение свойств численных методов решения этой задачи, а с другой стороны, наличие разрыва говорит о недостаточном качестве ограничений допустимого множества задачи ПбЛП [2]. Таким образом, анализ соотношения двойственности может оказаться полезным при анализе качества задания геометрического объекта: при задании геометрического объекта в виде допустимого множества системы ограничений задачи ПбЛП оценка качества определяется существованием разрыва двойственности на некоторых целевых направлениях.

Отметим, что задача ПбЛП с разрывом двойственности обычно рассматривается как нежелательная и экзотическая. Между тем в настоящей работе мы показываем, что задачи ПбЛП с разрывом двойственности образуют богатое семейство задач, и связываем наличие разрыва двойственности с незамкнутостью границы конической оболочки точек в многомерном пространстве. Размерность этого пространства на единицу больше количества переменных задачи. С помощью пары двойственных задач ПбЛП с разрывом двойственности осуществляется анализ структуры этой незамкнутой границы.

1. Двойственные задачи линейного программирования

Пара двойственных задач линейного программирования в топологических векторных пространствах имеет вид

$$\inf \{(x, y_0) : T \cdot x \in z_0 + Q_Z, x \in Q_X\}, \quad (1)$$

$$\sup \{(z_0, \omega) : -T^* \cdot \omega + y_0 \in Q_X^*, \omega \in Q_Z^*\}. \quad (2)$$

Здесь топологические векторные пространства X и Y , Z и W находятся в двойственности; T – линейный непрерывный оператор из X в Z ; $T^*: W \rightarrow Y$ – сопряжённый к нему; $Q_X \subset X$, $Q_Z \subset Z$ – замкнутые выпуклые конусы; $Q_X^* \subset Y$, $Q_Z^* \subset W$ – сопряжённые конусы.

Рассмотрим два частных случая пары двойственных задач (1)–(2).

1. Пространства X и Y – конечномерные пространства R^n , Z и W – бесконечномерные пространства, конусы Q_X и Q_Z являются неотрицательными ортантами. Тогда задача (1) принимает вид полубесконечной задачи линейного программирования:

$$v = \inf \{(x, c) : Ax \geq b, x \geq 0\}. \quad (3)$$

В этой задаче A – полубесконечная матрица, строки которой $a_\alpha \in R^n$, $\alpha \in \Omega$, Ω – некоторое счетное множество индексов, $b = (b_\alpha)_{\alpha \in \Omega}$ – вектор-столбец из пространства R^Ω , v – оптимальное значение задачи.

Двойственная задача (2) принимает вид

$$v^* = \sup \{(u, b) : A^T u \leq c, u \geq 0, u \in f\}, \quad (4)$$

где f – подпространство последовательностей из R^Ω , в которых лишь конечное число элементов отлично от нуля, v^* – оптимальное значение двойственной задачи.

Запишем пару двойственных задач (3)–(4) в другой, более удобной для нас форме:

$$v = \inf \{(x, c) : Ax \geq b\}, \quad (5)$$

$$v^* = \sup \{(u, b) : A^T u = c, u \geq 0, u \in f\}. \quad (6)$$

2. Пространства X и Y – конечномерные пространства R^n ; Z и W – конечномерные пространства R^m ; Q_X и Q_Z – замкнутые выпуклые конусы соответственно в пространствах R^n и R^m .

Преобразуем задачу (1) к виду (3). Ограничение $x \in Q_X$ задачи (1) эквивалентно системе линейных неравенств

$$(x, y) \geq 0, \quad \forall y \in Q_X^*. \quad (7)$$

Пусть $\{e_i: i = 1, \dots, n\}$ – базис единичных ортов пространства R^n . Тогда $x = \sum_{i=1}^n x_i e_i$ и ограничение $T \cdot x \in z_0 + Q_Z$ задачи (1) эквивалентно системе

$$(T \cdot (\sum_{i=1}^n x_i e_i) - z_0, \omega) \geq 0, \quad \forall \omega \in Q_Z^*,$$

что равносильно

$$\sum_{i=1}^n x_i (T \cdot e_i, \omega) \geq (z_0, \omega), \quad \forall \omega \in Q_Z^*. \quad (8)$$

Объединенная система ограничений (7) и (8) аналогична системе ограничений задачи (5).

2. Соотношение двойственности

Задачи (5) и (6) достаточно универсальны и изучаются во многих работах. В основном исследуются условия, при которых выполняется соотношение двойственности

$$v = v^*. \quad (9)$$

В [1] показано, в частности, что если $c \in \text{int}(\text{cone}\{a_\alpha, \alpha \in \Omega\})$, то равенство (9) имеет место. В [4, 5] в качестве достаточного условия для (9) требуется выполнение условия Слейтера для системы ограничений задачи (5) и замкнутость конической оболочки множества

$$K = \text{cone}\{[a_\alpha; b_\alpha] : \alpha \in \Omega\}.$$

Здесь и далее запись $[a_\alpha; b_\alpha]$ означает присоединение числа b_α к вектору-строке a_α .

В [6] для задачи (5) вводится понятие равномерной двойственности, при котором для каждого $c \in R^n$ должен иметь место один из следующих случаев: 1) $v = -\infty$, $v^* = -\infty$; 2) $v = +\infty$, $v^* = +\infty$; 3) $v = +\infty$, $v^* = -\infty$; 4) v конечно, задача (6) разрешима и $v = v^*$.

Необходимым и достаточным условием равномерной двойственности при $v < +\infty$ является замкнутость конуса

$$K \downarrow = \text{cone}\{[a_\alpha; b_\alpha], \alpha \in \Omega\}, [0; -1]\}.$$

3. Геометрический подход к анализу двойственных задач ЛП

В настоящей работе предлагается критерий, связывающий равенство (9) со свойствами некоторого промежутка. Фактически этот критерий позволяет взглянуть на задачи (5) и (6) с некоторой единой геометрической точки зрения, проясняет приведённые выше утверждения и приводит к новым результатам.

Введем обозначения

$$K^\uparrow = \text{cone} \{ \{[a_\alpha; b_\alpha], \alpha \in \Omega\}, [0; 1] \},$$

$$P = \{[c; r] : r \in R\}.$$

Покажем, что из геометрических соотношений между конусом K и прямой P можно получить ряд важных свойств задач (5) и (6). Возможны следующие случаи взаимного расположения K и P :

$$\overline{K} \cap P = \emptyset, \quad (10)$$

$$K \cap P = \emptyset, \quad \overline{K} \cap P \neq \emptyset, \quad (11)$$

$$K \cap P \neq \emptyset. \quad (12)$$

Найдем оптимальные значения v и v^* для каждого из этих случаев.

3.1. Бесконечность оптимальных значений пары двойственных задач

Пусть $\overline{K} \cap P = \emptyset$.

Тогда $K \cap P = \emptyset$, поэтому ограничения задачи (6) несовместны и, следовательно, $v^* = -\infty$.

Допустим система $Ax \geq b$ несовместна. Тогда, очевидно, $v = +\infty$.

Допустим, система $Ax \geq b$ совместна. Известно [5], что неравенство $(x, c) \geq r$ является следствием совместной системы $Ax \geq b$ тогда и только тогда, когда $[c; r] \in \overline{K}$. Допустим, для некоторого r_0 неравенство $(x, c) \geq r_0$ является следствием системы $Ax \geq b$. Тогда $r_0 \in \overline{K} \cap P = \emptyset$, что невозможно. Поэтому для любого r неравенство $(x, c) \geq r$ не может являться следствием совместной системы $Ax \geq b$. Отсюда $v = -\infty$.

Таким образом, в случае (10) возможны лишь две ситуации:

$$v = +\infty, v^* = -\infty \text{ или}$$

$$v = -\infty, v^* = -\infty.$$

3.2. Бесконечность оптимального значения двойственной задачи

Справедливо следующее утверждение.

Утверждение 1. Если выполняется (11), то

$$v = \sup \{r : [c; r] \in \overline{K}\}, \quad (13)$$

$$v^* = -\infty.$$

Доказательство. Обозначим через r_0 правую часть равенства (13).

Так как $\overline{K} \cap P \neq \emptyset$, то $r_0 > -\infty$.

Если $r_0 = +\infty$, то для любого достаточно большого r выполняется $[c; r] \in \overline{K}$, т.е. $(x, c) \geq r$ является неравенством-следствием системы $Ax \geq b$. Для совместной системы $Ax \geq b$ это невозможно. Поэтому система $Ax \geq b$ несовместна, откуда $v = r_0 = +\infty$.

Пусть теперь r_0 конечно.

Докажем, что система $Ax \geq b$ совместна. Допустим, что это не так. Тогда из [5] получаем $[0; 1] \in \overline{K}$. Так как $\overline{K} \cap P \neq \emptyset$, то при некотором r_1 имеем $[c; r_1] \in \overline{K}$. Тогда для любого $s > r_1$ имеем

$$[c; s] = (s - r_1) \cdot [0; 1] + [c; r_1] \in \overline{K}.$$

Отсюда вытекает $r_0 = +\infty$, что противоречит конечности r_0 . Таким образом, система $Ax \geq b$ совместна.

Возьмем произвольное r_1 , при котором $[c; r_1] \in \overline{K}$. Неравенство $(x, c) \geq r_1$ является следствием совместной системы $Ax \geq b$. Поэтому $v \geq r_1$, откуда

$$v \geq \sup \{r : [c; r] \in \overline{K} \cap P\} = r_0.$$

С другой стороны, так как $(x, c) \geq v$ является следствием совместной системы $Ax \geq b$ 2-го рода [7], то $[c; v] \in \bar{K}$, откуда $v \leq r_0$. Таким образом, $v = r_0$.

Вторая часть утверждения 1 вытекает из равенства $K \cap P = \emptyset$. В этом случае система ограничений задачи (6) несовместна, откуда $v^* = -\infty$. Утверждение 1 доказано.

3.3. Конечность оптимальных значений пары двойственных задач

Утверждение 2. Если выполняется (12), то

$$\begin{aligned} v &= \sup \{r : [c; r] \in \bar{K}\}, \\ v^* &= \sup \{r : [c; r] \in K\}. \end{aligned}$$

Доказательство. Первая часть утверждения совпадает с (13).

Обозначим $r_0 = \sup \{r : [c; r] \in K\}$. Тогда существует последовательность $\{r_j\}$, такая, что $\lim_{j \rightarrow \infty} r_j = r_0$ и $[c; r_j] \in K$. Отсюда каждый вектор $[c; r_j]$ можно представить как конечную положительную линейную комбинацию элементов множества $\{[a_\alpha; b_\alpha], \alpha \in \Omega\}$. Коэффициенты этих комбинаций являются допустимыми решениями двойственной задачи (6) со значениями целевой функции r_j . Следовательно, $v^* \geq r_0$.

Обратно: так как $K \cap P \neq \emptyset$, то система ограничений задачи (6) совместна. Тогда существует последовательность $\{u^i\}$ допустимых решений задачи (6), такая что $\lim_{i \rightarrow \infty} (b, u^i) = v^*$. Поэтому для каждого натурального i имеем $[c; (b, u^i)] \in K$, откуда $r_0 \geq v^*$.

Итак, $r_0 = v^*$. Утверждение 2 доказано.

3.4. Критерий наличия разрыва соотношения двойственности

Построим с помощью конуса K и прямой P одномерный интервал S , который назовем характеристическим интервалом задачи ПБЛП. С учетом утверждения 2 определим интервал S следующим образом:

$$S = [\sup \{r : [c; r] \in K\}, \sup \{r : [c; r] \in \bar{K}\}]. \quad (14)$$

Интервал S , если он не пуст, представляет собой точку, отрезок, луч или всю прямую P и расположено в пространстве R^{n+1} . Если интервал S не вырождается в точку, то он содержится на границе замыкания конуса K , но не содержится в самом конусе. В этом случае конус K не является замкнутым. Таким образом, интервал S содержится в незамкнутой части границы конуса K .

Интервал S позволяет определить оптимальные значения пары двойственных задач (5) и (6) и, следовательно, наличие разрыва двойственности.

Теорема 1. Для характеристического интервала S имеют место равенства

$$v = \sup \{r : [c; r] \in S\}, \quad (15)$$

$$v^* = \inf \{r : [c; r] \in S\}. \quad (16)$$

Доказательство. Теорема вытекает из определения интервала S и утверждений 1 и 2.

Следствие 1. (Критерий разрыва двойственности). Пусть $v^* > -\infty$. У пары задач (5) и (6) существует разрыв двойственности тогда и только тогда, когда множество S имеет непустую внутренность.

3.5. Существование оптимальных решений

Из вышесказанного следует, что конус K , прямая P и, вообще говоря, направление прямой P определяют оптимальные значения задач (5) и (6).

Исследуем вопрос о разрешимости двойственной задачи. Двойственная задача (6) достигает своего значения тогда и только тогда, когда супремум в утверждении 2 достигается.

Так как при переходе от множества $\{[a_\alpha; b_\alpha], \alpha \in \Omega\}$ к конусу K ограничения задачи (5) теряют свою индивидуальность, то восстановить оптимальное решение задачи (6), вообще говоря, невозможно.

Рассмотрим разрешимость исходной задачи. Известно, что в случае совместности системы ограничений задачи (5) функция оптимума $u(c)$ является собственной, вогнутой и по утверждению 2

$v(c) = \sup \{r : [c; r] \in \bar{K}\}$. Следствие 23.5.3 из [8] утверждает, что субдифференциал $\partial v(c)$ представляет собой всё множество оптимальных решений задачи (5). Из геометрического смысла субдифференциала получаем следующее утверждение.

Утверждение 3. Задача (5) имеет оптимальное решение тогда и только тогда, когда через точку $[c; v(c)]$ можно провести гиперплоскость, такую, что ее подграфик содержит конус $\bar{K}$.

3.6. Связь с ранее известными результатами

1. Из теоремы 3 статьи [9] вытекает следующее. Если конус K замкнут, то двойственная задача (6) достижима. Если конус $\{[Ax + z; (x, c) - r] : x \in R^n, z \geq 0, r \geq 0\}$ замкнут в декартовом произведении $R^\Omega \times R^1$ с топологией покоординатной сходимости, то исходная задача достижима. А из утверждения 2 вытекает, что из одного конуса K можно определить достижимость задач (5) и (6).

2. В пространстве R^Ω с покоординатной сходимостью положительный конус не имеет внутренности. Поэтому условия типа Слейтера (например, существует x_0 такой, что $(x_0, a_\alpha) > b_\alpha$ для каждого $\alpha \in \Omega$) недостаточны для анализа разрыва двойственности. Это и не удивительно, так как данное условие Слейтера равносильно $[x_0; -1] \in \text{int } K^*$ и не связано с границей конуса K , от которой, в соответствии с утверждением 2, зависит разрыв.

3. Из утверждений 1 и 2 легко вывести теорему о слабой двойственности [4] в несколько измененной формулировке: если $v(c_0)$ конечно, то

$$v^*(c_0) = \lim_{dB(c_0) \rightarrow 0} \sup_{c \in B(c_0)} v^*(c),$$

где $B(c_0)$ – шар с центром в точке c_0 и радиусом $dB(c_0)$.

3.7. Противоположная задача ЛП

Заменяем в исходной задаче (5) критерий и знаки неравенств в системе ограничений на противоположные. Полученную задачу назовем противоположной задачей ЛП:

$$v' = \sup \{(x, c) : Ax \leq b\}. \quad (17)$$

Двойственной к ней является задача

$$v'^* = \inf \{(b, u) : A^T u = c, u \geq 0\}. \quad (18)$$

Повторное применение процедуры построения противоположной задачи возвращает нас к исходной задаче (5). Таким образом, пара противоположных задач и пара двойственных задач ЛП обладают одним и тем же свойством взаимной обратимости.

Найдем v' . Используем известное правило замены критериев при оптимизации произвольной функции $f(x)$:

$$\sup f(x) = -\inf (-f(x)).$$

Тогда $v' = -\inf \{(x, -c) : -Ax \geq -b\}$. Используя утверждение 2, продолжаем:

$$v' = -\sup \{r : [-c; r] \in -\bar{K}\} = -(-\inf \{-r : [-c; r] \in -\bar{K}\}) = \inf \{-r : [-c; r] \in -\bar{K}\}.$$

Сделаем замену $t = -r$. Получаем

$$v' = \inf \{t : [-c; -t] \in -\bar{K}\} = \inf \{t : [c; t] \in \bar{K}\}.$$

Аналогично получаем

$$v'^* = \inf \{t : [c; t] \in K\} \leq \sup \{t : [c; t] \in K\} = v^*.$$

Теорема 2. Для противоположной системы выполняются соотношения

$$v' = \inf \{t : [c; t] \in \bar{K}\},$$

$$v'^* = \inf \{t : [c; t] \in K\}.$$

При этом справедливо тройное неравенство

$$v \geq v^* \geq v'^* \geq v'. \quad (19)$$

3.8. Допустимое множество противоположной задачи ЛП

Относительно ограничений противоположной задачи можно высказать следующие утверждения.

Утверждение 4. Если $M = \{x : Ax \geq b\}$ – непустое, ограниченное множество, не состоящее из одной точки, то $M' = \{x : Ax \leq b\} = \emptyset$.

Доказательство. Допустим противное: $Ax_0 \leq b$ при некотором x_0 . Тогда для любого x из $Ax \geq b$ вытекает $A(x - x_0) \geq 0$. Отсюда для любого $\lambda \geq 0$

$$A(x + \lambda(x - x_0)) = Ax + \lambda A(x - x_0) \geq b. \quad (20)$$

Так как $\{x : Ax \geq b\}$ ограничено, то из (20) получаем $x = x_0$, т.е. $\{x : Ax \geq b\} = \{x_0\}$, что противоречит условию.

Утверждение 5. Пусть $M = \{x : Ax \geq b\}$ неограничено. Если множество $M' = \{x : Ax \leq b\}$ непустое, то оно неограничено.

Доказательство. Пусть $x_0 \in M$. Из неограниченности M вытекает, что найдется ненулевой вектор e , такой, что для любого $t \geq 0$ выполняется $x_0 + te \in M$, т.е. $A(x_0 + te) \geq b$. С учетом $Ax_0 \geq b$ получаем $Ae \geq 0$. Допустим, $x' \in M'$. Тогда $A(x' - te) \leq b$, т.е. множество M' неограничено.

Из утверждений 4 и 5 следует, что противоположная задача может оказаться полезной именно при наличии конечного разрыва двойственности для задач (5) и (6), так как этот разрыв существует только тогда, когда $\{x : Ax \geq b\}$ неограничено.

4. Примеры

Пример 1. В следующем примере исходная и противоположная задачи имеют конечные разрывы двойственности:

$$\begin{cases} \inf x_2 \\ x_2 \geq 1, \\ x_2 \geq 2, \\ (1/n)x_1 + x_2 \geq 0, \\ (1/n)x_1 + x_2 \geq 3, \quad (n=1, 2, \dots) \end{cases}$$

Конус K натянут на следующие точки: $C = [0, 1; 1]$, $B = [0, 1; 2]$, $D^n = [1/n, 1; 0]$, $A^n = [1/n, 1; 3]$, не замкнут и содержится в пространстве R^3 (рис. 1).

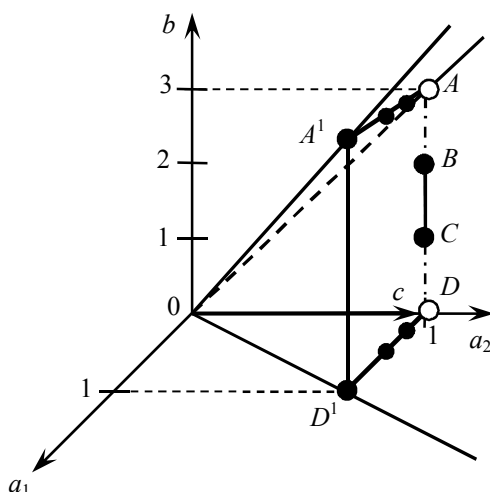


Рис. 1. Трехмерный конус K с незамкнутой границей

Последовательность $\{A^n\}$ сходится к точке $A = [0, 1; 3]$, $\{D^n\}$ сходится к точке $D = [0, 1; 0]$, причем точки A и D лежат на незамкнутой границе конуса K . Целевой вектор $c = [0, 1; 0]$. Прямая P прохо-

дит через точки A, B, C и D . Отрезок $[A, B]$ является характеристическим для исходной задачи, а отрезок $[C, D]$ является характеристическим для противоположной задачи. Используя теорему 2, получаем

$$v = 3, v^* = 2, v'^* = 1, v' = 0.$$

Пример 2. Приведем пример задачи ПБЛП с тремя переменными, для которой разрыв двойственности выполняется для неколлинеарных целевых векторов. Система ограничений задачи имеет вид

$$\begin{cases} x_1 + x_2 \geq 1, \\ x_1 - x_2 \geq 1, \\ (1 - 1/n)x_1 + x_2 + (1/n)x_3 \geq 2, \\ (1 - 1/n)x_1 - x_2 + (1/n)x_3 \geq 2, \quad n = (2, 3, \dots). \end{cases}$$

Для рассматриваемой системы конус K натянут на следующие точки: $A^n = [1 - 1/n, 1, 1/n; 2]$, $B^n = [1 - 1/n, -1, 1/n; 2]$, $C = [1, 1, 0; 1]$, $D = [1, -1, 0; 1]$.

Последовательность $\{A^n\}$ сходится к точке $A = [1, 1, 0; 2]$, $\{B^n\}$ сходится к точке $B = [1, -1, 0; 2]$. Рассмотрим трехмерную гиперплоскость $H: ([a; b], d) = 0$, где $d = [0, 0, 1; 0]$. Точки A, B, C, D , а также вектор h лежат в гиперплоскости H , а точки A^n и B^n – в полуплоскости $([a; b], d) > 0$. Поэтому точки A и B принадлежат замыканию конуса K , но не самому K , т.е. лежат на незамкнутой границе конуса K .

