

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 25

2019

№ 5

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

САПР

КОМПЬЮТЕРНАЯ ГРАФИКА

МЕТОДЫ ПРОГРАММИРОВАНИЯ

ОПЕРАЦИОННЫЕ СИСТЕМЫ И СРЕДЫ

ТЕЛЕКОММУНИКАЦИИ
И ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ

ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

НЕЙРОСЕТИ И
НЕЙРОКОМПЬЮТЕРЫ

СТРУКТУРНЫЙ СИНТЕЗ

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ
СИСТЕМЫ

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

ОПТИМИЗАЦИЯ И МОДЕЛИРОВАНИЕ

ИТ В ОБРАЗОВАНИИ

ГИС

Рисунки к статье Н. П. Деменкова, Е. А. Микрина, И. А. Мочалова
«НЕЧЕТКОЕ ОПТИМАЛЬНОЕ УПРАВЛЕНИЕ ЛИНЕЙНЫМИ СИСТЕМАМИ.
Часть 1. ПОЗИЦИОННОЕ УПРАВЛЕНИЕ»

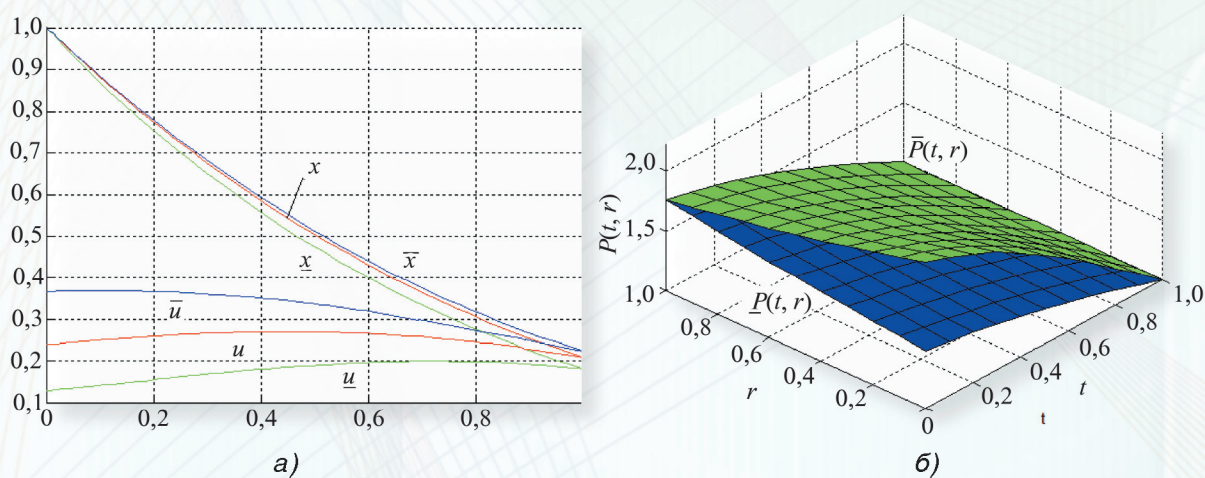


Рис.3. Оптимальные траектории и управления для нечеткого регулятора, синтезированного при использовании квадратичного функционала

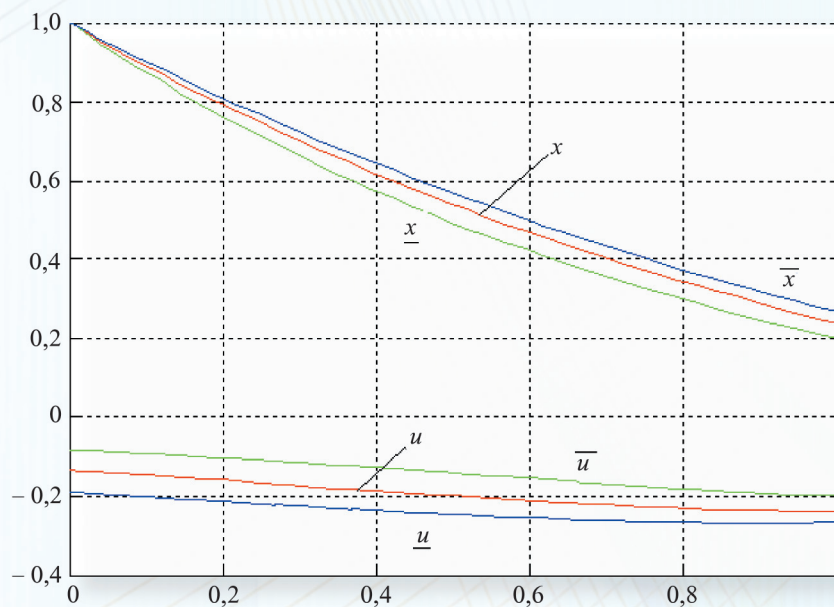


Рис. 4. Оптимальные траектории и вид $K(t)$ для нечеткого регулятора S-типа, синтезированного по критерию обобщенной работы

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 25
2019
№ 5

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

DOI 10.17587/issn.1684-6400

УЧРЕДИТЕЛЬ

Издательство "Новые технологии"

СОДЕРЖАНИЕ

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

Деменков Н. П., Микрин Е. А., Мочалов И. А. Нечеткое оптимальное управление линейными системами. Часть I. Позиционное управление 259

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

Кулагин В. П., Малых Е. С. Проектирование матричных вычислительных структур с использованием сетей Петри 271

Юсупов Р. М., Воробьев В. И., Петров М. Ю. Представление эволюции средств вычислительной техники в экспозиции Музея истории СПИИРАН 283

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Грушо Н. А., Тимонина Е. Е. Сравнение архитектур многоагентных систем . . . 293

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

Абдуллаев Н. Т., Дышин О. А., Ибрагимов И. Д., Ахмедова Х. Р. Сегментирование нестационарных физиологических сигналов с фрактальными свойствами 300

НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

Савченко Л. В. Система постановки производства на основе сверточных нейронных сетей и информационной теории восприятия речи 313

Главный редактор:

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

Зам. главного редактора:

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

Редакционный совет:

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

Редакционная коллегия:

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

Редакция:

БЕЗМЕНОВА М. Ю.

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

Журнал включен в систему Российского индекса научного цитирования и базу данных RSCI на платформе Web of Science.

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

INFORMATION TECHNOLOGIES

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Vol. 25
2019
No. 5

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

DOI 10.17587/issn.1684-6400

ISSN 1684-6400

CONTENTS

MODELING

Demenev N. P., Mikrin E. A., Mochalov I. A. Fuzzy Optimal Control of Linear Systems. Part 1. Positional Control 259

COMPUTING SYSTEMS AND NETWORKS

Kulagin V. P., Malych E. S. Design of Matrix Computing Structures using Petri Nets 271

Yusupov R. M., Vorobiev V. I., Petrov M. Yu. Presentation of Computer Evolution in the Exposition of SPIIRAS History Museum 283

INTELLIGENT SYSTEMS AND TECHNOLOGIES

Grusho N. A., Timonina E. E. Comparison of Architectures of Multi-Agent Systems 293

DIGITAL PROCESSING OF SIGNALS AND IMAGES

Abdullaev N. T., Dyshin O. A., Ibrahimova I. D., Ahmadova Kh. R. Segmentation of Non-Stationary Physiological Signals with Fractal Properties 300

NEUROTECHNOLOGIES

Savchenko L. V. Computer-Assisted Language Learning Based on Convolutional Neural Networks and Information Theory of Speech Perception 313

Editor-in-Chief:

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

Deputy Editor-in-Chief:

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

Chairman:

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

Editorial Board Members:

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

Editors:

Bezmenova M. Yu.

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

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

Н. П. Деменков, канд. техн. наук, доц., dnp@bmstu.ru,
Е. А. Микрин, д-р техн. наук, проф., evgeny.mikrin@bmstu.ru,
И. А. Мочалов, д-р техн. наук, проф., intelsyst@mail.ru,
Москва, МГТУ им. Н. Э. Баумана

Нечеткое оптимальное управление линейными системами. Часть 1. Позиционное управление

В части 1 рассмотрены начальные задачи (задачи Коши) для нечетких нелинейных дифференциальных уравнений типа Риккати, которые возникают при решении задач синтеза оптимальных линейных регуляторов методом динамического программирования, и для нечетких линейных дифференциальных уравнений, которые появляются при синтезе регуляторов по критерию обобщенной работы. Полагается, что первоначально решается четкая задача оптимизации управления, а далее, после получения соответствующего обыкновенного дифференциального уравнения, параметры этого уравнения заменяются нечеткими, и решается соответствующее нечеткое дифференциальное уравнение. В части 2 рассмотрены двухточечные краевые задачи, которые возникают при решении задач оптимального управления на основе принципа максимума и трансформируются в начальные задачи. Для всех случаев приведены примеры.

Ключевые слова: нечеткие краевые задачи, синтез нечетких оптимальных регуляторов, нечеткие дифференциальные уравнения, функция принадлежности, нечеткая начальная задача, принцип максимума, динамическое программирование, критерий обобщенной работы

Введение

Разнообразные задачи естествознания и техники приводятся к задачам оптимизации интегральных критериев, которые решаются методами вариационного исчисления (например, это задачи о преломлении света, о геодезических линиях и т. д.) [1–3]. Некоторые положения вариационного исчисления, примененные к задачам синтеза оптимального управления, приводят к использованию динамического программирования, критерия обобщенной работы и принципа максимума [4–7]. Эти методы имеют свои преимущества и недостатки при решении практических задач.

В методе динамического программирования и в методе на основе критерия обобщенной работы минимизируются функционалы качества управления для объекта управления (ОУ), представляемого в виде начальной задачи (задачи Коши) для обыкновенного дифференциального уравнения [2]. Эти методы отличаются формой минимизируемого функционала, но в обоих случаях для нахождения

оптимального управления необходимо решать уравнения в частных производных с крайевыми условиями.

При использовании принципа максимума для нахождения оптимального управления необходимо решать двухточечную краевую задачу для системы обыкновенных дифференциальных уравнений, которую в некоторых важных случаях (например, при оптимизации линейных систем с квадратичным функционалом) можно преобразовать к задаче Коши [8].

Важной особенностью задач оптимального управления является неопределенность динамических параметров ОУ и соответствующих начальных условий. В традиционной математической постановке неопределенность обычно описывается стохастическими моделями. В этом случае появляются задачи оптимального управления стохастическими системами [9, 10]. В последнее время неопределенность обычно представляют в нечетких терминах или в сочетании нечеткости и стохастичности.

Нечеткость в задачах управления используется в двух направлениях. В первом нечеткость

применяется для реализации нечетких логических систем управления и конструирования традиционных регуляторов с использованием нечетких алгоритмов [11]. Второе направление связано с представлением классических дисциплин в нечеткой интерпретации [12, 13]. Это приводит к появлению новых свойств при нечеткой реализации [14, 15]. При решении нечетких систем линейных алгебраических уравнений (НСЛАУ) появляются "сильные" и "слабые" решения [16, 17], у нечетких обыкновенных дифференциальных уравнений при определенных условиях имеет место не единственность решения [18, 19], для нечетких марковских случайных процессов, в частности для марковских цепей, возникают нечеткие состояния, что приводит к уменьшению размерности переходной матрицы [20, 21] и т. д.

В связи с этим важной проблемой является решение задачи оптимального управления в нечеткой трактовке, когда динамические параметры ОУ и краевые условия представляются нечеткими переменными. Весьма актуальной является задача определения типов нечетких оптимальных управлений с представлением их в сильной/слабой формах. Тип и форма управления дают новые качества нечеткого оптимального управления и определяют научную новизну предлагаемой ниже работы. В качестве методики синтеза оптимального управления, используемой в настоящей статье, реализуется прием, принятый в теории нечетких множеств, когда часть исходной задачи решается в четкой постановке, а другая ее часть — в нечеткой постановке. Объектами исследования здесь являются методы динамического программирования Р. Беллмана и обобщенной работы А. А. Красовского в задаче синтеза нечетких линейных регуляторов, представляемых в виде задачи Коши для нечетких нелинейных дифференциальных уравнений (не-

четкая начальная задача). К этой же проблеме относится и задача нахождения нечеткого оптимального управления с использованием принципа максимума Л. С. Понтрягина, когда соответствующая двухточечная краевая задача может быть трансформируема к задаче Коши.

1. Базовые определения и обозначения

В статье приняты следующие обозначения: нечеткие переменные имеют нижний индекс "н", например, x_n — нечеткая переменная (элемент); $y_n(\mathbf{x})$ — нечеткая функция многих переменных, где $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, $y'_{n x_i}(\mathbf{x})$ — нечеткая производная по переменной x_i ; $\dot{x}_n(t)$ — нечеткая производная вектора x_n по времени.

Для более полного понимания методов решения нечетких оптимизационных задач приведены основные понятия теории нечетких множеств, которые используются в данной работе. К ним относятся: нечеткая треугольная переменная, нечеткая функция, нечеткий функционал, нечеткая производная, нечеткая начальная задача, типы нечетких решений.

Нечеткая треугольная переменная x_n с функцией принадлежности $r(x)$ задается посредством трех чисел $a_1 < a_2 < a_3$, $a_i \in R$, $i = \overline{1,3}$. График $r(x)$ в плоскости (x, r) имеет форму треугольника с основанием (support) $\text{supp}x_n = [a_1, a_3]$ и высотой $\text{hgt}x_n = 1$, исходящей из точки с координатами $(x = a_2; r = 0)$ (рис. 1).

Для x_n в этом случае приняты следующие обозначения:

$$\begin{aligned} x_n = (a_1|a_2|a_3) &\Leftrightarrow (r(x) = (x - a_1)(a_2 - a_1)^{-1}; \\ \bar{r}(x) &= (-x + a_3)(a_3 - a_2)^{-1}|r \in [0; 1]) \Leftrightarrow \\ &\Leftrightarrow (\underline{x}(r) = (a_2 - a_1)r + a_1; \\ \bar{x}(r) &= (a_3 - a_2)r + a_3|r \in [0; 1]). \end{aligned}$$

Полагают, что если $a_1 \geq 0$, то $x_n \geq 0$; если $a_3 \leq 0$, то $x_n \leq 0$.

Нечеткая функция (отображение) четких переменных $y_n(x)$. Пусть E — множество всех нечетких переменных с заданной функцией принадлежности $r(x)$, $r \in [0; 1]$, $x \in R$. Тогда $y_n(x): R \rightarrow E$ определяет нечетко-значимую функцию. В параметрической форме имеет место представление

$$\begin{aligned} y_n(x) = y(x, r) = \\ = (\underline{y}(x, r), \bar{y}(x, r)|r \in [0; 1]). \end{aligned}$$



Рис. 1. Функции принадлежности нечетких треугольных переменных

Арифметические операции $(+, -, \times, :)$ трактуются как специальный тип четкого отображения f над нечеткими множествами A_1, \dots, A_n с многомерной функцией принадлежности $r_A(x_1, \dots, x_n)$, именуемый принципом расширения Заде:

$$f: A = A_1 * \dots * A_n \xrightarrow{r_A(x_1, \dots, x_n)} B \Leftrightarrow \\ \Leftrightarrow r_B(y) = f(r_A(x_1, \dots, x_n)) = \\ = \sup_{y=f(\cdot)} r_A(x_1, \dots, x_n) \sup_{y=f(\cdot)} \min(r_{A_1}(x_1), \dots, r_{A_n}(x_n)).$$

В частности, для двух нечетких треугольных переменных x_{H1}, x_{H2} имеем

$$x_H = x_{H1} * x_{H2} = (a_{11}|a_{12}|a_{13}) * (a_{21}|a_{22}|a_{23}) = \\ = (a_{11} * a_{21}|a_{12} * a_{22}|a_{13} * a_{23}),$$

$$x_H = x_{H1} * x_{H2} \Leftrightarrow$$

$$\Leftrightarrow r_B(y) = \max_{x_H = x_{H1} * x_{H2}} \min(r_{A_1}(x_1), r_{A_2}(x_2)) \Leftrightarrow \\ \Leftrightarrow r_B(y) = \begin{cases} \sum_i r_{A_1}(x_{1i}) | r_B(y_i) = r_{A_1}(x_{1i}) * r_{A_2}(x_{2i}), \\ A_1, A_2 - \text{дискретные;} \\ \int r_{A_1}(x_1) | r_B(y) = r_{A_1}(x_1) * r_{A_2}(x_2), \\ A_1, A_2 - \text{непрерывные,} \end{cases}$$

где Σ, \int — символы представления нечеткой переменной $r_B(y)$ в виде совокупности пар $\{r_{A_1}(x_1) | r_B(y)\}$.

Отношение порядка (операции сравнения \geq, \leq) в общем случае следует из следующего определения. Пусть имеем нечеткие переменные x_{H1}, x_{H2} . Тогда

$$x_{H1} \geq x_{H2},$$

$$\text{если } \int_0^1 r[\underline{x}_1(r) + \bar{x}_1(r)] dr \geq \int_0^1 r[\underline{x}_2(r) + \bar{x}_2(r)] dr,$$

где $\underline{x}_i(r), \bar{x}_i(r)$ — элементы параметрического представления нечетких переменных $x_{Hi}, i = 1, 2$.

Нечеткий функционал J_H определяется как отображение множества нечетких функций y_H в R . $J_H: y_H \rightarrow r \in [0; 1] \subset R$, т. е. нечеткой функции y_H соответствует нечеткая треугольная переменная. Совокупность $\{y_H\}$, на которой определен нечеткий функционал J_H , составляет нечеткую область определения. Нечеткость J_H обусловлена наличием нечетких параметров, которые характеризуют неточность в их задании.

Нечеткая числовая матрица $A_H = \{a_{Hij}\}, i, j = \bar{1}, n$, с треугольными переменными — это прямоугольная (квадратная) таблица, содержащая нечеткие элементы в виде нечетких переменных с треугольными функциями принадлежности

$r(a_{ij}) = (a_{1ij}|a_{2ij}|a_{3ij}), r \in [0; 1] \subset R$. Элементы $a_{1ij}, a_{2ij}, a_{3ij}$ определяют свойства симметричности и неотрицательности A_H , подобные свойствам традиционных (четких) матриц. Элемент матрицы $0_{-1}^{+1} = (-1 | 0 | +1)$ обозначает нечеткий нуль, для которого $\int_0^1 r[\underline{x}(r) + \bar{x}(r)] dr = 0$.

Поэтому все элементы матрицы нечетко неотрицательны, т. е. $a_{Hij} \geq 0$, что означает неотрицательность нечеткой матрицы.

Банахово пространство нечетких переменных вводится в соответствии с подходом, принятым в функциональном анализе. Для этого в совокупности $\{x_{Hi}\} = E$ задаются операции:

а) сложения x_{Hi} и x_{Hj} в виде

$$x_{Hi} + x_{Hj} = (\underline{x}_i(r) + \underline{x}_j(r), \bar{x}_i(r) + \bar{x}_j(r)) | r \in [0; 1];$$

б) умножения x_H на скаляр $k \in R$ по правилу

$$kx_{Hi} = \begin{cases} k \underline{x}_i(r), k \bar{x}_i(r) | r \in [0; 1], k \geq 0; \\ k \bar{x}_i(r), k \underline{x}_i(r) | r \in [0; 1], k < 0; \end{cases}$$

в) существования у x_{Hi} противоположного элемента x_{Hk} такого, что

$$x_{Hi} + x_{Hk} \equiv 0 \Leftrightarrow r_k(x) = r_i(-x).$$

Относительно операций сложения и умножения выполняются аксиомы: коммутативность и ассоциативность для операции сложения, дистрибутивность для операции умножения. Поэтому совокупность $\{x_{Hi}\}$ с операциями сложения и умножения, с существованием противоположного элемента образует векторное (линейное) пространство E .

В пространстве E определим метрику

$$S(x_{Hi}, x_{Hj}) = \\ = \sup_{r \in [0; 1]} \{ \max[|\underline{x}_i(r) - \bar{x}_j(r)|, |\bar{x}_i(r) - \underline{x}_j(r)|] \}$$

и норму

$$\|x_{Hi} - x_{Hj}\| = S(x_{Hi}, x_{Hj}).$$

Далее определяется нечеткая последовательность Коши

$$\{x_{Hn}\}: S(x_{Hn}, x_{Hm})_{n,m \rightarrow \infty} \rightarrow 0 \\ \text{и полнота } E: x_{Hn} \text{ } n \rightarrow \infty \rightarrow x_H, x_H \in E.$$

Это приводит к банахову пространству нечетких переменных (E, S) .

Нечеткая производная функции по ее четкому аргументу согласно общему подходу находится путем определения для некоторой нечеткой функции, определенной ранее, операций:

вычитания или существования противоположного элемента, умножения на константу, предельного перехода относительно заданной метрики. В статье используются следующие типы нечетких производных: $y'_H{}^S(x)$ — Сейккалы (Seikkala — S) и $y'_H{}^{BF}(x)$ — Баклей-Фейринга (Buckley-Feuring — BF). Имеет место утверждение: если нечеткие производные существуют при $x = x_*$ и непрерывны в этой точке, то обе нечеткие производные при $x = x_*$ равны между собой [20].

Нечеткий интеграл. Пусть имеется нечеткое отображение $f_H: [a, b] \subset R \rightarrow E$, где E — нечеткое множество. Для каждого разбиения $P = \{t_0, \dots, t_n\} \in [a, b]$ и $\forall \xi_i \in [t_{i-1}, t_i]$, $i = \overline{1, n}$, допускается $R_P = \sum_{i=1}^n f_H(\xi_i)(t_i - t_{i-1})$, $\Delta = \max\{|t_i - t_{i-1}|, i = \overline{1, n}\}$, тогда определяется нечеткий интеграл для f_H на отрезке $[a, b]$:

$$\int_a^b f_H(t)dt = \lim_{\Delta \rightarrow \infty} R_P,$$

где предел определяется в метрике S Хаусдорфа.

Если нечеткая функция $f_H(t) = f(t, r) = (\underline{f}(t, r), \overline{f}(t, r) | r \in [0; 1])$ непрерывна в метрике Хаусдорфа и существует предел $\lim_{\Delta \rightarrow \infty} R_P$, то

$$\int_a^b \underline{f}(t, r)dt = \int_a^b \underline{f}(t, r)dt; \int_a^b \overline{f}(t, r)dt = \int_a^b \overline{f}(t, r)dt;$$

$r \in [0; 1],$

где $\int_a^b \underline{f}(t, r)dt$ и $\int_a^b \overline{f}(t, r)dt$ — нижний и верхний нечеткие интегралы, а $\underline{f}(t, r)$ и $\overline{f}(t, r)$ — нижняя и верхняя подинтегральные нечеткие функции.

Нечеткая начальная задача (задача Коши) рассматривается в статье для нечетких производных $y'_H{}^S(x)$ и $y'_H{}^{BF}(x)$ [13, 24]. Пусть имеем нечеткую начальную задачу, описываемую нелинейным уравнением первого порядка

$$y'_H(x) = f(x, y_H(x), \mathbf{k}_H), y_H(0) = c_H, \quad (1)$$

где $y'_H(x)$ — некоторая нечеткая производная из перечисленных ранее, константа c_H и вектор параметров \mathbf{k}_H являются неточно заданными, т. е. неопределенными. Представим эту неопределенность посредством нечетких треугольных переменных $c_H = (c_1|c_2|c_3)$, $k_{Hi} = (k_{1i}|k_{2i}|k_{3i})$ или с использованием обратных отображений в параметрической форме $k_{Hi} = (\underline{k}_i(r), \overline{k}_i(r) | r \in [0; 1])$, $c_H = (\underline{c}(r), \overline{c}(r) | r \in [0; 1])$. Необходимо получить

решение (1) $y_H = \beta(x, \mathbf{k}_H, c_H)$, которое для любого x является нечеткой переменной.