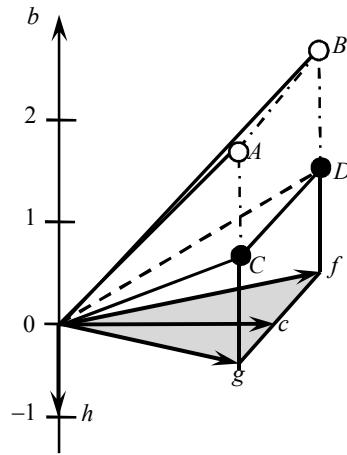


Рис. 2. Трехмерное сечение четырехмерного конуса K трехмерной гиперплоскостью

Построим сечение $K^1 = K \cap H$ (рис. 2). Возьмем два целевых вектора $g = [1, 1, 0; 0]$ и $f = [1, -1, 0; 0]$. Векторы g, f линейно независимы и принадлежат конусу K^1 . Возьмем вектор c как произвольную выпуклую комбинацию этих векторов $c = \alpha \cdot g + (1 - \alpha) \cdot f$, $0 \leq \alpha \leq 1$. Для целевого вектора c по утверждению 2 имеем:

- 1) $v = \sup \{ r : [c; r] \in \overline{K} \} = 2$, и супремум достигается в точке $\alpha \cdot A + (1 - \alpha) \cdot B$;
- 2) $v^* = \sup \{ r : [c; r] \in K \} = 1$, и супремум достигается в точке $\alpha \cdot C + (1 - \alpha) \cdot D$.

5. Применение MATLAB для анализа соотношений двойственности

Для проведения численного анализа соотношений двойственности (19) разработана MATLAB-программа [12]. В программе используются стандартные оптимизационные функции *fseminf* и *linprog*, предназначенные для приближенного решения полубесконечных нелинейных и конечномерных линейных задач оптимизации соответственно. Функция *fseminf* может быть использована для нахождения оптимальных значений v и v' задач (5), (17), а *linprog* – для нахождения оптимальных значений v^* и v'^* задач (6), (18).

Особенностью MATLAB-программы является указание конечного значения n в полубесконечных ограничениях исходной задачи, что приводит к замене задачи ПБЛП задачей ЛП с конечным числом

ограничений. Стоит отметить, что данная замена может привести к изменению оптимального значения обеих задач. В перспективе предполагается разработка новой MATLAB-программы с возможностью для пользователей указания бесконечной последовательности ограничений.

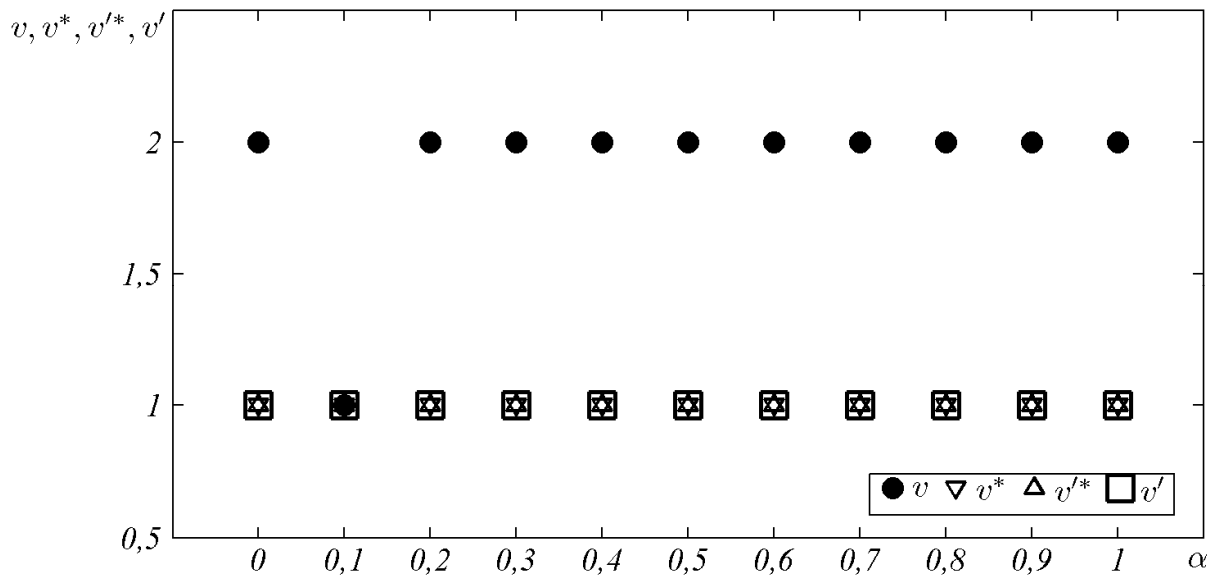


Рис. 3. Значения v, v^*, v', v''^* при $\{\alpha \mid \alpha = 0,1 \cdot k, 0 \leq k \leq 10\}$

На рис. 3 представлены результаты численного решения примера 2. Из рисунка видно, что практически при всех α $v = 2, v^* = 1, v' = 1, v''^* = 1$.

Заключение

В работе продемонстрирован геометрический подход к анализу полубесконечных задач ЛП. Данный подход позволяет моделировать задачи с заданными свойствами разрыва двойственности. Показано, что разрыв полностью определяется свойствами границы конуса коэффициентов. Ранее приводились примеры с разрывом двойственности для изолированных целевых векторов. В этом случае множество целевых векторов, для которых возникает конечный разрыв, образует луч. В данной статье показано, что множество целевых векторов с разрывом может представлять собой сложные многомерные области. Таким образом, задачи с разрывом двойственности не являются экзотическим нежелательным объектом, а должны исследоваться целенаправленно.

Разрыв двойственности возникает, когда ограничения множества допустимых решений имеют следующий недостаток. При бесконечно малых возмущениях правых частей ограничений допустимое множество на некоторых направлениях начинает аномально расширяться. Это проявляется в резком изменении оптимального значения исходной задачи. Таким образом, качество ограничений, которые определяют геометрический объект в виде допустимого множества, можно считать недостаточным. Наш подход позволяет находить направления, на которых проявляется аномальное расширение геометрических объектов. Подобные явления могут возникать в различных областях техники, экономики, социологии и политики.

Данный подход может использоваться также при анализе структуры незамкнутой границы выпуклых множеств в многомерных пространствах, поскольку мы с трудом можем представить себе многомерные множества, а также строить график функций с количеством переменных более двух.

ЛИТЕРАТУРА

1. Goberna M.A., Lopez M.A. Linear Semi-Infinite Optimization. Chichester : Wiley, 1998. 356 p.

2. Glashoff K., Gustafson S.A. Linear Optimization and Approximation: An Introduction to the Theoretical Analysis and Numerical Treatment of Semi-Infinite Programs. N.Y. : Springer, 1983. 212 p.
3. Karney D.F. Duality gaps in semi-infinite linear programming // Math. Progr. 1981. V. 20, No. 1. P. 129–143.
4. Duffin R.J., Karlovitz L.A. An infinite linear program with a duality gap // Management Sci. 1965. V. 12, No. 1. P. 122–134.
5. Черников С.Н. Линейные неравенства. М. : Наука, 1968. 489 с.
6. Jeroslow R.G. Uniform quality in semi-infinite convex optimization // Math. Progr. 1983. V. 27, No. 2. P. 144–155.
7. Еремин И.И., Астафьев Н.Н. Введение в теорию линейного и выпуклого программирования. М. : Наука, 1976. 192 с.
8. Rockafellar R.T. Convex analysis. Princeton, NJ : Princeton University Press, 1970. 260 p.
9. Kretschmer K.S. Programmes in paired spaces // Canad. J. Math. 1961. V. 13. P. 221–238.
10. Schechter M. Linear program in topological vector spaces // J. Math. Anal. Appl. 1972. V. 37. P. 492–500.
11. Трофимов С.П. Критерий разрыва двойственности для полубесконечных задач линейного программирования // Противоречивые модели оптимизации : сб. науч. тр. Свердловск : ИММ УНЦ АН СССР, 1987. С. 64–70.
12. Программа численного анализа соотношений двойственности. Репозиторий MATLAB-программы. URL: <https://github.com/re3burn/DGA> (дата обращения: 20.08.2016).

Трофимов Сергей Павлович, канд. физ.-мат. наук, доцент. E-mail: tsp61@mail.ru

Иванов Алексей Витальевич. E-mail: av.ivanov.2014@yandex.ru

Уральский федеральный университет (г. Екатеринбург)

Поступила в редакцию 20 марта 2016 г.

Trofimov Sergey P., Ivanov Aleksey V. (Ural Federal University, Russian Federation).

The duality gap in semi-infinite linear programming and the quality analysis of geometrical objects' constraints.

Keywords: linear programming problem; duality gap; feasible set; conical hull of coefficients; system of linear inequalities; duality relation criteria; convex unclosed cone.

DOI: 10.17223/19988605/38/6

The paper focuses on the universal problem of linear programming (LP) with an infinite number of linear constraints and a finite number of variables. The authors demonstrate how the LP problem, which limits use a convex closed cone, is reduced to the above-mentioned one. The geometric interpretation using the conical K shell of the system constraints coefficients is proposed for a pair of the dual LP problems. The other two cones (i.e. the K cone epigraph and hypograph) are additionally constructed to analyze the dual problems. The one-dimensional interval using three cones and the target vector of the original problem is also constructed. The characteristics of the interval allow us to determine the main characteristics of the original problem: the optimal values and the solvability of both problems. The interval lies on the boundary of the multidimensional K cone.

With regard to the results achieved, we suggest a qualitative analysis of the geometric method of the semi-infinite LP problem without using either the Gauss method being commonly used or the simplex method. It is shown that the basic information of a pair of the dual problems is contained in the characteristics of the surface of an unclosed convex K cone.

A new type of dual problem is put forward for the LP problem. It is different from the classic one based on the Lagrange multipliers. It is sufficient to replace the criterion and the inequalities of constraints signs with the opposite ones. The similarity between the two is in the fact that their characteristics are determined by the epigraph and the hypograph of the K cone. Thus, they equally provide information on the surface of an unclosed K cone.

The paper shows that the duality gap is not an unwanted extraordinary case. The existence of a gap indicates the places where the feasible set starts to expand abnormally at infinitesimal relaxation of constraints. Thus, gap analysis allows to determine the "weak" places by setting constraints of a feasible set. Similar problems arise in politics, economy and any other sphere in which various constraints are imposed on a set of objects.

We construct numerical example of a semi-infinite LP problem with three variables. The characteristics of the surface of an unclosed four-dimensional K cone are analyzed. It is shown that the target vectors with duality gap do not form an isolated ray and fill the two-dimensional region in.

The example of the code allowing to carry out the numerical analysis of duality relations by the use of MATLAB are given. The program is based on such standard optimization functions as *fseminf* and *linprog* designed to find approximate solutions of finite-dimensional linear and non-linear semi-infinite optimization problems respectively. A distinguished feature of the MATLAB program is replacing of a semi-infinite linear programming original problem by a LP problem with a finite number of constraints. It is noted that this substitution may change the optimum value of both problems. It is worth mentioning that the authors are planning to develop a new MATLAB-program with the ability for users to specify an infinite sequence of constraints.

REFERENCES

1. Goberna, M.A. & Lopez, M.A. (1998) *Linear Semi-Infinite Optimization*. Chichester: Wiley.
2. Glashoff, K. & Gustafson, S.A. (1983) *Linear Optimization and Approximation: An Introduction to the Theoretical Analysis and Numerical Treatment of Semi-Infinite Programs*. New York: Springer.
3. Karney, D.F. (1981) Duality gaps in semi-infinite linear programming – an approximation problem. *Math. Progr.* 20(1). pp. 129–143. DOI: 10.1007/BF01589340
4. Duffin, R.J. & Karlovitz, L.A. (1965) An infinite linear program with a duality gap. *Management Sci.* 12(1). pp. 122–134. DOI: 10.1287/mnsc.12.1.122
5. Chernikov, S.N. (1968) *Lineynye neravenstva* [Linear Inequalities]. Moscow: Nauka.

6. Jeroslow, R.G. (1983) Uniform quality in semi-infinite convex optimization. *Math. Progr.* 27(2). pp. 144–155.
7. Eremin, I.I. & Astafiev, N.N. (1976) *Introduction to the Theory of Linear and Convex Programming*. Moscow: Nauka.
8. Rockafellar, R.T. (1970) *Convex analysis*. New Jersey: Princeton University Press.
9. Kretschmer, K.S. (1961) Programmes in paired spaces. *Canad. J.Math.* 13. pp. 221–238. DOI: 10.4153/CJM-1961-019-2
10. Schechter, M. (1972) Linear program in topological vector spaces. *J. Math. Anal. Appl.* 37. pp. 492–500. DOI: 10.1016/0022-247X(72)90290-9
11. Trofimov, S.P. (1987) Kriteriy razryva dvoystvennosti dlya polubeskonechnykh zadach lineynogo programmirovaniya [A criterion for a discontinuity in the duality for semi-infinite problems of linear programming]. In: *Protivorechivye modeli optimizatsii* [Inconsistent Optimization Model]. pp. 64–70.
12. *Numerical analysis of duality relations program*. MATLAB Program Repository. [Online] Available from: <https://github.com/re3burn/DGA>. (Accessed: 20th August 2016). (In Russian).

ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

УДК 004.891

DOI: 10.17223/19988605/38/7

М.М. Горохов, Д.А. Переведенцев

ПРОЕКТИРОВАНИЕ БАЗЫ ЗНАНИЙ ЭКСПЕРТНОЙ СИСТЕМЫ УПРАВЛЕНИЯ НАУЧНЫМИ ПРОЕКТАМИ

Описан процесс разработки базы знаний экспертной системы управления научными и инновационными проектами, основывающийся на анализе конкретной ситуации, степени развития проекта в определенный момент времени, оценке его состояния, а также анализе предыдущих условий работы с данным проектом.

Ключевые слова: экспертная система; управление научными проектами; поле знаний; база правил; системный анализ.

С организационной и технологической точки зрения процесс оценки потенциала новых проектов сегодня остается еще достаточно трудоемким и слабоформализованным в рамках производственной деятельности предприятия. При выборе конкретного проекта имеет место совокупность научных, коммерческих, организационных, ресурсных и других составляющих успешного развития проекта, предопределенных спецификой его развития и определяющих качество описания проекта, и изменение любого из слагаемых в значительной мере сказывается на степени реализуемости проекта.

Необходимость переоценки внешних и внутренних факторов влияния на развитие проекта требует частого обращения к экспертам соответствующей области, что значительно осложняет процесс оценки и анализа научных и инновационных проектов, а следовательно, и принятия решений. Эффективное решение задачи прогнозирования изменения экономического и социального эффекта научных проектов на этапах НИР и ОКР заключается в использовании экспертных систем, построенных на совокупности формализованных и описанных знаний эксперта конкретной предметной области, структурированных в определенном порядке для построения процесса принятия решения.

В свою очередь, для успешного функционирования экспертной системы необходимо не только учитывать имеющиеся данные, но и иметь возможность преобразовать информацию в форму, позволяющую эффективно оценить описанные условия и ситуацию и выдать рекомендательные решения.

1. Формирование поля знаний

Исходным понятием на стадии проектирования экспертной системы (ЭС) является получение и структурирование знаний эксперта, именуемое полем знаний, определяемое как заданное в различных форматах неформальное описание основных понятий предметной области и взаимосвязей между ними [1. С. 59].

Поле знаний задается описанием состояния среды (ситуации) и совокупностью всех возможных шагов решения задачи – пространством состояний. Поиск решений основывается на формализованном, удобном для работы алгоритме представления задач и описании последовательности смены состояний, ведущих к достижению поставленной цели. Процесс поиска решения задачи также носит название вывода цели. Посредством заданных действий алгоритм переводит задачу из одного состояния в другое. Для описания указанного процесса воспользуемся деревом поиска [2. С. 50–51].

Исходя из вариантов представления результатов принятия решений и заданной многомерной модели [3. С. 59], построим поле знаний нашей предметной области, определяемой концептами – основными элементами системы СППР, и их атрибутами – допустимыми значениями, таким образом формируя концептуальную структуру поля знаний (рис. 1).

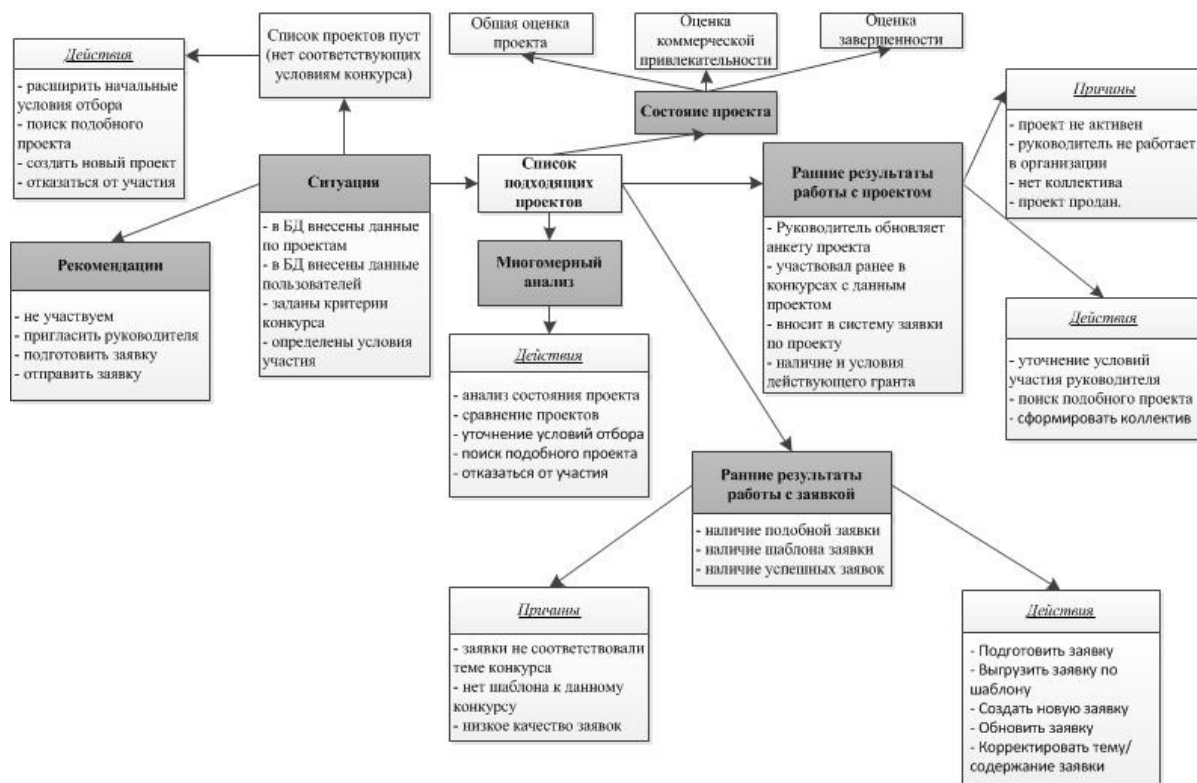


Рис. 1. Концептуальная составляющая поля знаний

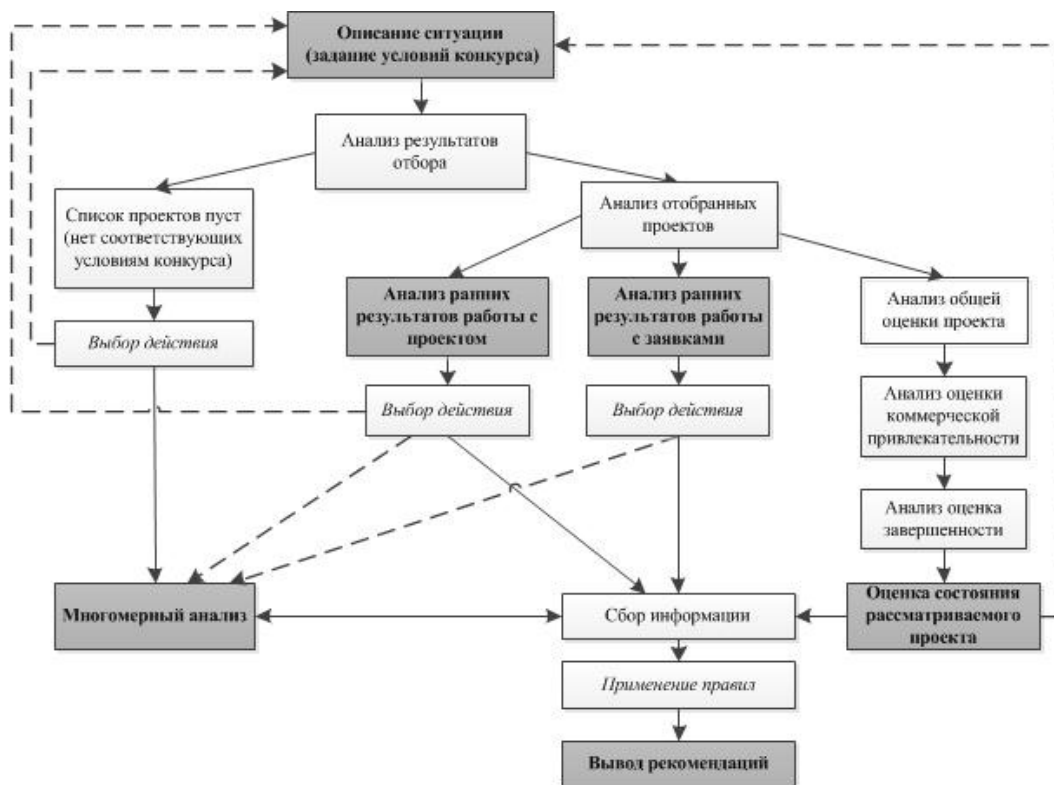


Рис. 2. Функциональная составляющая поля знаний

Структура функциональной составляющей поля знаний моделирует основные связи или отношения между концептами и их атрибутами, образующими поле знаний. Эти связи отражают стратегию принятия решений экспертом (рис. 2).

С целью подробного представления рассуждений ЭС выдаваемые рекомендации целесообразно представить в виде промежуточных выводов и подведения итогового решения, представляющих собой отдельные блоки информации.

Таким образом, накопление и организация специальных знаний – одна из самых важных задач ЭС, целям их расширения и накопления служит база знаний, которая, по сути, является ядром ЭС. При этом знания должны быть выражены в явном виде и организованы так, чтобы упростить принятие решений [4]. Это требование заставляет экспертные системы оперировать не только данными, но и понятиями концептуальными знаниями, выраженными на естественном языке и в терминах предметной области. Правила, содержащиеся в базе знаний, соответствуют заданной прикладной программе и могут быть различными по формату представления.

2. Формирование базы правил

Описание концептов и их атрибутов позволяет перейти к построению базы фактов и правил, для этого применяются формальные языки, обычно напоминающие естественный язык, но гораздо более строгие и ограниченные. Существует несколько базовых специализированных машинных языков представления знаний, сегодня наибольшее распространение в ЭС получил язык продукционных правил, имеющих форму ЕСЛИ «Условие» – ТО «Событие».

Продукционные правила обеспечивают естественный способ описания процессов, управляемых сложной и изменяющейся средой. Правила дают возможность описать ход решения задачи, не имея заранее алгоритма этого решения, а также корректировать способ решения путем добавления новых правил, не изменяя существующих, что обеспечивает высокую модульность базы знаний [2. С. 45].

В общем продукционная модель имеет следующий вид (1):

$$i = \langle T; L; A \rightarrow B; Q \rangle, \quad (1)$$

где T – описание ситуации; L – условие, при котором продукция активизируется; $A \rightarrow B$ – ядро продукции, представляющее собой условие: если A , то B ; Q – постусловие продукционного правила, описывает действия, совершаемые после реализации B .

В разработанной информационно-аналитической системе (ИАС) [3. С. 60] в качестве исходной информации для формирования списка рекомендуемых действий используются:

- значения оценок проекта, полученных при введении проекта в базу данных;
- данные предыдущих результатов подачи заявок по проекту;
- описание ситуации, в условиях которой реализуется проект.

Конструирование вывода в ЭС обычно подразумевает следующие операции: хранение значений всех атрибутов и проверка на истинность в процессе решения задачи – истинным считается факт, если он может быть выведен по законам формальной логики из имеющейся базы фактов и правил. При обращении к правилам согласно заданной ситуации обновляется содержимое рабочей памяти, при нахождении правила (B), сопоставимого факту (A), правило срабатывает и выводится соответствующее решение.

Пусть y – конкурс; $D(y)$ – условия конкурса; (x) – проект, отображаемый в списке подходящих для участия в конкурсе (y); $S(x)$ – состояние проекта; $F(x)$ – ситуация, в которую заключен проект; $T(x)$ – результаты участия в предыдущих конкурсах и $R(x)$ – окончательное решение об участии в конкурсе. При этом наличие и отсутствие каких-либо данных задаются понятиями «истина» (true) и «ложь» (false) соответственно.

Источником данных для оценки проекта служит специально разработанная база данных научных сотрудников, проектов и конкурсов, реализованная на платформе SQL Microsoft SQL Server [5. С. 34].

На основе введенных переменных опишем правила БЗ и условия их срабатывания.

Начальным условием актуализации базы правил является выполнение условия соответствия проекта заданным параметрам отбора: $x = D(y)$.

Далее срабатывают остальные правила.

Блок «Решение» выводит итоговые рекомендации исходя из заданных условий, состояния проекта и оценки активности по проекту. Обозначим ее функцией (2):

$$R(x) = \{ \langle S(x), F(x), T(x), \rangle \}, \quad (2)$$

где $R(x)$ – решение по участию проекта (x) в выбранном конкурсе, $\forall(x)\exists R(x) = \{ \text{решение} \}$; $S(x)$ – вывод о состоянии проекта (x), $\forall(x)\exists S(x) = \{ \text{вывод} \}$; $F(x)$ – рекомендации по работе с проектом (x) в случае участия в выбранном конкурсе, $\forall(x)\exists F(x) = \{ \text{рекомендация} \}$; $T(x)$ – рекомендации по подготовке заявки по проекту (x) на выбранный конкурс, $\forall(x)\exists T(x) = \{ \text{рекомендация} \}$.

Правило базы знаний для формирования окончательного решения по проекту представляется в следующем виде:

«Выдача решения» = <«Решение», блок актуализирован, $S(x) \wedge F(x) \wedge T(x) \rightarrow \text{вывод} = \{ \text{решение} \}$, иначе = «недостаточно данных», решение об участии в конкурсе».

В случае необходимости получения дополнительной информации аналитиком для выбора проекта на конкурс он обращается к многомерному анализу (см. рис. 2), обозначим его как $M(x)$ – результаты многомерного анализа проекта (x), $\forall(x)\exists M(x) \in \{ \text{false}; \text{true} \}$.

Тогда правило базы знаний для формирования окончательного решения по проекту представляет в следующем виде:

«Выдача решения» = <«Решение», блок актуализирован, $S(x) \wedge (F(x) \wedge M(x)) \wedge (T(x) \wedge M(x)) \rightarrow \text{вывод} = \{ \text{решение} \}$, иначе = «недостаточно данных», решение об участии в конкурсе».

Далее в таблице приведен пример записи правил с учетом концептуального представления знаний эксперта.

Пример записи правил

Блок	Условия			Промежуточный вывод / решение
	If	and	or	
1	Проект соответствует условиям конкурса	ОО = {201;300} ∧ ИКП = {0;0,5} ∧ ОЗ = {81;100}	ОО = {301;400} ∧ ИКП = {0,51;0,8} ∧ ОЗ = {21;40}	Проект имеет высокий научный потенциал, подходит для участия в выбранном конкурсе
2	Не имеет действующего гранта	Ранее подготовлено 3 и более заявок		Руководитель активно участвует в конкурсах. Проект готов к участию
3	Имеется заполненная заявка	Имеется шаблон заявки		В системе имеется шаблон заявки для данного конкурса, есть возможность автоматически заполнить заявку
4	«Высокий научный потенциал»	«Готов к участию»	«В системе есть шаблон»	Сформировать и подать заявку

Примечания. ОО – общая оценка проекта, ИКП – индекс коммерческой привлекательности, ОЗ – оценка завершенности проекта.

Таким образом, процесс получения решения в разработанной ЭС имеет прямую цепочку вывода: поиск ведется на основе данных (data-driven search), процесс решения задачи начинается с исходных фактов, затем, применяя допустимые правила изменения состояний, осуществляется переход к новым фактам, и так до тех пор, пока цель не будет достигнута, в нашем случае это формирование эффективной рекомендации специалисту по работе с проектами.

Заключение

Разработанная база знаний ЭС повышает эффективность анализа и обработки данных о научных проектах, способствуя снижению информационной нагрузки, а значит, и временных затрат соответствующих специалистов и, как следствие, повышается объем выполненной ими работы, что в силу ускорения процесса накопления опыта и автоматизации работы с проектами, несомненно, положительно скажется на результатах реализации научных проектов, а также их коммерциализации.

А повышение эффективности обработки информации о состоянии проекта на определенный момент времени в условиях заданной ситуации и выбор форм ее представления лицу, принимающему решение, приводят к росту определенности выбора управленческого решения.

ЛИТЕРАТУРА

1. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. СПб. : Питер, 2000. 384 с.
2. Муромцев Д.И. Введение в технологию экспертных систем. СПб. : СПб ГУ ИТМО, 2005. 93 с.
3. Переведенцев Д.А. Разработка UML – модели информационно-аналитической системы перспективных научных проектов // Вестник ИжГТУ имени М.Т. Калашникова. 2015. № 4. С. 58–60.
4. Суменков М.С., Суменков С.М. Экспертные системы при принятии решений на предприятии // Бизнес. Менеджмент. Право. 2003. № 2. URL: http://www.bmpravo.ru/show_stat.php?stat=193 (дата обращения: 15.05.16).
5. Благодатский Г.А., Переведенцев Д.А. Информационно-аналитическая система поддержки научной деятельности предприятий и ВУЗов «UNIPROJECT» // Сборник материалов XX Республиканской выставки-сессии студенческих инновационных проектов. Ижевск : Иннова, 2015. С. 31–37.

Горохов Максим Михайлович, д-р физ.-мат. наук, профессор. E-mail: insys2005@mail.ru

Переведенцев Денис Алексеевич. E-mail: perevedentsev@mail.ru

ФГБОУ ВО «ИжГТУ им. М.Т. Калашникова» (г. Ижевск)

Поступила в редакцию 26 сентября 2016 г.

Gorokhov Maksim M., Perevedentsev Denis A. (Izhevsk State Technical University, Russian Federation).

Design of the knowledge base of expert system for management scientific projects.

Keywords: Expert System; management of science project; system analysis; IAS «UNIPROJECT».

DOI: 10.17223/19988605/38/7

This article describes the process of developing the knowledge base of the expert system for management scientific and innovative projects based on an analysis of the specific situation, the degree of development of the project at any given time, assessment of its condition, as well as the analysis of the previous conditions of work with the project. We present the algorithm and method of bring to the information from the database to the form, which allows to describe the conditions and assess the situation in the form of a rule base for obtainment interim findings and final decisions.

On the base of the analysis and formalization of the decision-making process for the management of scientific projects are determined the directions and content analysis of the available data, next, based on the ER model was developed multidimensional data model, allowing to determine the possible dimensions of the cube and their hierarchies. Dataset values are database tables that include quantitative and qualitative data.

Formalization and description of the expert knowledge of the studied subject area in the form of conceptual and functional models in the field of knowledge allowed us to determine the conditions and procedure for decision-making and develop a base of rules for the formation of interim findings and final decisions on working with a specific project.

With the purpose of detailed presentation of the reasoning ES issued recommendations it was agreed to submit an interim findings and summing the final decision as separate blocks of information. Thus, the final decision in each project depends on a General assessment of the project, a situation in which it is at the moment, as well as conditions of participation in the contest.

Block "project Status" contains information obtained from the processing of the calculated estimates of the project, the output status of the project is based on the analysis of the overall assessment, of index of commercial viability and the assessment of completeness.

Block "Work with project" outputs guidelines for working with the project, based on the results of the analysis of the situation the project development.

Block "Work with application" also lists recommendations for working with the project based on the analysis of previous results of the preparation of applications.

Block "Solution" displays a summary of recommendations based on the given conditions, the state of the project and evaluation activity for the project.

Thus, the process of obtaining the solutions in ES is a direct chain of inference: the search is based on data (data-driven search), the process of solving the problem begins with the source of the facts, then, by applying valid rules of state change is the transition to new facts, and so until then, until the goal is reached, in this case the results of the recommendations for the specialist for working with the projects.

REFERENCES

1. Gavrilova, T.A. & Khoroshevskiy, V.F. (2000) *Bazy znaniy intellektual'nykh sistem* [Knowledge bases of intelligent systems]. St. Petersburg: Peter.
2. Muromtsev, D.I. (2005) *Vvedenie v tekhnologiyu ekspertnykh sistem* [Introduction to expert systems]. St. Petersburg: ITMO.
3. Perevedentsev, D.A. (2015) Development of UML-Model of Information Analytical System of Promising Scientific Projects. *Vestnik IzhGTU imeni M.T. Kalashnikova – Vestnik of Kalashnikov Izhevsk State Technical University*. 4. pp. 58-60. (In Russian).
4. Semenov, M.S. & Sumenkov, S.M. (2003) Expert system in decision-making at the enterprise. *Business, Management, Law*. 2. [Online] Available from: http://www.bmpravo.ru/show_stat.php?stat=193. (Accessed: 15th May 2016).
5. Blagodatskiy, G.A. & Perevedentsev, D.A. (2015) [Information-analytical system of support of scientific activities of the enterprises and institutions "UNIPROJECT"]. *Proc. of the Republican exhibition of the 20th session of innovative student projects*. Izhevsk: Innova. pp. 31–37. (In Russian).

УДК 004.272

DOI: 10.17223/19988605/38/8

А.А. Пазников**ОПТИМИЗАЦИЯ ДЕЛЕГИРОВАНИЯ ВЫПОЛНЕНИЯ КРИТИЧЕСКИХ СЕКЦИЙ
НА ВЫДЕЛЕННЫХ ПРОЦЕССОРНЫХ ЯДРАХ***Работа выполнена при поддержке РФФИ (гранты № 15-07-02693, 15-37-20113, 15-07-00653, 16-07-00712, 15-07-00048).*

В работе рассмотрена задача оптимизации метода делегирования выполнения критических секций на выделенных процессорных ядрах (Remote Core Locking – RCL). Данный метод позволяет повысить масштабируемость многопоточных программ за счёт снижения конкурентности доступа к критическим секциям и пространственной локализации обращений к памяти. Предложен алгоритм оптимизации выделения памяти в программах на основе RCL в многоядерных системах с неоднородным доступом к памяти. Выполнено исследование на кластерной вычислительной системе. Показано, что оптимизация достигается благодаря выделению памяти на NUMA-узлах, на которых функционирует RCL-сервер.

Ключевые слова: RCL; многопоточное программирование; критические секции; масштабируемость.

В настоящее время имеет место значительное увеличение количества процессорных ядер в вычислительных системах (ВС) с общей памятью [1], относящихся к классам SMP и NUMA. В связи с этим возрастает потребность в создании масштабируемых параллельных программ, эффективных при большом количестве параллельных потоков.

Среди перспективных подходов, позволяющих обеспечить высокую масштабируемость параллельных программ, следует в первую очередь выделить алгоритмы и структуры данных, свободные от блокировок (lock-free) [2], и программную транзакционную память (software transactional memory) [3]. Недостатками неблокируемых алгоритмов и структур данных являются ограниченная область их применения и существенные трудозатраты, связанные с разработкой параллельных программ. Кроме того, пропускная способность таких структур часто сопоставима с аналогичными структурами данных, основанными на блокировках. Программная транзакционная память на сегодняшний день не обеспечивает достаточный уровень производительности многопоточных программ и не получила широкого применения в реальных приложениях.

Традиционный подход организации критических секций в многопоточных программах с помощью блокировок сегодня остаётся наиболее распространённым в многопоточном программировании. Блокировки проще в использовании по сравнению с неблокируемыми алгоритмами и структурами данных и обеспечивают высокий уровень производительности. Кроме того, основная часть существующих многопоточных программ разработана с использованием блокировок. Поэтому востребованным является создание масштабируемых алгоритмов реализации взаимного исключения.

Масштабируемость блокировок существенно зависит от решения проблем, связанных с конкурентным доступом (access contention) параллельных потоков в разделяемых областях памяти и пространственной локализацией обращений к кэш-памяти (cache locality). Конкурентный доступ имеет место при одновременном обращении нескольких потоков к критической секции, защищённой одним объектом синхронизации, что приводит к увеличению загруженности кэш-памяти. Локальность кэша имеет существенное значение в том случае, когда внутри критической секции выполняется обращение к разделяемым данным, которые перед этим использовались на другом ядре. Данное обстоятельство приводит к промахам по кэшу (cache misses) и значительному увеличению времени выполнения критических секций.

Среди наиболее эффективных подходов к реализации масштабируемых блокировок можно выделить CAS spinlocks, MCS-locks [4], Flat combining [5], CC-Synch [6], DSM-Synch [6], Oyama lock [7]. От-

дельно необходимо выделить ряд методов, предполагающих делегирование выполнения критических секций выделенному процессорному ядру для повышения локальности кэша [5, 8–10]. Работы [9, 10] посвящены разработке потокобезопасных структур данных (списков и хеш-таблиц) на основе делегирования выполнения критических секций. В статье [8] предлагается универсальное аппаратное решение, включающее набор процессорных инструкций для передачи управления выделенному ядру процессора. Flat Combining [5] относится к программно-реализуемым методам. В роли сервера, выполняющего критические секции, выступают все потоки. Однако выполнение процедуры передачи управления критической секцией между потоками приводит к снижению производительности даже при незначительном конкурентном доступе. Кроме того, данные алгоритмы не поддерживают блокировки потоков внутри критических секций как вследствие активного ожидания, так и из-за блокировки потока на уровне ядра операционной системы.

1. Метод делегирования выполнения критических секций на выделенных процессорных ядрах

Время выполнения критических секций в многопоточных программах складывается из времени передачи права выполнения блокировкой и времени выполнения инструкций критической секции (рис. 1). При передаче права владения блокировкой возникают накладные расходы вследствие переключения контекста, загрузки кэш-линии, содержащей глобальную переменную синхронизации, из оперативной памяти, активизации потока, выполняющего критическую секцию. На время выполнения инструкций критической секции существенно влияет пространственная локализация обращений к глобальным переменным. Традиционные алгоритмы взаимного исключения предполагают частое переключение контекста, которое приводит к вытеснению разделяемых переменных из кэш-памяти.

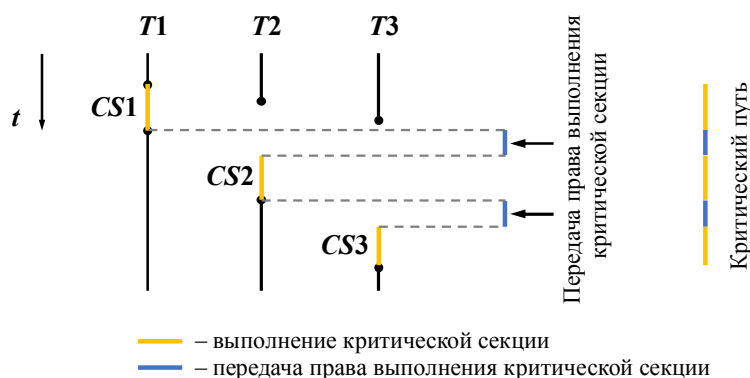


Рис. 1. Критический путь выполнения критических секций

В работе рассматривается метод выполнения критических секций на выделенных узлах процессора (Remote Core Locking – RCL) [11], позволяющий минимизировать время выполнения существующих многопоточных программ за счет уменьшения критического пути выполнения критических секций.

Данный метод предполагает замену в существующих программах высоконагруженных критических секций на удалённый вызов процедур для выполнения на выделенных процессорных ядрах (рис. 2). Все критические секции, защищённые одной блокировкой, выполняются потоком-сервером, работающим на выделенном ядре. Поскольку критические секции выполняются в отдельном потоке, отсутствуют накладные расходы на передачу права владения критической секцией.

RCL также позволяет сократить время выполнения инструкций критических секций. Это достигается тем, что разделяемые данные, защищённые блокировкой, с большой вероятностью находятся в локальной кэш-памяти ядра, на котором функционирует RCL-сервер, что обеспечивает снижение числа промахов по кэшу. Конкурентность доступа снижается за счёт использования отдельной выделенной кэш-линии для каждого потока-клиента, которая отвечает за хранение информации о критической секции и активного ожидания потока-клиента (рис. 3).

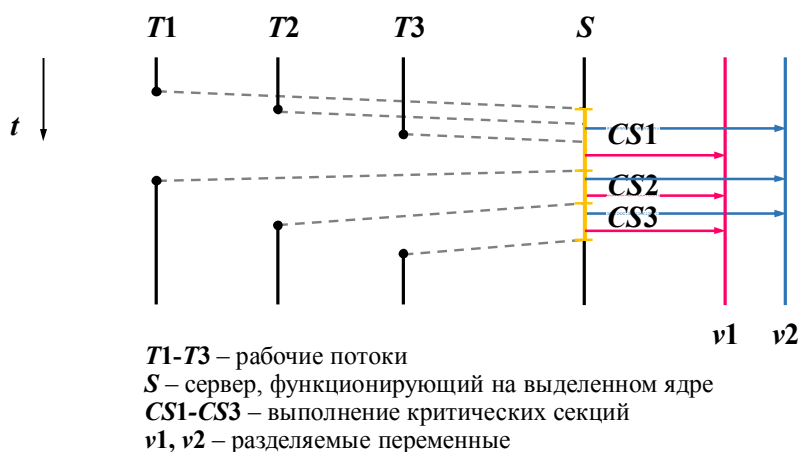


Рис. 2. Делегирование выполнения критических секций на выделенном процессорном ядре



Рис. 3. Таблица запросов на выполнение критических секций в RCL

В существующей реализации RCL не учитывается неоднородный доступ к памяти в NUMA-системах. В данной работе предлагаются алгоритмы, позволяющие минимизировать время доступа к памяти в системах с неоднородным доступом к памяти при выполнении многопоточных программ на базе RCL.

2. Оптимизация алгоритмов делегирования выполнения критических секций в NUMA-системах

В настоящее время вычислительные системы с неоднородным доступом к общей памяти (Non-uniform memory access – NUMA) (рис. 4) получили широкое распространение. Такие системы представляют собой композиции многоядерных процессоров, каждый из которых связан с локальной областью глобальной памяти. Процессор вместе с его локальной памятью формирует NUMA-узел (NUMA-node). Взаимодействие между процессорами и обращение к памяти осуществляются через шину (AMD HyperTransport, Intel Quick Path Interconnect). Обращение к «локальной» области памяти выполняется непосредственно, обращение к «удалённой» области памяти является более трудоёмким, поскольку выполняется через транзитные узлы.

В многопоточных программах при выполнении критических секций на выделенных процессорных ядрах существенную роль играет латентность при обращении RCL-сервера к участкам памяти внутри критических секций. Поэтому в таких программах важно выделять память в области памяти, локальной узлу, на котором функционирует сервер RCL. Это позволяет сократить время обращений к памяти и минимизировать время выполнения многопоточной программы.