Соответствующая (1) четкая начальная задача имеет вид

$$y'(x) = f(x, y(x), \mathbf{k}), y(0) = c, \quad (2)$$

где $c = \text{const}$, $\mathbf{k} = (k_1, \dots, k_n)$ — вектор четких параметров. Если для задачи (2) выполнены все условия существования и единственности, то $y = \alpha(x, \mathbf{k}, c)$ есть решение уравнения (2).

Типы нечетких решений определяются типом производной. В соответствии с работами [13, 24] для нечеткой начальной задачи (1) имеет место два типа решений: $y_H{}^S(x)$ — решение Сейккалы (S) и $y_H{}^{BF}(x)$ — решение Buckley-Feuring (BF). Последнее существует при одновременном выполнении следующих условий:

$$f'_y > 0, \alpha'_c > 0, f'_{k_i} \alpha'_{k_i} > 0, i = \overline{1, n}, \quad (3)$$

где $f(\cdot)$ — правая часть, $\alpha(\cdot)$ — решение уравнения (2) соответственно; k_i — компоненты вектора \mathbf{k} .

Для уравнения (1) определяется существование BF -решения путем проверки условий (3), которые эквивалентны условиям одновременного возрастания (убывания) функций $f(\cdot)$ и $\alpha(\cdot)$ относительно параметров. Если условия (3) выполняются, то BF -решение существует и имеет вид

$$y_H{}^{BF}(x) = \left(\min_{r \in [0; 1]} \alpha(x, \mathbf{k}_H(r), c_H(r)), \max_{r \in [0; 1]} \alpha(x, \mathbf{k}_H(r), c_H(r)) \right). \quad (4)$$

Если хотя бы одно из условий (3) не выполняется, то BF -решение не существует и далее ищется S -решение $y_H{}^S(x)$.

Если четкое уравнение (2) имеет решение, то всегда существует решение $y_H{}^S(x)$, и оно находится из следующей системы уравнений:

$$\begin{cases} \underline{y}'^S(x) = f(x, \underline{y}(x, r), \overline{y}(x, r), \underline{k}), \underline{y}^S(0) = \underline{c}, \\ r \in [0; 1]; \\ \overline{y}'^S(x) = f(x, \underline{y}(x, r), \overline{y}(x, r), \overline{k}), \overline{y}^S(0) = \overline{c}, \\ r \in [0; 1], \end{cases} \quad (5)$$

а $y_H{}^S(x) = y(x, r) = (\underline{y}(x, r), \overline{y}(x, r) | r \in [0; 1])$ — нечеткая функция, в параметрическом представлении имеющая нижнюю $\underline{y}(x, r)$ и верхнюю $\overline{y}(x, r)$ ветви.

Если же решения $y_H{}^S(x)$ не существует, то не существует и решения (1).

Между нечеткими решениями имеется следующая взаимосвязь [24]:

$$\left. \begin{array}{l} \exists S \Rightarrow \exists BF \\ \exists BF \Rightarrow \exists S \end{array} \right\} \Rightarrow \exists BF \not\Rightarrow \exists S,$$

т. е. если существует $y_n^S(x)$, то существует решение $y_n^{BF}(x) = y_n^S(x)$. Если хотя бы одно из условий (3) не выполняется, то $y_n^{BF}(x) \neq y_n^S(x)$ и $y_n^{BF}(x)$ не существует.

2. Постановка задачи

Требуется определить при использовании метода динамического программирования, критерия обобщенной работы и принципа максимума типы S или BF оптимальных управлений для динамической модели ОУ с нечеткими динамическими параметрами, нечеткими начальными условиями и заданным качеством управления в виде нечеткого функционала.

2.1. Применительно к синтезу нечеткого оптимального управления методом динамического программирования имеем:

нечеткую линейную динамическую модель ОУ с нечеткими начальными условиями

$$\dot{\mathbf{x}}_n(t) = A_n(t)\mathbf{x}_n(t) + B_n(t)\mathbf{u}_n(t), \mathbf{x}_n(t_0) = \mathbf{x}_{n0}, \quad (6)$$

где $\mathbf{x}_n = (x_{n1}, \dots, x_{nn})^T \in E^n$ — нечеткий вектор состояния; $\mathbf{u}_n = (u_{n1}, \dots, u_{nm})^T \in E^m$ — нечеткий вектор управления; E^n, E^m — нечеткие множества размерности n и m соответственно, а нечеткая производная каждой компоненты вектора \mathbf{x}_n понимается в смысле Сейккала, для которой индекс "S" опущен для упрощения записи,

и нечетким квадратичным функционалом с заданными моментами времени начала t_0 и окончания t_k процесса соответственно

$$J_n = 0,5\mathbf{x}_n^T(t_k)G_n\mathbf{x}_n(t_k) + 0,5\int_{t_0}^{t_k} (\mathbf{x}_n^T(t)Q_n(t)\mathbf{x}_n(t) + \mathbf{u}_n^T(t)R_n(t)\mathbf{u}_n(t))dt, \quad (7)$$

где $A_n(t), B_n(t), G_n, Q_n(t)$ и $R_n(t)$ — матрицы согласованных размеров с нечеткими элементами и треугольными функциями принадлежности; симметрические матрицы G_n и $Q_n(t)$ неотрицательно определены; симметрическая матрица $R_n(t)$ положительно определена; $\mathbf{x}_n(t_k)$ — нечеткий свободный правый конец задачи.

В условиях (6) необходимо найти нечеткое оптимальное управление $\mathbf{u}_n^*(t, \mathbf{x}_n)$, которое минимизирует функционал (7). В четком (традиционном) случае (индекс "н" отсутствует)

это управление принято называть управлением с полной обратной связью.

Здесь и далее полагается, что нечеткая задача (6), (7) реализуется по методике, когда первоначально решается задача четкой оптимизации, т. е. индекс "н" в соотношениях (6), (7) исключается, а далее, после получения соответствующего обыкновенного дифференциального уравнения, полагается, что параметры этого уравнения являются нечеткими, появляется индекс "н" и решается соответствующая нечеткая задача [25, 26].

2.2. Применительно к синтезу нечеткого оптимального управления по критерию обобщенной работы функционал задается в виде нечеткой обобщенной работы

$$J_n = 0,5\mathbf{x}_n^T(t_k)G_n\mathbf{x}_n(t_k) + 0,5\int_{t_0}^{t_k} (\mathbf{x}_n^T(t)Q_n(t)\mathbf{x}_n(t) + \mathbf{u}_n^T(t)R_n(t)\mathbf{u}_n(t) + \Psi_n(t, \mathbf{x}_n(t)))dt, \quad (8)$$

где $\Psi_n(t, \mathbf{x}_n(t))$ — нечеткая функция, доопределяющая функционал (8).

По аналогии с работами [27, 28] сформулированные выше проблемы можно называть задачами аналитического конструирования нечетких оптимальных регуляторов.

2.3. Применительно к синтезу нечеткого оптимального управления на основании принципа максимума задаются нечеткая линейная модель ОУ с нечеткими начальными условиями (6) и с нечетким квадратичным функционалом (7) с заданными моментами времени начала t_0 и окончания t_k процесса соответственно, но с нечетким правым концом:

$$\mathbf{x}_n(t_k) = \mathbf{x}_{nk}. \quad (9)$$

В этих условиях, используя принцип максимума, необходимо найти нечеткое оптимальное управление, минимизирующее функционал (7), путем преобразования соответствующей двухточечной краевой задачи в задачу Коши [8].

3. Методика решения нечетких оптимизационных задач

Для сформулированных выше нечетких оптимизационных задач по пп. 2.1—2.2 используется прием, принятый в теории решения нечеткой начальной задачи [13, 24, 29, 30].

Методика решения нечеткой оптимизационной задачи управления заключается в следующем.

1. Составляется четкая оптимизационная задача соответствующей нечеткой задачи оптимизации управления, и находится ее решение.

2. Для полученной при решении четкой оптимизационной задачи четкой начальной задачи выполняется процедура фаззификации посредством представления нечеткими треугольными переменными ее параметров и решения.

3. Проверяются условия, определяющие существование *S*- или *BF*-типов решений нечеткой начальной задачи. Типы нечетких решений определяются типом производной. Если условия выполняются, то *BF*-решение существует.

4. Если хотя бы одно из условий не выполняется, то *BF*-решение не существует, и ищется *S*-решение, которое существует всегда, если существует решение четкой начальной задачи.

5. Вычисляются типы *S* или *BF* оптимальных управлений исходной нечеткой оптимизационной задачи.

4. Метод нечеткого динамического программирования

В соответствии с предложенной методикой получим нечеткие решения задачи 2.1, которая является нечеткой задачей Больца. Для этого первоначально, исключив в соотношениях (6) и (7) нижний индекс "н", получим традиционные уравнения. Затем стандартным способом составляется для некоторой функции $W(t, \mathbf{x})$, имеющей непрерывные частные производные, уравнение Беллмана [4]:

$$-\frac{\partial W(t, \mathbf{x})}{\partial t} = \min_{\mathbf{u}} \left\{ 0, 5[\mathbf{x}^T(t)Q(t)\mathbf{x}(t) + \mathbf{u}^T(t)R(t)\mathbf{u}(t)] + \left(\frac{\partial W(t, \mathbf{x})}{\partial \mathbf{x}} \right)^T [A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t)] \right\}. \quad (10)$$

Известно, что если функция $W(t, \mathbf{x})$ существует и удовлетворяет уравнению Беллмана с граничным условием

$$W(t_k, \mathbf{x}_k) = 0, 5\mathbf{x}_k^T G \mathbf{x}_k(t_k), \quad (11)$$

а управление удовлетворяет условию

$$\mathbf{u}^*(t, \mathbf{x}) = \arg \min_{\mathbf{u}} \left\{ \left(\frac{\partial W(t, \mathbf{x})}{\partial \mathbf{x}} \right)^T (A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t)) + 0, 5[\mathbf{x}^T(t)Q(t)\mathbf{x}(t) + \mathbf{u}^T(t)R(t)\mathbf{u}(t)] \right\},$$

то $\mathbf{u}^*(t, \mathbf{x}) = -R^{-1}(t)B^T(t)P(t)\mathbf{x}(t)$ является оптимальным управлением для четкого функционала, соответствующего (7).

Процесс минимизации приводит к начальной задаче для стандартного дифференциального уравнения Риккати в матричной форме:

$$-\dot{P}(t) = P(t)A(t) + A^T(t)P(t) - P(t)B(t)R^{-1}(t)B^T(t)P(t) + Q(t), P(t_k) = G, \quad (12)$$

которое после операции фаззификации, заключающейся в замене четких элементов матриц P , A , B , R и Q на нечеткие, имеет вид

$$-\dot{P}_H = P_H(t)A_H(t) + A_H^T(t)P_H(t) - P_H(t)B_H(t)R_H^{-1}(t)B_H^T(t)P_H(t) + Q_H(t), P_H(t_k) = G_H, \quad (13)$$

а управление находится из выражения $\mathbf{u}^*(t, \mathbf{x}) = -R_H^{-1}(t)B_H^T(t)P_H(t)\mathbf{x}_H(t)$.

В результате решения нечеткого уравнения Риккати (13) могут быть получены *S*- и *BF*-типы решений: $P_H^S(t)$ или $P_H^{BF}(t)$. Это приводит к соответствующим типам нечеткого оптимального управления $\mathbf{u}_H^{*S}(t)$ или $\mathbf{u}_H^{*BF}(t)$:

$$\mathbf{u}_H^{*S}(t) = -R_H^{-1}(t)B_H^T(t)P_H^S(t)\mathbf{x}_H(t);$$

$$\mathbf{u}_H^{*BF}(t) = -R_H^{-1}(t)B_H^T(t)P_H^{BF}(t)\mathbf{x}_H(t),$$

которые справедливы при любом начальном $\mathbf{x}(t_0) = \mathbf{x}_{H0}$.

Пример 1. Синтезировать нечеткий оптимальный регулятор для электродвигателя постоянного тока методом динамического программирования.

Математическая модель ОУ в виде электродвигателя постоянного тока без учета действия противоЭДС получается следующим образом. Нелинейная механическая характеристика $m = m(x)$ в координатах (m, x) аппроксимируется линейной зависимостью

$$m = ax + b, x > 0,$$

где m — момент на валу двигателя; x — угловая скорость вращения вала двигателя; a , b — параметры электродвигателя (рис. 2).

Обозначим $m_0 = k_0 u_0$ — пусковой момент электродвигателя; u_0 — пусковое напряжение на обмотках электродвигателя (вход); k_0 — коэффициент пропорциональности; x_0 — угловая скорость холостого хода.

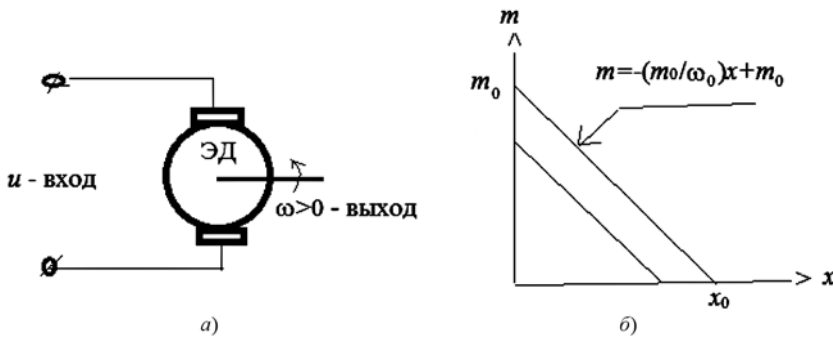


Рис. 2. Схема электродвигателя (а) и его механическая характеристика (б)

С учетом того, что угловое ускорение вала подчиняется уравнению

$$f\ddot{x} = m,$$

где f — момент инерции вала относительно его оси вращения, получим модель в виде

$$\dot{x} = m/f = m_0/f - m_0x/(fx_0) = -ax + bu_0,$$

в котором параметры $a = m_0/(fx_0)$, $b = k_0/f$.

Управление скоростью вращения вала электродвигателя осуществляется путем варьирования величины u_0 , поэтому после переобозначения $u_0 = u$, полагая, что параметры a , b являются нечеткими треугольными переменными a_n , b_n и начальное условие также является нечетким $x(t_0) = x_{n0}$, получим нечеткую модель электродвигателя:

$$\dot{x}_n = -a_n x_n + b_n u_n, \quad x_n(t_0) = x_{n0}, \quad (14)$$

где

$$a_n = (a_1|a_2|a_3), \quad a_1 > 0, \quad b_n = (b_1|b_2|b_3), \quad b_1 > 0, \\ x_{n0} = (x_{01}|x_{02}|x_{03}), \quad x_{01} > 0.$$

Нечеткость параметров электродвигателя обусловлена значительными колебаниями температуры в обмотках электропривода при его длительной непрерывной эксплуатации. Это приводит к значительным изменениям электро-механической постоянной времени электродвигателя и, значит, к изменениям параметров a и b . Одной из моделей варьирования этих параметров могут служить нечеткие треугольные переменные, что приводит к нечеткой модели (14).

Для упрощения дальнейших вычислений функционал (7) зададим в следующем виде:

$$J = 0,5g_n x_n^2(1) + 0,5 \int_0^1 u_n^2(t) dt. \quad (15)$$

В условиях (14) необходимо найти нечеткое оптимальное управление $u_n^*(t)$, которое

минимизирует функционал (15) и существует при любом начальном $x_n(t_0) = x_{n0}$.

Для этого первоначально найдем традиционное решение задачи (14), (15). Решая уравнение Риккати:

$$\dot{P}(t) = -2aP(t) + b^2P^2(t), \quad P(1) = g, \quad (16)$$

являющееся частным случаем уравнения (12), получим

$$P(t) = \frac{2a}{b^2 - C_k \exp(2at)}, \quad (17)$$

где $C_k = (b^2P(t_k) - 2a) \exp(-2at_k)/P(t_k)$ — постоянная интегрирования, определяемая при $t_k = 1$ из конечных условий.

При значениях $b = 1$, $a = 1/T$, где T — электро-механическая постоянная времени электродвигателя, уравнения (16) и (17) примут вид

$$\dot{P}(t) = \beta(t, P(t), T) = -2P(t)/T + P^2(t); \\ P(t) = \alpha(t, T, g) = \frac{2}{T - (T - 2/g) \exp(2(t-1)/T)},$$

так как $C_k = (1 - 2/Tg) \exp(-2/T)$.

Соответствующее (16) нечеткое уравнение Риккати:

$$\dot{P}_n(t) = -2a_n P_n(t) + b_n^2 P_n^2(t), \quad P_n(1) = g_n, \quad (18)$$

с параметрами $b_n = 1_n$, $a_n = 1/T_n$, где T_n — нечеткая треугольная переменная для электро-механической постоянной времени электродвигателя; 1_n — нечеткая единица, определяющая нечеткий параметр с функцией принадлежности синглтон (singleton).

Для определения типа решения нечеткой начальной задачи (18) необходимо определить знаки производных β'_P и α'_g . Вычисления дают:

$$\beta'_P = \frac{\partial \beta}{\partial P} = \frac{\partial(-2P(t)/T + P^2(t))}{\partial P} = \\ = \begin{cases} -2/T + 2P(t) > 0, & \text{где } P(t) > 1/T; \\ -2/T + 2P(t) \leq 0, & \text{где } P(t) \leq 1/T; \end{cases} \\ \alpha'_g = \frac{\partial \alpha}{\partial g} = \frac{\partial}{\partial g} \left[\frac{2}{T - (T - 2/g) \exp(2(t-1)/T)} \right] = \\ = \frac{-4 \exp(2(t-1)/T)/g^2}{[T - (T - 2/g) \exp(2(t-1)/T)]^2} < 0.$$

Здесь $P(t)$ определена выше.

Возможны следующие два варианта существования нечетких решений для (18):

- вариант 1, когда $P(t) > 1/T$, тогда $\beta'_p > 0$, $\alpha'_g < 0$ и не существует BF -решение для (18), но существует его S -решение. Из уравнения (18) с нечеткими параметрами $a_n = 1/T_n$, $b_n = 1_n$ с функцией принадлежности синглтон:

$$\dot{P}_n(t) = -2P_n(t)/T_n + P_n^2(t), P_n(1) = g_n$$

или

$$\begin{bmatrix} \dot{\underline{P}} \\ \dot{\bar{P}} \end{bmatrix} = \begin{bmatrix} -2/T & 0 \\ 0 & -2/\bar{T} \end{bmatrix} \begin{bmatrix} \underline{P} \\ \bar{P} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \underline{P}^2 \\ \bar{P}^2 \end{bmatrix}, \quad (19)$$

$$\underline{P}(1) = \underline{g}, \bar{P}(1) = \bar{g},$$

где $P_n(t) = (\underline{P}(r,t)|P(t)|\bar{P}(r,t))$, $g_n = (\underline{g}(r)|g|\bar{g}(r))$, $T_n = (\underline{T}(r,t)|T(t)|\bar{T}(r,t))$, $r \in [0;1]$ — нечеткие треугольные переменные, получим $\underline{P}(r, t)$, $\bar{P}(r, t)$ и, следовательно, S -решение:

$$P_n^S(t) = (\underline{P}(r,t), \bar{P}(r,t)|r \in [0;1]),$$

а нечеткое оптимальное управление S -типа будет равно

$$\begin{aligned} u_n^S(t, x) &= -R_n^{-1}(t)B_n^T(t)P_n^S(t)x(t)|_{R_n=1_n, B_n=1_n} = \\ &= (\underline{u}(r,t) = \min_{r \in [0;1]} [-P(r,t)x(t)], \bar{u}(r,t) = \\ &= \max_{r \in [0;1]} [-P(r,t)x(t)]). \end{aligned}$$

На рис. 3, а (см. вторую сторону обложки) показаны оптимальные траектории и управления для нечеткого регулятора S -типа.

Нечеткий функционал (15) в этом случае равен

$$J_n^S(\mathbf{x}, t_k) = (0,0326; 0,0543; 0,0719),$$

т. е. значения нечеткого критерия оптимальности лежат в пределах от 0,0326 до 0,0719, что практически совпадает с критерием оптимальности, равным 0,0486, вычисленным для аналогичного примера из работы [28];

- вариант 2, когда $P(t) \leq 1/T$, тогда $\beta'_p < 0$, $\alpha'_g < 0$, и существует BF -решение:

$$P_n^{BF}(t) = \left[\min_{r \in [0;1]} P(r,t), \max_{r \in [0;1]} P(r,t) \right]$$

и нечеткое оптимальное управление BF -типа будет равно

$$\begin{aligned} u_n^{BF}(t, x) &= -R_n^{-1}(t)B_n^T(t)P_n^{BF}(t)x(t)|_{R_n=1_n, B_n=1_n} = \\ &= (\underline{u}(r,t) = \min_{r \in [0;1]} [-P(r,t)x(t)], \bar{u}(r,t) = \\ &= \max_{r \in [0;1]} [-P(r,t)x(t)]). \end{aligned} \quad (20)$$

На рис. 3, б (см. вторую сторону обложки) показаны оптимальные поверхности $\min_{r \in [0;1]} P(r,t)$ и $\max_{r \in [0;1]} P(r,t)$. Оптимальные траектории и управления для нечеткого регулятора BF -типа с большой точностью совпадают с аналогичными траекториями на рис. 3, а.

Нечеткий функционал (15) в этом случае:

$$J_n^{BF}(\mathbf{x}, t_k) = (0,0318; 0,0534; 0,0708).$$

Момент времени переключения t_n вариантов нечетких решений находится из условия $P(t_n) = 1/T$, откуда после подстановки функции $P(t)$ из (17) и решения соответствующего уравнения относительно t_n будем иметь:

$$t_n = 0,5T \ln |(2 - b^2)/(b^2 - 2 \exp(-2/T)/(TP(t_k)))|, \\ P(t_k) \neq 0.$$

Таким образом, рассмотрена и решена задача синтеза нечеткого регулятора методом динамического программирования в виде BF - и S -типов.

5. Нечеткий метод обобщенной работы

При решении задачи синтеза нечеткого оптимального регулятора для классических форм задания функционалов появляется проблема практической реализации алгоритма оптимизации [7].

А. А. Красовским предложен подход к проблеме синтеза оптимальных регуляторов с использованием функционала обобщенной работы, который облегчает решение задачи.