В настоящей статье исследуется влияние политики выделения памяти в NUMA-системах на эффективность метода делегирования выполнения критических секций. Предложен алгоритм RCL-awareMalloc, позволяющий минимизировать время выполнения параллельных программ за счет оптими-

зации выделения памяти в NUMA-системах. Суть алгоритма заключается в следующем. При выполнении многопоточной программы память выделяется на том NUMA-узле, на котором функционирует RCL-сервер. Основные шаги алгоритма RCLAwareMalloc:

1. При выполнении функции выделения памяти проверяется, инициализированы ли в программе RCL-блокировки (запущены ли RCL-серверы).

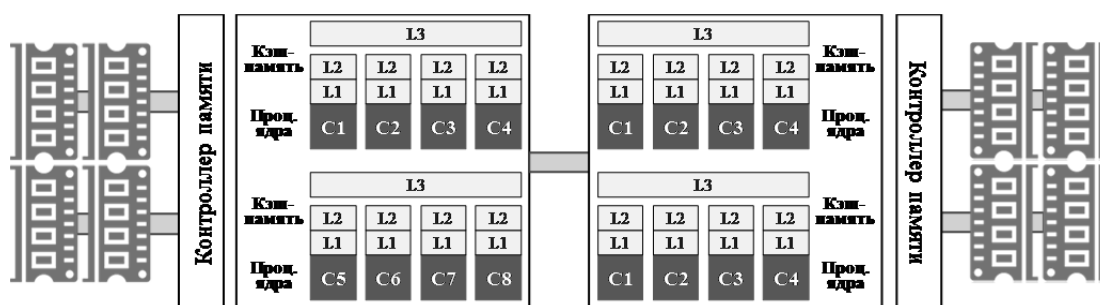


Рис. 4. Функциональная схема многопроцессорной NUMA-системы

2. Если в программе не инициализированы RCL-блокировки, то используется функция DefaultMalloc выделения памяти по умолчанию и указатель на выделенный сегмент памяти добавляется в список AllocList.

3. Если в программе инициализированы RCL-блокировки, то проверяется, выполняются ли все RCL-серверы на одном NUMA-узле.

4. Если все RCL-серверы выполняются на одном NUMA-узле, то устанавливается политика выделения памяти на данном NUMA-узле. В противном случае используется функция DefaultMalloc.

5. Если после очередного запуска RCL-блокировки обнаружено, что не все RCL-серверы выполняются на одном узле, то осуществляется привязка выделения памяти к NUMA-узлу.

6. При инициализации первой RCL-блокировки проверяется, пустой ли список AllocList. Если список не пустой, то выполняется миграция блоков выделенной памяти на NUMA-узел, на котором функционирует RCL-сервер.

3. Результаты экспериментов

Эксперименты проводились на узле вычислительного кластера Oak Центра параллельных вычислительных технологий Федерального государственного бюджетного образовательного учреждения высшего образования «Сибирский государственный университет телекоммуникаций и информатики». Узел укомплектован двумя четырёхъядерными процессорами Intel Xeon E5420, 24 Гб памяти. Соотношение скорости доступа к локальному и удалённому сегментам памяти 2:1.

Моделирование алгоритмов выполнялось на основе синтетического теста. Тест реализует итеративный доступ к элементам массива длины $b = 5 \times 10^8$. Использовались три шаблона доступа к элементам:

- случайный: на каждой итерации выбирается случайный элемент массива;
- последовательный: на каждой новой итерации выбирается элемент, следующий за предыдущим;
- с интервалом: на каждой новой итерации выбирается элемент, расположенный на расстоянии $s = 1000$ элементов от предыдущего.

Число p параллельных потоков варьировалось от 1 до 8 (количество процессорных ядер на вычислительном узле). Каждый поток выполнял $n = 10^8 / p$ операций инкремента соответствующего элемента тестового массива. Индекс элемента массива определялся шаблоном доступа к массиву.

Проводилось сравнение двух алгоритмов выделения памяти:

1. Выделение памяти по умолчанию (DefaultMalloc).
2. Выделение памяти с учётом расположения RCL-сервера (RCLAwareMalloc).

Выполнялась привязка потоков к процессорным ядрам вычислительного узла:

- поток, выполняющий выделение памяти: NUMA-узел 0, процессорное ядро 0;

- RCL-сервер: NUMA-узел 1, процессорное ядро 0;
- тестовый поток 0: NUMA-узел 0, процессорное ядро 0;
- тестовый поток 1: NUMA-узел 0, процессорное ядро 1;
- тестовый поток 2: NUMA-узел 0, процессорное ядро 2;
- тестовый поток 3: NUMA-узел 0, процессорное ядро 3;
- тестовый поток 4: NUMA-узел 1, процессорное ядро 1;
- тестовый поток 5: NUMA-узел 1, процессорное ядро 2;
- тестовый поток 6: NUMA-узел 1, процессорное ядро 3.

В качестве показателей эффективности использовались пропускная способность $b = n / t$ (t – время выполнения экспериментов) и m – процент промахов по кэшу (cache misses).

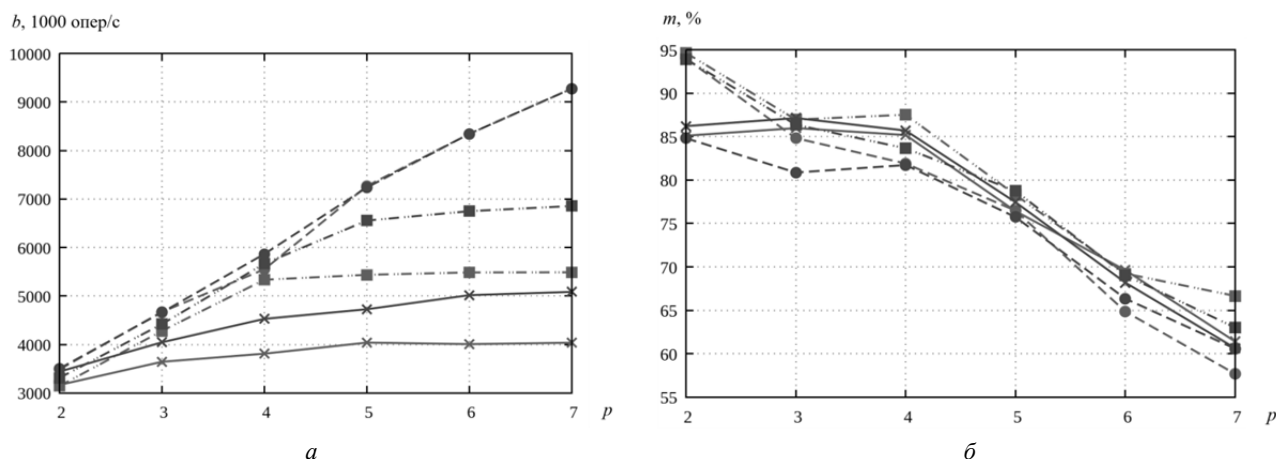


Рис. 5. Эффективность выполнения тестовой параллельной программы: a – пропускная способность, b – промахи по кэшу.

- $\times$ – DefaultMalloc, случайный доступ; – $\bullet$ – DefaultMalloc, последовательный доступ;
- $\square$ – DefaultMalloc, доступ с интервалом; – $\times$ – RCLAwareMalloc, случайный доступ;
- $\bullet$ – RCLAwareMalloc, последовательный доступ; – $\square$ – RCLAwareMalloc, доступ с интервалом

Эксперименты показали (рис. 5), что алгоритм RCLAwareMalloc выделения памяти, учитывающий размещение RCL-сервера в NUMA-системе, позволяет повысить (по сравнению с алгоритмом DefaultMalloc) пропускную способность критической секции за счёт сокращения времени выполнения обращений RCL-сервера к памяти. Предложенный подход наиболее эффективен при случайном доступе к элементам массива и в случае доступа к элементам массива с интервалом. Эффективность RCLAwareMalloc увеличивается с ростом числа потоков.

Заключение

Разработан алгоритм выделения памяти в вычислительных системах с неоднородным доступом к памяти при выполнении критических секций на выделенных процессорных ядрах (RCL). Алгоритм позволяет минимизировать время выполнения критических секций по сравнению с выделением памяти по умолчанию за счёт минимизации количества обращений к удалённым NUMA-сегментам памяти. Алгоритм реализован в виде библиотеки и может быть использован с целью минимизации существующих многопоточных программ.

ЛИТЕРАТУРА

1. Хорошевский В.Г. Распределённые вычислительные системы с программируемой структурой // Вестник СибГУТИ. 2010. № 2 (10). С. 3–41.
2. Herlihy M., Shavit N. The Art of Multiprocessor Programming, Revised Reprint. Elsevier, 2012. 528 с.
3. Herlihy M., Moss J. E.B. Transactional memory: Architectural support for lock-free data structures // Proceedings of the 20th annual international symposium on computer architecture ACM. ACM, 1993. V. 21. No. 2. P. 289–300.

4. Mellor-Crummey J.M., Scott M.L. Algorithms for scalable synchronization on shared-memory multiprocessors // *ACM Transactions on Computer Systems (TOCS)*. 1991. V. 9. No. 1. P. 21–65.
5. Hendler D. et al. Flat combining and the synchronization-parallelism tradeoff // *Proceedings of the twenty-second annual ACM symposium on Parallelism in algorithms and architectures*. ACM, 2010. P. 355–364.
6. Fatourou P., Kallimanis N.D. Revisiting the combining synchronization technique // *ACM SIGPLAN Notices*. ACM, 2012. V. 47. No. 8. P. 257–266.
7. Oyama Y., Taura K., Yonezawa A. Executing parallel programs with synchronization bottlenecks efficiently // *Proceedings of the International Workshop on Parallel and Distributed Computing for Symbolic and Irregular Applications, PDSIA '99*. 1999. P. 1–24.
8. Suleman M.A. et al. Accelerating critical section execution with asymmetric multi-core architectures // *ACM SIGARCH Computer Architecture News*. ACM, 2009. V. 37. No. 1. P. 253–264.
9. Metreveli Z., Zeldovich N., Kaashoek M.F. Cphash: A cache-partitioned hash table // *ACM SIGPLAN Notices*. ACM, 2012. V. 47. No. 8. P. 319–320.
10. Calciu I., Gottschlich J.E., Herlihy M. Using elimination and delegation to implement a scalable NUMA-friendly stack // *Proc. Unix Workshop on Hot Topics in Parallelism (HotPar)*. 2013. P. 1–7.
11. Lozi J.P. et al. Remote Core Locking: Migrating Critical-Section Execution to Improve the Performance of Multithreaded Applications // *USENIX Annual Technical Conference*. 2012. P. 65–76.

Пазников Алексей Александрович, канд. техн. наук. E-mail: apaznikov@gmail.com
Институт физики полупроводников им. А.В. Ржанова СО РАН (г. Новосибирск)

Поступила в редакцию 20 марта 2016 г.

Paznikov Alexey A. (Rzhanov Institute of Semiconductor Physics Siberian Branch of Russian Academy of Sciences. Novosibirsk, Russian Federation).

Optimization method of remote core locking.

Keywords: RCL; multithreading programming; critical sections; scalability.

DOI: 10.17223/19988605/38/8

The number of processor cores in shared memory computer systems is increasing nowadays. Therefore, the need of scalable parallel programs rises up. These programs must be efficient on large number of parallel threads.

Among the perspective scalable approaches of multithreading programming, one can emphasize lock-free algorithms and data structures and software transactional memory. The main drawbacks of lock-free algorithms and data structures are their restrained area of applying and essential complexity of parallel programs development. Besides the throughput of those structures is comparable with lock-based structures. Current implementations of software transactional memory do not provide high performance of multithreading programs and have not common in real applications yet.

Classical approach of lock-based critical sections in multithreading programs remains the most prevalent. Locks are easier than lock-free algorithms and data structures and ensures high level of productivity. Besides that, the most of present software is lock-based. Thus, development of scalable locks is substantial now.

Locks scalability depends on access contention while accessing of parallel threads to share memory and space cache locality. Access contention takes place while simultaneous access of multiple thread to one critical section, protected by one synchronization object. This leads to cache become inefficiency. Cache locality is significant when the critical section contains the access to shared data, which was accessed previously on the other core. This circumstance leads to cache misses and severe critical section execution time increasing.

The most efficient approaches of scalable locks implementation are CAS spinlocks, MCS-locks, Flat combining CC-Synch, DSM-Synch, Oyama lock. However, these methods cannot guarantee the minimum of critical section execution.

The critical section execution time includes time of lock ownership transfer and critical section execution time. In this work, we consider the method of critical section execution on remote processor cores (Remote Core Locking, RCL). RCL minimizes the execution time of legacy multithreading programs by reduction critical section execution critical path. This method replaces critical sections in legacy programs by procedures remote calls. All critical sections protected by one lock are performed by server thread launched on remote core. Therefore, by execution of the critical section by the particular thread minimizes the overheads of lock ownership transfer.

In the current implementation RCL, they do not consider the non-uniform access to memory in NUMA-systems. These systems are the compositions multicore processors connected with local area of global access. Time of critical section execution in RCL depends on RCL-server latency while accessing to memory blocks within critical section. Thus, in these programs memory should be allocated on NUMA node with the RCL-server. This reduces time of memory accesses and minimizes of programs execution time.

In this work, we propose the algorithm minimizing the memory access time in NUMA systems while RCL-programs execution. The algorithm optimizes the memory allocation on NUMA-systems. The essential of the algorithms is allocating on the NUMA-node, on which RCL-server is executing. The results of experiments show that the memory allocation algorithm increases the throughput of critical sections execution by reducing the time of RCL-server access to memory.

REFERENCES

1. Khoroshevsky, V.G. (2010) Distributed programmable structure computer systems. *Vestnik SibGUTI*, 2(10). pp. 3–41. (In Russian).
2. Herlihy, M. & Shavit, N. (2012) *The Art of Multiprocessor Programming*. Revised Reprint. Elsevier.
3. Herlihy, M. & Moss, J.E.B. (1993) Transactional memory: Architectural support for lock-free data structures. *Proceedings of the 20th annual international symposium on computer architecture ACM*. 21(2). pp. 289–300. DOI: 10.1145/173682.165164
4. Mellor-Crummey, J.M. & Scott, M.L. (1991) Algorithms for scalable synchronization on shared-memory multiprocessors. *ACM Transactions on Computer Systems (TOCS)*. 9(1). pp. 21–65. DOI: 10.1145/103727.103729

5. Hendler, D. et al. (2010) Flat combining and the synchronization-parallelism tradeoff. *Proceedings of the twenty-second annual ACM symposium on Parallelism in algorithms and architectures*. ACM. pp. 355–364. DOI: 10.1145/1810479.1810540
6. Fatourou, P. & Kallimanis, N.D. (2012) Revisiting the combining synchronization technique. *ACM SIGPLAN Notices*. ACM. 47(8). pp. 257–266. DOI: 10.1145/2370036.2145849
7. Oyama, Y., Taura, K. & Yonezawa, A. (1999) Executing parallel programs with synchronization bottlenecks efficiently. *Proceedings of the International Workshop on Parallel and Distributed Computing for Symbolic and Irregular Applications, PDSIA '99*. pp. 1–24.
8. Suleman, M.A. et al. (2009) Accelerating critical section execution with asymmetric multi-core architectures. *ACM SIGARCH Computer Architecture News*. ACM. 37(1). pp. 253–264. DOI: 10.1145/2528521.1508274
9. Metreveli, Z., Zeldovich, N. & Kaashoek, M.F. (2012) Cphash: A cache-partitioned hash table. *ACM SIGPLAN Notices*. ACM. 47(8). pp. 319–320. DOI: 10.1145/2370036.2145874
10. Calciu, I., Gottschlich, J.E. & Herlihy, M. (2013) Using elimination and delegation to implement a scalable NUMA-friendly stack. *Proc. Usenix Workshop on Hot Topics in Parallelism (HotPar)*. pp. 1–7.
11. Lozi, J.P. et al. (2012) Remote Core Locking: Migrating Critical-Section Execution to Improve the Performance of Multithreaded Applications. *USENIX Annual Technical Conference*. pp. 65–76.

УДК 629.7.05

DOI: 10.17223/19988605/38/9

А.В. Сарапулов, А.Б. Уманский

РЕКОНФИГУРИРОВАНИЕ БОРТОВОЙ ВЫЧИСЛИТЕЛЬНОЙ МАШИНЫ ДЛЯ ПОВЫШЕНИЯ ОТКАЗОУСТОЙЧИВОСТИ

Рассматривается задача повышения отказоустойчивости беспилотного летательного аппарата, неспособного по массе и габаритам иметь в своем составе резервированные элементы системы управления. Показаны особенности нерезервированной вычислительной машины на базе специализированных вычислителей. Проведена модификация традиционных алгоритмов работы межмодульного взаимодействия, предложен алгоритм реконфигурации с перераспределением задач между исправными элементами. Поясняется, для каких случаев метод окажется эффективным.

Ключевые слова: система управления; бортовая цифровая вычислительная машина; система контроля и функциональной защиты; реконфигурация.

В настоящее время становится актуальным использование беспилотных летательных аппаратов (БПЛА) в различных сферах жизнедеятельности человека, которые выполняют различные функции, такие как доставка посылок, организация телекоммуникационных сетей, мониторинг и фотографирование местности и т.д.

При проектировании современных систем управления (СУ) БПЛА полагаются на следующие принципы: надежность и малые габариты. Важным требованием также является возможность продолжения функционирования при отказах элементов СУ, чтобы повысить вероятность спасения БПЛА для повторной эксплуатации.

В основе современных СУ ЛА находятся цифровые вычислительные машины (ЦВМ). Они проектируются как сложные системы, содержащие несколько цифровых вычислителей, которые могут работать как в режиме независимого решения разных задач, так и в режиме резервирования [1].

1. Описание бортовой вычислительной машины

В разрабатываемой СУ используется ЦВМ, строящаяся по магистрально-модульному принципу, в основе которого лежит управляющий трехканальный системный модуль (СМ), обеспечивающий повышенную отказоустойчивость ЦВМ. СМ соединен с остальными модулями ЦВМ (вычислители (далее – ВМ), модули обмена (далее – МС) с внешними абонентами (ВА)) резервированной системной магистралью [2, 3]. СМ отвечает за обмен с внешними устройствами и исполнительными органами, ВМ отвечает за отработку логической части функциональных алгоритмов (рис. 1).

С учетом требований по стойкости для цифровой части СУ при работе в экстремальных условиях (широкий диапазон изменения температур, механические воздействия в виде ударов и широкополосной вибрации, полей ионизирующего излучения и т.п.) и возможностей отечественной элементной базы зачастую возникает проблема нехватки производительности одного вычислителя для решения всего спектра задач СУ. Исходя из этого, для обеспечения решения необходимого перечня задач СУ прибегают к использованию специализированных вычислителей, составляющих распределенную вычислительную систему.

Одновременно с этим, при отсутствии требований работы при одной возможной неисправности из-за невозможности применения структурного резервирования аппаратуры ЦВМ вследствие малых габаритов ЛА, предъявляются требования по обеспечению спасения аппаратуры ЛА в случае сбоев и отказов аппаратуры СУ. Такая задача требует выработки новых алгоритмов функционирования вычислительного процесса ЦВМ СУ БПЛА.

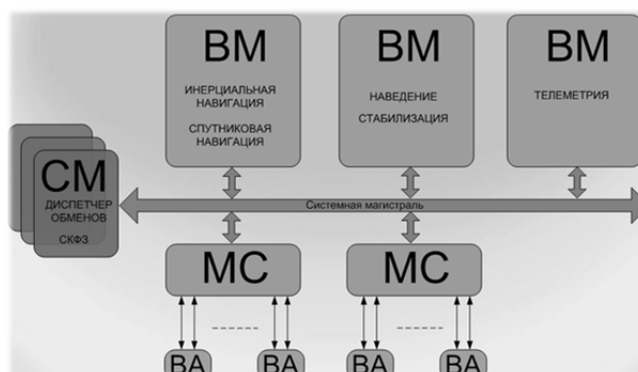


Рис. 1. Структурная схема магистрально-модульной вычислительной машины с распределением задач по вычислителям

2. Процесс реконфигурации

В целях повышения отказоустойчивости и возможности обеспечения аварийного спасения СУ БПЛА предлагается следующая организация вычислительного процесса ЦВМ. Каждый вычислитель содержит программный код всех функциональных задач, решаемых ЦВМ. СМ, работающий циклически, выдает в каждый вычислитель команду, запускающую следующую задачу. Эти команды расположены в СМ последовательно в порядке убывания их приоритета. В начале первого цикла СМ раздает задачи всем вычислителям и циклически анализирует регистр состояния каждого вычислителя на простой. При обнаружении свободного вычислителя СМ определяет ему следующую задачу, предварительно списывая его стек, в котором находятся промежуточные результаты вычислений, необходимые для решения этой же задачи в следующем цикле с новыми данными от внешних абонентов. Стек копируется по математическому номеру во все вычислители одновременно. В случае, если один из вычислителей выйдет из строя, другой сможет продолжить решение этой задачи, имея всю актуальную информацию за предыдущий цикл.

Фактически выдача команды выглядит как запись в ячейку памяти ВМ номера задачи из таблицы задач, что обеспечит минимальное время от момента считывания информации из ячейки до перехода через соответствующую строку таблицы к выполнению нужной задачи.

Поскольку циклограмма определяется динамически, требуется следующее условие: цикл ЦВМ должен быть в два раза меньше цикла работы внешних абонентов. Таким образом, ЦВМ будет получать новую информацию из внешних систем, например от инерциальной системы навигации, практически без пропусков. В случае единичных сбоев пропущенные данные могут быть экстраполированы, поскольку информация, полученная от внешних абонентов, для решения задач навигации представляет собой изменение по сравнению с прошлой итерацией.

Такая организация вычислительной системы дает возможность равномерного распределения производительности между вычислителями. Механизм выдачи команд по приоритетности будет отбрасывать задачи, наименее влияющие на функции полета, при снижении общей производительности системы. Эти задачи все также будут находиться в очереди, но, не успев на выдачу до завершения цикла, будут перемещены обратно в конец. Наименьший приоритет, например, могут иметь задачи, корректирующие основные вычисления для повышения точности. Вследствие отбрасывания этих задач точность полета по заданной траектории аппарата упадет, но в то же время обеспечится функционирование с задействованием необходимых алгоритмов спасения, что окажется эффективным в случаях, когда нежелательна потеря всего изделия или отдельных его частей. При этом основной сложностью будет обеспечение точного определения сбоев нерезервированной части аппаратуры, что потребует реализации дополнительного функционального контроля программно-алгоритмическими средствами.

3. Система контроля и функциональной защиты

Функционирование каждого вычислительного модуля контролируется системным модулем. СМ синхронизирует работу вычислительных модулей раз в цикл с помощью одновременной записи в их

триггеры прерываний. Вычислительные модули самостоятельно ведут счет выполненных циклов, который периодически списывается системным модулем для сравнения со своим счетчиком.

Помимо этого в каждом вычислителе на основании выполнения функционального алгоритма формируется признак, подтверждающий работоспособность. СМ каждый цикл считывает для анализа в специальную ячейку с данным признаком и после этого обнуляет ее. Признаки работоспособности ВМ сравниваются в СМ с эталоном. Алгоритмический контроль основан на том, что каждый ВМ, выполняя расчеты, проверяет вхождение результатов в допустимые зоны. Также периодически ВМ выполняет тестовый пример, сравнивая результат с эталоном-константой, записанной в область постоянной памяти. Тестовыми примерами могут являться простейшие арифметические задачи, например сложение целых чисел или с плавающей точкой, если позволяет установленное время цикла. Выход за расчетные границы или ошибки в тестовых примерах формируют признак, обозначающий возникновение сбоя в полетной программе.

Достоверность передаваемой информации при межмодульном обмене дополнительно обеспечивается проверкой контрольной суммы, которой сопровождается массив информации.

Для повышения надежности функционирования ЛА программы контроля находятся в заблокированной на запись части постоянной памяти для исключения возможности непреднамеренной коррекции или случайного искажения.

Заключение

В работе рассмотрен вариант решения актуальной научно-технической задачи: обеспечение повышенной живучести БПЛА с нерезервированной аппаратурой СУ в условиях возникновения сбоев и отказов.

Предложены модель организации работы ЦВМ и алгоритмы перераспределения задач по вычислителям в соответствии с их состоянием. Данный метод перераспределения целесообразно применять только в оговоренных случаях с определенными требованиями, поскольку при отказах аппарат может потерять значительную часть функционала, реализуя выполнение только базовых функций, например стабилизацию в атмосфере. В частности, такое решение может оказаться эффективным в случаях, когда не допускается потеря аппаратуры и БПЛА должен перейти, например, на выполнение программы по возвращению на базу.

ЛИТЕРАТУРА

1. Уманский А.Б., Антимиров В.М., Шалимов Л.Н. Бортовые цифровые вычислительные системы семейства «Малахит» для работы в экстремальных условиях // Вестник СГАУ. 2013. № 4 (42). С. 19–27.
2. Сарапулов А.В., Уманский А.Б. Эффективность перераспределения задач бортовой вычислительной системы для повышения надежности систем автоматического управления подвижными объектами // Международная конференция студентов, аспирантов и молодых ученых «Информационные технологии, телекоммуникации и системы управления»: сб. докл. Екатеринбург, 2015. С. 231–238.
3. Антимиров В.М., Телицин В.В. Проектирование бортовой аппаратуры систем управления : курс лекций. 2010. 146 с.

Сарапулов Алексей Валерьевич. E-mail: aesee@mail.ru

Уральский федеральный университет (г. Екатеринбург)

Уманский Алексей Борисович, канд. техн. наук. E-mail: pdwn1982@yandex.ru

АО «Научно-производственное объединение автоматики им. академика Н.А. Семихатова» (г. Екатеринбург)

Поступила в редакцию 17 марта 2016 г.

Sarapulov Alexey V., Umansky Alexey B. (SPA of Automatics named after Academician N.A. Semikhatov. Russian Federation).

Reconfiguring the on-board computer to increase stability in case of failure.

Keywords: friction control system; unmanned aerial vehicle; system of control and functional protection; on-board digital computer system; rescue system; algorithmic control.

DOI: 10.17223/19988605/38/9

The article examines the problem of increasing the stability of the onboard computer compact unmanned aerial vehicle in case of failure. The problem of the use of traditional redundant structure and its disadvantages was under review. We offer organization of com-

puting process on the basis of already used the onboard digital computer (OBDC) with a clear interaction. Board computer is based on the main-modular. It is based on a reliable three channel system module (SM).

We pose the following problem: used board computer does not allow the use of computing modules (CM) required performance. Thus, there is a need for multiple CM, which are distributed parallel system. The algorithm of redistribution of tasks requires a dynamic sequence diagrams, and SM distributes tasks to perform in real time. Each sequence diagram is given strictly defined time limit. With the condition of the length of the cycle SM is half the length of the external subscribers cycles, achieved no omissions of relevant information.

The programming code of the functional tasks is written in all the computing modules. The ability to solve the same tasks on different devices is achieved through the distribution of the stack with intermediate data between all CM.

Completing the performance of a tasks computer generates a signal for claim line. It needs to check the status registers of CM seeking the released. His stack SM copies to all the other devices and sets the following priorities tasks.

This algorithm makes sense in cases where the unit, which has a failure, is expected by the operator at the base, instead of continuing the full functioning. The return of the entire apparatus enables reuse of its constituent parts. This decision sets another problem: the precise definition of equipment failures.

The control system includes a number of ways to determine the fault. Algorithmic methods include counting and comparing the cycle number of the modules, the analysis of the results of calculations for entry into certain limits or repeated calculations on a simplified algorithm, as well as the decision calculators test examples and results are compared with the standards. The transfer of accurate information is analyzed with the help of checksums.

Thus, we propose an algorithm for the realization of functioning of small-sized unmanned aerial vehicle in case of faults and failures.

REFERENCES

1. Umansky, A.B., Antimirov, V.M. & Shalimov, L.N. (2013) Onboard digital computer systems of the “Malachite” family for extreme conditions. *Vestnik SGAU – Vestnik of the Samara State Aerospace University*. 4(42). pp. 19–27. (In Russian).
2. Sarapulov, A.V. & Umansky, A.B. (2015) [The efficiency of the redistribution of tasks onboard computer system to improve the reliability of automatic control system of moving object]. *Informatsionnye tekhnologii, telekommunikatsii i sistemy upravleniya* [Information technology, telecommunications and control systems]. International Conference of Students, Graduates and Young Scientists. Ekaterinburg. pp. 231–238. (In Russian).
3. Antimirov, V.M. & Telitsyn, V.V. (2010) *Proektirovanie bortovoy apparatury sistem upravleniya* [Design of onboard equipment control systems].

ДИСКРЕТНЫЕ ФУНКЦИИ И АВТОМАТЫ

УДК 004

DOI: 10.17223/19988605/38/10

И.Б. Бурдонов, А.С. Косачев

ТЕСТИРОВАНИЕ СИСТЕМЫ АВТОМАТОВ

Статья посвящена проблеме тестирования составных систем, компоненты которых моделируются конечными автоматами, а взаимодействие между ними – обменом сообщениями по симплексным каналам связи. Система описывается ориентированным графом связей, вершины которого соответствуют автоматам компонентов, а дуги – каналам связи. При тестировании возможно наблюдение состояний автоматов и передаваемых сообщений. Гипотеза о связях предполагает, что граф связей статический и не содержит ошибок. Это позволяет существенно сократить время тестирования детерминированной системы вплоть до экспоненциального уменьшения числа тестовых воздействий.

Ключевые слова: ориентированный граф; покрытие графа; взаимодействующие автоматы; тестирование; сети.

Большинство сложных, особенно распределённых систем представляет собой набор взаимодействующих компонентов. В данной статье компоненты моделируются конечными автоматами, а взаимодействие – обменом сообщениями между автоматами. Структура связей между компонентами моделируется ориентированным графом (*графом связей*), в вершинах которого находятся автоматы, а дуги соответствуют симплексным каналам передачи сообщений. *Внутренние* дуги соединяют автоматы между собой, а *внешние* дуги связывают систему с её *окружением*. *Внешняя входная дуга* ведёт из внешней среды в один из автоматов, а *внешняя выходная дуга* ведёт из автомата во внешнюю среду. При тестировании тест играет роль внешней среды: передаёт сообщения в систему по внешним входным дугам и принимает от системы сообщения по внешним выходным дугам.

Если граф связей статический, т.е. не меняющийся в процессе работы системы, система (так же как её компоненты) может моделироваться конечным автоматом, получающимся из автоматов-компонентов с помощью подходящего оператора композиции, учитывающего граф связей.

Система работает правильно, если структура её связей правильна и каждый автомат в системе работает правильно. Обратное, вообще говоря, не верно, если требования к системе неоднозначно определяют её структуру, например, функциональные требования к системе, связывающие сообщения, получаемые от системы, с сообщениями, посылаемыми в систему. В данной статье целью тестирования является покрытие переходов автоматов системы, достижимых при работе этих автоматов в системе. Поэтому если в структуре связей нет ошибок (*гипотеза о связях*), т.е. граф связей автоматов совпадает с заданным, то такое тестирование сводится к проверке правильности переходов каждого автомата. Проблема в том, что автомат может тестироваться только как часть системы, т.е. тест не имеет непосредственного доступа к автомату, и вынужден осуществлять тестовые воздействия с помощью сообщений, посылаемых по внешним входным дугам, которые ведут, быть может, в другие автоматы. Тестирование компонента такой системы похоже на тестирование в контексте [1–5], когда этот компонент рассматривается как тестируемая система, а остальные – как контекст. Существенное отличие, однако, в том, что в таком контексте тоже возможны ошибки, но если верна гипотеза о связях, то только в компонентах, а не в графе связей. С другой стороны, при тестировании проверяется работа сразу нескольких компонентов, через которые проходят сообщения. Поскольку автомат тестируется как часть системы, могут быть проверены не все его переходы, которые проверяются при автономном тестировании с прямым доступом к автомату. Речь идет только о переходах автоматов, *достижимых* при его работе в составе системы.

Тестирование композиционного автомата системы, получающегося композицией автоматов по заданному графу связей, обеспечивает проход по всем его достижимым переходам. При этом, конечно, проверяются все достижимые переходы автоматов-компонентов, но может делаться много «лишней работы». Гипотеза о связях позволяет существенно сократить время тестирования.

В статье предлагается алгоритм построения набора тестов, который является полным (проверяет все достижимые переходы автоматов-компонентов) при выполнении двух условий: 1) верна гипотеза о связях; 2) система является детерминированной. Дополнительно алгоритм определяет недостижимые переходы автоматов. Предполагается, что нам известно, каким должен быть каждый автомат (задан граф переходов автомата с точностью до изоморфизма), и именно это проверяется при тестировании. Кроме того, тест может наблюдать как состояния автоматов в вершинах графа связей, так и сообщения на его дугах. Поскольку не налагается ограничений на связность графов переходов автоматов, полный набор тестов может содержать более одного теста. При переходе от одного теста к другому требуется рестарт системы. Такие предположения могут быть оправданы, например, при имитационном тестировании аппаратуры (simulation-based verification) (см. например, [6]).

1. Модель

Дуга рассматривается как очередь сообщений длины 1. По дуге можно послать сообщение, если она *пуста* (на ней нет сообщений), и с дуги можно принять сообщение, если она не пуста (на ней есть сообщение). Если сообщение передаётся по внутренней дуге $a \rightarrow b$, то оно было послано автоматом в вершине a и будет принято автоматом в вершине b . Если $a \rightarrow b$ – внешняя входная дуга, то a – внешняя среда. Если $a \rightarrow b$ – внешняя выходная дуга, то b – внешняя среда. Автомат может принимать одновременно несколько сообщений по нескольким (не обязательно всем) входным дугам, но не более одного сообщения по каждой дуге, и посылать несколько сообщений по нескольким (не обязательно всем) выходным дугам, но не более одного сообщения по каждой дуге. Автомат предполагается детерминированным. В графе связей мы допускаем кратные дуги, для различения которых будем пометать их символами из алфавита Z . Для простоты будем считать, что все вершины перенумерованы $0, 1, 2, \dots, k$, где k – число вершин, в которых находятся автоматы, а номер 0 соответствует внешней среде. Введём формальные определения и обозначения.

Граф связей определяется как набор $G = (V, Z, E)$, где $V = \{0, 1, \dots, k\}$ – конечное множество вершин, $E \subseteq (V \times Z \times V) \setminus (\{0\} \times Z \times \{0\})$ – конечное множество дуг (у внешней среды 0 нет дуг-петель). Дуга (v, z, w) задана начальной вершиной v , пометкой z и конечной вершиной w . Дуга $(0, z, w)$ – внешняя входная дуга, $(v, z, 0)$ – внешняя выходная дуга. Кроме того, $I_v = \{(a, z, v) \mid (a, z, v) \in E\}$ – множество дуг, заканчивающихся в вершине v , $O_v = \{(v, z, b) \mid (v, z, b) \in E\}$ – множество дуг, начинающихся в вершине v . I_0 содержит все внешние входные дуги, O_0 – внешние выходные дуги, $E \setminus (I_0 \cup O_0)$ – внутренние дуги.

Пусть M – множество всех возможных *сообщений*, общее для всех автоматов в системе.

Для множеств A и B через A^B обозначим множество всех отображений из B в A .

Система автоматов определяется как граф связей G , в котором каждой вершине v поставлен в соответствие автомат $A_v = (M, S_v, X_v, Y_v, T_v, s_{0v})$, где S_v – конечное множество *состояний* автомата, $X_v = \{x \mid \exists f (x \subseteq f \ \& \ f \in M^{I_v})\}$ – множество *стимулов* (входных символов), $Y_v = \{y \mid \exists f (y \subseteq f \ \& \ f \in M^{O_v})\}$ – множество *реакций* (выходных символов), $T_v \subseteq S_v \times X_v \times Y_v \times S_v$ – конечное множество *переходов*, $s_{0v} \in S_v$ – *начальное состояние автомата*. Обозначим: $s \xrightarrow{?x!y} t \triangleq (s, x, y, t) \in T_v$, $s \xrightarrow{?x!y} \triangleq \exists t (s, x, y, t) \in T_v$. Там, где это не приведёт к недоразумению, мы будем в неформальном тексте сам переход (s, x, y, t) обозначать стрелкой $s \xrightarrow{?x!y} t$. Состояние t будем называть *постсостоянием* перехода. Если постсостояние несущественно, то переход (s, x, y, t) будем обозначать стрелкой $s \xrightarrow{?x!y} \rightarrow$. Дугу из I_v будем называть *входной дугой* автомата в вершине v , а дугу из O_v – *выходной дугой* этого автомата. Стимул автомата – это частично определённое отображение $x: I_v \rightarrow M$, которое каждой входной дуге $i \in \text{Dom}(x)$ ставит в соответствие сообщение $x(i)$, принимаемое по этой дуге. Реакция автомата – это частично определённое отображение $y: O_v \rightarrow M$, которое каждой выходной дуге $j \in \text{Dom}(y)$ ставит в соответствие сообщение $y(j)$, посылаемое по этой дуге.

Для описания состояния входных и выходных дуг автомата в вершине v введём два частично-определённых отображения $x_v^\# : I_v \rightarrow M$ и $y_v^\# : O_v \rightarrow \{M\}$. Первое отображение каждой непустой входной дуге $i \in I_v$ ставит в соответствие сообщение $m \in M$, находящееся на этой дуге. Второе отображение каждой пустой выходной дуге $j \in O_v$ ставит в соответствие множество сообщений M . Эти отображения $x_v^\#$ и $y_v^\#$ порождают множества *потенциальных* стимулов и *потенциальных* реакций, которые автомат может, соответственно, принять и послать при данном состоянии входных и выходных дуг, если, конечно, в текущем состоянии автомата определены соответствующие переходы:

$$X_v^\# = \{x : I_v \rightarrow M \mid \text{Dom}(x) \subseteq \text{Dom}(x_v^\#) \ \& \ \forall i \in \text{Dom}(x) (x(i) = x_v^\#(i))\},$$

$$Y_v^\# = \{y : O_v \rightarrow M \mid \text{Dom}(y) \subseteq \text{Dom}(y_v^\#)\}.$$

Определим формальное условие выполнения перехода $s - ?x!y \rightarrow t : x \in X_v^\# \ \& \ y \in Y_v^\#$.

2. Детерминизм

Для того чтобы система автоматов была детерминированной, потребуем детерминированности каждого из этих автоматов. Прежде всего, состояние и стимул должны однозначно определять реакцию и постсостояние перехода:

$$1) \ \forall s, x, x', y, y', t, t' \quad s - ?x!y \rightarrow t \ \& \ s - ?x'!y' \rightarrow t' \Rightarrow y = y' \ \& \ t = t'.$$

Распределение $x_v^\#$ сообщений по входным дугам автомата неоднозначно определяет стимул, принимаемый автоматом, поскольку автомат может как принимать, так и не принимать сообщения с непустых входных дуг. Будем говорить, что стимулы x и x' *совместимы*, и обозначать $x \approx x'$, если при некотором отображении $x_v^\#$ автомат может принять любой из этих стимулов, т.е. $x \in X_v^\#$ и $x' \in X_v^\#$. Формально $x \approx x' \triangleq \forall i \in \text{Dom}(x) \cap \text{Dom}(x') (x(i) = x'(i))$. Заметим, что все стимулы во множестве $X_v^\#$ совместимы друг с другом. Отсюда вытекает второе требование детерминизма автомата:

2) нет переходов из одного состояния по разным совместимым стимулам:

$$\forall s, x, x', y, y', t, t' \quad x \neq x' \ \& \ x \approx x' \Rightarrow \neg(s - ?x!y \rightarrow t \ \& \ s - ?x'!y' \rightarrow t').$$

Теорема 1. Если выполнены оба требования детерминизма, то состояние s_v автомата в вершине v и отображения $x_v^\#$ и $y_v^\#$ однозначно определяют, выполняет ли автомат какой-либо переход, и если выполняет, то сам переход, т.е. однозначно определяют принимаемый стимул $x_v^\wedge$, посылаемую реакцию $y_v^\wedge$ и постсостояние $t_v^\wedge$.

Доказательство. По определению отображения $x_v^\#$ и $y_v^\#$ однозначно определяют множества $X_v^\#$ и $Y_v^\#$. Из второго требования детерминизма следует, что при любом распределении $x_v^\#$ сообщений на входных дугах автомата, порождающем множество $X_v^\#$ потенциальных стимулов, не более одного из этих стимулов $x_v \in X_v^\#$ может быть принят автоматом в данном состоянии s_v . Такой стимул x_v будем называть *выбираемым*, он может отсутствовать. Правда, это не означает, что выбираемый стимул x_v (если он есть) обязательно будет принят автоматом, поскольку выполнение перехода с приёмом этого стимула обусловлено возможностью послать реакцию. Рассмотрим все случаи поведения автомата в состоянии s_v при заданных $x_v^\#$ и $y_v^\#$ (однозначно определяющих $X_v^\#$ и $Y_v^\#$), определяя, будет ли выполнен переход, и если будет, то сам переход, т.е. принимаемый стимул $x_v^\wedge$, посылаемую реакцию $y_v^\wedge$ и постсостояние $t_v^\wedge$.

Если имеется выбираемый стимул $x_v \in X_v^\#$, то он единственный по второму требованию детерминизма. При этом, если для некоторой реакции y есть переход $s_v - ?x!y \rightarrow t$ и $y \in Y_v^\#$, т.е. реакция y может быть послана (нужные выходные дуги пусты), то автомат выполнит этот переход (он единственный по первому требованию детерминизма), $x_v^\wedge = x$, $y_v^\wedge = y$, $t_v^\wedge = t$. В противном случае автомат не выполнит никакого перехода, $x_v^\wedge = \emptyset$, $y_v^\wedge = \emptyset$, $t_v^\wedge = s_v$. Последнее имеет место и в случае отсутствия выбираемого стимула. Теорема 1 доказана.

3. Композиция

Определим композицию детерминированных автоматов по заданному графу связей. Результатом композиции будет автомат (S, X, Y, T, s_0) , отражающий работу системы в целом, включая все автоматы-компоненты и все дуги.

Состояние системы есть набор $s = (s_1, s_2, \dots, s_k, D)$, где $s_1, s_2, \dots, s_k$ есть набор состояний её автоматов, а $D: E \rightarrow M$ – частично-определённое отображение, задающее распределение сообщений по дугам графа связей: оно для каждой непустой дуги указывает находящееся на ней сообщение. Начальное состояние системы $s_0 = (s_{10}, s_{20}, \dots, s_{k0}, \emptyset)$, в котором каждый автомат находится в своём начальном состоянии, а сообщений на дугах нет.

Определение переходов композиции зависит от предполагаемого режима работы. В синхронном режиме за один такт срабатывают все автоматы, которые могут выполнить переход, а в асинхронном – только один такой автомат (вообще говоря, некоторое подмножество автоматов), выбираемый недетерминированным образом. Поскольку в рамках данной статьи нас интересуют только детерминированные системы, асинхронный режим далее не рассматривается.