В соответствии с критерием обобщенной работы четкий функционал, соответствующий нечеткому функционалу (8), задается в виде

$$\begin{aligned} J &= 0,5\mathbf{x}^T(t_k)G\mathbf{x}(t_k) + 0,5 \int_{t_0}^{t_k} (\mathbf{x}^T(t)Q(t)\mathbf{x}(t) + \\ &+ \mathbf{u}^T(t)R(t)\mathbf{u}(t) + \Psi(t, \mathbf{x}(t)))dt, \end{aligned} \quad (21)$$

в котором функция $\Psi(t, \mathbf{x}(t))$ находится из соотношения

$$\Psi(t, \mathbf{x}(t)) = \left(\frac{\partial z(t, \mathbf{x})}{\partial \mathbf{x}} \right)^T B(t)R^{-1}(t)B^T(t) \frac{\partial z(t, \mathbf{x})}{\partial \mathbf{x}}, \quad (22)$$

где $z(t, \mathbf{x})$ определяется из решения линейного уравнения в частных производных

$$\frac{\partial z(t, \mathbf{x})}{\partial t} + \frac{\partial z(t, \mathbf{x})}{\partial \mathbf{x}} A(t)\mathbf{x}(t) - 0,5\mathbf{x}^T(t)Q(t)\mathbf{x}(t) = 0, \quad (23)$$

$$z(t_k, \mathbf{x}) = -0,5\mathbf{x}^T G\mathbf{x}.$$

Оптимальное управление, найденное исходя из критерия обобщенной работы, имеет вид

$$\mathbf{u}^*(t, \mathbf{x}) = R^{-1}(t)B^T(t) \frac{\partial z(t, \mathbf{x})}{\partial \mathbf{x}}. \quad (24)$$

Решение для уравнения (23) ищется методом разделения переменных в следующем виде:

$$z(t, \mathbf{x}) = 0,5\mathbf{x}^T K(t)\mathbf{x}, \quad (25)$$

где $K(t)$ — неизвестная симметричная матрица размером $n \times n$. Если $z(\mathbf{x}, t)$ подставить в (23) и использовать правило

$$\mathbf{x}^T A\mathbf{x} \equiv 0 \Leftrightarrow A + A^T = 0,$$

то получим уравнение для нахождения $K(t)$:

$$\dot{K}(t) = -A^T(t)K(t) - K(t)A(t) + Q(t) \quad \text{при } K(t_k) = -G. \quad (26)$$

Это уравнение, в отличие от уравнения Риккати, является линейным. С учетом соотношений (25), (26) получим из (24) оптимальный регулятор

$$\begin{aligned} \mathbf{u}^*(t, \mathbf{x}) &= R^{-1}(t)B^T(t) \frac{\partial z(t, \mathbf{x})}{\partial \mathbf{x}} = \\ &= R^{-1}(t)B^T(t)K(t)\mathbf{x}. \end{aligned} \quad (27)$$

После операции фаззификации уравнений (26) и (27), заключающейся в замене четких элементов матриц K , A , B , R и Q на нечеткие, получим нечеткую начальную задачу для нахождения $K_H(t)$:

$$\begin{aligned} \dot{K}_H(t) &= -A_H^T(t)K_H(t) - K_H(t)A_H(t) + Q_H(t) \\ \text{при } K_H(t_k) &= -G_H \end{aligned} \quad (28)$$

и уравнение для нахождения нечеткого оптимального управления

$$\mathbf{u}_H^*(t, \mathbf{x}_H) = R_H^{-1}(t)B_H^T(t)K_H(t)\mathbf{x}_H. \quad (29)$$

В результате решения нечеткого линейного уравнения (28) могут быть получены S - и BF -типы решений: $K_H^S(t)$ или $K_H^{BF}(t)$, что приведет к соответствующим типам нечеткого оптимального управления $\mathbf{u}_H^{*S}(t, \mathbf{x}_H)$ или $\mathbf{u}_H^{*BF}(t, \mathbf{x}_H)$ в уравнении (29).

Пример 2. Необходимо синтезировать нечеткий оптимальный регулятор для электродвигателя постоянного тока по критерию обобщенной работы.

Объект управления описывается нечетким уравнением (14), а нечеткий критерий — выражением (15)

Доопределим функционал (15) нечеткой функцией $\Psi_H(t, x(t))$:

$$J_H = 0,5g_H x^2(1) + 0,5 \int_0^1 [u_H^2(t) + \Psi_H(t, x(t))] dt. \quad (30)$$

Нечеткая оптимизационная задача (30) в соответствии с методикой ее решения по критерию обобщенной работы дает из (28) следующее уравнение:

$$\begin{aligned} \dot{K}_H(t) &= 2a_H K_H(t) \\ \text{при } K_H(1) &= C_H = -1_H, C_H < 0. \end{aligned} \quad (31)$$

Для нечеткой задачи Коши (31) имеем следующие показатели типов решений:

1. $\dot{K} = \alpha(K, a, t) = 2aK \Rightarrow \frac{\partial \alpha}{\partial K} = 2a|_{a>0} > 0;$
1. $\frac{\partial \alpha}{\partial a} = 2K = 2Ce^{2at}|_{C<0} < 0;$
2. $K = \beta(K, C, t) = Ce^{2at} \Rightarrow \frac{\partial \beta}{\partial K} = e^{2at} > 0;$
- $\frac{\partial \beta}{\partial a} = 2tCe^{2at}|_{C<0} < 0;$
3. $\frac{\partial \alpha}{\partial K} \frac{\partial \beta}{\partial a} < 0.$

Эти показатели указывают на то, что для уравнения (31) при $C < 0$ не существует BF -решения, но существует S -решение. Для его нахождения положим:

$$\begin{aligned} a_H &= (a_1|a_2|a_3) = (\underline{a}(r), \bar{a}(r))|r \in [0;1], a_1 > 0, \\ C_H &= (C_1|C_2|C_3) = (\underline{C}(r), \bar{C}(r))|r \in [0;1], C_3 < 0. \end{aligned}$$

Так как $a_H > 0$, $C_H < 0$, то согласно свойствам арифметических операций в банаховом пространстве над нечеткими переменными [31] для $K_H(t)$ имеем

$$K_H(t) = \begin{cases} (a\underline{K}(r,t), a\bar{K}(r,t)) | r \in [0;1], a \geq 0; \\ (a\bar{K}(r,t), a\underline{K}(r,t)) | r \in [0;1], a < 0. \end{cases}$$

Это означает, что S -решение находится из уравнений:

$$\begin{aligned} \dot{K}_H^S(t) &= a_H K_H^S(t), \quad \dot{\underline{K}}(r,t) = \underline{a}(r)\underline{K}(r,t), \\ \underline{K}(r,t_k) &= \underline{C}(r); \\ K_H^S(1) &= C_H = -1_H, \quad \dot{\bar{K}}(r,t) = \bar{a}(r)\bar{K}(r,t), \\ \bar{K}(r,t_k) &= \bar{C}(r). \end{aligned} \quad (32)$$

Нечеткая форма $z_H(x_H, t)$ из выражения (25) соответственно равна

$$\begin{aligned} z_H^S(t, x_H) &= 0, 5K_H^S(t)x_H^2 = 0, 5(\underline{K}(r,t), \\ \bar{K}(r,t))x_H^2 &= 0, 5(\underline{K}(r,t)\underline{x}^2(r), \bar{K}(r,t)\bar{x}^2(r)) | r \in [0;1], \end{aligned}$$

а нечеткое оптимальное управление из соотношения (29) имеет вид

$$\begin{aligned} u_H^S(t, x_H) &= R_H^{-1}(t)B_H^T(t)K_H^S(t)x_H(t) \Big|_{\substack{R_H=1_H \\ B_H=b_H}} = \\ &= (\underline{b}(r), \bar{b}(r))(\underline{K}(r,t), \bar{K}(r,t))(\underline{x}(r), \bar{x}(r)) = \\ &= (\underline{b}(r)\underline{K}(r,t)\underline{x}(r), \bar{b}(r)\bar{x}(r)\bar{K}(r,t)) | r \in [0;1], \end{aligned}$$

где $\underline{K}(r, t)$ и $\bar{K}(r, t)$ определены из соотношений (32).

Оптимальные траектории и управления для нечеткого регулятора S -типа, синтезированного по критерию обобщенной работы, приведены на рис. 4 (см. вторую сторону обложки).

Нечеткий функционал (15) в этом случае:

$$J_H^S(x, t_k) = (0, 0304; 0, 0481; 0, 0651).$$

Таким образом, рассмотрена и решена задача синтеза нечеткого регулятора по методу обобщенной работы в виде S -типа.

Заключение

Рассмотрено решение задач оптимального управления в нечеткой постановке, когда являются начальными задачи (задачи Коши) для нечетких дифференциальных уравнений.

Сформулированы три типа нечетких оптимизационных задач управления. Первая из них — это задача синтеза оптимального нечеткого регулятора с полной обратной связью. Во второй задаче в качестве критерия опти-

мальности используется нечеткий функционал обобщенной работы. Третья задача формулируется в виде нечеткой оптимизационной задачи на принцип максимума и с нечетким правым концом.

Для первой нечеткой оптимизационной задачи, решаемой методом динамического программирования, получены BF - и S -типы нечеткого оптимального управления. Рассмотрен пример оптимального управления электродвигателем постоянного тока с нечеткими динамическими параметрами и сформулированы условия существования BF - и S -типов управлений.

Для второй оптимизационной задачи решается нечеткое дифференциальное уравнение в частных производных, которое приводит к нечеткой задаче Коши для линейного дифференциального уравнения. На примере управления электродвигателем с нечеткими параметрами показано отсутствие BF -оптимального управления, однако получен оптимальный регулятор S -типа с полной обратной связью.

Список литературы

1. Ванько В. И., Ермошина О. В., Кувыркин Г. Н. Вариационное исчисление и оптимальное управление. М.: Изд-во МГТУ им. Н. Э. Баумана, 2006. 487 с.
2. Эльсгольц Л. Э. Дифференциальные уравнения и вариационное исчисление. М.: Наука, 1969. 425 с.
3. Цлаф Л. Я. Вариационное исчисление и интегральные уравнения. М.: Наука, 1969. 192 с.
4. Сю Д., Мейер А. Современная теория автоматического управления и ее применения. М.: Машиностроение, 1972. 544 с.
5. Брайсон А., Хо Ю-Ши. Прикладная теория оптимального управления. М.: Мир, 1972. 544 с.
6. Иванов В. А., Медведев В. С. Математические основы теории оптимального и логического управления. М.: Изд-во МГТУ им. Н. Э. Баумана, 2011. 599 с.
7. Справочник по теории автоматического управления. Под ред. Красовского А. А. М.: Наука, 1987. 712 с.
8. Моисеев Н. Н. Элементы теории оптимальных систем. М.: Физматлит, 1975. 576 с.
9. Черноусько Ф. Л., Колмановский В. Б. Оптимальное управление при случайных возмущениях. М.: Физматлит, 1978. 352 с.
10. Флеминг У., Ршел Р. Оптимальное управление детерминированными и стохастическими системами. М.: Мир, 1978. 316 с.
11. Методы робастного, нейро-нечеткого и адаптивного управления. Под ред. К. А. Пупкова, Н. Д. Егупова. М.: Изд-во МГТУ им. Н. Э. Баумана, 2001. 743 с.
12. Buckley J. J., Feuring T. Introduction to fuzzy partial differential equations // Fuzzy sets and systems. 1999, no. 5, pp. 241–248.
13. Buckley J. J., Feuring T. Fuzzy differential equations // Fuzzy sets and systems. 2000. N. 11. P. 43–54.
14. Latif Ahmad, Muhammad Farooq, Salem Abdullah. Solving nth order fuzzy differential equation by fuzzy Laplace Transform // Open Access Library Journal Mathematics. 2014. ID on Arxiv:1403.0242.

15. Venhata Ramu G., Padmanabhan K. and Ananti S. Optimal control with fuzzy State Space Modeling using Riccati equation // International Journal of Information and Electronics Engineering. 2012. Vol. 2, N. 5. P. 800–805.
16. Friedman M., Ming M., Kandel A. Fuzzy linear systems // Fuzzy sets and systems. 1988. N. 96. P. 201–209.
17. Деменков Н. П., Мочалов И. А. Нечеткие сплайны // Вестник Московского государственного технического университета им. Н. Э. Баумана. Серия "Приборостроение". 2012. № 2(87). С. 48–59.
18. Асмолова Ю. Е., Мочалов И. А. Элементы нечеткого вариационного исчисления // Вестник Университета дружбы народов. Сер. Инженерные исследования. 2010. № 4. С. 37–44.
19. Мочалов И. А., Хрисат М. С. Оценивание параметров модели по нечетким случайным данным // Информационные технологии. 2014. № 2(210). С. 14–22.
20. Мочалов И. А. и др. Нечеткие вероятностно-статистические методы // Приложение к журналу "Информационные технологии". 2003. № 4. 48 с.
21. Hamideh Masoumi, Behrouz Fathi Vajargah. Study on behavior of fuzzy Marcov chains // Advances in computer science and its applications. 2012. Vol. 2, N. 2. P. 373–376.
22. Деменков Н. П., Мочалов И. А. Динамика нечеткой системы автоматической оптимизации // Вестник МГТУ им. Н. Э. Баумана. Сер. "Приборостроение". 2016. № 1. С. 59–74.
23. Деменков Н. П., Мочалов И. А. Нейросетевая оценка динамики системы автоматической оптимизации // Мехатроника, автоматизация, управление. 2015. Т. 16, № 10. С. 659–663.
24. Seikkala S. On the fuzzy initial value problem // Fuzzy sets and systems. 1987. Vol. 24, N. 3. P. 319–330.
25. Amir Sadeghi, Ahmad Izani Md. Ismail, Ali F. Jameel. Solving systems of fuzzy differential equation // International Mathematical Forum. 2011. Vol. 6, N. 42. P. 2087–2100.
26. Parandin N. Numerical solution of fuzzy differential equations of 2nd-order by Runge-Kutta method // Journal of mathematical extension. 2013. Vol. 7, N. 3. P. 47–62.
27. Афанасьев В. Н., Колмановский В. Б., Носов В. Р. Математическая теория конструирования систем управления. М.: Высшая школа, 2003. 615 с.
28. Пантелеев А. В., Бортаковский А. С. Теория управления в примерах и задачах. М.: Высшая школа, 2003. 585 с.
29. Мочалов И. А., Хрисат М. С., Шихаб Еддин М. Я. Нечеткие дифференциальные уравнения в задачах управления. Часть 1 // Информационные технологии. 2015. Т. 21, № 3. С. 171–178.
30. Мочалов И. А., Хрисат М. С., Шихаб Еддин М. Я. Нечеткие дифференциальные уравнения в задачах управления. Часть II // Информационные технологии. 2015. Т. 21, № 4. С. 243–250.
31. Saadi R., Vaezpour S. M. Some results on fuzzy Banach spaces // Journal of applied mathematics and computing. 2005. Vol. 17, N. 1. P. 475–484.

N. P. Demenkov, Ph. D., Associate Professor, dnp@bmstu.ru,
 E. A. Mikrin, D. Sc., Professor, evgeny.mikrin@bmstu.ru,
 I. A. Mochalov, D. Sc., Professor, intelsyst@mail.ru,
 Moscow, MSTU named after N. E. Bauman

Fuzzy Optimal Control of Linear Systems. Part 1. Positional Control

Single-point boundary tasks are considered in part 1 for the fuzzy nonlinear differential equations like Rikkati and for the fuzzy linear differential equations. The fuzzy nonlinear differential equations arise at the solution of problems of synthesis of optimal linear regulators by method of dynamic programming. The fuzzy linear differential equations appear at synthesis of regulators by criterion of the generalized work. Three types of fuzzy optimization control problems are formulated. The first of these is the problem of synthesizing an optimal fuzzy regulator with full feedback. In the second task, the fuzzy functional of the generalized work is used as the optimality criterion. The third problem is formulated as a fuzzy optimization problem on the maximum principle and with a fuzzy right end. For the first fuzzy optimization problem solved by the dynamic programming method, Buckley-Feuring and Seikkala types of fuzzy optimal control are obtained. An example of optimal control of a DC motor with fuzzy dynamic parameters is considered, and conditions for the existence of Buckley-Feuring and Seikkala types of controls are formulated. For the second optimization problem, a fuzzy partial differential equation is solved, which leads to a fuzzy Cauchy problem for a linear differential equation. By the example of motor control with fuzzy parameters, the absence of a Buckley-Feuring optimal control is shown, however, an optimal Seikkala type controller with full feedback is obtained.

Keywords: Fuzzy boundary value problems, synthesis of fuzzy optimal controllers, fuzzy differential equations, membership function, fuzzy initial problem, fuzzy systems of linear algebraic equations, maximum principle, dynamic programming, criterion of generalized work

DOI: 10.17587/it.25.259-270

References

1. Van'ko V. I., Ermoshina O. V., Kuvyrkin G. N. *Variacionnoe ischislenie i optimal'noe upravlenie* (Calculus of variations and optimal control), Moscow, Publishing house of MGTU im. N. E. Bauman, 2006, 487 p. (in Russian).
2. Ehl'sgol'c L. Eh. *Differencial'nye uravneniya i variacionnoe ischislenie* (Differential equations and calculus of variations), Moscow, Nauka, 1969, 425 p. (in Russian).
3. Claf L. Ya. *Variacionnoe ischislenie i integral'nye uravneniya* (Calculus of variations and integral equations), Moscow, Nauka, 1969, 192 p. (in Russian).

4. **Syu D., Mejer A.** *Sovremennaya teoriya avtomaticheskogo upravleniya i ee primeneniya* (Modern theory of automatic control and its application), Moscow, Mashinostroenie, 1972, 544 p. (in Russian).
5. **Brajson A., Ho Yu-Shi.** *Prikladnaya teoriya optimal'nogo upravleniya* (Applied theory of optimal control), Moscow, Mir, 1972, 544p. (in Russian).
6. **Ivanov V. A., Medvedev V. S.** *Matematicheskie osnovy teorii optimal'nogo i logicheskogo upravleniya* (Mathematical foundations of the theory of optimal and logical control), Moscow, Publishing house of MGTU im. N. Eh. Bauman, 2011, 599 p. (in Russian).
7. **Krasovskii A. A.** ed. *Spravochnik po teorii avtomaticheskogo upravleniya* (Handbook of automatic control theory), Moscow, Nauka, 1987, 712 p. (in Russian)
8. **Moiseev N. N.** *Ehlementy teorii optimal'nyh sistem* (Elements of the theory of optimal systems), Moscow, Fizmatlit, 1975, 576 p. (in Russian).
9. **Chernous'ko F. L., Kolmanovskij V. B.** *Optimal'noe upravlenie pri sluchajnyh vozmushcheniyah* (Optimal control for random disturbances), Moscow, Fizmatlit, 1978, 352 p. (in Russian).
10. **Fleming U., Rishel R.** *Optimal'noe upravlenie determinirovannymi i stohasticheskimi sistemami* (Optimal control of deterministic and stochastic systems), Moscow, Mir, 1978, 316 p. (in Russian).
11. **Pupkov K. A., Egupov N. D.** ed. *Metody robustnogo, nejro-nechetkogo i adaptivnogo upravleniya* (Robust, neuro-fuzzy and adaptive control methods). Moscow, Publishing house of MGTU after name N. Eh. Bauman, 2001, 743 p. (in Russian).
12. **Buckley J. J., Feuring T.** Introduction to fuzzy partial differential equations, *Fuzzy sets and systems*, 1999, no. 5, pp. 241–248.
13. **Buckley J. J., Feuring T.** Fuzzy differential equations, *Fuzzy Sets and Systems*, 2000, no. 11, pp. 43–54.
14. **Latif Ahmad, Muhammad Farooq, Salem Abdullah.** Solving nth order fuzzy differential equation by fuzzy Laplace Transform, *Open Access Library Journal Mathematics*, 2014, ID on Arxiv:1403.0242.
15. **Venhata Ramu G., Padmanabhan K., Ananti S.** Optimal control with fuzzy State Space Modeling using Riccati equation, *International Journal of Information and Electronics Engineering*, 2012, vol. 2, no. 5, pp. 800–805.
16. **Friedman M., Ming M., Kandel A.** Fuzzy linear systems, *Fuzzy Sets and Systems*, 1988, no. 96, pp. 201–209.
17. **Demenkov N. P., Mochalov I. A.** *Nechetkie splajny* (Fuzzy Splines), *Vestnik Moskovskogo gosudarstvennogo tekhnicheskogo universiteta im. N.EH.Baumana. Seriya "Priborostroenie"*, 2012, no. 2(87), pp. 48–59 (in Russian).
18. **Asmolova Yu. E., Mochalov I. A.** *Ehlementy nechetkogo variacionnogo ischisleniya* (Elements of fuzzy variational calculus), *Vestnik Universiteta Druzhyby narodov, s. Inzhenernye issledovaniya*, 2010, no. 4, pp. 37–44 (in Russian).
19. **Mochalov I. A., Hrisat M. S.** *Ocenivanie parametrov modeli po nechetkim sluchajnym dannym* (Elements of fuzzy variational calculus), *Informacionnye Tekhnologii*, 2014, no. 2(210), pp. 14–22 (in Russian).
20. **Mochalov I. A.** et al. *Nechetkie veroyatnostno-statisticheskie metody* (Fuzzy probabilistic statistical methods), *Prilozhenie, Informacionnye Tekhnologii*, 2003, no. 4, 48 p. (in Russian).
21. **Hamideh Masoumi, Behrouz FathiVajargah.** Study on behavior of fuzzy Marcov chains, *Advances in computer science and its applications*, 2012, vol. 2, no. 2, pp. 373–376.
22. **Demenkov N. P., Mochalov I. A.** *Dinamika nechetkoj sistemy avtomaticheskoy optimizacii* (The dynamics of a fuzzy automatic optimization system), *Vestnik MGTU im. N. Eh. Bauman. Ser. "Priborostroenie"*, 2016, no. 1, pp. 59–74 (in Russian).
23. **Demenkov N. P., Mochalov I. A.** *Nejrosetevaya oценка dinamiki sistemy avtomaticheskoy optimizacii* (Neural network assessment of the automatic optimization system), *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2015, vol. 16, no. 10, pp. 659–663 (in Russian).
24. **Seikkala S.** On the fuzzy initial value problem, *Fuzzy Sets and Systems*, 1987, vol.24, no.3, pp.319–330.
25. **Amir Sadeghi, Ahmad Izani Md. Ismail and Ali F. Jameel.** Solving systems of fuzzy differential equation, *International Mathematical Forum*, 2011, no. 42, vol. 6, pp. 2087–2100.
26. **Parandin N.** Numerical solution of fuzzy differential equations of 2nd-order by Runge-Kutta method, *Journal of Mathematical Extension*, 2013, vol. 7, no. 3, pp. 47–62.
27. **Afnas'ev V. N., Kolmanovskij V. B., Nosov V. R.** *Matematicheskaya teoriya konstruirovaniya sistem upravleniya* (Mathematical theory of designing control systems), Moscow, Vysshaya shkola, 2003, 615 p. (in Russian).
28. **Panteleev A. V., Bortakovskij A. S.** *Teoriya upravleniya v primerah i zadachah* (Theory of control in examples and problems), Moscow, Vysshaya shkola, 2003, 585 p. (in Russian).
29. **Mochalov I. A., Hrisat M. S., Shihab Eddin M. Ya.** *Nechetkie differencial'nye uravneniya v zadachah upravleniya. Chast' I* (Fuzzy differential equations in control problems. Part 1), *Informacionnye Tekhnologii*, 2015, no. 3, vol. 21, pp. 171–178 (in Russian).
30. **Mochalov I. A., Hrisat M. S., Shihab Eddin M. Ya.** *Nechetkie differencial'nye uravneniya v zadachah upravleniya. Chast' II* (Fuzzy differential equations in control problems. Part II), *Informacionnye Tekhnologii*, 2015, no. 4, vol. 21, pp. 243–250 (in Russian).
31. **Saadi R., Vaezpour S. M.** Some results on fuzzy Banach spaces, *Journal of applied mathematics and computing*, 2005, vol. 17, no. 1, pp. 475–484.