Определим переходы композиции формально. В состоянии системы s внешняя среда может послать в систему сообщения по любым пустым внешним входным дугам и принять сообщения с любых занятых внешних выходных дуг. Это определяет допустимые внешние стимулы и реакции. Стимул $x: I_0 \rightarrow M$ допустим, если $\text{Dom}(x) \cap \text{Dom}(D) = \emptyset$, в частности всегда допустим пустой стимул $x = \emptyset$.

Реакция $y: O_0 \rightarrow M$ допустима, если $y \subseteq D$, в частности всегда допустима пустая реакция $y = \emptyset$. Если стимул x и реакция y допустимы, то в композиции определяется переход $s \xrightarrow{x!y} \hat{t}$, где $\hat{t} = (t_1^\wedge, t_2^\wedge, \dots, t_k^\wedge, D^\wedge)$. Определим для каждого v постсостояние $t_v^\wedge$ и распределение сообщений $D^\wedge$.

Рассмотрим вершину v . Для состояния системы s однозначно определяются отображение $x_v^\#$ как сужение отображения D на множество I_v входных дуг v -го автомата: $x_v^\# = \{(i, m) | (i, m) \in D \ \& \ i \in I_v\}$, и отображение $y_v^\#$, которое каждой пустой выходной дуге v -го автомата ставит в соответствие множество M всех сообщений: $y_v^\# = \{(j, M) | j \in O_v \setminus \text{Dom}(D)\}$. По теореме 1 при заданных s_v , $x_v^\#$ и $y_v^\#$ автомат в вершине v выполняет не более одного перехода, и однозначно определяются принимаемый стимул $x_v^\wedge$ и посылаемая реакция $y_v^\wedge$, а также постсостояние $t_v^\wedge$, которое и становится частью состояния $\hat{t}$.

Определим $D^\wedge$. Сначала положим $D^\wedge = D$. Рассмотрим, как должно меняться расположение сообщений на дуге $e = (i, z, j)$.

1. В состоянии s дуга e была пустой, т.е. $e \notin \text{Dom}(D)$. Если $j \neq 0$, то j -й автомат не принимает с неё сообщения, т.е. $e \notin \text{Dom}(x_j^\wedge)$. Если $i \neq 0$, то i -й автомат посылает по этой дуге сообщение m , если $e \in \text{Dom}(y_i^\wedge) \ \& \ y_i^\wedge(e) = m$, тогда пара (e, m) добавляется в $D^\wedge$. Если $j = 0$, то $e \notin \text{Dom}(y)$. Если $i = 0$, то $e \in \text{Dom}(x) \Rightarrow e \in \text{Dom}(D^\wedge) \ \& \ D^\wedge(e) = x(e)$, т.е. если внешняя среда посылает по внешней входной дуге e сообщение $x(e)$, то пара $(e, x(e))$ добавляется в $D^\wedge$.

2) В состоянии s на дуге e было сообщение m , т.е. $e \in \text{Dom}(D) \ \& \ D(e) = m$. Если $j \neq 0$, то j -й автомат принимает это сообщение, если $e \in \text{Dom}(x_j^\wedge) \ \& \ x_j^\wedge(e) = m$, тогда пара $(e, x_j^\wedge(e))$ удаляется из $D^\wedge$. Если $i \neq 0$, то i -й автомат не может послать по этой дуге никакого сообщения, поскольку дуга в состоянии s занята, т.е. $e \notin \text{Dom}(y_i^\wedge)$. Если $j = 0$, то $e \in \text{Dom}(y) \Rightarrow e \notin \text{Dom}(D^\wedge) \ \& \ D(e) = y(e)$, т.е. если внешняя среда принимает по внешней выходной дуге e сообщение $y(e)$, то пара $(e, y(e))$ удаляется из $D^\wedge$. Если $i = 0$, то $e \notin \text{Dom}(x)$.

Тем самым, $D^\wedge = (D \cup x \cup y_1^\wedge \cup \dots \cup y_k^\wedge) \setminus (y \cup x_1^\wedge \cup \dots \cup x_k^\wedge)$.

Будем говорить, что такая композиция детерминированных автоматов детерминирована, если в каждом достижимом (из начального состояния) состоянии каждая пара допустимых стимула и реакции однозначно определяет постсостояние системы, т.е. выполняемый переход.

Теорема 2. Композиция детерминированных автоматов детерминирована.

Доказательство. Нужно показать, что 1) каждый автомат вершины графа связей может выполнить не более одного перехода и 2) распределение $D^\wedge$ определяется однозначно. И то и другое следует из теоремы 1 и определения композиции. Теорема 2 доказана.

4. Генерация тестов

В данной статье целью тестирования системы автоматов является покрытие всех достижимых переходов автоматов. На каждом такте тест посылает в тестируемую систему сообщения по пустым

внешним входным дугам (не обязательно всем) и принимает от системы по занятым внешним выходным дугам (не обязательно всем) имеющиеся на них сообщения. Определим композицию системы и теста. Состояние композиции есть пара состояний системы и теста. Переход композиции соответствует паре (допустимый стимул, допустимая реакция), что определяет возможные постсостояния системы и теста, т.е. возможные постсостояния композиции. Заметим, что приём или неприём тестом сообщений с внешних выходных дуг системы, по сути, является дополнительным тестовым воздействием на систему, поскольку меняет выполнимость тех или иных переходов. Сам переход композиции является внутренним, т.е. ничем не помечен, поскольку композиция системы и теста замкнута и ни с чем не взаимодействует. Формально переходы композиции системы и теста определяются следующим правилом вывода: $s - ?x!y \rightarrow t \ \& \ s' \text{---} ?y!x \rightarrow t' \vdash ss' \rightarrow tt'$.

Если тест и система детерминированы, то их композиция тоже будет детерминирована в следующем смысле: в каждом её состоянии определено не более одного перехода.

Тестовая последовательность есть конечная последовательность пар $(x_1, y_1), \dots, (x_n, y_n)$, которой в тестируемой системе соответствует маршрут (цепочка смежных переходов) $s_0 - ?x_1!y_1 \rightarrow s_1 - \dots \rightarrow s_{n-1} - ?x_n!y_n \rightarrow s_n$, а в тесте – маршрут $s'_0 - ?y_1!x_1 \rightarrow s'_1 - \dots \rightarrow s'_{n-1} - ?y_n!x_n \rightarrow s'_n$. Каждое состояние s_i и s'_i соответствует префиксу тестовой последовательности длиной i . Для такой тестовой последовательности детерминированный тест состоит только из указанной выше цепочки переходов.

Какие проверки выполняются при прогоне теста на каждом i -м такте? 1) Выполнился ли в тесте переход $s'_i - ?y_{i+1}!x_{i+1} \rightarrow s'_{i+1}$; если не выполнен, то фиксируется ошибка. 2) Для каждого автомата в системе: правильно ли изменилось его состояние, правильно ли он выполнил приём стимула (с каких входных дуг принял сообщения), правильно ли он выполнил выдачу реакции (на какие выходные дуги послал сообщения, и правильные ли эти сообщения).

Прогон тестовой последовательности покрывает некоторое множество переходов автоматов системы. Поскольку система детерминирована, это множество одно и то же при разных прогонах данной тестовой последовательности (с рестартом между прогонами), поэтому тестовую последовательность достаточно прогонять один раз. Мы будем рассматривать только конечные наборы тестов, после завершения прогона одного теста выполняется рестарт системы и прогоняется следующий тест из набора. Набор тестов покрывает множество переходов автоматов, которое является объединением множеств переходов автоматов, покрываемых тестами из этого набора. Набор тестов будем называть *полным*, если он покрывает все переходы всех автоматов, достижимые при работе этих автоматов в системе. Ставится задача генерации полного набора тестов.

Для решения этой задачи мы предлагаем использовать любой алгоритм генерации полного набора тестов для одного автомата. Таких алгоритмов предложено довольно много, по сути, они сводятся к построению набора маршрутов, покрывающих граф переходов автомата, достижимых из его начального состояния (см., например, [7]). В качестве такого автомата для наших целей берётся автомат системы, получаемый с помощью композиции, описанной в предыдущем разделе. Покрывая все достижимые переходы композиционного автомата системы, мы, конечно, покрываем все достижимые переходы автоматов-компонентов. Однако такой набор тестов может быть избыточным для решения нашей задачи: покрытие всех достижимых переходов автоматов-компонентов не обязательно требует покрытия всех достижимых переходов композиционного автомата системы.

Поэтому предлагается в процессе генерации полного набора тестов для композиционного автомата системы применять *процедуру фильтрации*, которая будет отбрасывать «лишние» тесты. Эта процедура работает следующим образом. С самого начала создаётся пустое множество T генерируемого набора тестов и множество P непокрытых переходов автоматов, которое сначала равно множеству всех переходов всех автоматов. Когда генерируется очередной i -й тест T_i для композиционного автомата системы, вычисляется множество P_i переходов автоматов, покрываемое этим тестом. Тесту соответствует маршрут в композиционном автомате. Каждому переходу этого маршрута соответствует множество переходов в автоматах-компонентах (не более одного в каждом автомате); объединение этих множеств по всем переходам маршрута и есть множество P_i . Далее алгоритм фильтрации проверяет, покрывает ли i -й тест какой-либо новый, ещё не покрытый переход какого-либо автомата компонента. Если $P_i \cap P = \emptyset$, то никаких новых переходов

i -й тест не покрывает, и он отбрасывается. В противном случае тест добавляется к набору тестов $T := T \cup \{T_i\}$, а из множества непокрытых переходов удаляются новые переходы $P := P \setminus P_i$. После того как все тесты сгенерированы и отфильтрованы, получившееся множество T является полным набором тестов, а множество P – множеством недостижимых переходов автоматов компонентов.

5. О размере тестового набора

Теорема 3. Для любого числа состояний автоматов компонентов $n_1, n_2, \dots, n_k$ существует такая система, что время тестирования (в тактах), необходимое для покрытия всех достижимых переходов композиции, равно $\Omega(n_1 n_2 \dots n_k)$, а минимальное время, достаточное для покрытия всех достижимых переходов автоматов-компонентов, равно $O(n_1 + n_2 + \dots + n_k)$.

Доказательство. Рассмотрим систему, изображённую на рис. 1.

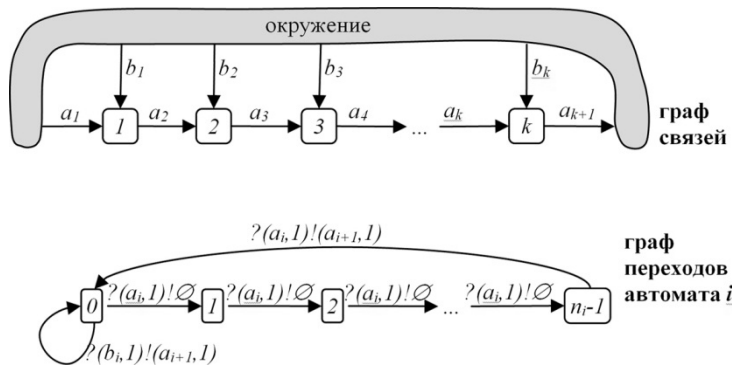


Рис. 1. Пример системы автоматов

Здесь имеется единственное сообщение: $M = \{1\}$. Все автоматы в вершинах однотипные и различаются числом состояний n_i . Автомат в вершине i имеет две входные дуги, обозначенные a_i и b_i , и одну выходную дугу a_{i+1} . Соответственно, имеются два стимула $(a_i, 1)$ и $(b_i, 1)$ и единственная непустая реакция – $(a_{i+1}, 1)$. Автомат i в состоянии $j \neq n_i - 1$, получая по входной дуге a_i сообщение 1, переходит в следующее состояние $j+1$ без выдачи реакции, а в состоянии $n_i - 1$ переходит в начальное состояние 0 с посылкой сообщения 1 по единственной выходной дуге a_{i+1} . Тем самым автомат i на каждую порцию из принятых им n_i сообщений по входной дуге a_i посылает одно сообщение следующему автомату $i + 1$ или внешней среде, если $i = k$. Кроме того, в состоянии 0 есть переход-петля по приёму сообщения по дуге b_i с выдачей сообщения следующему автомату $i + 1$ или окружению, если $i = k$.

Рассмотрим состояние композиции вида $s = (s_1, s_2, \dots, s_k, \emptyset)$, где состояние i -го автомата $s_i = 0, \dots, n_{i-1}$. Начальное состояние $s_0 = (0, 0, \dots, 0, \emptyset)$. Обозначим $s_{\max} = (n_1 - 1, n_2 - 1, \dots, n_k - 1, \emptyset)$. Состояние s можно понимать как пару $s = (\tilde{s}, \emptyset)$, где $\tilde{s}$ – число, записанное в позиционной системе счисления слева направо от младшей позиции 1 к старшей позиции k : 1, 2, ..., k , и n_i – основание системы счисления в позиции i . Число таких состояний равно $n_1 n_2 \dots n_k$. Поскольку все они достижимы из начального состояния, время тестирования композиционной системы равно $\Omega(n_1 n_2 \dots n_k)$.

Как наиболее быстро покрыть все переходы автомата i ? Для этого достаточно: 1) n_i раз послать сообщение по дуге a_i и 2) один раз – по дуге b_i . Для автомата 0 пункт 1 тест может выполнить непосредственно, поскольку дуга a_1 внешняя. Для автомата $i > 1$ пункт 1 можно выполнить, посылая n_i раз сообщение по внешней дуге b_{i-1} в предыдущий автомат с номером $i - 1$. Пункт 2 тест также может выполнить непосредственно, поскольку дуга b_i внешняя. Тем самым, время тестирования всех переходов автоматов-компонентов равно $O(n_1 + \dots + n_k)$. Теорема 3 доказана.

Заметим, что для $n_1 = n_2 = \dots = n_k = n$ имеем соотношение n^k и nk , т.е. для фиксированного числа k компонентов получаем экспоненциальное уменьшение времени тестирования.

Заключение

Сформулируем направления дальнейших исследований.

1. Оптимизация. Предложенный алгоритм фильтрации строит набор тестов не обязательно оптимальный по времени тестирования и / или числу тестов. Возникает задача поиска оптимального набора, которая, вообще говоря, сводима к задаче о поиске минимального покрытия [8, 9].

2. Недетерминизм. Нужно определить такие ограничения на недетерминизм системы и / или составляющих её автоматов, которые позволяли бы выполнять полное тестирование за конечное время и разработать соответствующие алгоритмы тестирования. Неплохие решения этой задачи предложены для автономного тестирования, когда автомат находится под непосредственным управлением теста [10–13] (не в контексте окружающей его части системы).

3. Конформность. В данной статье при тестировании проверяется изоморфизм автомата-компонента реализации его спецификации, заданной как автомат. В общем случае между автоматом компонента в реализации и автоматом компонента в спецификации задаётся отношение конформности, которое слабее изоморфизма: квази-редукция, симуляция и т.п. Это требует более сложного алгоритма тестирования. В то же время, если при тестировании мы можем наблюдать состояние реализации (как предполагается в данной статье), то возможно полное автономное тестирование за конечное время для конформности типа редукции или слабой симуляции [2, 3, 10–20]. Для составной системы возникает проблема декомпозиции системных требований, известная также как проблема несохранения конформности. Она заключается в том, что композиция реализаций компонентов, конформных спецификациям этих компонентов, в общем случае неконформна спецификации системы, в частности композиции спецификаций компонентов. Этой проблеме посвящён ряд работ [2, 4, 21], но возникает задача переосмысления предложенных решений для тестирования компонентов составной системы, когда верна гипотеза о связях.

4. Обобщение. В данной работе дуга графа связей реализует очередь длины 1. Но могут быть и другие дуги: очереди большей, в том числе неограниченной длины, очереди с приоритетами, стеки и т.п. Нужно обобщить понятие дуги с помощью определения автомата дуги. Более того, автомат дуги мог бы иметь несколько входов и выходов, как автомат вершины. Композиция должна быть определена для пары автоматов, выход одного из которых соединён с входом другого. На таком соединении происходит синхронное взаимодействие автоматов, где один автомат посылает сообщение тогда и только тогда, когда другой автомат это сообщение принимает. Для детерминизма системы, по-видимому, к автомату дуги нужно предъявить дополнительные (по сравнению с автоматом вершины) требования.

ЛИТЕРАТУРА

1. Revised Working Draft on “Framework: Formal Methods in Conformance Testing”. JTC1/SC21/WG1/Project 54/1, ISO Interim Meeting. ITU-T on. Paris, 1995.
2. Бурдонов И.Б., Косачев А.С., Кулямин В.В. Теория соответствия для систем с блокировками и разрушением. М. : Наука, 2008. 412 с.
3. Бурдонов И.Б. Теория конформности (функциональное тестирование программных систем на основе формальных моделей). LAP Lambert Academic Publishing, 2011. 428 с.
4. Бурдонов И.Б., Косачев А.С. Пополнение спецификации для ioco // Программирование. 2011. № 1. С. 3–18.
5. Petrenko A., Yevtushenko N., Von Bochmann G., Dssoul Ri. Testing in context: framework and test derivation // Computer Communications. 1996. V. 19(14). P. 1236–1249.
6. Камкин А., Чупилко М. Обзор современных технологий имитационной верификации аппаратуры // Программирование. 2011. № 3. С. 42–49.
7. Бурдонов И.Б., Косачев А.С., Кулямин В.В. Неизбыточные алгоритмы обхода ориентированных графов. Детерминированный случай // Программирование. 2003. № 5. С. 59–69.
8. Левитин А.В. Алгоритмы: введение в разработку и анализ. М. : Вильямс, 2006. С. 160–163.
9. Кормен Т.К., Лейзерсон Ч.И., Ривест Р.Л., Штайн К. Алгоритмы: построение и анализ. 2-е изд. М. : Вильямс, 2006. С. 456–458.
10. Бурдонов И.Б., Косачев А.С. Полное тестирование с открытым состоянием ограниченно недетерминированных систем // Программирование. 2009. № 6. С. 3–18.
11. Бурдонов И.Б., Косачев А.С. Семантики взаимодействия с отказами, дивергенцией и разрушением. Ч. 2. Условия конечного полного тестирования // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2011. № 2 (15). С. 89–98.

12. Бурдонов И.Б., Косачев А.С. Тестирование конформности на основе соответствия состояний // Труды ИСП РАН. 2010. № 18. С. 183–220.
13. Бурдонов И.Б., Косачев А.С. Безопасное тестирование симуляции систем с отказами и разрушением // Моделирование и анализ информационных систем. 2010. Т. 17(4). С. 27–40.
14. Bourdonov I.B., Kossatchev A.S., Kulyamin V.V. Formal Conformance Testing of Systems with Refused Inputs and Forbidden Actions // Proceedings of the Workshop on Model Based Testing (MBT 2004). Elsevier, 2006.
15. Бурдонов И.Б., Косачев А.С., Кулямин В.В. Формализация тестового эксперимента // Программирование. 2007. № 5. С. 3–32.
16. Бурдонов И.Б., Косачев А.С., Кулямин В.В. Безопасность, верификация и теория конформности // Материалы Второй международной научной конференции по проблемам безопасности и противодействия терроризму. МГУ, 2006. М. : МЦНМО, 2007. С. 135–158.
17. Бурдонов И.Б., Косачев А.С. Системы с приоритетами: конформность, тестирование, композиция // Труды ИСП РАН. 2008. Т. 14 (1). С. 23–54.
18. Бурдонов И.Б., Косачев А.С. Тестирование с преобразованием семантик // Труды ИСП РАН. 2009. Т. 17. С. 193–208.
19. Kossachev A., Burdonov I. Formal Conformance Verification, Short Papers of the 22nd IFIP ICTSS / eds. by Alexandre Petrenko, Adenilso Simao, Jose Carlos Maldonado. Natal, Brazil, 2010. P. 1–6.
20. Бурдонов И.Б., Косачев А.С. Семантики взаимодействия с отказами, дивергенцией и разрушением // Программирование. 2010. № 5. С. 3–23.
21. Бурдонов И.Б., Косачев А.С. Согласование конформности и композиции // Программирование. 2013. № 6. С. 3–15.

Бурдонов Игорь Борисович, д-р физ.-мат. наук. E-mail: igor@ispras.ru
Косачев Александр Сергеевич, канд. физ.-мат. наук. E-mail: kos@ispras.ru
 Институт системного программирования РАН (ИСП РАН) (г. Москва)

Поступила в редакцию 25 июня 2016 г.

Igor Burdonov, Alexander Kossatchev (Institute for System Programming of the Russian Academy of Sciences, Moscow, Russian Federation).

Testing of automata system.

Keywords: directed graph; graph coverage; communicating automata; testing; networks.

DOI: 10.17223/19988605/38/10

The problem of testing of aggregate systems is considered. The system is described as an oriented graph where the nodes correspond to component automata while edges correspond to simplex communication channels. The following hypothesis is assumed: the graph of links is static and the link structure is error-free. At each state, a component automaton can accept and send multiple messages through incoming and outgoing edges (at most one message through each edge). The goal of testing is to cover transitions of component automata reachable during the system work. It is assumed that during testing it is possible to observe the state changes of automata and the messages on the edges. The general model is considered when the system can simultaneously contain multiple messages, but not more than one for each edge. An automata composition is defined and the restrictions on automata making the system deterministic are described. An algorithm of test generation is proposed basing on test filtration generated for covering all transitions of the deterministic composition system. A test is rejected if it covers only transitions of the components that are covered by other tests. A simplified system model with only one message in transit is considered at the end. Using this example, we show that the hypothesis on links allows considerably reduce the number of required testing actions from the product of state numbers of the component automata to the sum of these numbers. If all the automata have the same number of states then it gives the exponential reduction of the number of test actions. In conclusion, some directions of future research are considered.