В. П. Кулагин, д-р техн. наук, проф., зав. кафедрой "Аппаратное, программное и математическое обеспечение вычислительных систем", e-mail: Kulagin_VP@mgupl.ru,
Е. С. Малых, магистрант кафедры "Аппаратное, программное и математическое обеспечение вычислительных систем", e-mail: kattya.miller.95@yandex.ru,
МИРЭА — Российский технологический университет, Москва

Проектирование матричных вычислительных структур с использованием сетей Петри

На примере проектирования матричных вычислительных структур предложен алгоритм синтеза моделирующих сетей Петри. Описан подход к синтезу новых структур на основе использования векторов проекции. Показано, что данный подход не позволяет получить все множество возможных сетей Петри (СП), моделирующих выполнение заданной на матричной структуре функции. Описан оригинальный алгоритм проектирования полного множества возможных СП-моделей матричных структур для заданной функции. Алгоритм основан на использовании структурного анализа и тензорной методологии. Результатом работы алгоритма является генерация программ синтеза структур СП-моделей. Показано, что среди генерируемых программ существуют такие, которые синтезируют эквивалентные СП-модели. Сформулированы свойства, выявляющие "эквивалентные" программы. В целях количественного и качественного анализа построенных матричных структур введены критерии оценки структурных и динамических свойств синтезируемых СП-моделей. Показано, что в силу экспоненциального роста числа синтезируемых СП-моделей от размерности исходной примитивной системы необходим поиск новых приемов и подходов, которые исключали бы из рассмотрения модели, не отвечающие наперед заданным требованиям.

Ключевые слова: матричные структуры, параллельные вычисления, сети Петри, тензорный анализ, программы синтеза, оценка структурных свойств

Введение

Актуальность построения параллельных структур, хорошо приспособленных для реализации вычислительно емких алгоритмов обработки сигналов и изображений, в настоящее время не вызывает сомнения. Подобные структуры обладают такими достоинствами, как модульность, регулярность, наличие локальных соединений и высокая степень конвейерной обработки. Использование для построения указанных структур систолических массивов не всегда приводит к успеху из-за недостатка систолических массивов, вызванного глобальной синхронизацией. Подобная синхронизация приводит к проблемам расфазировки и, в целом, к нарушению отказоустойчивости. Эффективным решением описанных проблем является использование параллельных структур, в которых процесс вычислений управляется потоком данных. Другими словами, в вычис-

лительном устройстве с управлением потоком данных команда начинает выполняться только в том случае, когда получены все активизирующие ее операнды [11]. Такой подход исключает необходимость в глобальном управлении и в глобальной синхронизации. Благодаря этому для матричной обработки можно использовать подход, основанный на управлении данными и автосинхронизации. Свойства сетей Петри как математической теории полностью отвечают требованиям математической модели, предназначенной для описания и дальнейшего исследования матричных структур [1, 2].

В последние годы наблюдается неослабевающий интерес к использованию аппарата сетей Петри (СП) [4–6] для решения задач, связанных с исследованием сложных систем. Можно отметить, что современные подходы, используемые при анализе и синтезе структур сложных систем, основаны на методах декомпозиции, координации и агрегации, структурном подхо-

де, подходе, основанном на теории сложности, и др. Возникающие противоречия между сложностью создаваемых современных систем и традиционными подходами к их проектированию определяют на сегодня одну из основных задач теории систем — задачу разработки методологии и создания систем автоматизированного проектирования, решающих основные задачи синтеза. Данная проблема включает в себя разработку методов и алгоритмов, помогающих вести направленный поиск оптимальных характеристик системы, а также позволяющих контролировать изменение этих характеристик в процессе проектирования. Указанные потребности привели к возникновению и интенсивному развитию системного подхода к проектированию сложных систем, в концепцию которого органически вписывается структурный подход, приобретающий еще большее значение на современном этапе [13—15, 17].

Другим важным вопросом, требующим решения при проектировании сложных систем, является представление имеющихся данных и параллельных процессов в виде специальных формальных объектов, удобных для проведения над ними вычислительных и имитационных экспериментов. С большой долей достоверности можно отметить, что большинство современных дискретных динамических систем характеризуются такими свойствами, как параллелизм, недетерминированность, многоуровневость представления, сочетание синхронных и асинхронных процессов, однородность и др. Поэтому выбор формализованного языка, в наибольшей степени учитывающего особенности современных систем, является основной задачей начального этапа проектирования.

Преимущества СП при описании указанных сложных систем очевидны и широко описаны в литературе. Недостатком данного аппарата является отсутствие инструментария для количественной оценки моделей, выраженных в терминах СП. Вместе с тем теория структур и теория сложности [7—9] предоставляют методы и алгоритмы, позволяющие строить оценочные шкалы для получения количественных характеристик элементов дискретного пространства. В работе [13] были сделаны некоторые шаги, позволяющие в рамках пространства структур СП ввести оценочные шкалы и получить количественные характеристики синтезируемых СП-моделей. В настоящей работе представлено дальнейшее развитие описанного направления исследований.

1. Описание матричной структуры

Одним из методов достижения высокой производительности систем обработки данных реального времени является использование процессорных матриц — специализированных вычислительных структур, состоящих из множества относительно простых устройств (процессорных элементов), объединенных в однородные параллельные системы. Данные процессоры предназначены для решения узкого круга задач, характеризующихся большим объемом обрабатываемой информации. Число процессорных элементов (ПЭ) определяется размерами входного массива данных, а также возможностью разбиения исходных данных на подмассивы, позволяющие проводить независимую обработку [11, 12].

В данном примере рассматривается подход к проектированию матричных процессоров, основанный на структурном анализе СП и тензорной методологии [13, 14].

2.1. Описание примера

Пусть имеется матрица $A = [a_{ij}]$ и вектор $B = [b_j]$ размерности J , необходимо найти вектор $C = AB$. Матрица A может быть разложена на сумму матриц A_i и, следовательно,

$$C = A_1 B + A_2 B + \dots + A_J B,$$

где A_i — матрица, у которой i -я строка совпадает с i -й строкой матрицы A , а остальные элементы — нули.

Матричное умножение может быть осуществлено за J шагов (рекурсий):

$$c_i^{(k)} = c_i^{(k-1)} + \alpha_i^{(k)} b_j^{(k)},$$

при этом

$$a_i^{(k)} = a_{ik}, b_j^{(k)} = b_k, k = 1, 2, \dots, J; c_i^{(0)} = 0.$$

Процессорная матрица Q , представленная графом потока данных [11, 16] и отображающая процедуру умножения $A[3,3]$ на вектор $B[3]$, приведена на рис. 1, а. Предполагается, что элементы матрицы A находятся в узлах графа (процессорные элементы — ПЭ), элементы вектора B подаются на ПЭ слева, результирующие значения вектора C считываются с верхней строки матричной структуры. Регистры ПЭ, содержащие промежуточные суммы $c_i^{(k)}$, в началь-

ный момент времени установлены в ноль. Структура ПЭ представлена на рис. 1, б.

Процесс начинается с ПЭ, имеющего координаты (1,1) и содержащего элемент a_{11} :

$$q_1^{(1)} = q_1^{(0)} + a_{11}b_1.$$

Вычислительный фронт распространяется к ближайшим соседним ПЭ с координатами (1,2) и (2,1), которые параллельно будут выполнять операции:

$$q_1^{(2)} = q_1^{(1)} + a_{12}b_2 \text{ и } q_2^{(1)} = q_2^{(0)} + a_{21}b_1.$$

Очевидно, что $q_j^{(0)} = 0$.

Следующий фронт активности будет у процессорных элементов (1,3), (2,2) и (3,1). Так создается движущийся вверх и вправо по процессорной матрице вычислительный волновой фронт. После того как волновой фронт проходит через все ПЭ, первая рекурсия считается выполненной.

2.2. Синтез структур

Назовем тактом интервал времени t , в течение которого происходит передача данных между соседними ПЭ, а также выполнение операций умножения и сложения в самом ПЭ. Тогда можно отметить, что первый элемент $c_1 = q_1^{(3)}$ вектора C будет получен через три такта, а вся процедура умножения матрицы на вектор (получение последнего элемента $c_3 = q_3^{(3)}$) завершится через пять тактов.

Так как в каждый момент времени выполняющие рекурсию ПЭ будут различны, на процессорной матрице (рис. 1, а) возможна организация конвейерных вычислений. Это важное замечание, позволяющее обосновать проектирование конвейерных процессоров различной структуры.

Анализ количественных и качественных характеристик процессорных матриц (временные оценки, корректность протекающих параллельных процессов, эффективность загрузки ПЭ и др.), представленных графовыми моделями, достаточно эффективно можно осуществлять с использованием СП.

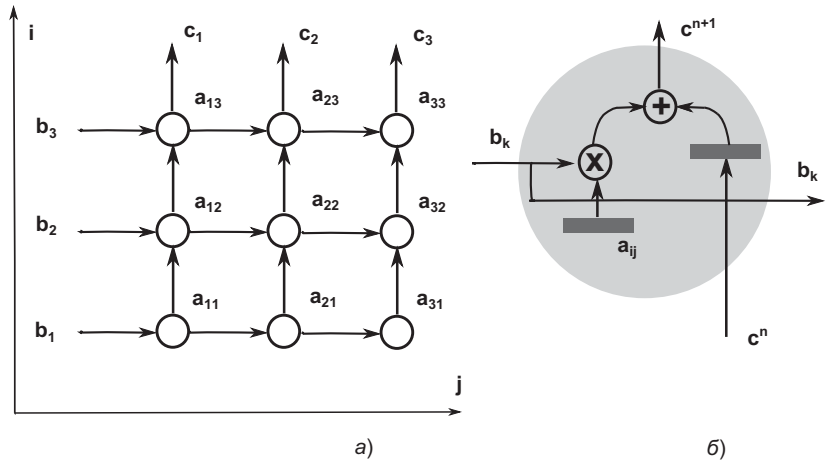


Рис. 1. Граф потока данных умножения матрицы $A(3 \times 3)$ на вектор $B(3)$ (а) и структура процессорного элемента (б)

СП N_1 , моделирующая матричную структуру Q , приведена на рис. 2. Переходы СП-модели t_{ij} соответствуют процессорным элементам структуры, позиции b^*_{ij} моделируют движение элементов вектора B , а позиции c^*_{ij} — движение частичных сумм при получении результата умножения. Предполагается, что элементы матрицы A находятся во внутренней памяти ПЭ. При анализе СП-модели видно, что на первом шаге (такте) активным является лишь переход t_{11} .

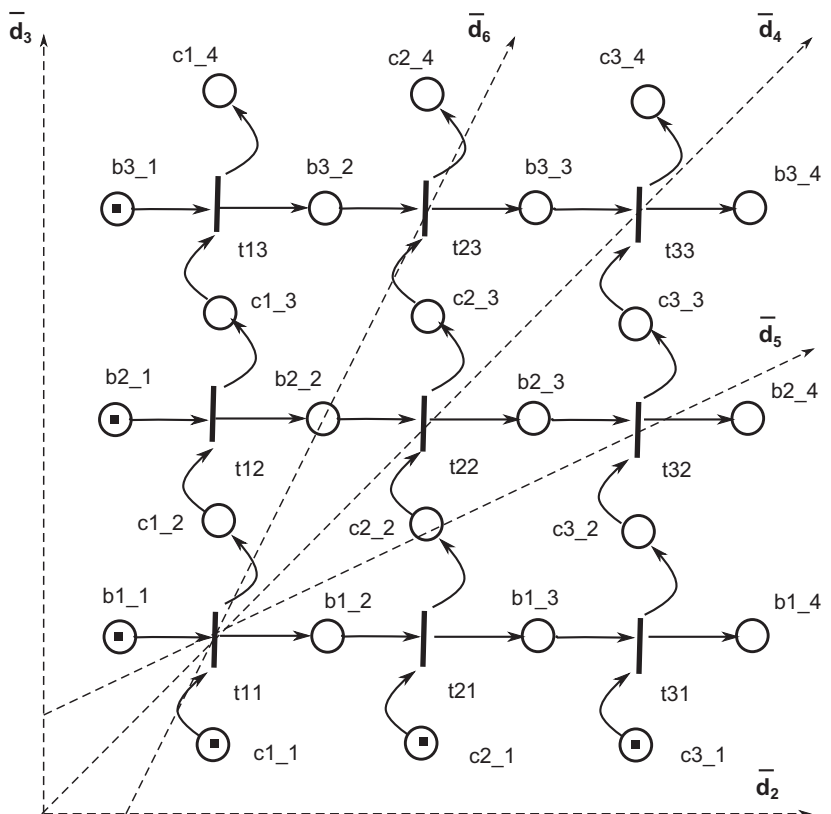


Рис. 2. СП-модель N_1 двумерной параллельной структуры умножения матрицы $A(3 \times 3)$ на вектор $B(3)$

Срабатывание данного перехода активизирует переходы t_{12} и t_{21} , затем активизируются переходы t_{13} , t_{22} , t_{31} (моделирование фронта активности ПЭ) и т. д. Процесс моделирования умножения матрицы на вектор заканчивается, когда в позициях c_{1_4} , ..., c_{3_4} появляются метки.

Анализ функционирования СП-модели N_1 показывает, что при моделировании операции умножения каждый переход срабатывает лишь один раз. Это говорит о том, что ПЭ матрицы (при выполнении одиночной операции умножения матрицы на вектор) будут заняты только в течение одного такта, и их коэффициент занятости будет составлять в среднем 1/5 часть от общего времени выполнения операции. Возникает задача проектирования других параллельных структур, выполняющих ту же функцию (умножение матрицы на вектор), но имеющих более высокие показатели загрузки ПЭ. С этой целью введем понятие вектора проекции \vec{d} ПЭ матричной структуры [11]. При описании нашего подхода мы будем придерживаться линейных проекций, при выполнении которых вершины графа, расположенные на заданной прямой, проецируются в один элемент.

Рассмотрим проекции вершин СП, моделирующей структуру Q , заданные штриховыми линиями и обозначенные $\vec{d}_2, \dots, \vec{d}_6$ (рис. 2). На рис. 3 представлены СП-модели N_2, \dots, N_6 , которые получены в результате выполнения указанных проекций. Отметим некоторые особенности интерпретации вершин данных СП-моделей:

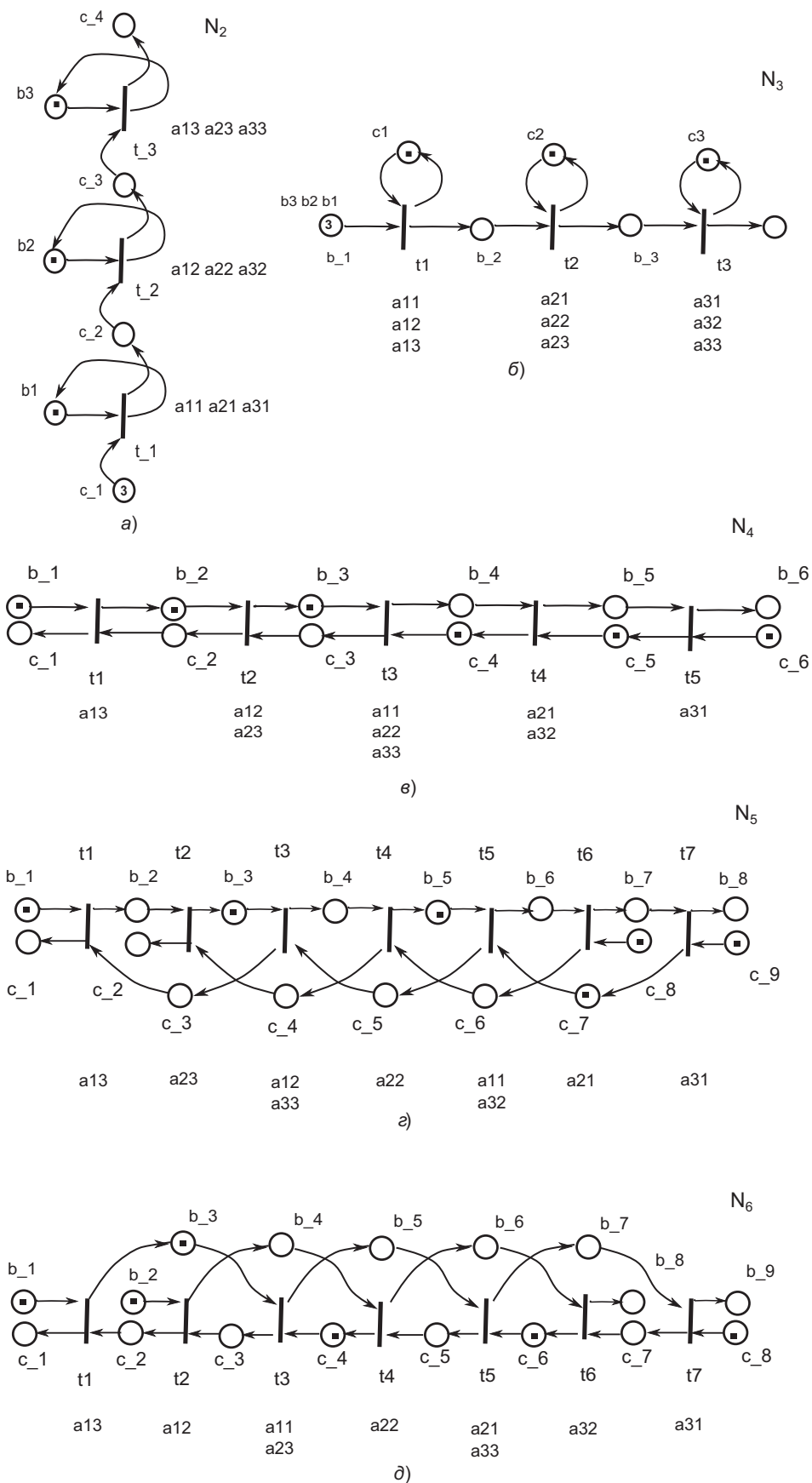


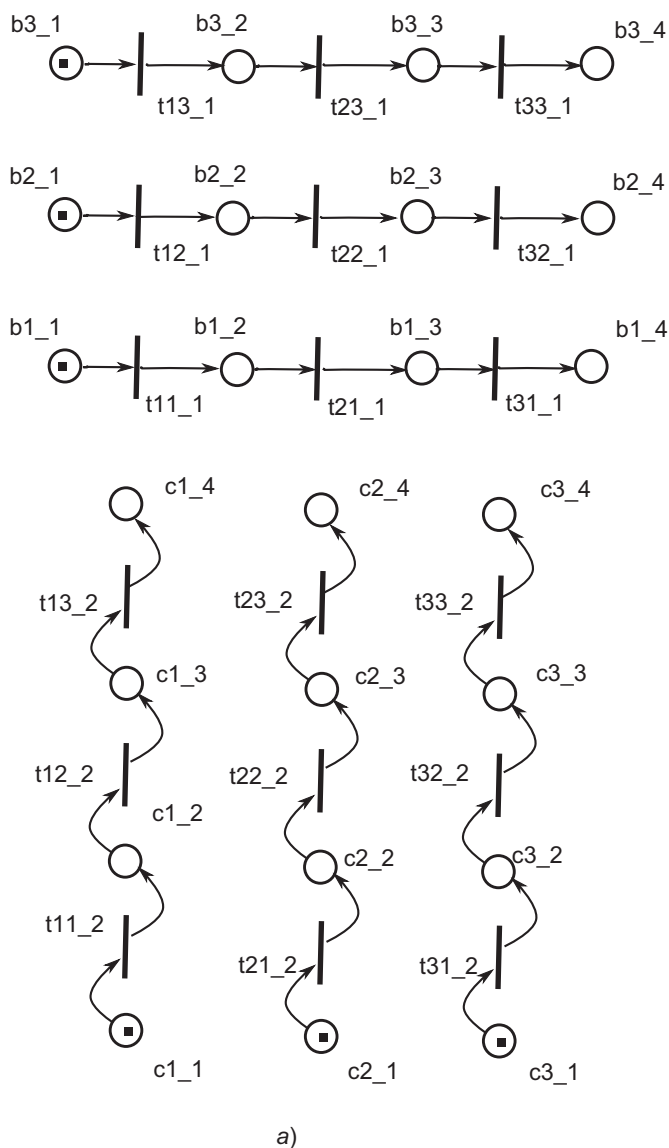
Рис. 3. СП-модели параллельных структур умножения матрицы $A(3 \times 3)$ на вектор $B(3)$, полученные в результате проекций d_2 (а), d_3 (б), d_4 (в), d_5 (г), d_6 (д)

а) на рис. 3, $a-d$ каждому переходу поставлены в соответствие элементы матрицы A , которые должны храниться во внутренней памяти ПЭ матричных структур, моделируемых данными СП-моделями;

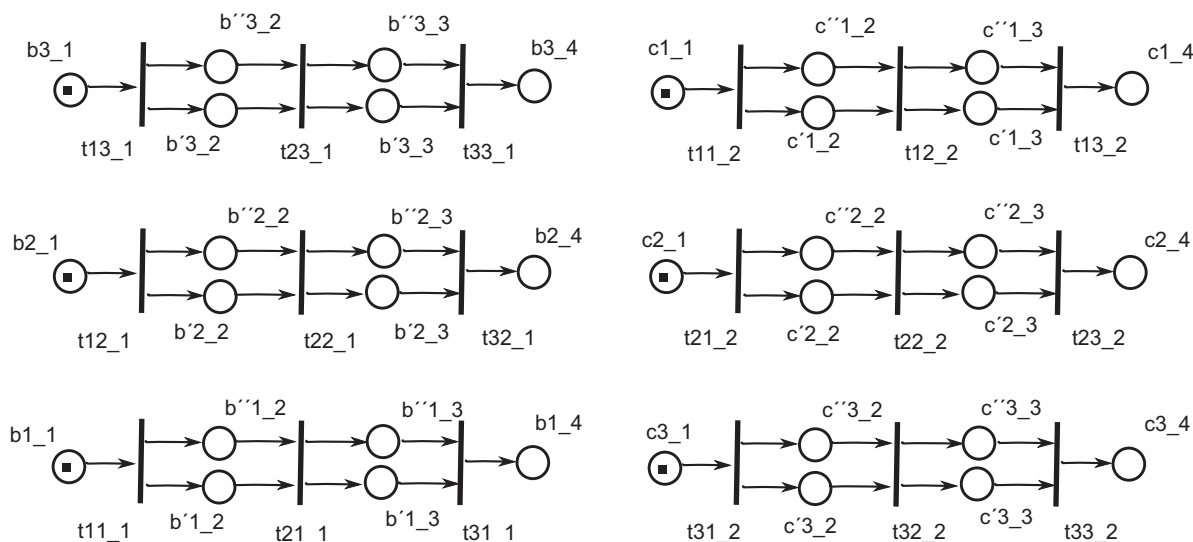
б) петли, полученные в СП-моделях N_2 и N_3 , указывают на образование в новых матричных структурах рекурсивных связей, моделирующих подачу элементов вектора B (рис. 3, a) и получение частичных сумм (рис. 3, b).

Проводя предварительный анализ СП-моделей, представленных на рис. 3, можно отметить, что они имеют одинаковое число тактов на выполнение операции умножения матрицы на вектор, но отличаются числом ПЭ и организацией входных и выходных данных.

В целях демонстрации предлагаемого подхода к проектированию сложных систем рассмотрим процедуру построения СП-моделей, представленных на рис. 3, с помощью структурного анализа и тензорной методологии. Для этого воспользуемся тензорными методами исследования СП [14]. Используя операции деления вершин СП [13], представим исходную СП-модель (см. рис. 2) в виде множества линейных базовых фрагментов (ЛБФ) (рис. 4, a). Данному множеству поставим в соответствие примитивную систему, представленную на рис. 5. В целях формирования одинаковой размерности матриц, описывающих систему ЛБФ и примитивную систему, дополним линейные базовые фрагменты эквивалентными позициями (см. рис. 4, b). Обозначим $D_{\beta'}^{\gamma}$ и D_{β}^{γ} матрицы инцидентности системы ЛБФ и примитивной системы соответственно.



а)



б)

Рис. 4. Система ЛБФ без эквивалентных позиций (a), с эквивалентными позициями (b)

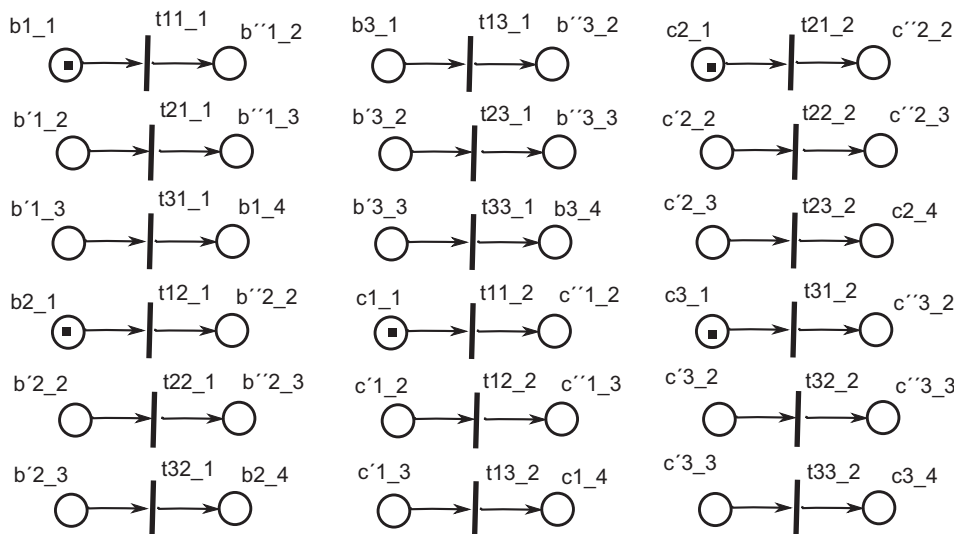


Рис. 5. Прimitивная система N_{PR}

Подставляя матрицы $D_{\beta'}^{\gamma'}$ и D_{β}^{γ} в выражение (1), можно построить тензор преобразования $C_{\beta'\gamma}^{\beta\gamma'}$ [14, 15]:

$$D_{\beta'}^{\gamma'} = C_{\beta'\gamma}^{\beta\gamma'} \times D_{\beta}^{\gamma}. \quad (1)$$

Рассмотрим программы синтеза СП-моделей, которые были представлены на рис. 3. Данные программы построены эмпирически и отображают лишь некоторые из множества возможных программ синтеза каждой из рассматриваемых СП-моделей.

Опишем общий метод построения программ синтеза всех возможных СП-моделей. В качестве исходных данных для синтеза СП-моделей выберем примитивную систему N_{PR} , состоящую из m переходов и n позиций, а также операции объединения вершин примитивной системы, описанные в работе [13].

Поставим в соответствие примитивной системе N_{PR} некоторую последовательность V ($V_T + V_P$), состоящую из $(m + n)$ элементов. Будем считать, что элемент последовательности V равен нулю, если соответствующая вершина примитивной системы N_{PR} не принадлежит подмножеству объединяемых вершин, и элемент последовательности V больше нуля (имеет положительное значение), если соответствующая вершина принадлежит подмножеству объединяемых вершин. При выполнении операции объединения вершин элементы последовательности V должны содержать следующую информацию: во-первых, участвует ли данная вершина в процессе объединения, а во-вторых, какому объединяемому подмножеству данная вершина принадлежит.

Пусть каждый элемент частей V_T и V_P последовательности V меняется в пределах от 1 до z_t и от 1 до z_p , где z_t и z_p — максимальные числа подмножеств вершин, на которое может быть разделено множество переходов и множество позиций для объединения, соответственно.

Так как минимальное число элементов, входящих в подмножество объединяемых вершин, равно 2, то величина z_p определяется следующим образом:

$$z_p = |P|/2,$$

а величина z_t :

$$z_t = \begin{cases} \frac{|T|}{2}, & \text{если } |T| \text{ — четно;} \\ \frac{|T|-1}{2}, & \text{если } |T| \text{ — нечетно.} \end{cases}$$

Пусть части V_T и V_P последовательности V содержат некоторые числа разрядности $m = |T|$ и $n = |P|$ в системах счисления z_t и z_p , соответственно. Тогда каждый разряд данных чисел будет содержать информацию как об участии вершины СП N_{PR} в процессе объединения (значение соответствующего разряда больше нуля), так и о том, какому объединяемому подмножеству данная вершина принадлежит (разряды чисел V_T или V_P , которым соответствуют вершины, входящие в одно объединяемое подмножество, имеют одинаковые значения).

В итоге значения элементов последовательности V задают комбинацию, определяющую программу синтеза СП-модели. Отсюда алгоритм генерации программ синтеза всех СП-моделей, входящих в СП-структуру с рангом $r = m$, можно представить счетчиком, состоящим из двух частей V_T и V_P , каждая из которых работает в системах счисления z_t и z_p соответственно.

Среди комбинаций, генерируемых описанным алгоритмом, существуют такие, которые задают программы синтеза эквивалентных СП-моделей. Для определения таких комбинаций рассмотрим следующие свойства последовательности V .

Свойство 1. Комбинация V , содержащая лишь один положительный элемент, отличный от остальных элементов в части V_T или в части V_P , описывает эквивалентную СП-модель.

Свойство 2. Если существует некоторая подстановка X с областью определения $\{1, 2, \dots, z_x\}$ и подстановка Y с областью определения $\{1, 2, \dots, z_p\}$, которые переводят комбинацию V в комбинацию V' :

$$V_T \xrightarrow{X} V'_T \text{ и } V_P \xrightarrow{Y} V'_P,$$

то комбинация V' описывает программу синтеза эквивалентной СП-модели.

Рассмотрим программы синтеза СП-моделей N_1, \dots, N_6 , представленных на рис. 2 и рис. 3.

Программа синтеза исходной СП-модели N_1 имеет следующий вид:

$$\begin{aligned} t11 &= t11_1 + t11_2 \\ t12 &= t12_1 + t12_2 \\ t13 &= t13_1 + t13_2 \\ t21 &= t21_1 + t21_2 \\ t22 &= t22_1 + t22_2 \\ t23 &= t23_1 + t23_2 \\ t31 &= t31_1 + t31_2 \\ t32 &= t32_1 + t32_2 \\ t33 &= t33_1 + t33_2 \end{aligned} \quad (2)$$

1	4	7	2	5	8	3	6	9	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

t11_1 t21_1 t31_1 t12_1 t22_1 t32_1 t13_1 t23_1 t33_1 t11_2 t12_2 t13_2 t21_2 t22_2 t23_2 t31_2 t32_2 t33_2

Рис. 6. Программа синтеза исходной СП-модели N_1 (часть V_T)

	t11	t21	t31	t12	t22	t32	t13	t23	t33	μ_0
b1_1	-1									1
b1_2	1									
b1_3		-1								
b1_4		1								
b2_1			-1							1
b2_2			1							
b2_3				-1						
b2_4				1						
b3_1					-1					1
b3_2					1					
b3_3						-1				
b3_4						1				
c1_1	-1									1
c1_2	1									
c1_3			-1							
c1_4			1							
c2_1		-1								1
c2_2		1								
c2_3				-1						
c2_4				1						
c3_1			-1							1
c3_2			1							
c3_3					-1					
c3_4					1					

a)

	t11	t21	t31	t12	t22	t32	t13	t23	t33	μ_0
b1_1	-1									1
b1_2	1	-1								
b1_3		1	-1							
b1_4			1	-1						
b2_1				1	-1					1
b2_2					1	-1				
b2_3						1	-1			
b2_4							1	-1		
b3_1								1	-1	1
b3_2								1	-1	
b3_3									1	-1
b3_4										1
c1_1	-1									1
c1_2	1									
c1_3			-1							
c1_4			1							
c2_1		-1								1
c2_2		1								
c2_3				-1						
c2_4				1						
c3_1			-1							1
c3_2			1							
c3_3					-1					
c3_4					1					

б)

	t11	t21	t31	t12	t22	t32	t13	t23	t33	μ_0
b1_1	-1									1
b1_2	1	-1								
b1_3		1	-1							
b1_4			1							
b2_1				-1						1
b2_2				1	-1					
b2_3					1	-1				
b2_4						1				
b3_1							-1			1
b3_2							1	-1		
b3_3								1	-1	
b3_4									1	
c1_1	-1									1
c1_2	1									
c1_3			-1							
c1_4			1							
c2_1		-1								1
c2_2		1								
c2_3				-1						
c2_4				1						
c3_1			-1							1
c3_2			1							
c3_3					-1					
c3_4					1					

в)

Рис. 7. Матрицы инцидентности исходной СП-модели N_1 :

a — матрица D_1 ; б — матрица D'_1 ; в — матрица D'_1 после удаления эквивалентных вершин

Последовательность V , соответствующая данной программе, представлена на рис. 6. Ввиду того что при синтезе СП-модели N_1 объединения позиций не происходит (разряды части V_P равны нулю), последовательность представлена только частью V_T . Рассмотрим процедуру синтеза данной СП-модели с использованием тензорного подхода.

На рис. 7, а представлена матрица инцидентности D_1 исходной СП-модели N_1 (с учетом исключения эквивалентных переходов), выраженная в системе координат примитивной системы. На рис. 7, б представлена матрица инцидентности D'_1 в исходной системе координат (после умножения матрицы D_1 на тензор $C_{\beta\gamma}^{\beta\gamma}$). На рис. 7, в представлена матрица инцидентности D'_1 после исключения эквивалентных позиций. Можно убедиться, что полученная матрица инцидентности D'_1 в точности описывает СП, представленную на рис. 2. Следует отметить, что пустые ячейки матриц инцидентности соответствуют нулевым значениям. Этот прием использован для наглядности и будет использоваться в дальнейшем.

В силу того что дальнейший синтез СП-моделей N_2, \dots, N_6 будет базироваться на исходной СП-модели N_1 , программы синтеза СП-моделей N_2, \dots, N_6 будем строить относительно матриц инцидентности D_1 и D'_1 , описывающих СП-модель N_1 .

Программа синтеза СП-модели N_2

Один из возможных вариантов программы синтеза СП-модели N_2 , дополняющий программу (2), имеет следующий вид:

$$\begin{aligned}
 t_{_1} &= t11 + t21 + t31 \\
 t_{_2} &= t12 + t22 + t32 \\
 t_{_3} &= t13 + t23 + t33 \\
 b1 &= b1_{_1} + b1_{_4} \\
 b2 &= b2_{_1} + b2_{_4} \\
 b3 &= b3_{_1} + b3_{_4}
 \end{aligned}
 \tag{3}$$

Последовательность V , задающая синтез СП-модели N_2 , представлена на рис. 8.

На рис. 9, а представлена матрица инцидентности D_2 после выполнения программы (3) (выражена в системе координат примитивной системы). Следует отметить, что матрица инцидентности в силу своего построения имеет недостаток, который проявляется в отсутствии возможности отображать петли. После выполнения программы (3) такие связи возникают. В целях отображения петель в матрице инцидентности примем обозначение $\langle \text{число} \rangle^*$, в котором знак "*" указывает на наличие петли, а параметр $\langle \text{число} \rangle$ указывает на номер операции объединения вершин в последовательности синтеза, в результате выполнения которой данная петля образовалась. Например, на рис. 9 знаками $1^*, 2^*$ и 3^* обозначено, что в результате операций объединения позиций, отмеченных в части V_p программы синтеза СП-модели N_2 значениями 1, 2 и 3, получены петли.

На рис. 9, б представлена матрица инцидентности D'_2 , полученная после умножения матрицы D_2 на тензор $C_{\beta\gamma}^{\beta\gamma}$ (перевод в исходную систему координат). Эквивалентные переходы в данной матрице не показаны. На рис. 9, в представлена матрица инцидентности D'_2 после исключения эквивалентных вершин СП-модели. Данная матрица полностью соответствует СП-модели N_2 (см. рис. 3, а).

Аналогичным образом могут быть построены СП-модели N_3, \dots, N_6 , программы синтеза которых представлены на рис. 8.

	N_2	N_3	N_4	N_5	N_6	N_7
b1_1	1		2	4	1	1
b"1_2				5	2	4
b'1_2			3	5	2	4
b"1_3				6	3	5
b'1_3			4	6	3	5
b1_4	1					2
b2_1	2		1	2		2
b"2_2			2	3	4	6
b'2_2				3	4	6
b"2_3				4	5	7
b'2_3				4	5	7
b2_4	2		4	5		3
b3_1	3					3
b"3_2			1	1	1	8
b'3_2				1	1	8
b"3_3				2	2	9
b'3_3				2	2	9
b3_4	3		3	3	3	1
c1_1		1	6	11	8	10
c"1_2				9	7	13
c'1_2			7	9	7	13
c"1_3				7	6	14
c'1_3			8	7	6	14
c1_4		1				12
c2_1		2	5		10	12
c"2_2			6	10	9	15
c'2_2				10	9	15
c"2_3				8	8	16
c'2_3				8	8	16
c2_4		2	8		7	11
c3_1		3				11
c"3_2			5	11	11	17
c'3_2				11	11	17
c"3_3				9	10	18
c'3_3				9	10	18
c3_4		3	7	7	9	10
t11	1	1	2	2	1	
t21	1	2	3		2	
t31	1	3				
t12	2	1	1	1		
t22	2	2	2			
t32	2	3	3	2		
t13	3	1				
t23	3	2	1		1	
t33	3	3	2	1	2	

Рис. 8. Программы синтеза СП-моделей

Программа синтеза СП-модели N_7

СП-модели N_2, \dots, N_6 построены на основе алгоритма свертки линейно расположенных вершин СП-модели N_1 . Покажем, что существуют другие программы синтеза СП-моделей, моделирующих работу матричной структуры при умножении матрицы A на вектор B .

Рассмотрим программу синтеза (4). Выполняя последовательность шагов, описанную при синтезе структур N_2, \dots, N_6 , можно получить новую СП-модель N_7 , матрица инцидентности которой представлена на рис. 10.

	t11	t21	t31	t12	t22	t32	t13	t23	t33		μ_0
b1_1	1*	1*	1*							1	1
b*1_2	1	1	1							2	
b'1_2	-1	-1	-1							3	
b''1_3	1	1	1							4	
b'1_3	-1	-1	-1							5	
b1_4	1*	1*	1*							6	
b2_1				2*	2*	2*				7	1
b''2_2				1	1	1				8	
b'2_2				-1	-1	-1				9	
b''2_3				1	1	1				10	
b'2_3				-1	-1	-1				11	
b2_4				2*	2*	2*				12	
b3_1							3*	3*	3*	13	1
b''3_2							1	1	1	14	
b'3_2							-1	-1	-1	15	
b''3_3							1	1	1	16	
b'3_3							-1	-1	-1	17	
b3_4							3*	3*	3*	18	
c1_1	-1	-1	-1							19	1
c'1_2	1	1	1							20	
c''1_2				-1	-1	-1				21	
c'1_3				1	1	1				22	
c''1_3							-1	-1	-1	23	
c1_4							1	1	1	24	
c2_1	-1	-1	-1							25	1
c'2_2	1	1	1							26	
c''2_2				-1	-1	-1				27	
c'2_3				1	1	1				28	
c''2_3							-1	-1	-1	29	
c2_4							1	1	1	30	
c3_1	-1	-1	-1							31	1
c'3_2	1	1	1							32	
c''3_2				-1	-1	-1				33	
c'3_3				1	1	1				34	
c''3_3							-1	-1	-1	35	
c3_4							1	1	1	36	
	t1	t1	t1	t2	t2	t2	t3	t3	t3		

а)

		t11	t21	t31	t12	t22	t32	t13	t23	t33		μ_0
b1	b1_1	1*									1	1
	b''1_2										2	
	b'1_2										3	
	b''1_3										4	
	b'1_3										5	
b2	b1_4	1*			2*						6	
	b2_1										7	1
	b''2_2										8	
	b'2_2										9	
	b''2_3										10	
	b'2_3										11	
	b2_4				2*						12	
b3	b3_1							3*			13	1
	b''3_2										14	
	b'3_2										15	
	b''3_3										16	
	b'3_3										17	
	b3_4							3*			18	
c_1	c1_1	-1									19	1
c_2	c'1_2	1			-1						20	
	c''1_2	1			1						21	
	c'1_3				1			-1			22	
	c''1_3							-1			23	
	c1_4							1			24	
	c2_1	-1									25	1
	c'2_2	1			-1						26	
	c''2_2	1			-1						27	
c_3	c'2_3				1			-1			28	
	c''2_3				1			-1			29	
	c2_4							1			30	
	c3_1	-1									31	1
	c'3_2	1			-1						32	
	c''3_2	1			-1						33	
	c'3_3				1			-1			34	
	c''3_3				1			-1			35	
c_4	c3_4							1			36	
		t_1			t_2			t_3				

б)

Рис. 9. Матрицы инцидентности СП-модели N_2 :

a — матрица D_2 ; b — матрица D'_2 ; c — матрица D''_2 после удаления эквивалентных вершин

	t_1	t_2	t_3	μ_0
b1	1*			1
b2		2*		1
b3			3*	1
c_1	-1			3
c_2	1	-1		
c_3		1	-1	
c_4			1	

в)

$$\begin{aligned}
 b_{_1} &= b1_{_1} + b3_{_4} \\
 b_{_2} &= b1_{_4} + b2_{_1} \\
 b_{_3} &= b2_{_4} + b3_{_1} \\
 b1_{_2} &= b'1_{_2} + b''1_{_2} \\
 b1_{_3} &= b'1_{_3} + b''1_{_3} \\
 b2_{_2} &= b'2_{_2} + b''2_{_2} \\
 b2_{_3} &= b'2_{_3} + b''2_{_3} \\
 b3_{_2} &= b'3_{_2} + b''3_{_2} \\
 b3_{_3} &= b'3_{_3} + b''3_{_3} \\
 c_{_1} &= c1_{_1} + c3_{_4} \\
 c_{_3} &= c2_{_4} + c3_{_1} \\
 c_{_2} &= c1_{_4} + c2_{_1} \\
 c1_{_2} &= c'1_{_2} + c''1_{_2} \\
 c1_{_3} &= c'1_{_3} + c''1_{_3} \\
 c2_{_2} &= c'2_{_2} + c''2_{_2} \\
 c2_{_3} &= c'2_{_3} + c''2_{_3} \\
 c3_{_2} &= c'3_{_2} + c''3_{_2} \\
 c3_{_3} &= c'3_{_3} + c''3_{_3}
 \end{aligned}$$

(4)

	t11	t21	t31	t12	t22	t32	t13	t23	t33	μ_0
b_1	-1								1	1
b1_2	2	-2								
b1_3		2	-2							
b_2			1	-1						1
b2_2				2	-2					
b2_3					2	-2				
b_3						1	-1			1
b3_2							2	-2		
b3_3								2	-2	
c_1	-1								1	1
c1_2	2			-2						
c1_3				2			-2			
c_2		-1					1			1
c2_2		2			-2					
c2_3					2			-2		
c_3			-1					1		1
c3_2			2			-2				
c3_3						2			-2	

Рис. 10. Матрица инцидентности СП-модели N_7 (D'_7) после удаления эквивалентных вершин

Такты работы СП-модели N_2

№ ПЭ	Такты				
	1	2	3	4	5
1 ($t_{_1}$)	$a_{11} \cdot b_1$	$a_{21} \cdot b_1$	$a_{31} \cdot b_1$		
2 ($t_{_2}$)		$a_{11} \cdot b_1 + a_{12} \cdot b_2$	$a_{21} \cdot b_1 + a_{22} \cdot b_2$	$a_{31} \cdot b_1 + a_{32} \cdot b_1$	
3 ($t_{_3}$)			$a_{11} \cdot b_1 + a_{12} \cdot b_2 + a_{13} \cdot b_3$	$a_{21} \cdot b_1 + a_{22} \cdot b_2 + a_{23} \cdot b_3$	$a_{31} \cdot b_1 + a_{32} \cdot b_2 + a_{33} \cdot b_3$

Таблица 2

Такты работы СП-модели N_4

№ ПЭ	Такты				
	1	2	3	4	5
1 ($t_{_1}$)			$a_{11} \cdot b_1 + a_{12} \cdot b_2 + a_{13} \cdot b_3$		
2 ($t_{_2}$)		$a_{11} \cdot b_1 + a_{12} \cdot b_2$		$a_{21} \cdot b_1 + a_{22} \cdot b_2 + a_{23} \cdot b_3$	
3 ($t_{_3}$)	$a_{11} \cdot b_1$		$a_{21} \cdot b_1 + a_{22} \cdot b_2$		$a_{31} \cdot b_1 + a_{32} \cdot b_2 + a_{33} \cdot b_3$
4 ($t_{_4}$)		$a_{21} \cdot b_1$		$a_{31} \cdot b_1 + a_{32} \cdot b_2$	
5 ($t_{_5}$)			$a_{31} \cdot b_1$		

Таблица 3

Такты работы СП-модели N_5

№ ПЭ	Такты				
	1	2	3	4	5
1 ($t_{_1}$)			$a_{11} \cdot b_1 + a_{12} \cdot b_2 + a_{13} \cdot b_3$		
2 ($t_{_2}$)				$a_{21} \cdot b_1 + a_{22} \cdot b_2 + a_{23} \cdot b_3$	
3 ($t_{_3}$)		$a_{11} \cdot b_1 + a_{12} \cdot b_2$			$a_{31} \cdot b_1 + a_{32} \cdot b_2 + a_{33} \cdot b_3$
4 ($t_{_4}$)			$a_{21} \cdot b_1 + a_{22} \cdot b_2$		
5 ($t_{_5}$)	$a_{11} \cdot b_1$			$a_{31} \cdot b_1 + a_{32} \cdot b_2$	
6 ($t_{_6}$)		$a_{21} \cdot b_1$			
7 ($t_{_7}$)			$a_{31} \cdot b_1$		

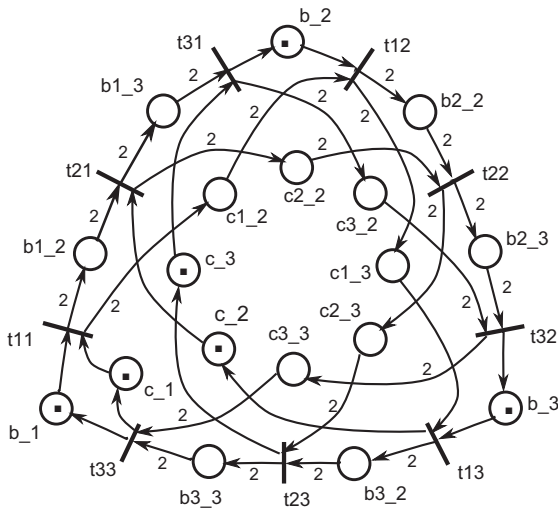


Рис. 11. СП-модель N_7 параллельной структуры умножения матрицы $A(3 \times 3)$ на вектор $B(3)$, полученной в результате выполнения программы синтеза (4)

Последовательность V , задающая синтез СП-модели N_7 , представлена на рис. 8.

2.3. Анализ структур

В ходе анализа построенных структур оценим характеристики процессорных матриц, моделируемых СП-моделями N_1, \dots, N_7 . Потактовую работу СП-моделей можно представить в виде специальных таблиц, показывающих очередность срабатывания переходов и их загрузку, организацию потока входных и выходных данных, а также условия окончания выполняемой операции. Табл. 1—4 демонстрируют функционирование некоторых построенных СП-моделей.

В целях количественного и качественного анализа построенных матричных структур введем некоторые критерии оценки структурных и динамических свойств соответствующих СП-моделей. Обозначим P_i — число тактов, в течение которых ПЭ был занят; P — общее число тактов, в течение которых выполняется операция; M — число ПЭ, входящих в анализируемую структуру.