REFERENCES

1. Revised Working Draft on “Framework: Formal Methods in Conformance Testing”. (1995). *JTC1/SC21/WG1/Project 54/1, ISO Interim Meeting, ITU-T on. Paris.*
2. Burdonov, I.B., Kosachev, A.S. & Kulyamin, V.V. (2008) *Teoriya sootvetstviya dlya sistem s blokirovkami i razrusheniyem* [Conformance theory of the systems with refused inputs and forbidden actions]. Moscow: Nauka.
3. Bourdonov, I.B. (2011) *Teoriya konformnosti (funktional'noe testirovanie programnykh sistem na osnove formal'nykh modeley)* [Conformance theory (functional testing on formal model base)]. LAP Lambert Academic Publishing.
4. Bourdonov, I.B. & Kosachev, A.S. (2011) Popolnenie spetsifikatsii dlya ioco [Specification Completion for IOCO]. *Programirovanie – Programming and Computer Software*. 37(1). pp. 3–18.
5. Petrenko, A., Yevtushenko, N., Von Bochmann, G. & Dssoul, Ri. (1996) Testing in context: framework and test derivation. *Computer Communications*. 19(14). pp. 1236–1249. DOI: 10.1016/S0140-3664(96)01157-7
6. Kamkin, A. & Chupilko, M. (2011) Obzor sovremennykh tekhnologiy imitatsionnoy verifikatsii apparatury [Survey of modern technologies of simulation-based verification of hardware]. *Programirovanie – Programming and Computer Software*. 37(3). pp. 42–49.
7. Bourdonov, I.B., Kossachev, A.S. & Kulyamin, V.V. (2005) Neizbytochnye algoritmy obkhoda orientirovannykh grafov. Determinirovanny sluchay [Irredundant Algorithms for Traversing Directed Graphs: The Deterministic Case]. *Programirovanie – Programming and Computer Software*. 29(5). pp. 59–69.

8. Levitin, A.V. (2006) *Algoritmy: vvedenie v razrabotku i analiz* [Introduction to the design and analysis of algorithms]. Moscow: Vil'yams. pp. 160–163.
9. Cormen, T., Leiserson, C., Rivest, R. & Stein, C. (2001) *Algoritmy: postroenie i analiz* [Introduction to Algorithms]. 2nd ed. Translated from English by I. Krasikov. Moscow: Vil'yams. pp. 456–458.
10. Bourdonov, I.B. & Kossachev, A.S. (2009) Polnoe testirovanie s otkrytym sostoyaniem ogranichenno nedeterminirovannykh sistem [Complete open-state testing of limitedly nondeterministic systems]. *Programmirovaniye – Programming and Computer Software*. 6. pp. 3–18.
11. Bourdonov, I.B. & Kossachev, A.S. (2001) Semantics of Interaction with Refused Inputs, Divergence and Forbidden Actions. Part 2. The condition of finite complete testing. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 2(15). pp. 89–98. (In Russian).
12. Bourdonov, I.B. & Kossachev, A.S. (2010) Testirovanie konformnosti na osnove sootvetstviya sostoyaniy [Conformance testing based on a state relation]. *Trudy ISP RAN – Proceedings of the Institute for System Programming of the RAS*. 18. pp. 183–220.
13. Bourdonov, I.B. & Kossachev, A.S. (2010) Safe simulation testing of systems with refusals and destructions. *Automatic Control and Computer Sciences* 17(4). pp. 27–40. DOI: 10.3103/S0146411611070042
14. Bourdonov, I.B., Kossachev, A.S. & Kuli Amin, V.V. (2006) Formal Conformance Testing of Systems with Refused Inputs and Forbidden Actions. *Electronic Notes in Theoretical Computer Science*. 164(4). pp. 1–128. DOI: 10.1016/j.entcs.2006.09.008
15. Bourdonov, I.B., Kossachev, A.S. & Kuli Amin, V.V. (2007) Formalizatsiya testovogo eksperimenta [Formalization of Test Experiments]. *Programmirovaniye – Programming and Computer Software*. 5. pp. 3–32.
16. Bourdonov, I.B., Kossachev, A.S. & Kuli Amin, V.V. (2006) [Safety, verification and conformance theory]. *The Proceedings of the Second International Conference on the Problems of Safety and Counteraction Against Terrorism*. Moscow: MTsNMO. pp. 135–158. (In Russian).
17. Bourdonov, I.B. & Kossachev, A.S. (2008) Sistemy s prioriteta mi: konformnost', testirovanie, kompozitsiya [Systems with priority: conformance, testing, composition]. *Trudy ISP RAN – Proceedings of the Institute for System Programming of the RAS*. 14(1). pp. 23–54.
18. Bourdonov, I.B. & Kossachev, A.S. (2009) Testirovanie s preobrazovaniem semantik [Testing with semantics conversion]. *Trudy ISP RAN – Proceedings of the Institute for System Programming of the RAS*. 17. pp. 193–208.
19. Kossachev, A. & Bourdonov, I. (2010) Formal Conformance Verification. In: Petrenko, A., Simao, A. & Maldonado, J.C. (2010) *Short Papers of the 22nd IFIP ICTSS*. Natal, Brazil. pp. 1–6.
20. Bourdonov, I.B. & Kossachev, A.S. (2010) Semantiki vzaimodeystviya s otkazami, divergentsiey i razrusheniem [Interaction semantics with refusals, divergence, and destruction]. *Programmirovaniye – Programming and Computer Software*. 5. pp. 3–23.
21. Bourdonov, I.B. & Kossachev, A.S. (2013) Soglasovanie konformnosti i kompozitsii [Agreement between conformance and composition]. *Programmirovaniye – Programming and Computer Software*. 6. pp. 3–15.

ПРОЕКТИРОВАНИЕ И ДИАГНОСТИКА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

УДК 004.312

DOI: 10.17223/19988605/38/11

Н.Б. Буторина

О СВОЙСТВАХ ПРОГРАММИРУЕМЫХ ЛОГИЧЕСКИХ БЛОКОВ, РЕАЛИЗУЮЩИХ КОДОВЫЕ СЛОВА РАВНОВЕСНЫХ КОДОВ

Исследование поддержано Российским научным фондом (исследовательский проект № 14-19-00218).

Рассматривается проблема синтеза самотестируемого детектора для подмножества кодовых слов. Выделяются свойства тестируемости программируемых логических блоков, на входы которых поступают подмножества кодовых слов специального вида.

Ключевые слова: ПЛБ; самотестируемый детектор; равновесный код.

Увеличивающиеся сложность и значимость дискретных (цифровых) систем требуют их высокой надежности. Сложность систем повышает вероятность возникновения неисправностей в них. Обнаружение неисправности в первый же момент ее проявления на выходах устройства позволяет защитить систему, в которую устройство встроено. Такое обнаружение неисправности может достигаться, например, за счет использования самопроверяемых схем, позволяющих обнаруживать неисправности в режиме нормального функционирования схемы [1]. Обнаружение осуществляется с помощью детектора кодов, который обычно является самотестируемым.

Если при появлении неисправности из некоторого класса неисправностей в процессе нормального функционирования схемы (в рабочей области функционирования) данная неисправность либо обнаружится на выходах схемы в момент ее первого проявления, либо никак не влияет на функционирование схемы (остается необнаружимой в рабочей области функционирования), то такую схему будем называть самопроверяемой относительно рассматриваемого класса неисправностей.

Самотестируемость же означает, что в рамках рассматриваемого класса неисправностей для каждой из неисправностей существует тестовый набор среди множества кодовых слов детектора. В частности, если мы имеем дело с детектором (m, n) -кода (детектором равновесных кодовых слов), то множество кодовых слов включает в себя C_n^m булевых векторов длины n , содержащих m единичных компонент.

Автором рассматриваются детекторы, построенные с использованием программируемых логических блоков.

Ранее [2] предлагался метод проектирования самотестируемого детектора для множества всех кодовых слов равновесного кода (детектора (m, n) -кодов), основанный на использовании программируемых логических блоков (ПЛБ). Предполагалось, что программируемые логические блоки реализованы в рамках LUT-технологий, т.е. могут быть запрограммированы на реализацию любых интересующих нас булевых функций заданного числа переменных.

Рассматривалось множество V неисправностей детектора, которое включает в себя все кратные константные неисправности на входах ПЛБ. При этом в схеме детектора неисправным может быть только один ПЛБ. Предполагалось, что в системе самопроверяемая схема, детектор (m, n) -кодов неисправными могут быть либо схема, либо детектор, но не оба вместе.

К самотестируемому детектору предъявлялись следующие требования:

1) при появлении на выходе схемы (или на входе детектора) в некоторый момент времени t кодового слова детектор должен выдать соответствующий сигнал;

2) в самом детекторе может произойти неисправность из рассматриваемого множества неисправностей V , которая должна быть обнаружима в рабочей области функционирования детектора, т.е. на множестве всех кодовых слов. Это означает, что должен существовать входной набор (кодированное слово (m, n) -кода), на котором эта неисправность проявляется на выходах детектора.

Самотестируемый детектор имеет два выхода, причём комбинации значений сигналов имеют следующие интерпретации:

- а) (01) или (10) означают, что входной набор является кодовым словом (m, n) -кода и детектор исправен;
- б) (00) или (11) означают, что либо входной набор не является кодовым, либо детектор неисправен.

Кодовые слова (m, n) -кода могут быть представлены дизъюнкцией конъюнкций ранга n . Обозначим эту ДНФ $D_n^m(X)$, где $X = \{x_1, x_2, \dots, x_n\}$ – множество переменных. Для представления всевозможных (m, n) -кодированных слов ранее [2] была предложена специальная формула разложения

$$D_n^m(X) = \sum_{i=0}^m D_g^i(X^1) D_s^{m-i}(X^2). \quad (1)$$

Назовем некоторое число k *основой разложения* (k может быть числом входов в ПЛБ), ДНФ D_g^i, D_s^{m-i} – *функциями разложения*. Первоначальное множество переменных X мощности n делится на два подмножества X^1 и X^2 : $|X^1| = g = \lceil n/2 \rceil$, $|X^2| = s = n - g$. Если $g > k$ и $s > k$, то формула (1) снова используется для каждой функции разложения D_g^i, D_s^{m-i} , $i = \overline{0, m}$, и т.д. В результате мы имеем формулу A для всех кодовых слов (m, n) -кода, для наглядного представления которой удобно использовать соответствующее дерево.

Рассмотрим пример. Получим формулу A для D_{14}^7 , $k = 7$:

$$\begin{aligned} X^1 &= \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}, X^2 = \{x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}\}. \\ D_{14}^7 &= D_7^0(X^1)D_7^7(X^2) \vee D_7^1(X^1)D_7^6(X^2) \vee D_7^2(X^1)D_7^5(X^2) \vee D_7^3(X^1)D_7^4(X^2) \vee \\ &\vee D_7^4(X^1)D_7^3(X^2) \vee D_7^5(X^1)D_7^2(X^2) \vee D_7^6(X^1)D_7^1(X^2) \vee D_7^7(X^1)D_7^0(X^2). \end{aligned}$$

При выбранных значениях m, n и k достаточно одного шага разложения с использованием формулы (1).

Обычно детектор строится на той же элементной базе, что и схема. Часто построение детектора основано на определении веса входных кодовых слов. Для этой цели применяются схемы, основанные либо на пороговых схемах, либо на параллельных счетчиках [3, 4]. Такие подходы концептуально исключают построение детектора, ориентированного на подмножества кодовых слов, в то время как на выходах самопроверяемого устройства, к которому подключен детектор равновесных кодовых слов, не всегда достигаются всевозможные кодовые слова. Число l достигаемых кодовых слов может быть существенно меньше числа всевозможных кодовых слов. Из сказанного следует, что некоторые неисправности самотестируемого детектора, подключенного к выходам самопроверяемого устройства, могут оказаться необнаружимыми, так как обнаруживающие неисправности кодовые слова не поступают на входы детектора во время работы самопроверяемого устройства. Необнаруженная неисправность детектора может привести в дальнейшем к тому, что детектор не обнаружит некодовое слово, поступающее на него с выходов самопроверяемого устройства, вследствие неисправности, возникшей в устройстве. Эта неисправность появилась после неисправности в детекторе. Такая ситуация в дискретной системе, состоящей из самопроверяемого устройства и детектора кодов, недопустима.

Выходом из нее может быть дублирование детекторов и сравнение их реакций. Предполагается, что каждый раз неисправность появляется либо в одной из копий детектора, либо в самопроверяемом устройстве. Поскольку возникновение одинаковых неисправностей в копиях детектора маловероятно, такая система способна надежно работать в условиях, когда не все кодовые слова достигаются на выходах самопроверяемого устройства. Недостатком дублирования являются большие аппаратные затраты (более 100%).

Итак, если мы не хотим дублировать детекторы равновесных кодов, то необходимо обеспечение свойства самотестируемости на тех кодовых словах, которые достигаются в процессе работы самопроверяемого дискретного устройства. Реализация такого подхода возможна при синтезе самопроверяемого последовательностного дискретного устройства на этапе кодирования его состояний. Зная, для каких именно кодовых слов построен самотестируемый детектор, мы можем закодировать состояния именно этими кодовыми словами.

Построение самотестируемого детектора (m, n) -кодов для произвольного числа l кодовых слов основано на решении двух подзадач:

1. Представление множества кодовых слов мощности l объединением подмножеств равновесных кодовых слов, так что каждое из подмножеств задается существенным поддеревом дерева формулы А.
2. Построение самотестируемого детектора на базе полученного представления.

Метод представления множества мощности l описан в [5, 6]. Метод построения самотестируемого детектора для подмножества кодовых слов тот же, что и метод построения самотестируемого детектора для полного множества кодовых слов, и основан на покрытии дерева разложения программируемыми логическими блоками [1]. В данной статье выделяются свойства программируемых логических блоков, на входы которых поступают подмножества кодовых слов специального вида.

В дальнейшем будем предполагать, что число k входов в ПЛБ не меньше 6.

1. Выявление свойств программируемых логических блоков и подсхем из них, реализующих кодовые слова равновесного кода

Определение 1. Назовем функцию, представляемую ДНФ D_p^q , $0 < q < p$ функцией типа 1.

Пример такой функции для D_3^2 приведен в табл. 1. В ней столбцы сопоставлены с булевыми переменным, а строки представляют единичные наборы булевой функции (булевы векторы).

Т а б л и ц а 1

Пример функции типа 1

x_1	x_2	x_3
0	1	1
1	0	1
1	1	0

Определение 2. Функцией типа 2 назовём функцию, которая удовлетворяет следующим условиям: в каждом столбце таблицы, представляющей множество ее единичных наборов, единичная компонента присутствует один и только один раз, и в каждой строке число единичных компонент одинаково.

Пример функции типа 2 приведен в табл. 2, которая построена аналогично табл. 1.

Т а б л и ц а 2

Пример функции типа 2

x_1	x_2	x_3	x_4	x_5	x_6
0	0	0	0	1	1
0	0	1	1	0	0
1	1	0	0	0	0

Определение 3. Функцией типа 3 назовём функцию, область единичных значений которой представлена двумя взаимно инверсными векторами.

Пример функции типа 3 приведен в табл. 3, построенной аналогично таблице функции типа 1.

Т а б л и ц а 3

Пример функции типа 3

x_1	x_2	x_3	x_4
0	0	0	0
1	1	1	1

Теорема 1. Для выхода одновыходного ПЛБ, реализующего функцию типа 1, 2 или 3, и кратной неисправности на входных и выходном полюсах ПЛБ либо существует тестовый набор из области единичных значений функции типа 1, 2 или 3, либо кратная неисправность проявляет себя как неисправность «константа 1» на выходе ПЛБ. Доказательство теоремы приведено в [7].

Теорема 2. Если на одном из выходов двухвыходного блока реализуется функция типа 1, 2 или 3, то какой бы ни была вторая функция, сопоставляемая блоку, кратная неисправность на входных полюсах этого блока либо обнаруживается на соответствующем функции типа 1, 2 или 3 тестовом наборе из области единичных значений функции, либо проявляется на этом выходе как константа 1. Доказательство теоремы приведено в [7].

Рассмотрим систему из двух булевых функций с непересекающимися множествами наборов их единичных значений. Будем говорить, что система состоит из ортогональных булевых функций. Обозначим через M объединение множеств единичных наборов функций этой системы.

Теорема 3. Если множество M системы ортогональных булевых функций, реализуемой двухвыходным блоком, представляется таблицей функции типа 1 (2), то кратная неисправность на входах этого блока либо обнаруживается на одном из его выходов при поступлении на входы блока тестового набора из M , либо проявляется на одном из выходов как константная неисправность 1. Доказательство теоремы приведено в [7].

Отметим, что при покрытии программируемыми блоками дерева формулы A [1] с целью построения схемы самотестируемого детектора для всех кодовых слов (покрытие выполняется при просмотре дерева от концевых вершин к корню) сначала используются ПЛБ, реализующие функции D_k^q для соответствующих значений k и q . Здесь k – количество входов в программируемый логический блок, а q – число единичных компонент вектора из области единичных значений функции, реализуемой блоком. Затем используются ПЛБ, реализующие функции типа 2, и только они.

При покрытии подмножества существенных поддеревьев дерева формулы A программируемыми блоками, реализующими функции D_k^q , может оказаться, что на входы блока поступают только некоторые кодовые слова из множества, представляемого D_k^q . В этом случае мы не можем гарантировать свойство самотестируемости построенной схемы детектора, опираясь на приведенные выше теоремы и метод синтеза [1]. В такой ситуации приходится исследовать подмножества кодовых слов, поступающих на входы ПЛБ. Будем использовать как одновыходные, так и двухвыходные ПЛБ.

Обозначим через M всё множество кодовых слов (q, k) -кода, реализуемого программируемым логическим блоком, а через M_l – его подмножество мощности l , такое, что в каждом столбце таблицы, содержащей все векторы подмножества M_l , присутствуют как нулевые, так и единичные значения.

Определение 4. Назовем функцию, область единичных значений которой совпадает с подмножеством M_l , функцией типа 1.

Пример такой функции приведен в табл. 4.

Т а б л и ц а 4

Пример функции типа 1' ($l = 11$)

x_1	x_2	x_3	x_4	x_5	x_6
0	0	0	1	1	1
0	0	1	0	1	1
0	1	0	1	0	1
0	1	0	1	1	0
0	1	1	0	0	1
1	0	1	0	1	0
1	0	1	1	0	0
1	1	0	0	0	1
1	1	0	0	1	0
1	1	0	1	0	0
1	1	1	0	0	0

Пусть ПЛБ реализует функцию D_k^q . Здесь k – количество входов в программируемый логический блок, а q – число единичных компонент, $0 < q < k$. Пусть на вход программируемого логического блока приходит некоторое подмножество M_l кодовых слов (q, k) -кода.

Теорема 4. Если на входы ПЛБ, реализующего множество M всех кодовых слов (q, k) -кода, $0 < q < k$, подаётся их подмножество M_l , то для кратной неисправности на входах ПЛБ либо существует тестовый набор из M_l , либо кратная неисправность проявляет себя как неисправность «константа 1» на выходе ПЛБ.

Доказательство теоремы приведено в [8].

Теорема 5. Если на одном из выходов двухвыходного блока реализуется функция типа 1', то какой бы ни была вторая функция, сопоставляемая блоку, для кратной неисправности на входах ПЛБ либо существует тестовый набор из M_l , такой что неисправность проявляется на выходе блока, реализующего функцию типа 1', либо неисправность проявляется на этом выходе как константа 1.

Доказательство теоремы приведено в [8].

Рассмотрим систему из двух ортогональных булевых функций, в которой M есть объединение множеств единичных наборов функций этой системы.

Теорема 6. Если множество M системы ортогональных булевых функций, реализуемой двухвыходным блоком, представляется таблицей функции типа 1', то для кратной неисправности на входах ПЛБ либо существует тестовый набор из M , такой что неисправность проявляется на одном из выходов блока, либо неисправность проявляется на одном из выходах блока как константа 1.

Доказательство. Утверждение следует из того факта, что множества переменных функций одного и того же блока совпадают, а сами функции реализуются на отдельных блоках памяти (LUT). На втором выходе блока кратная неисправность может вообще не проявляться. Это зависит от вида функции, реализуемой вторым выходом.

Теорема доказана.

Воспользуемся приведенными выше теоремами для анализа заданного множества кодовых слов на возможность построения для него самотестируемого детектора.

Алгоритм анализа подмножества кодовых слов.

1) Разбиваем множество n переменных на w подмножеств так, как это делается при построении формулы A для заданного k . В результате подмножество кодовых слов разбивается на w множеств фрагментов булевых векторов. Длина каждого фрагмента булева вектора не более k . Анализируем векторы каждого фрагмента X^i , $i = \overline{1, w}$.

2) Рассмотрим фрагмент, сопоставляемый X^i . Разбиваем его векторы на части по весу.

3) Для каждой части, которая не совпадает ни с функцией типа 1, ни с функцией типа 1', находим такую часть, что пара представляющих обе части ортогональных функций задает единичные наборы функции типа 1'. Каждая часть может входить только в одну пару.

4) Если необходимое количество пар построить не удастся, то самотестируемость детектора рассматриваемым методом синтеза невозможно обеспечить. **ВЫХОД.**

5) Иначе $i = i + 1$.

6) Если $i < w$, переходим в п. 2.