Введем коэффициент использования i -го ПЭ:

$$k_i = \frac{P_i}{P}.$$

Тогда средний коэффициент использования ПЭ матричной структуры можно представить следующей формулой:

$$k_{cp} = \frac{M}{\sum_{i=1}^M P_i} (PM).$$

Например, для структур, представленных на рис. 3, δ и рис. 3, ϵ ,

$$k_{cp} = (1 + 2 + 3 + 2 + 1) / (5 \cdot 5) = 0,36$$

и $k_{cp} = (1 + 1 + 2 + 1 + 2 + 1 + 1) / (5 \cdot 7) = 0,26$ соответственно.

Таблица 4

Такты работы СП-модели N_7

№ ПЭ	Такты				
	1	2	3	4	5
1 (t_11)	$a_{11} * b_1$				
2 (t_21)		$a_{21} * b_1$			
3 (t_31)			$a_{31} * b_1$		
4 (t_12)		$a_{11} * b_1 + a_{12} * b_2$			
5 (t_22)			$a_{21} * b_1 + a_{22} * b_2$		
6 (t_32)				$a_{31} * b_1 + a_{32} * b_2$	
7 (t_13)			$a_{11} * b_1 + a_{12} * b_2 + a_{13} * b_3$		
8 (t_23)				$a_{21} * b_1 + a_{22} * b_2 + a_{23} * b_3$	
9 (t_33)					$a_{31} * b_1 + a_{32} * b_2 + a_{33} * b_3$

Таблица 5

Количественные показатели синтезированных СП-моделей

№	Идентификатор СП-модели	Число ПЭ	Число тактов	Коэффициент использования ПЭ (k_i)	Средний коэффициент использования ПЭ ($k_{ср}$)
1	N_1	9	5	1/5	0,11
2	N_2	3	5	3/5	0,60
3	N_3	3	5	3/5	0,60
4	N_4	5	5	Величина переменная	0,36
5	N_5	7	5	Величина переменная	0,26
6	N_6	7	5	Величина переменная	0,26
7	N_7	9	5	Величина переменная	0,2
8	N_0	1	9	1	1

Вычисленные коэффициенты для синтезированных СП-моделей сведены в табл. 5. Для более полного сравнения в последней строке таблицы приведены значения коэффициентов для структуры N_0 , включающей один ПЭ.

На рис. 12 приведена зависимость коэффициента использования ПЭ синтезированных структур от числа ПЭ. На основе данной зависимости можно оценить, например, какими значениями будут характеризоваться структуры, полученные

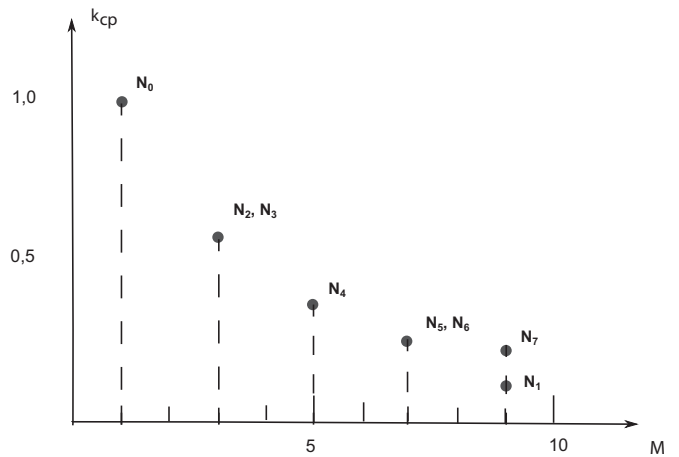


Рис. 12. Зависимость коэффициента использования ПЭ от конфигурации матричной структуры

в результате выполнения других программ синтеза, не рассмотренных в данной работе.

Заключение

Описанные в данной работе результаты дают предпосылки для построения метода синтеза новых структур параллельных вычислительных систем. В основу метода положены модели, выраженные в терминах сетей Петри. Возможность существования подобных методов проектирования продемонстрирована на синтезе структур, реализующих операцию матричного умножения. Однако класс задач, для решения которых предложенный подход актуален, более широк.

Операция проекции расположенных линейно вершин СП-моделей в одну вершину (операция свертки) позволяет не только получить структуры с новыми свойствами, но и перейти к анализу моделей, имеющих меньшую размерность. Данный переход позволяет применять известные методы анализа СП к сетям меньшей размерности, что в итоге повышает эффективность и результативность процедур анализа.

Использование возможностей структурного анализа сложных систем, и, в частности, СП-моделей, позволяет на множестве построенных моделей задать систему оценочных шкал и провести количественный анализ свойств построенных моделей, что дает основу для построения программных средств поиска оптимальных структур по заданным критериям.

Основными препятствиями, мешающими непосредственному использованию предложенных методов, являются ограничения, связанные с числом синтезируемых СП-моделей. Это число растет экспоненциально в зависи-

мости от размерности примитивной системы. Для того чтобы описанные методы можно было бы использовать на практике, *необходим поиск новых приемов и подходов, которые ограничили бы множество рассматриваемых СП-моделей.* Необходимы средства, позволяющие исключать из рассмотрения модели, не отвечающие наперед заданным требованиям. Данные требования могут быть представлены, например, в виде множества правил, определяющих доступные или запрещенные связи между позициями и переходами синтезируемых СП-моделей, или в виде некоторых численных ограничений, отсекающих неудовлетворяющие им СП-модели. Набор подобных правил мог бы составить некоторую экспертную базу данных (базу знаний), позволяющую существенно сократить число СП-моделей, представляемых для окончательного анализа.

Список литературы

1. **Котов В. Е.** Сети Петри. М.: Наука, 1984.
2. **Питерсон Дж.** Теория сетей Петри и моделирование систем. М.: Мир, 1984.
3. **Кулагин В. П.** Проблемы анализа и синтеза структур параллельных вычислительных систем // Информационные технологии. 1997. № 1. С. 2—8.
4. **32nd International Conference on Application and Theory of Petri Nets and Concurrency**, 2011. URL: <http://www.jaist.ac.jp/acsd-petrinets2011/>.
5. **33rd International Conference on Application and Theory of Petri Nets and Concurrency (Petri Nets 2012)**, Hamburg, Germany, June 25—29, 2012. URL: <http://www.informatik.uni-hamburg.de/TGI/PetriNets/meetings/pn2012/>
6. **34th International Conference on Application and Theory of Petri Nets and Concurrency (Petri Nets 2013)**, Milano, Italy, June 24—28, 2013. URL: <http://www.mc3.disco.unimib.it/petrinets2013/>.
7. **Анкудинов Г. И.** Синтез структуры сложных объектов. Логико-комбинаторный подход. Л.: ЛГУ, 1986.
8. **Биркгоф Г.** Теория структур. М.: Иностранная литература, 1952.
9. **Скорняков Л. А.** Элементы теории структур. М.: Наука, 1982.
10. **Cabasino M. P., Giua A., Seatzu N.** Structural analysis of Petri nets // Control of Discrete-Event Systems. Lecture Notes in Computer Science. № 433. Springer-Verlag, London, 2013. P. 213—233.
11. **Кун С.** Матричные процессоры на СБИС: Пер. с англ. М.: Мир, 1991. 672 с.
12. **Сверхбольшие** интегральные схемы и современная обработка сигналов: Пер. с англ. / Под ред. С. Гуна, Х. Уайтхауса, Т. Кайлата. М.: Радио и связь, 1989. 472 с.
13. **Кулагин В. П., Дубинин В. Н.** Структурный анализ сетей Петри // Информационные технологии. 2016. № 1 (22). С. 3—13.
14. **Кулагин В. П.** Тензорные методы исследования структур сетей Петри // Информационные технологии. 2015. Т. 21, № 2. С. 83—94.
15. **Кулагин В. П.** Методы построения тензоров преобразования для сетевых моделей сложных систем // Информатизация образования и науки. 2015. № 4 (28). С. 133—147.
16. **Kulagin V.** Design of control systems for parallel computing structures based on net models // 2016 International Siberian Conference on Control and Communications (SIBCON). Proceedings. M.: HSE, 2016.
17. **Drozdov D., Dubinin V., Kulagin V.** Petri nets behavioral equivalence checking in SMV // 2016 International Siberian Conference on Control and Communications (SIBCON). Proceedings. M.: HSE, 2016.

V. P. Kulagin, D. Sc., Professor, Head of the Department "Hardware, software and mathematical support of computer systems", e-mail: Kulagin_VP@mgupi.ru,
E. S. Malych, Master Student of the Department "Hardware, software and mathematical support of computer systems"; e-mail: katya.miller.95@yandex.ru,
 MIREA-Russian University of Technology, Moscow

Design of Matrix Computing Structures using Petri Nets

In article on the example of projection of matrix computing structures the synthesis algorithm of the simulativng Petri nets is offered. Approach to synthesis of new structures on the basis of use of vectors of a projection is described. It is shown that this approach doesn't allow to receive all set of the possible Petri nets modeling realization of the function set on matrix structure. The original algorithm of projection of the complete set of the possible joint ventures models of matrix structures for the given function is described. The algorithm is based on use of a structure analysis and tensor methodology. Oscillation of programs of synthesis of structures of the joint venture models is result of work of an algorithm. It is shown that among the generated programs exist such which synthesize the equivalent joint ventures models. The properties revealing the "equivalent" programs are formulated. For the purpose of the quantitative and qualitative analysis of the constructed matrix structures criteria for evaluation of structural and dynamic properties of the synthesizable joint ventures models are injected. It is shown that owing to exponential body height of number of the synthesizable joint ventures models from dimension of an initial primitive system, searching of new receptions and approaches which would exclude from consideration of model is necessary, not meeting beforehand the given requirements.

Keywords: matrix structure, parallel computations, Petri nets, tensor analysis, program synthesis, evaluation of structural properties

DOI: 10.17587/it.25.271-283

References

1. **Kotov V. E.** Seti Petri, Moscow, Nauka, 1984 (in Russian).
2. **Piterson Dzh.** Teoriya setej Petri i modelirovanie sistem, Moscow, Mir, 1984.
3. **Kulagin V. P.** Problemy` analiza i sinteza struktur parallel`ny`x vy`chislitel`ny`x sistem, *Informacionny`e Tekhnologii*, 1997, no. 1, pp. 2–8.
4. **32nd International Conference on Application and Theory of Petri Nets and Concurrency**, 2011, available at: [//www.jaist.ac.jp/acsd-petrinets2011/](http://www.jaist.ac.jp/acsd-petrinets2011/).
5. **33rd International Conference on Application and Theory of Petri Nets and Concurrency (Petri Nets 2012)**, Hamburg, Germany, June 25–29, 2012, available at: [//www.informatik.uni-hamburg.de/TGI/PetriNets/meetings/pn2012/](http://www.informatik.uni-hamburg.de/TGI/PetriNets/meetings/pn2012/)
6. **34th International Conference on Application and Theory of Petri Nets and Concurrency (Petri Nets 2013)**, Milano, Italy, June 24–28, 2013, available at: [//www.mc3.disco.unimib.it/petrinets2013/](http://www.mc3.disco.unimib.it/petrinets2013/).
7. **Ankundinov G. I.** Sintez struktury` slozhny`x ob`ektov. Logiko-kombinatorny`j podxod. L.: LGU, 1986.
8. **Birkhof G.** Teoriya struktur. M.: Inostrannaya literatura, 1952.
9. **Skorniyakov L. A.** E`lementy` teorii struktur, Moscow, Nauka, 1982.
10. **Cabasio M. P., Giua A., Seatzu N.** Structural analysis of Petri nets, *Control of Discrete-Event Systems. Lecture Notes in Computer Science*, № 433, Springer-Verlag, London, 2013, pp. 217–233.
11. **Kun S.** Matrichny`e processory` na SBIS: Per. s angl, Moscow, Mir, 1991, 672 p.
12. **Sverxhol`shie integral`ny`e sxemy` i sovremennaya obrabotka signalov:** Per. s angl. / Pod red. S. Guna, X. Uajtxausa, T. Kajlata, Moscow, Radio i svyaz`, 1989, 472 p.
13. **Kulagin V. P., Dubinin V. N.** Strukturny`j analiz setej Petri, *Informacionny`e Tekhnologii*, 2016, no. 1 (22), pp. 3–13.
14. **Kulagin V. P.** Tenzorny`e metody` issledovaniya struktur setej Petri, *Informacionny`e texnologii*, 2015, T. 21, № 2, pp. 83–94.
15. **Kulagin V. P.** Metody` postroeniya tenzorov preobrazovaniya dlya setevy`x modelej slozhny`x sistem, *Informatizaciya obrazovaniya i nauki*, 2015, № 4 (28), pp. 133–147.
16. **Kulagin V.** Design of control systems for parallel computing structures based on net models, in: 2016 International Siberian Conference on Control and Communications (SIBCON). Proceedings, Moscow, HSE, 2016.
17. **Drozdov D., Dubinin V., Kulagin V.** Petri nets behavioral equivalence checking in SMV, in: 2016 International Siberian Conference on Control and Communications (SIBCON). Proceedings, Moscow, HSE, 2016.

УДК 001.89

DOI: 10.17587/it.25.283-292

Р. М. Юсупов, чл.-корр. РАН, науч. рук. института, e-mail: yusupov@iias.spb.su,
В. И. Воробьев, д-р техн. наук, гл. науч. сотр., e-mail: vvi@iias.spb.su,
М. Ю. Петров, вед. программист, e-mail: miha@iias.spb.su,
Санкт-Петербургский институт информатики и автоматизации РАН (СПИИРАН)

Представление эволюции средств вычислительной техники в экспозиции Музея истории СПИИРАН

Описана история развития средств вычислительной техники в СПИИРАН. Приведен обзор основных экспонатов: отдельных устройств и компонентов вычислительной техники. Экспозиция отражает отдельные этапы становления и развития вычислительной техники в стране и возникновения информатики как фундаментальной науки. Систематизирован опыт использования вычислительной техники в научных исследованиях.

Ключевые слова: экспозиция музея, средства вычислительной техники, математическое обеспечение ЭВМ, компьютерные сети, центр коллективного пользования

Введение

Появление средств вычислительной техники следует считать одним из важнейших достижений цивилизации, обусловивших переход от индустриального к информационному обществу. Создание научно-технического музея истории вычислительной техники направлено на формирование понимания основных тенденций развития информационных технологий от истоков до современного их состояния. Экспозиция музея предназначена для различных возрастных групп школьников, студентов, пе-

дагогов и научных работников. Она знакомит посетителей с важнейшими вехами развития информатики и средствами вычислительной техники на различных этапах их развития.

Музей истории СПИИРАН является одним из источников научно-образовательной деятельности в области истории информатики, сохранения и демонстрации средств вычислительной техники в северо-западном регионе России [1,2]. Коллекция содержит более 1000 объектов, представленных в экспозиции. Выставленные образцы отражают основные вехи метасистемного перехода в развитии средств вычислитель-

ной техники. Динамика развития такова, что раз в два года происходит удвоение сложности вычислительной установки (информационной системы), соответственно сокращается время возникновения новых типов систем [3].

Роль музея в популяризации достижений науки и техники на заре цифровой экономики трудно переоценить. Музей выполняет также культурно-нравственную функцию, осуществляя связь поколений, передачу знаний и опыта [4]. Экспонаты, представленные в хронологическом порядке, позволяют наглядно представить эпоху становления информационного общества.

Музеи вычислительной техники, реальные и виртуальные, широко представлены на просторах сети Интернет. Живой интерес к истории развития вычислительной техники неуклонно растет, что подтверждается развитием и увеличением числа музеев вычислительной техники, среди которых: Computer History Museum в Калифорнии [5], в котором представлены часть экспонатов тех же семейств и производителей, что и в музее СПИИРАН; музей науки Великобритании [6], экспозиция которого включает отечественную ЭВМ БЭСМ-6; вропейский виртуальный компьютерный музей (ЕСМ) [7], позволяющий раздвинуть рамки экспозиции любого музея; обновленная экспозиция в Политехническом музее в Москве [8]. В сети Интернет множество материалов представлено фирмами-производителями и частными коллекциями [5–18]. Современные информационно-вычислительные системы неразрывно связаны с сетями передачи данных, поэтому важна история их развития и демонстрация устройств [9].

Любой современный исследовательский центр рано или поздно сталкивается с дефицитом, неполнотой и утратой информации. На данный момент наши знания о достижениях в области вычислительных и коммуникационных технологий дискретны и не всегда достоверны. Очевидно, эта область науки уже требует обобщения, классификации и анализа накопленных знаний, понимания динамики развития в целях установления эмпирических связей и соотношений, что позволит впоследствии выявить закономерности развития.

Первым шагом на этом пути является создание научного, общедоступного, многопланового музея вычислительной техники, информационных и коммуникационных систем, который предназначен для решения следующих задач:

- повышения общеобразовательного, культурного уровня и компьютерной грамотности;

- стимулирования интереса к вопросам практического повсеместного использования компьютерных средств;
- сохранения историко-культурного наследия общества, связанного с созданием и развитием средств работы с одним из важнейших национальных богатств общества — информационными ресурсами;
- пополнения, восстановления и сохранения информации об утраченных экспонатах — вычислительных машинах различных моделей и сопутствующих им устройствах, в частности периферийных, и средствах связи.

История института — предпосылки для создания коллекции

Сбор коллекции музея начался в 2000 г., когда при проведении ремонта помещений сотрудниками были сохранены интересные экземпляры вычислительной техники. В последующие несколько лет был продолжен целенаправленный сбор экспонатов, и в 2008 г. директором СПИИРАН членом-корреспондентом РАН Р. М. Юсуповым было принято решение о создании музея института. В ходе работы над экспозицией сотрудники стремились сохранить и показать ушедшие в историю устройства. Собранные и систематизированы экспонаты, отражающие сорокалетнюю историю института, связанную с развитием вычислительной техники в стране и формированием информатики как фундаментальной науки.

История коллекции музея неразрывно связана с историей создания 7 октября 1974 г. Отдела вычислительной техники физико-технического института (далее ЛНИВЦ, ЛИИАН и СПИИРАН), руководителем которого был назначен д-р техн. наук профессор В. М. Пономарев.

Мощь государства определяется не только добываемыми ресурсами или количеством выработанной энергии, но и научной мощью, включающей вычислительные и информационные ресурсы. К 1974 г. в Ленинграде назрела критическая ситуация, связанная с острой нехваткой вычислительных мощностей. Ресурса ЛО ЦЭМИ АН СССР, который пытался взять на себя роль вычислительного центра коллективного пользования, явно не хватало. В Москве и Новосибирске уже работали системы коллективного пользования. Встала такая задача и в Ленинграде [19]. Пакетный режим, в котором эксплуатировались ЭВМ, обеспечивал

лишь более или менее эффективную загрузку, но время решения задачи, т. е. время отладки и получения содержательного результата, составляло месяцы, а для некоторых задач и годы.

Перед коллективом ЛВЦ (отдел ФТИ), руководимым д-ром техн. наук В. М. Пономаревым, была поставлена задача создания вычислительных ресурсов в виде системы коллективного пользования и далее, на ее базе, разработки системы автоматизации научных исследований.

Первая проблема — установка мощных ЭВМ — была сложной инженерной задачей, с которой справлялись не все институты. Так, один из институтов АН СССР, получив дефицитнейшее оборудование (БЭСМ-6), не сумел запустить его в работу, а молодой коллектив ЛВЦ успешно с этим справился. В 1975 г. были введены в эксплуатацию МИР-2 и М-6000, первая ЭВМ БЭСМ-6 (рис. 1, *а—в*, см. третью сторону обложки), в 1976 г. — вторая ЭВМ БЭСМ-6, которые сразу были объединены в многомашинный вычислительный комплекс, что позволило на некоторое время удовлетворить потребности ленинградских учреждений АН СССР.

Эффективность решения задач определялась не только мощностью процессора, но и математическим обеспечением ЭВМ, включающим операционную систему, набор компиляторов для языков программирования (ас-семблеры, АЛГОЛ, ФОРТРАН, PL-1, SIMULA, APL, GPSS и др.), библиотек стандартных программ, СУБД и средств автоматизации программирования. За короткое время были установлены и сданы в эксплуатацию практически все компоненты программного и математического обеспечения, которые были доступны и работоспособны, включая библиотеки стандартных программ общего пользования: IBM, МГУ, Уилкинсона и др.

Анализ отечественного и зарубежного рынка вычислительной техники того времени показал, что наиболее перспективным следовало считать приобретение современной вычислительной системы высокой производительности, ориентированной на использование в научных исследованиях и работу в системе коллективного пользования. На выпуске систем такого класса специализировалась американская фирма ControlData Corporation.

Следующим шагом по расширению ресурсов и внедрению новых сервисов была установка в 1978 г. вычислительной системы CYBER-172-6 фирмы CDC (рис. 2, *а*, см. третью сторону обложки), которая отличалась от отечественных ЭВМ рациональной и сбалансиро-

ванной архитектурой, большим разнообразием и качественным периферийным оборудованием и развитым математическим обеспечением. В частности, были установлены и успешно эксплуатировались библиотека MSFLIB, которая содержала большое число уникальных алгоритмов и программ, а также система управления базами данных (DMS-170). Система CYBER включала кроме центрального процессора 10 периферийных процессоров и, что особенно важно, сетевой процессор. Данные особенности архитектуры позволяли повысить производительность, поскольку центральный процессор занимался только решением задач пользователей. Наличие сетевого процессора открывало возможности для создания компьютерных сетей. Первые эксперименты по установлению межкомпьютерных связей для создания компьютерных сетей и подключения ресурсов европейских компьютерных центров были проведены с Будапештом (ИИВТА) и с Финляндией (Технический исследовательский центр Финляндии). В этом направлении институт был одним из лидеров в стране. Успехи института в настоящее время обусловлены в том числе своевременным выбором руководством института сетевых технологий.

Дальнейшим шагом в развитии системы коллективного пользования стала установка в 80-е годы ЕС-1052 (рис. 1, *г*, см. третью сторону обложки), ЕС-1060, объединенных в единый многомашинный комплекс, СМ ЭВМ и развитие терминальной сети. Удаленные терминалы были установлены и подключены по телефонным проводам более чем в 50 территориально разнесенных учреждениях. В качестве сетевых ЭВМ использовались компьютеры серии СМ ЭВМ, которые также применялись для автоматизации эксперимента в академических институтах, ФИН, ФТИ и др., а также для управления производственными системами. В отличие от системы CYBER, в которой для обеспечения сетевых телекоммуникаций использовался метод коммутации каналов, отечественные компьютеры объединялись на базе метода коммутации пакетов, что и было использовано в проекте АКАДЕМСЕТЬ, реализованном в 1985 г. С этого момента ЛНИВЦ полностью взял на себя выполнение заявок академических учреждений на выполнение вычислительных работ.

С помощью вычислительной техники и математического и программного обеспечения решались научные задачи в разных областях науки, таких как: физика плазмы, твердого тела и газовой динамики; квантово-механические

расчеты электронных структур молекул, спектров и структур полимерных систем; расчеты эфемерид астрономии и небесной механики; планетарная геофизическая гидродинамика теории климата; математическое моделирование фитоценозов; прогноз продуктивности в экосистемах; моделирование системы "Человек — Биосфера"; создание региональных моделей рационального использования ресурсов и т. д.

В 1990-х гг. на волне всеобщего увлечения персональными ЭВМ и идолопоклонства перед западными технологиями институт сохранил накопленный опыт и обеспечил последовательный переход к современным информационным технологиям.