7) При $i = w$ самотестируемость детектора обеспечивается. **ВЫХОД.**

Покажем, что для приведенного выше подмножества кодовых слов, $l = 11$, можно построить детектор, используя данный алгоритм. Это подмножество кодовых слов является частью кодовых слов, представляемых ДНФ D_6^3 . Воспользуемся разложением по формуле (1):

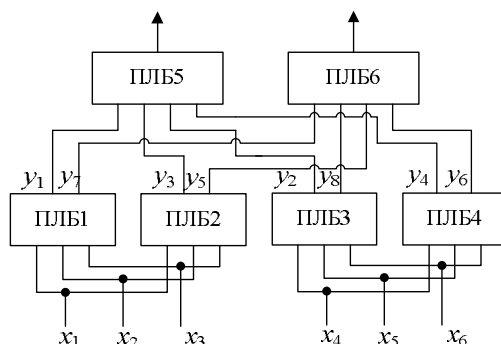
$$D_6^3(x_1, x_2, x_3, x_4, x_5, x_6) = D_3^0(x_1, x_2, x_3)D_3^3(x_4, x_5, x_6) \vee D_3^1(x_1, x_2, x_3)D_3^2(x_4, x_5, x_6) \vee D_3^2(x_1, x_2, x_3)D_3^1(x_4, x_5, x_6) \vee D_3^3(x_1, x_2, x_3)D_3^0(x_4, x_5, x_6). \quad (2)$$

Анализируя табл. 5, видим, что на некоторые ПЛБ, реализующие кодовые слова ДНФ D_k^q , поступают не все кодовые слова.

Описание элементов самотестируемого детектора для подмножества мощности $l = 11$

№ ПЛБ	Функция, реализуемая на 1-м выходе	Функция, реализуемая на 2-м выходе	Обозначение 1-го выхода	Обозначение 2-го выхода	Обеспечение самотестируемости
1	$D_3^0(x_1, x_2, x_3)$	$D_3^3(x_1, x_2, x_3)$	y_1	y_7	Система функций, реализуемая ПЛБ, удовлетворяет теореме 6
2	$D_3^1(x_1, x_2, x_3)$	$D_3^2(x_1, x_2, x_3)$	y_3	y_5	На вход ПЛБ, реализующего ДНФ $D_3^1(x_1, x_2, x_3)$, приходит неполное множество кодовых слов. На вход ПЛБ, реализующего ДНФ $D_3^2(x_1, x_2, x_3)$, приходит полное множество кодовых слов, т.е. реализуется система функций, удовлетворяющая теореме 2
3	$D_3^3(x_4, x_5, x_6)$	$D_3^0(x_4, x_5, x_6)$	y_2	y_8	Система функций, реализуемая ПЛБ, удовлетворяет теореме 6
4	$D_3^2(x_4, x_5, x_6)$	$D_3^1(x_4, x_5, x_6)$	y_4	y_6	На обоих выходах ПЛБ реализуются функции типа 1 (теорема 2)
5	$\overline{y_1 y_2 y_3 y_4} \vee \overline{y_1 y_2 y_3 y_4}$	–	Выход детектора		ПЛБ реализует функцию типа 2 (теорема 1)
6	$\overline{y_5 y_6 y_7 y_8} \vee \overline{y_5 y_6 y_7 y_8}$	–	Выход детектора		ПЛБ реализует функцию типа 2 (теорема 1)

Схема детектора представлена на рис. 1.

Рис. 1. Схема самотестируемого детектора для подмножества мощности $l = 11$

Однако не для любого подмножества кодовых слов можно обеспечить самотестируемость детектора предложенным подходом. Рассмотрим подмножество кодовых слов в табл. 6. Мощность l -го подмножества также равна 11. Для этого подмножества нельзя построить самотестируемый детектор.

Таблица 6

Пример подмножества, на котором нельзя обеспечить свойство самотестируемости детектора ($l = 11$)

x_1	x_2	x_3	x_4	x_5	x_6
0	0	0	1	1	1
0	0	1	0	1	1
0	0	1	1	0	1
0	0	1	1	1	0
0	1	0	0	1	1
0	1	0	1	0	1
0	1	0	1	1	0
1	0	0	0	1	1
1	0	0	1	0	1
1	0	0	1	1	0
1	1	0	1	0	0

Очевидно, свойство самотестируемости нарушается по пункту 4 алгоритма анализа подмножества кодовых слов.

Заключение

Выявлены свойства ПЛБ, позволяющие построить самотестируемый детектор для заданного подмножества кодовых слов. Предложен алгоритм анализа подмножества кодовых слов с целью построения самотестируемого детектора, реализующего это подмножество.

ЛИТЕРАТУРА

1. Lala P.K. Self-Checking and Fault-Tolerant Digital Design. Morgan Kaufmann Pub. 2000.
2. Матросова А.Ю., Никитин К.В. Синтез самотестируемого детектора (m, n) -кодов на программируемых логических блоках // Вестник Томского государственного университета. Приложение. 2003. № 6. С. 124–136.
3. Anderson D.A., Metze G. Design of totally self-checking check circuits for m -out-of- n codes // IEEE Trans. Computers, C-22. 1973. P. 263–269.
4. Marouf M.A., Friedman A.D. Efficient design of sel-checking checker for any m -out-of- n code // IEEE Trans. Computers. 1978. C-27. P. 482–490.
5. Butorina N., Burkatovskaya J. The properties of the essential subtrees of a graph representation of the (m, n) -codewords // Proceedings of the 7th International Forum on Strategic Technology (IFOST 2012) IEEE Computer Society, 2012. P. 472–476.
6. Butorina N., Ostanin S. Implementation by the Special Formula of an Arbitrary Subset of Code Words of (m, n) -code for Designing of a Self-Testing Checker // Proceeding of IEEE East-West Design & Test Symposium. IEEE. 2011. P. 255–258.
7. Буркатовская Ю.Б., Буторина Н.Б., Матросова А.Ю. Синтез самотестируемых детекторов произвольного числа равновесных кодов // Вестник Томского государственного университета. Приложение. 2006. № 17. С. 190–197.
8. Буторина Н.Б. Свойства программируемых логических блоков, реализующих кодовые слова равновесных кодов // Динамика сложных систем – XXI век. 2015. Т. 9, № 4. С. 50–53.

Буторина Наталья Борисовна. E-mail: nnatta07@mail.ru

Национальный исследовательский Томский государственный университет

Поступила в редакцию 21 сентября 2016 г.

Butorina Natalia B. (National Research Tomsk State University, Russian Federation).

The properties of programmable logic blocks realizing code words of (m, n) -code.

Keywords: self-checking circuit; (m, n) -code; self-testing checker; CLB.

This paper deals with a problem of self-testing checker (STC) design for a subset of code words of (m, n) -code.

A self-testing (m, n) -code checker is used in concurrent error detection (CED) technique under suggestion that all code words appear on its inputs. It is supposed that either the circuit or the checker may be fault but not both. In practice only some code words are reachable on outputs of self-checking circuit and consequently on inputs of self-testing checker. In that case some faults of the checker can be undetectable as the proper code words do not appear on the circuit outputs. As a result, wrong code word on the fault self-checking circuit may be not detected by the checker. That is why it is necessary to provide the self-testing property of the checker for the corresponding sub-set of all code words of the same weight.

A method of a self-testing (m, n) -code checker design based on using of CLBs was proposed before. CLBs were supposed to be realized in the frame of the LUT-technology and, hence, could be programmed to implement any Boolean functions of a fixed number of variables. The set V of the checker faults includes all multiple stuck-at faults at the CLB inputs. Only one CLB in the checker can be fault.

The self-testing checker has to satisfy the following conditions:

- 1) if a non-code word appears at the output of the circuit (or at the input of the checker) then the checker should give the appropriate signal;
- 2) any fault from the set V should be detected in the set of all code words. It means that there should be a code word for which the fault manifests itself at the checker outputs.

The self-testing checker has two outputs with the following meanings of signals combinations:

- a) (01) or (10) mean that the input word is a word of the (m, n) -code and the checker is faultless;
- b) (00) or (11) mean that either the input word is non code one or the checker is fault.

To represent all possible (m, n) -code words, a special decomposition formula was proposed

$$D_n^m(X) = \sum_{i=0}^m D_g^i(X^1) D_s^{m-i}(X^2).$$

The decomposition formula is used several times. The result of its application is represented by the corresponding tree. The design of a self-testing checker for arbitrary number l of code words of (n, m) -code is based on the decision of two tasks:

1. Getting representation of l code words by sum of code words presented by proper sub-tree of the tree.
2. Deriving the self-testing checker based on the received representation.

Here the second task of the problem is considered.

The properties of programmable logic blocks which inputs receive a sub-set of code words of special type are described. Based on the properties the algorithm of analysis of the given sub-set is developed. It allows finding out if it is possible to design a self-testing checker or not.

REFERENCES

1. Lala, P.K. (2000) *Self-Checking and Fault-Tolerant Digital Design*. Morgan Kaufmann.
2. Matrosova, A.Yu., Nikitin, K.V. (2003) Sintez samotestiruемого детектора (m, n)-кодов на программируемых логических блоках [Design of a self-testing (m,n)-code checkers using configurable logic blocks]. *Vestnik Tomskogo gosudarstvennogo universiteta. Prilozhenie – Tomsk State University Journal. Supplement*. 6. pp. 124–136.
3. Anderson, D.A. & Metze, G. (1973) Design of totally self-checking check circuits for m-out-of-n codes. *IEEE Trans. Computers*. C-22. pp. 263–269. DOI: 10.1109/T-C.1973.223705
4. Marouf, M.A. & Friedman, A.D. (1978) Efficient design of sel-checking checker for any m-out-of-n code. *IEEE Trans. Computers*. C-27. pp. 482–490. DOI: 10.1109/TC.1978.1675138
5. Butorina, N. & Burkatovskaya, J. (2012) The properties of the essential subtrees of a graph representation of the (m,n) –codeword. *Proceedings of the 7th International Forum on Strategic Technology (IFOST 2012)*. IEEE Computer Society. pp. 472–476
6. Butorina, N. & Ostanin, S. (2011) Implementation by the Special Formula of an Arbitrary Subset of Code Words of (m, n)-code for Designing of a Self-Testing Checker. *Proceeding of IEEE East-West Design&Test Symposium*. IEEE. pp. 255–258
7. Burkatovskaya, Yu.B., Butorina, N.B. & Matrosova, A.Yu. (2006) Sintez samotestiruemykh детекторов произвольного числа равновесных кодов [Design of self-testing checkers of an arbitrary number of (m,n)-code words]. *Proceedings of the Vth Siberian scientific school-seminar with international participation "Computer Security and Cryptography" - SIBECRYPT'06*. Tomsk. pp. 190-197. (In Russian).
8. Butorina, N.B. & Sidendirzieva, S.P. (2008) Svoystva программируемых логических блоков, realizuyushchikh kodovye slova равновесных кодов [Design of self-testing checkers of (m,n)-code words]. *Dinamika slozhnykh sistem–XXI vek – Dynamics of Dif-ficult Systems–XXI*. pp. 44–44.

МЕМУАРЫ, ПАМЯТНЫЕ ДАТЫ, ПЕРСОНАЛИИ

85 лет профессору Тарасенко Феликсу Петровичу

6 марта 2017 г. исполнилось 85 лет доктору технических наук, профессору, академику МАНВШ, члену-корреспонденту РАЕН, заслуженному деятелю науки РФ, почетному работнику высшего профессионального образования РФ, заслуженному профессору Томского государственного университета Феликсу Петровичу Тарасенко.

После окончания мужской средней школы № 41 г. Красноярска с золотой медалью в 1950 г. он поступил на радиофизический факультет ТГУ. В 1955 г. окончил с отличием университет по специальности «радиофизика-электроника». С 1955 г. Ф.П. Тарасенко продолжил свою деятельность в ТГУ: аспирант, ассистент, доцент (1955–1960), зав. кафедрой электронной вычислительной техники и автоматики ТГУ (1960–1964), зав. кафедрой статистической радиофизики ТГУ (1964–1965), зав. отделом кибернетики Сибирского физико-технического института им. акад. В.Д. Кузнецова (1970–1977), зав. кафедрой теоретической кибернетики ТГУ (1977–1998), декан Международного факультета управления (МФУ) ТГУ (1992–2013).

В разные годы читал в ТГУ курсы: физики; кибернетики; статистики; теории систем, системного анализа; менеджмента. Научная деятельность Ф.П. Тарасенко связана с проблемами теории информации, кибернетики, непараметрической и робастной статистики, системного анализа. В 1975 г. в совете при ТГУ защитил диссертацию «Проблемы передачи информации по каналам с частично или полностью неизвестными распределениями шумов» на соискание ученой степени доктора технических наук. Является основоположником научной школы, развивающейся в двух направлениях: «Непараметрические и робастные статистические методы в кибернетике» и «Статистический анализ данных и разработка моделей социально-экономических систем». По этим направлениям им подготовлено 7 докторов и 36 кандидатов наук. Феликс Петрович опубликовал 4 монографии и более 165 научных работ в российских и зарубежных научных изданиях. Научное признание выразилось в избрании Ф.П. Тарасенко в состав: Научного совета АН СССР по кибернетике (секция теории информации), Советского комитета по автоматическому управлению ИФАК, рабочей группы по советско-американскому сотрудничеству по кибернетике АН СССР. Ф.П. Тарасенко удостоен премии им. В.М. Глушкова с вручением золотой медали лауреата за разработку теоретических основ и системного синтеза сложных наукоемких территориальных информационных систем.

Феликс Петрович – прекрасный преподаватель, инициатор многих образовательных проектов ТГУ. Он активно популяризирует и развивает идеи системного анализа и системного подхода в менеджменте. Свой курс по прикладному системному анализу он читал в вузах в разных городах для студентов различных специальностей. Им написано учебное пособие «Прикладной системный анализ (Наука и искусство решения проблем)» (Москва: Кнорус, 2010, испр. и доп. в 2017 г.) и переведены на русский язык шесть книг классиков системного анализа (2007–2016), две из них – в рамках проекта НИУ ТГУ «Монографии выдающихся зарубежных исследователей: впервые на русском языке». Эти книги стали руководством по выработке решений для многих современных руководителей.



Своей научной и практической работой Ф.П. Тарасенко внес существенный вклад в развитие Томской области и г. Томска. Об этом свидетельствует ряд фактов из его профессиональной биографии. В 1970-х гг. он входил в комплексную научно-техническую группу, которая занималась разработкой основ технического проекта по созданию автоматизированной системы управления хозяйством Томской области (был заместителем генерального конструктора АСУ Томской области); участвовал в координации программно-целевого комплекса «Лес и лесопереработка» – одного из 19 народнохозяйственных комплексов Томской области (1970–1975 гг.). Будучи деканом МФУ организовал подготовку высококвалифицированных специалистов по Государственному и муниципальному управлению для г. Томска и Томской области. В числе первых слушателей по программе второго высшего образования были главы муниципальных образований, их заместители, сотрудники администрации области. Феликс Петрович принимал участие в разработке законодательной основы научно-инновационных процессов в сотрудничестве с ТО СО РАН и вузов г. Томска (2000–2001). Он был инициатором создания и первым президентом Ротарианского клуба в г. Томске, председателем комитета международных проектов томского Ротари-клуба, председателем Всероссийского фонда образования (Томское городское отделение), инициатором создания научно-образовательного канала (НОК) на телевидении в г. Томске. Участвовал в реализации международной программы «Образование и бизнес в Сибири». Осуществлял научное руководство работ по оценке перспектив нефтегазовых районов Томской области (2000–2001) и работы по теме «Отработка методики разработки программ социально-экономического развития муниципальных образований на примере Томского района» (2003 г.). Оказывал благотворительную помощь ряду томских детских и медицинских учреждений. Результаты его работы были отмечены бронзовой медалью ВДНХ.

Ф.П. Тарасенко являлся главным редактором научно-практического журнала «Проблемы управления в социальных системах», издаваемого в ТГУ. В течение почти 40 лет входил (часто возглавляя его) в оргкомитет Всесоюзной (позднее Международной) школы-семинара по непараметрическим и робастным статистическим методам в кибернетике. Неоднократно приглашался для чтения лекций на международные симпозиумы. В 1967–1968 гг. как эксперт ЮНЕСКО был лектором Дар-эс-Саламского университета в Танзании. Он является членом Американского математического общества. Его биография включена в справочники «Кто есть кто в мире», «Кто есть кто в России». В 1994 г. Американским биографическим институтом назван человеком года. Награжден орденом Трудового Красного Знамени, медалями «За доблестный труд в ознаменование 100-летия со дня рождения В.И. Ленина», «Ветеран труда» и «За заслуги перед Томским университетом». Отмечен нагрудными знаками «За заслуги перед городом Томском» и «Отличник высшей школы».

И в настоящее время Феликс Петрович полон сил и энергии. Читает свой курс по прикладному системному анализу для иностранных студентов на английском языке, а также кампусные курсы для студентов вузов и предпринимателей г. Томска, переводит на русский язык новые зарубежные монографии и публикации, участвует в работе международных конференций как за рубежом, так и в России.

Дорогой Феликс Петрович!
Поздравляем Вас с юбилеем и желаем здоровья на многие годы!
Творческих Вам успехов!

*Факультет прикладной математики и кибернетики ТГУ.
Редакционная коллегия журнала «Вестник ТГУ. Управление,
вычислительная техника и информатика».
Кафедра теоретической кибернетики*

СВЕДЕНИЯ ОБ АВТОРАХ

БУРАКОВ Дмитрий Петрович – кандидат технических наук, доцент кафедры математики и моделирования факультета промышленного и гражданского строительства Петербургского государственного университета путей сообщения Императора Александра I. E-mail: bdsu@yandex.ru

БУРДОНОВ Игорь Борисович – доктор физико-математических наук, ведущий научный сотрудник Института системного программирования РАН (г. Москва). E-mail: igor@ispras.ru

БУТОРИНА Наталья Борисовна – старший преподаватель кафедры программирования факультета прикладной математики и кибернетики Национального исследовательского Томского государственного университета. E-mail: nnatta07@mail.ru

ВАСИЛЬЕВ Вячеслав Артурович – профессор, доктор физико-математических наук, профессор кафедры высшей математики и математического моделирования факультета прикладной математики Национального исследовательского Томского государственного университета. E-mail: vas@mail.tsu.ru

ГАРАЕВА Эсмירה Акиф кызы – научный сотрудник Института систем управления НАН Азербайджана (г. Баку, Азербайджан). E-mail: esmira.qarayeva@mail.ru

ГАРИНА Марина Игоревна – кандидат технических наук, доцент кафедры математики и моделирования факультета промышленного и гражданского строительства Петербургского государственного университета путей сообщения Императора Александра I. E-mail: migarina@gmail.com

ГОРОХОВ Максим Михайлович – профессор, доктор физико-математических наук, заведующий кафедрой информационных систем инженерно-экономического факультета Ижевского государственного технического университета им. М.Т. Калашникова. E-mail: insys2005@mail.ru

ДОГАДОВА Татьяна Валерьевна – аспирантка кафедры высшей математики и математического моделирования факультета прикладной математики Национального исследовательского Томского государственного университета. E-mail: auroa1900@mail.ru

ИВАНОВ Алексей Витальевич – аспирант Института радиоэлектроники и информационных технологий ИРИТ–РТФ Уральского федерального университета (г. Екатеринбург). E-mail: av.ivanov.2014@yandex.ru

КОСАЧЕВ Александр Сергеевич – кандидат физико-математических наук, ведущий научный сотрудник Института системного программирования РАН (г. Москва). E-mail: kos@ispras.ru

МАНСИМОВ Камил Байрамали оглы – доктор физико-математических наук, профессор, заведующий кафедрой математической кибернетики Бакинского государственного университета (г. Баку, Азербайджан). E-mail: mansimovbkamil@gmail.com

МЕДВЕДЕВ Геннадий Алексеевич – профессор, доктор физико-математических наук, профессор факультета прикладной математики и информатики Белорусского государственного университета (г. Минск, Беларусь). E-mail: MedvedevGA@bsu.by

ПАЗНИКОВ Алексей Александрович – кандидат технических наук, научный сотрудник Федерального государственного бюджетного учреждения науки Институт физики полупроводников им. А.В. Ржанова Сибирского отделения Российской академии наук (г. Новосибирск). E-mail: apaznikov@gmail.com

ПЕРЕВЕДЕНЦЕВ Денис Алексеевич – аспирант кафедры «Информационные системы» инженерно-экономического факультета Ижевского государственного технического университета им. М.Т. Калашникова. E-mail: perevedencew@mail.ru

РУБАН Анатолий Иванович – доктор технических наук, профессор, профессор кафедры информатики Института космических и информационных технологий Сибирского федерального университета (г. Красноярск). E-mail: ai-rouban@mail.ru

САРАПУЛОВ Алексей Валерьевич – магистрант кафедры радиотехнического факультета Уральского федерального университета (г. Екатеринбург). E-mail: aesev@mail.ru

ТРОФИМОВ Сергей Павлович – кандидат физико-математических наук, доцент Института радиоэлектроники и информационных технологий ИРИТ–РТФ Уральского федерального университета (г. Екатеринбург). E-mail: tsp61@mail.ru

УМАНСКИЙ Алексей Борисович – кандидат технических наук, сотрудник АО «Научно-производственное объединение автоматики им. академика Н.А. Семихатова» (г. Екатеринбург). E-mail: pdwn1982@yandex.ru

Научный журнал

**ВЕСТНИК
ТОМСКОГО
ГОСУДАРСТВЕННОГО
УНИВЕРСИТЕТА**

**УПРАВЛЕНИЕ,
ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА
И ИНФОРМАТИКА**

**TOMSK STATE UNIVERSITY
JOURNAL OF CONTROL AND COMPUTER SCIENCE**

2017. № 38

Редактор Н.А. Афанасьева
Корректор Е.Г. Шумская
Оригинал-макет А.И. Лелоюр
Редакторы-переводчики: Г.М. Кошкин; В.Н. Горенинцева
Дизайн обложки Л.Д. Кривцовой

Подписано к печати 16.03.2017 г. Формат 60х84¹/₈.
Гарнитура Times. Печ. л. 10,2; усл. печ. л. 9,7.
Тираж 250 экз. Заказ № 2421. Цена свободная.

Дата выхода в свет 01.04.2017 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)–53-15-28; 8(382-2)–52-96-75
Сайт: <http://publish.tsu.ru>; E-mail: rio.tsu@mail.ru