Способ коллективного использования информационных и вычислительных ресурсов в современном мире приобрел новые более развитые формы организации в концепции Data-центров, предоставляющих облачные сервисы. Идеи, косвенно соотносящиеся с тем, что используются в технологиях облачных вычислений, и описывающие возможность вычислений с использованием удаленных вычислительных центров, в том числе концепция виртуальных машин [20], зародились еще в 70...80-х годах прошлого века. В условиях дороговизны, дефицита оборудования и необходимости обслуживания техники высококвалифицированным инженерным персоналом существовала практика создания вычислительных центров коллективного пользования (ВЦКП) [21, 22]. Именно тогда разработчики программного обеспечения предложили модель приложений, при которых все вычисления и обработка информации осуществляются не на компьютере пользователя, а на удаленных серверах. Однако глобальной сети в то время не существовало, поэтому первые идеи "облаков" оказались трудно реализуемыми и практически не использовались при создании новых программ.

Тематические разделы коллекции музея

В коллекции музея можно выделить несколько тематических разделов, показывающих основные направления развития средств вычислительной техники.

1. Электронно-вычислительные машины разных поколений

Вычислительный комплекс CYBER-172-6 (рис. 2, а, см. третью сторону обложки) пред-

ставляет тот период развития, когда вычислительные машины были большими (1960...1980 гг.). Основу элементной базы комплекса составляют интегральные микросхемы низкой и средней степени интеграции. Комплекс состоит из центрального процессора с оперативной памятью, консоли оператора, устройств управления периферийными устройствами и самих периферийных устройств (устройств ввода-вывода). Устройствами ввода-вывода являются накопители на магнитных дисках, накопители на магнитных лентах, удаленные терминалы, принтеры, устройства ввода и вывода информации на перфокарты. Важным моментом для того времени является наличие сетевого процессора, работающего по принципу коммутации каналов, что позволило проводить работы по взаимодействию удаленных машин. Комплекс обладал весьма высокой по тому времени (1978 г.) производительностью порядка 1 млн операций в секунду. Комплекс был построен по блочно-модульному принципу, что позволяло увеличивать или уменьшать его состав в зависимости от решаемых задач, а также способствовало быстрой диагностике неисправностей и их устранению.

Программное обеспечение комплекса состояло из сбалансированной сетевой операционной системы (NOS), компиляторов языков программирования Фортран, Алгол, APL, Лисп, Симула, комплекса библиотек научных программ, системы управления базами данных и позволяло решать широкий круг инженерных и научных задач. В частности, в 1982 г. были проведены расчеты о влиянии строительства комплекса защитных сооружений Ленинграда от наводнений на экологическое состояние "Маркизовой лужи".

В экспозиции музея комплекс представлен следующими устройствами:

- центральный процессор, который представляет собой металлический шкаф (габаритные размеры 85 × 230 см, высотой 2 м), в котором смонтированы четыре поворотные рамы, блоки преобразователей питания и компрессор холодильной установки. В рамы вставлены платы с электронными компонентами (типовые элементы замены — ТЭЗ) — 17 рядов, по 42 платы в каждом ряду (рис. 2, б, см. третью сторону обложки). С обратной стороны рамы платы соединены между собой проводами методом навесного монтажа (рис. 2, в, см. третью сторону обложки). Каждая плата представляет собой отдельный узел-компонент, из которых состоит

центральный процессор, например: плата-регистр, плата-сумматор, плата-сегмент оперативной памяти, плата-тактовый генератор. Во всем процессоре имеется всего около 70 различных видов плат. Для удобства ремонта неисправных узлов прилагался комплект запасных плат по одной каждого вида для того, чтобы можно было временно заменить неисправную плату, запустить весь комплекс в работу, а неисправную плату починить с помощью специального устройства — тестера;

- консоль оператора, которая представляет собой тумбу-контроллер устройства. На тумбе размещен графический дисплей;

- накопитель на магнитных дисках (рис. 3, б, см. четвертую сторону обложки), представляющий собой ящик-тумбу с размещенным внутри контроллером устройства и шпинделем для установки сменного пакета магнитных дисков. Над поверхностью каждого диска находится магнитная головка, которая читает или записывает данные на концентрическую дорожку. Все головки имеют возможность перемещаться по поверхности дисков к центру и обратно, что позволяет создать на дисках множество дорожек. Таким образом, общая информационная емкость (объем) всего пакета дисков состоит из объема одной дорожки, умноженного на число дорожек и еще раз умноженного на число дисков (точнее их поверхностей).

Несмотря на наличие очень сложного и точного механизма устройство обладает высокой надежностью и скоростью работы, поэтому похожие устройства (только уменьшенных габаритов) используются в современных компьютерах до сих пор (рис. 3, а, см. четвертую сторону обложки);

- накопитель на магнитных лентах (рис. 3, в, см. четвертую сторону обложки), представляющий собой шкаф, в котором находится магнитофон (внешне похожий на бытовой) и устройство управления. В магнитофон устанавливается катушка с магнитной лентой. Ширина магнитной ленты позволяет организовать на ней несколько одновременных дорожек (в разных моделях от 5 до 11). Длина магнитной ленты составляет несколько сотен метров (до 650), что позволяет записать на нее огромный объем информации. Основной (и весьма существенный) недостаток заключается в невысокой скорости доступа (до нескольких минут) к информации. Поэтому подобные устройства использовались в основном для хранения архивов данных;

- удаленный терминальный комплекс (пакетный терминал) (рис. 3, з, см. четвертую сторону обложки) — рабочее место пользователя — представляет собой стол-тумбу с размещенным внутри нее контроллером удаленного терминала. На столе размещены модем для связи с ЭВМ, дисплей, устройство ввода информации с перфокарт, графопостроитель, алфавитно-цифровое печатающее устройство.

Появление интегральных микросхем высокой и сверхвысокой степени интеграции позволило "упаковать" весь процессор в одну микросхему, соответственно, существенно уменьшились размеры вычислительной техники. Началась эра персональных компьютеров.

В экспозиции музея класс персональных ЭВМ является самым многочисленным. Наибольший интерес представляет легендарная машина — "IBM PC XT", которая является первой массовой персональной ЭВМ. Фактически, эта машина произвела переворот в информатизации общества, поскольку для большинства современников являлась первой доступной ЭВМ. В музее имеется также машина "Правец-16", которая является совместимым аналогом IBM PC XT и была широко распространена в нашей стране.

По мере развития аппаратных возможностей вычислительных машин и создания программного обеспечения для решения самых разнообразных задач появлялись специализированные ЭВМ, предназначенные для решения определенного класса задач. К таким задачам, в частности, относится компьютерная обработка изображений, а предназначенные для этого машины называются "графическими станциями". В экспозиции представлены графические станции таких известных фирм, как "Control Data Corporation" (Silicon Graphics), "Sun Microsystems" (Sparc), "Apple" (Macintosh).

Появление однокристалльных микропроцессоров не обошли стороной и радиолюбители. В различных журналах печатались схемы домашних персональных компьютеров. Нередко такая разработка представляла собой приставку к телевизору (рис. 4, а). На корпусе приставки размещалась клавиатура, под клавиатурой находилась плата с электроникой. Программное обеспечение таких ЭВМ, как правило, оставляло желать лучшего и, в основном, ограничивалось наличием пары десятков компьютерных игр и одного или двух языков программирования для проведения несложных инженерных расчетов, вследствие чего эти машины

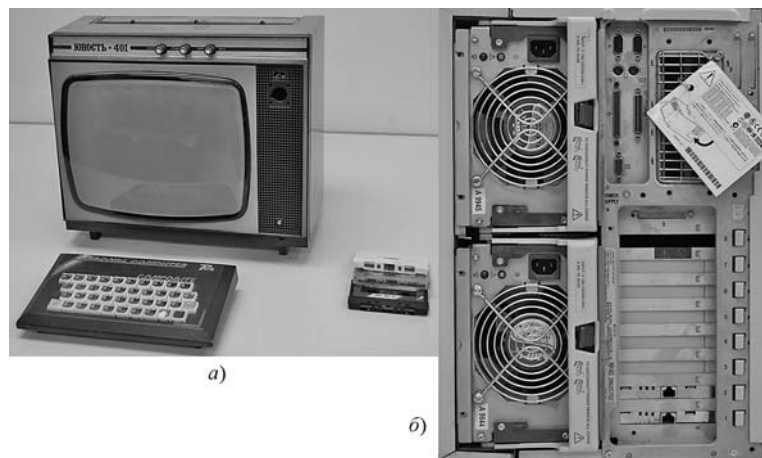


Рис. 4. "Домашний компьютер" и дублирование в серверах

заняли нишу так называемых игровых приставок. В экспозиции музея такие компьютеры также представлены.

Развитие сетевых возможностей и появление сети Интернет привело к созданию особого, специализированного класса машин — так называемых серверных ЭВМ. Эти машины полностью предоставляют свои ресурсы пользователям других машин через сеть. Работа пользователя непосредственно на консоли такой машины возможна, но не желательна. Отличительной особенностью таких машин является наличие ресурсов, существенно превосходящих возможности обычных персональных ЭВМ. Такими ресурсами могут быть дисковое пространство (хранилище) большого объема для серверов баз данных или множество процессоров для вычислительных серверов. Кроме того, создатели серверов обращают особое внимание на надежность работы такой техники. В частности, дублируются блоки электропитания, каналы ввода-вывода (рис. 4, б). В экспозиции музея представлена серверная ЭВМ "Corollary", содержащая в себе несколько процессоров и предназначенная для выполнения большого объема вычислительной работы.

2. Элементная база

Создание электронно-вычислительной машины стало возможным лишь тогда, когда для этого появилась соответствующая элементная база. В начале XX века была изобретена электронная лампа и открыты полупроводниковые явления. Стремительное развитие электроники, а также математической основы и, в частности, теории алгоритмов в 1920—1940-х гг. сделали возможным создание электронной вычис-

лительной машины. Первая ЭВМ была построена в 1946 г. и содержала 18 тыс. электронных ламп. А к 1952 г. в ведущих научных странах было создано уже около двух десятков разных машин. Большие размеры электронных ламп и значительное энергопотребление ограничивали возможности для дальнейшего развития. И в 1955 г. были созданы первые "транзисторные" ЭВМ. В 1960-х гг. появились интегральные схемы, и развитие пошло по пути миниатюризации. Уменьшение размеров и энергопотребления электронных устройств позволило развивать функциональные возможности ЭВМ.

В экспозиции музея представлены электронные лампы, транзисторы, интегральные микросхемы разной степени интеграции, использовавшиеся в вычислительной технике тех лет (рис. 5, а, см. четвертую сторону обложки). В экспозиции имеются также различные узлы и устройства (электронные платы), созданные с использованием перечисленных компонентов (рис. 5, б—г, см. четвертую сторону обложки).

3. Копировально-множительная техника и средства ввода-вывода информации

Самые первые ЭВМ имели крайне примитивные средства ввода-вывода. Информация вводилась побайтно в двоичном виде с переключателей, а выводилась также побайтно на линейки лампочек. Для пользователя ЭВМ это было крайне неудобно. В свою очередь, копировально-множительная техника исторически развивалась как самостоятельное направление технического прогресса. Но на определенном этапе развития выяснилось, что эта техника хорошо подходит для осуществления функций ввода и вывода информации из ЭВМ. Ведь под клавиши механической печатной машины можно поставить микропереключатели, и через специальный шифратор вводить информацию о нажатой клавише непосредственно в ЭВМ. А молоточками печатной машины можно управлять из ЭВМ с помощью электромагнитов. С этого момента развитие вычислительных машин и средств печати все чаще и чаще пересекалось. Современные печатные машины содержат в себе множество контроллеров и управляются полноценными ЭВМ.

Аналогично механическим печатным машинам связаны с ЭВМ и другие средства ввода-вывода информации. Информация о состоянии

устройства поступает в ЭВМ с микропереключателями или других датчиков, использующих самые разнообразные физические принципы (датчики Холла, фототранзисторы, пьезо- и тензодатчики), а работа механизмов управляется с помощью электромагнитов, реле или электронных усилителей сигналов.

В коллекции музея имеются механические и электрические печатные машины, телетайп, матричные, струйные, лазерные принтеры. Кроме того, представлены: графопостроитель (рис. 3, з, см. четвертую сторону обложки), устройства ввода информации с перфокарт и перфоленты, устройство вывода на перфоленту, световое перо, манипуляторы типа мышь, устройства хранения информации на магнитных дисках и магнитных лентах.

4. Средства для выполнения арифметических вычислений

В коллекции музея имеются арифмометр, простые и программируемые калькуляторы, логарифмические линейки, конторские счета.

5. Средства связи и телекоммуникации

С появлением первых ЭВМ перед инженерами возникли две проблемы. Первая заключалась в том, что некоторые задачи решались непозволительно долго даже на самых мощных по тем временам машинах. Для ее решения стали создавать средства комплексирования, которые позволяли объединить мощности двух или более ЭВМ в единый вычислительный комплекс. Вторая проблема заключалась в том, что первые машины занимали значительную площадь и часто находились в отдельных, специально построенных для этого зданиях, так называемых вычислительных центрах (ВЦ). Для решения задач пользователи были вынуждены "наносить визит" в эти ВЦ, что было очень неудобно и занимало много времени. Для облегчения пользователю доступа к ЭВМ были созданы так называемые "терминалы", связанные с ЭВМ с помощью модемов через телефонные линии. В отдельных случаях связь с ЭВМ осуществлялась через спутниковые каналы.

По мере уменьшения габаритных размеров вычисли-

тельных машин и увеличения их числа появились средства для построения локальных вычислительных сетей — сетевые адаптеры, хабы, трансиверы — преобразователи среды передачи данных. Появление и развитие сети Интернет привело к необходимости создания специальных роутеров, спутниковых устройств связи.

В коллекции музея представлены разнообразные терминалы, синхронные и асинхронные модемы с различной скоростью передачи данных (рис. 6, б), сетевые устройства (платы, коммутаторы), телепроцессоры для построения магистральной опорной сети, оборудование спутниковой связи.

6. Устройства для наладки и контроля средств вычислительной техники

ЭВМ — это очень сложный комплекс электронных устройств с большим числом самых разнообразных компонентов и деталей. Компоненты первых ЭВМ обладали не очень высокой надежностью и значительным разбросом параметров. Для того чтобы добиться четкой и слаженной работы этого комплекса использовались специальные настроечные устройства — тестеры, позволявшие контролировать правильность работы отдельных компонентов системы и облегчавшие их ремонт и настройку.

В коллекции музея представлены тестеры для настройки разнообразных устройств ввода-вывода (рис. 6, а), а также средства измерительной техники — вольтметры, амперметры, осциллографы, частотомеры, генераторы стандартных и специальных сигналов, использовавшиеся при настройке устройств вычислительной техники (рис. 6, в).

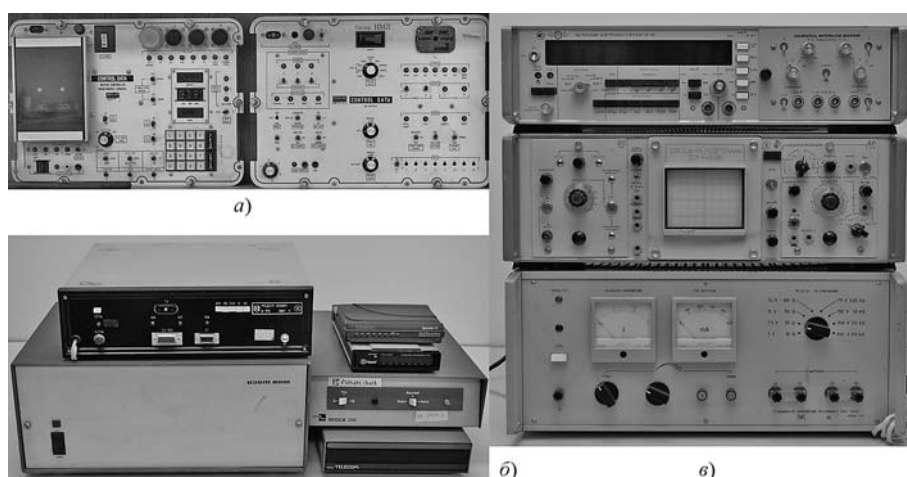


Рис. 6. Модемы и устройства для наладки и контроля средств вычислительной техники

7. Вспомогательная радиоэлектронная аппаратура и устройства

В коллекции музея имеются различные экспонаты, не имеющие, на первый взгляд, прямого отношения к вычислительной технике. Однако без них эксплуатация средств вычислительной техники была затруднена, а в отдельных случаях и невозможна. К таким экспонатам относятся:

- бытовые телевизоры (рис. 4, а) и магнитофоны, которые использовались в комплекте с домашним игровым компьютером (нередко собранным самостоятельно). Телевизор являлся средством отображения информации, магнитофон использовался для хранения программ и данных;
- установка пожаротушения, предназначенная для тушения возможного возгорания в помещении, где находятся средства вычислительной техники. Состоит из набора датчиков дыма, электронного блока управления и баллона с фреоном (рис. 7, а). В случае срабатывания датчика фреон из баллона выпускался, и газ за счет расширения резко снижал температуру в помещении (помещение фактически замораживалось), а кислород в помещении замещался негорючим газом, что способствовало прекращению горения;
- установка климатконтроля. Вычислительная техника весьма чувствительна к параметрам температуры и влажности. Высокая температура приводит к преждевременному выходу из строя электронных компонентов, а высокая влажность — к окислению контактов и плохой работе механических устройств. Установка климатконтроля (рис. 7, б) позволяет контролировать параметры температу-

ры и влажности и, в случае выхода их за пределы допустимого, выдает сигнал оператору;

- дырокол для перфоленты, приспособление для контроля правильности набивки перфокарт. "Когда машины были большими", подготовка данных для решения на ЭВМ выполнялась с помощью специальных устройств "подготовки данных"; это было долгим процессом, а при наборе допускалось много ошибок. Перфокарты и перфоленты, использовавшиеся тогда в качестве носителей информации, допускали "ручное" чтение информации, чем и занималось большинство пользователей, вручную проверяя правильность набора данных, и также вручную их корректировали;
- источники бесперебойного питания (ИБП). Даже кратковременное отключение питания приводит к пропаданию информации в оперативной памяти ЭВМ. ИБП поддерживает работу ЭВМ в течение получаса даже при полном отключении питания, что позволяет сохранить все данные и корректно завершить работу;
- аппарат для чтения микрофишей (рис. 7, в). Любая ЭВМ — это очень сложный комплекс электронных, механических устройств и программного обеспечения. Документация, описывающая работу вычислительной машины и ее составных частей, состоит из десятков, а иногда и сотен томов и занимает много места. Для сокращения физического объема документации использовались так называемые "микрофиши" — уменьшенные фотографии бумажных документов. Одна фотокарточка размерами 9 на 12 см содержит до 24 страниц текста формата А4;
- устройство для чистки магнитной ленты. Наличие пыли на магнитной ленте и магнитных дисках приводит к ошибкам при чтении и записи. Для очистки магнитной ленты от пыли использовалось специальное устройство;

чтении и записи. Для очистки магнитной ленты от пыли использовалось специальное устройство;

- установочные комплекты пакетов прикладных программ для персональных ЭВМ. Комплекты включают носители (дискеты) с самими программами, а также описание программ и руководство по их использованию на бумажных носителях. Такие комплекты создавались и использовались до повсеместного развития сети Интернет.

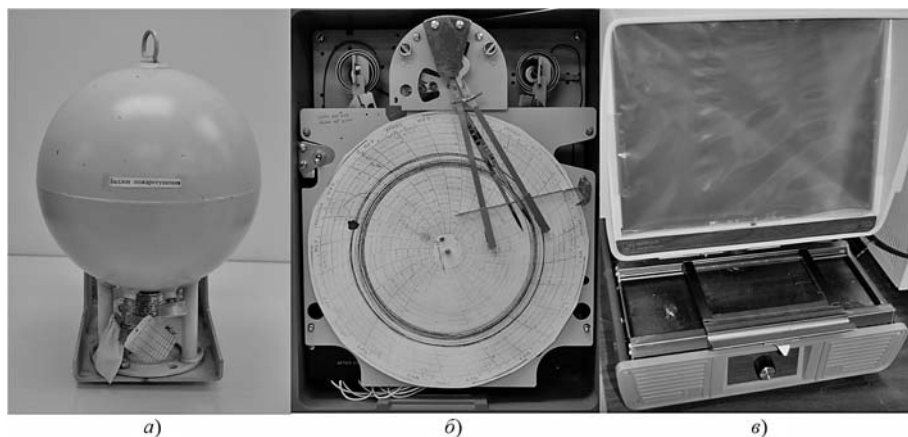


Рис. 7. Вспомогательная аппаратура и устройства

Заключение

Экспозиция музея направлена на формирование понимания места и роли средств вычислительной техники в жизни современного общества. Она отображает реальную жизнь и деятельность института. Историческое наследие является отправной точкой создания информационных технологий будущего.

Развитие музея планируется в следующих направлениях: дальнейшее пополнение и структуризация коллекции, использование робототехнических средств для ведения экскурсий, применение технологий искусственного интеллекта для выбора требуемой траектории конкретной экскурсии с учетом уровня осведомленности посетителей, их интересов и предпочтений. Важным элементом планируемой системы обслуживания посетителей будет интерактивная связь с близкими по тематике музеями, включающая словари, репозиторий метаданных описания коллекции, позволяющие реализовать такие функции, как запрос, поиск, управление и взаимосвязь между словарями и метаданными. Через репозиторий пользователь сможет искать и использовать словари или наборы данных на основе разных критериев и ключевых слов, оценивать запросы SPARQL, находить связанные словари, загружать дампы данных и многое другое. Метаданные могут значительно упростить и улучшить процессы сбора, интеграции и анализа источников данных об экспонатах.

В любой самостоятельной области науки рано или поздно наступает этап структуризации. В области информационных технологий этот процесс активно происходит в настоящее время. Параллельно из разрозненных во времени и пространстве фрагментов складывается целостная история развития, в обобщенном и агрегированном виде передаваемая последующим поколениям. Одним из таких фрагментов является музей истории СПИИРАН.

Список литературы

1. Юсупов Р. М. Из истории развития информатики и информационных технологий в Санкт-Петербурге // Научно-технические ведомости Санкт-Петербургского государственного

политехнического университета. Информатика. Телекоммуникации. Управление. 2011. Т. 6-2. № 138. С. 184—190.

2. Юсупов Р. М., Заболотский В. П. Концептуальные и научно-методологические основы информатизации. СПб.: Наука, 2009. 544 с.

3. Потапов А. С. Искусственный интеллект и универсальное мышление. СПб.: Политехника, 2012. 711 с.

4. Серебрякова Е. В музей — сегодня, в науку — завтра // Санкт-Петербургский вестник высшей школы. 2018. № 4(137).

5. Computer History Museum в Калифорнии. URL: <http://www.computerhistory.org/>

6. Музей науки Великобритании. URL: <https://www.sciencemuseum.org.uk/>

7. The Virtual Museum of Manchester Computing. URL: <http://www.computer50.org/kgill/index.html>

8. Политехнический музей в Москве. URL: <https://poly-mus.ru/ru/>

9. История и классификация глобальных сетей. URL: <https://sites.google.com/site/komptellek/istoria-i-klassifikaciya-globalnyh-setej>

10. Все о старых компьютерах. URL: <http://www.danbigras.ru/>

11. The Virtual Museum of Computing (VMoC). URL: <http://www.historisches-centrum.de/vlmp/computing.html>

12. Музей информатики, Франция. URL: <http://mo5.com/>

13. Музей калифорнийского колледжа. URL: <http://www.computer-museum.org/>

14. Когда и кем был придуман первый арифмометр? URL: <https://eponim2008.livejournal.com/206281.html>

15. История развития ЭВМ. URL: https://ru.wikibooks.org/wiki/История_развития_ЭВМ

16. История информатики и кибернетики в Санкт-Петербурге (Ленинграде). Вып. 1. Яркие фрагменты истории / Под общ. ред. чл.-кор. РАН Р. М. Юсупова; составитель М. А. Вус; Ин-т информатики и автоматизации РАН. СПб.: Наука, 2008. 356 с. (URL: <http://window.edu.ru/resource/804/74804/files/sbornik1.pdf>)

17. Колин К. К. Информатика как наука: История и перспективы развития // Открытое образование 2011. № 6. С. 77—88. URL: <https://cyberleninka.ru/article/v/informatika-kak-nauka-istoriya-i-perspektivy-razvitiya>

18. Виртуальный компьютерный музей. URL: <http://computer-museum.ru/>

19. История СПИИРАН — 40 лет научной деятельности / Под общ. ред. Р. М. Юсупова, А. Л. Ронжина. СПб.: СПИИРАН, 2017.

20. Прат Т. Языки программирования. М.: Мир, 1979. 574 с.

21. История информационных технологий в СССР / Под общей ред. Ю. В. Ревича. М.: Книма (ИП Берег Е. В.), 2016. 416 с.

22. Азеев А. А., Саприн В. В., Чеботарев Ю. А. Организация и функционирование вычислительного центра. М.: Статистика, 1977. 216 с.

R. M. Yusupov, Corr. RAS, Scientific Head of Institute, e-mail: yusupov@iias.spb.su,
V. I. Vorobiev, D. Sc., Chief Researcher, e-mail: vvi@iias.spb.su,
M. Yu. Petrov, Leading Programmer, e-mail: miha@iias.spb.su,
St. Petersburg Institute of Informatics and Automation RAS (SPIIRAN)

Presentation of Computer Evolution in the Exposition of SPIIRAS History Museum

The paper gives consideration to the history of the development of SPIIRAS computer facilities. The main artifacts such as separate devices and components of computer equipment are supplied by an overview. The exposition shows the stages of formation and development of computer technology in the country and of the emergence of informatics as a fundamental science. The experience of using computer technologies in scientific research is aggregated.

Keywords: museum exposition, computer hardware, computer software, computer networks, data center

DOI: 10.17587/it.25.283-292

References

1. **Yusupov R. M.** *Iz istorii razvitiya informatiki i informacionnyh tekhnologij v Sankt-Peterburge* (The history of Informatics and information technologies in St. Petersburg), *Nauchno-tekhnicheskie vedomosti Sankt-Peterburgskogo gosudarstvennogo politekhnicheskogo universiteta. Informatika. Telekommunikacii. Upravlenie*, 2011, vol. 6-2, no. 138, pp. 184–190 (in Russian).
2. **Yusupov R. M., Zabolotskij V. P.** *Konceptual'nye i nauchno-metodologicheskie osnovy informatizacii* (Conceptual, scientific and methodological foundations of informatization), SPb, Nauka, 2009, 544 p. (in Russian).
3. **Potapov A. S.** *Iskusstvennyj intellekt i universal'noe myshlenie* (Artificial intelligence and universal thinking), SPb, Politehnika, 2012, 711 p. (in Russian).
4. **Serebryakova E.** *V muzej — segodnya, v nauku — zavtra* (To the museum — today, to science — tomorrow), *Sankt-Peterburgskij Vestnik Vysshej Shkoly*, 2018, no. 4(137) (in Russian).
5. **Computer History Museum in California**, available at: <http://www.computerhistory.org/>
6. **The science museum of Great Britain**, available at: <https://www.sciencemuseum.org.uk/>
7. **The Virtual Museum of Manchester Computing**, available at: <http://www.computer50.org/kgill/index.html>
8. **Polytekhnical museum in Moscow**, available at: <https://polymus.ru/ru/> (in Russian).
9. **History and classification of global networks**, available at: <https://sites.google.com/site/komptellek/istoria-i-klassifikacija-globalnyh-setej> (in Russian).
10. **All about old computers**, available at: <http://www.danbig-ras.ru/> (in Russian).
11. **The Virtual Museum of Computing (VMoC)**, available at: <http://www.historisches-centrum.de/vlmp/computing.html>
12. **Museum of Informatics, France**, available at: <http://mo5.com/> (in Russian).
13. **San Diego Computer Museum**, available at: <http://www.computer-museum.org/> (in Russian).
14. **When and by whom was the first arithmometer invented?** available at: <https://eponim2008.livejournal.com/206281.html> (in Russian).
15. **History of computer development**, available at: https://ru.wikibooks.org/wiki/История_развития_ЭВМ (in Russian).
16. **Yusupov R. M.** ed. *Istoriya informatiki i kibernetiki v Sankt-Peterburge (Leningrade). Vyp.1. YArkie fragmenty istorii. In-t informatiki i avtomatizacii RAN*, SPb., Nauka, 2008, 356 p., available at: <http://window.edu.ru/resource/804/74804/files/sbornik1.pdf> (in Russian).
17. **Kolin K. K.** *Informatika kak nauka: Istoriya i perspektivy razvitiya, Otkrytoe Obrazovanie*, 2011, no. 6, pp. 77–88, available at: <https://cyberleninka.ru/article/v/informatika-kak-nauka-istoriya-i-perspektivy-razvitiya> (in Russian).
18. **Russian virtual computer museum**, available at: <http://computer-museum.ru/> (in Russian).
19. **Yusupov R. M., Ronzhin A. L.** ed. *Istoriya SPIIRAN — 40 let nauchnoj deyatel'nosti* (SPIIRAS history — 40 years of scientific activity), SPb., SPIIRAN, 2017 (in Russian).
20. **Prat T.** *Yazyki programmirovaniya* (Programming language). Moscow, Mir, 1979, 574 p. (in Russian).
21. **Yu. V. Revich** ed. *Istoriya informacionnyh tekhnologij v SSSR* (History of information technology in the USSR), Moscow, Knima (IP Brege E. V.), 2016, 416 p. (in Russian).
22. **Azeev A. A., Saprin V. V., Chebotarev Yu. A.** *Organizaciya i funkcionirovanie vychislitel'nogo centra* (The organization and operation in the computer data center.), Moscow, Statistika, 1977, 216 p. (in Russian).

Н. А. Грушо, канд. физ.-мат. наук, e-mail: info@itake.ru,
Е. Е. Тимонина, д-р техн. наук, проф., e-mail: eltimon@yandex.ru,
Институт проблем информатики, Федеральный исследовательский центр
"Информатика и управление" Российской академии наук, г. Москва

Сравнение архитектур многоагентных систем¹

Эксплуатация новых или существующих информационных систем связана с необходимостью оперативного получения данных об инцидентах, возникающих в них, а также с оперативным выявлением причин этих инцидентов. Если информационные системы имеют множество узлов, а также присутствуют связи информационных систем между собой, то журналы событий, накапливаемые за небольшой промежуток времени, могут иметь огромное число записей. Естественно, что администраторы информационных систем не в состоянии за короткий промежуток времени проводить полный и подробный анализ таких журналов. В данной работе рассматриваются архитектуры многоагентных систем, осуществляющие сбор данных об инцидентах в информационных системах, в целях их дальнейшего автоматического анализа.

Ключевые слова: агенты, многоагентные системы, архитектуры многоагентных систем, информационные технологии

Введение

Цифровая экономика подразумевает разработку и внедрение большого числа информационных систем (ИС), а также организацию связи между ними. Естественно, что по многим причинам могут возникать сбои в работе ИС. Сбои могут влиять на ИС локально, а могут приводить к глобальным сбоям, если речь идет о взаимосвязанных ИС. Для устранения таких сбоев необходимо оперативно определять источник сбоев. Это возможно, если имеется система, которая собирает данные с элементов ИС о запущенных процессах, действиях пользователей, сетевой активности, записях в журналах операционных систем и т. д. Активная работа пользователей или программного обеспечения, выполняющего основной функционал ИС, порождает большой объем данных, который необходимо оперативно доставлять для анализа в некое хранилище всех событий.

Для сбора данных о событиях, происходящих на элементах ИС, необходимо использовать специальное программное обеспечение

(далее по тексту — Агенты). Агенты собирают данные о событиях, но сами не обрабатывают их. Обработка событий Агентами может привести к повышенной нагрузке на систему, в которой он работает. Поэтому Агенты должны передать собранные данные в единую базу данных событий (БДС). Только после этого отдельное программное обеспечение (SIEM) может провести анализ накопленных данных в БДС для получения результата — выявления причины сбоя [1]. Для построения такой системы сбора данных необходимо рассмотреть возможные простейшие способы построения многоагентных систем (МС), определить их слабые и сильные стороны, возможность их реализации.

Исследования в области разработки МС проводились многими исследователями [2], но в этих работах основное внимание акцентируется на способе и языке взаимодействия Агентов [3], интеллекте Агентов [4], групповом поведении многоагентной системы. Определение МС в этих работах примерно одинаковое [3] — это описание элементов этой системы и описание их взаимодействия между собой для достижения определенной цели. При этом чаще всего ставится вопрос о решении множества сложных задач: взаимодействие Агента с

¹Работа выполнена при частичной поддержке РФФИ (грант 18-29-03081-мк).

окружающей средой и реакция на ее изменение, демонстрация определенного поведения Агентов для достижения цели, организация "социальных" отношений между Агентами [5]. В работах приводятся примеры МС для сложных инфраструктур, но нет описания способа их построения (архитектуры).

Одним из направлений исследований в области архитектур МС (АМС) является возможность ее реализации. В таких работах приводится классификация [6] или описание [7] готовой архитектуры без сравнения ее с другими возможными реализациями.

В найденных работах рассмотрены теоретические аспекты функционирования сложных интеллектуальных МС, а также приведены примеры готовых реализаций таких систем. В данной работе сравнение АМС начинается с рассмотрения простейшей архитектуры, определения ее преимуществ и недостатков, а также способов устранения выявленных недостатков с помощью дополнительных элементов и взаимосвязей, т. е. расширения архитектуры.

Построение простейших архитектур многоагентных систем

Для построения АМС определим функции ее элементов и минимальные требования, предъявляемые к ним:

- сбор данных о действиях пользователей, программного обеспечения, сетевой активности, записях в журналах выполняет Агент, находящийся на наблюдаемом персональном компьютере, сервере (далее по тексту — ПК) или на выделенном персональном компьютере, сервере (далее — Внешний Агент). Агент способен накапливать события, но объем накопленных данных должен быть минимален, чтобы не повышать нагрузку на ПК. Необходимость наличия такого элемента рассмотрена в работах [8, 9];
- БДС находится на сервере, имеющем ограничение по числу входящих подключений, вычислительной мощности, но не имеющем ограничений по объему хранимых данных;
- линии связи между элементами АМС имеют одинаковую задержку и пропускную способность. Резервирование линий связи может существовать или отсутствовать.

После определения элементов и их функций можно создать первую, простейшую АМС.

На рис. 1 схематически изображена простейшая АМС. Агент, находящийся на ПК,

проводит сбор данных и отправляет их в БДС. В случае если число подключаемых к БДС Агентов невелико, а линии связи гарантированно функционируют, то такая АМС является оптимальной. Но в случае если число Агентов превысит число максимально возможных подключений к БДС, а линии связи не будут гарантированно функционировать, то возникает ситуация двойного сбоя: БДС не способна принять подключения и данные от Агентов, а Агенты переполняют свое локальное хранилище накопленных данных. В таком случае АМС не сможет обеспечить гарантированный сбор, целостность данных и доставку их в БДС.

Для преодоления ограничения на число одновременных подключений необходимо ввести дополнительный, масштабируемый элемент в АМС, который бы обеспечил неограниченный рост числа подключаемых агентов. На рис. 2, 3 представлена АМС с таким элементом — Веб-сервером или Хранилищем. Этот элемент может быть представлен в виде кластеров серверов, он масштабируем и обеспечивает гарантированное принятие подключений от Агентов.

Важно учитывать, что введение в АМС промежуточных элементов приводит к увеличению стоимости ее реализации. Обслуживать такую АМС сложнее, так как серверы со специализированным программным обеспечением требуют наличия штата квалифицированных специалистов, запасных компонентов для оперативного ремонта и т. д.

Вернемся к рассмотрению ситуации, в которой линии связи могут отказать, что приведет к переполнению локального хранилища накопленных Агентом данных на ПК. Решений этой проблемы существует несколько:

- увеличение объема локального хранилища данных на ПК, на котором функционирует

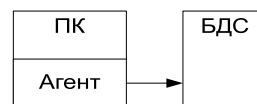


Рис. 1. Простейшая АМС

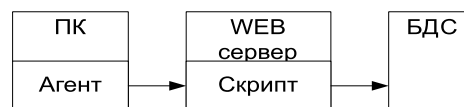


Рис. 2. Дополнительный элемент АМС — Веб-сервер

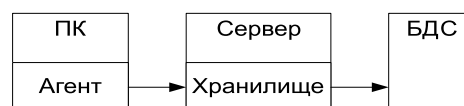


Рис. 3. Дополнительный элемент АМС — Хранилище

Агент. В случае же длительного отказа линий связи такое увеличение все равно может привести к переполнению;

- создание промежуточных локальных серверов для хранения накопленных данных (далее — Локальное Хранилище). Данный способ подразумевает, что линии связи в локальной сети не нарушены. Более того, это приводит к увеличению стоимости реализации всей АМС. Если же таких локальных сетей много, то увеличение стоимости будет значительным из-за большого числа дополнительных серверов — Локальных Хранилищ;
- выделение Агента на отдельный отказоустойчивый ПК (Внешний Агент) и осуществление удаленного сбора данных с остальных элементов АМС. Данный вариант приводит к значительному росту стоимости реализации АМС, увеличивает сложность обслуживания всей АМС, приводит к значительному усложнению программного кода Агента. Но он также имеет ряд преимуществ. В случае если ПК, на котором работает Агент, выходит из строя или подключение его к локальной сети нарушено, выявить неработающего Агента можно только по тому, что он перестает посылать данные в БДС. Если же Агент установлен на выделенном отказоустойчивом ПК, то он способен наблюдать не только за состоянием остальных ПК, но и может выявлять проблемы с сетевым оборудованием. Если линия связи Агента с БДС не функционирует, то такой Агент способен длительное время проводить сбор данных с элементов АМС в локальной сети.

Описанные выше варианты реализации АМС можно изобразить схематически (рис. 4, 5).

Если использовать все сильные стороны рассмотренных АМС, можно создать еще одну АМС, обеспечивающую избыточность получаемых от ИС данных (рис. 6).

В этой АМС должны присутствовать Агенты с увеличенным хранилищем накопленных данных, Внешние Агенты на отказоустойчивых серверах, Локальные Хранилища, промежуточные серверы — Хранилища или Веб-серверы, а также резервные линии связи между ключевыми элементами: Локальным Хранилищем — Хранилищем/Веб-сервером — БДС. Такая АМС в случае отказа отдельных элементов продолжит функционировать, по-

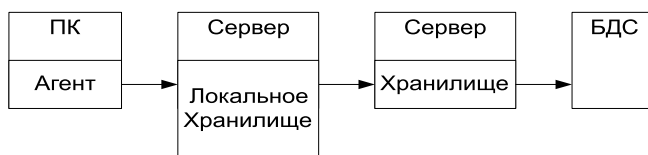


Рис. 4. Локальное Хранилище в АМС

зволяя проводить быстрое выявление причины возникших сбоев.

Рассмотрим подробнее вопрос сравнения надежности архитектур. Оценка надежности архитектуры основана на оценке надежности каждого элемента. Например, существуют способы построения отказоустойчивых аппаратных платформ, но чаще всего к сбоям приводят не аппаратные платформы, а ошибки программного обеспечения (ПО) и нарушение в линиях связи между элементами. Рассмотрим отдельно каждый элемент архитектур.

Надежность работы Агента зависит от надежности ПО Агента, операционной системы (ОС), оборудования, на котором он работает. Выход из строя Агента, ОС, оборудования может быть определен исходя из того, что Агент перестает отправлять данные в БДС.

Надежность линии связи от элемента с Агентом до БДС, Веб-сервера, Хранилища или Локального хранилища зависит от сетевого оборудования в локальной сети. Если сеть не имеет специального отказоустойчивого оборудования, например, это простая офисная сеть, то обнаружить сбой в линии связи — это трудновыполнимая задача. Профессиональное сетевое оборудование имеет возможность диагностики сетевых портов и наличия подключения к узлу. В таком случае в сети должен

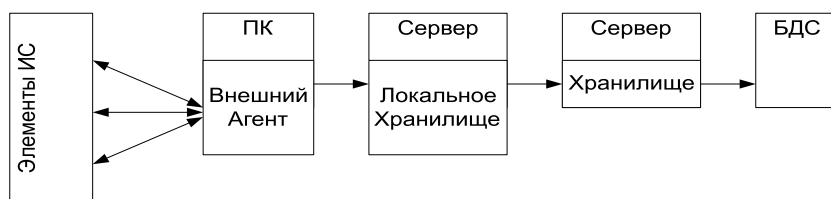


Рис. 5. Внешний Агент в АМС

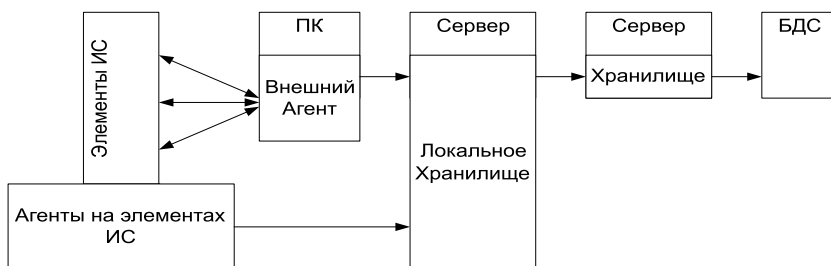


Рис. 6. Избыточная АМС

находиться Внешний Агент, наблюдающий за работой сетевого оборудования.

Надежность Веб-сервера, Хранилища, Локального Хранилища, БДС зависит от надежности аппаратной платформы и специального программного обеспечения, обеспечивающего предоставление сервисов. На данный момент существует множество решений, позволяющих организовать безотказную работу таких серверов. Благодаря резервированию и виртуализации можно создавать группы территориально удаленных серверов. В случае выхода из строя одной группы серверов или потери связи с ними Агенты могут использовать другую группу для передачи данных в БДС. Сами серверы БДС также могут быть выполнены в виде распределенной системы хранения данных, что позволит продолжить работу даже в случае отказа части серверов.

В итоге использование отказоустойчивых аппаратно-программных комплексов для реализации сервисов приема и обработки данных, резервирование линий связи и наличие Внешних Агентов, осуществляющих наблюдение за работой сетей, минимизация длины линии связи между Агентом и БДС позволяют реализовать любую из предложенных архитектур с необходимым уровнем надежности.

Теперь рассмотрим вопрос защищенности предложенных архитектур. Допустим, что одинаковые элементы в разных АМС имеют одинаковую степень защищенности. В таком случае отличия в степени защищенности АМС будут определяться степенью защищенности отличающихся элементов. Более подробно архитектурные уязвимости и способы их устранения рассмотрены в работе [10]. Например, сравнивая АМС на рис. 1 и АМС на рис. 2, можно сделать следующий вывод: АМС на рис. 2 имеет дополнительные элементы — это линия связи Веб-сервер — БДС и сам Веб-сервер. В таком случае АМС на рис. 2 имеет больше потенциальных уязвимостей и, следовательно, является менее защищенной. Такое же сравнение можно провести с другими рассмотренными архитектурами. Таким образом, можно сделать вывод, что увеличение числа элементов в АМС снижает защищенность, так как каждый новый элемент привносит в систему свои потенциальные уязвимости.

Сравнивая рассмотренные АМС между собой, важно учитывать время доставки данных от Агента в БДС. Допустим, что все линии связи между элементами архитектуры имеют одинаковые характеристики и обеспечивают ми-

нимальную задержку передачи данных, а серверы имеют достаточную производительность, чтобы на входе не образовывались очереди данных. В таком случае число соединений между Агентом и БДС определяет скорость доставки данных. Чем меньше линий связи, тем меньше задержка. В таком случае наилучший результат будет показывать архитектура "Агент — БД". Но если предположить, что БДС не может одновременно обрабатывать запросы от всех Агентов одновременно, то необходимо создавать промежуточные Хранилища, которые бы реализовывали принцип FIFO. В таком случае задержка определяется не только числом линий связи, но и длиной формируемой очереди на обработку данных. В случае, когда очередь разрастается, время задержки может быть существенным, однако это снижает нагрузку на БДС.

Таким образом, скорость доставки зависит не столько от АМС, сколько от вычислительной мощности БДС, Веб-сервера, Хранилища, а также от пропускной способности линий связи. При наличии достаточной вычислительной мощности можно реализовывать архитектуру "Агент — БД". Если же вычислительной мощности недостаточно или же линии связи не могут обеспечить необходимую пропускную способность, то можно использовать АМС типа "Агент — Хранилище — БДС". Аналогичная ситуация может возникать при использовании систем виртуализации для построения всей архитектуры ИС. Исследования влияния таких архитектур на скорость обмена данными между узлами приведены в работе [11].

Определение параметров архитектур многоагентных систем

Каждая из рассмотренных АМС имеет свои преимущества и недостатки. Недостатки, выявленные у самой простой АМС, обуславливают необходимость добавить элементы, которые бы могли компенсировать их. Это приводит к усложнению АМС и, как следствие, к появлению новых недостатков. Следовательно, требуется добавление элементов, которые будут устранять уже их. Такое рассуждение приводит к созданию АМС из большого числа дорогостоящих элементов (рис. 6).

Перед разработчиком АМС возникает вопрос, какие параметры являются ключевыми при выборе той или иной АМС. Это может быть стоимость реализации АМС, ее надежность, защищенность, сложность обслуживания

ния, время передачи данных, время разработки Агентов и т. д.

Существует несколько способов определения параметров, влияющих на выбор АМС.

- *Аналитический.* Этот способ требует построения сложной модели, учитывающей все параметры АМС. Если же модель не будет учитывать каких-либо параметров (упрощенная модель), то выбор, сделанный с помощью нее, может оказаться не реализуемым на практике.
- *Экспериментальный.* Проведение экспериментов на макете АМС позволяет получить большое число результатов о поведении АМС в реальных условиях. В процессе создания макета АМС можно выявить множество ограничений, накладываемых реальностью: задержки и непредвиденные обрывы линий связи, искажение передаваемых данных, сбои и задержки в работе самих Агентов, перегрузки серверов и т. д.

Рассмотрим вопрос создания макета АМС, приведенной на рис. 1. Для создания такого макета требуется:

- ПК;
- программное обеспечение — Агент;
- подключение по локальной сети к серверу БДС;
- программное обеспечение — СУБД для организации БДС.

Реализация такого макета позволяет провести эксперименты, выявляющие нагрузку Агента на ПК, порождаемый им сетевой трафик, нагрузку на сервер БДС, а также дает возможность отработать ситуации с прерыванием связи. Для того чтобы этот макет реализовать в глобальной сети, потребуется установка сервера БДС на каком-либо хостинге. При этом сразу возникнет иная проблема — чаще всего хостинговые компании не предоставляют прямой доступ к СУБД из сети Интернет. Таким образом, макет по АМС на рис. 1 невозможно реализовать.

Рассмотрим возможность реализации макета АМС, рассмотренной на рис. 2. Для реализации этого макета требуется:

- ПК;
- программное обеспечение — Агент;
- подключение ПК Агента к сети Интернет;
- Веб-сервер с программным обеспечением, принимающим данные от Агентов;
- программное обеспечение — СУБД для организации БДС.

Чаще всего на хостингах используют язык программирования PHP и СУБД MySQL. Разработанный макет позволяет оценить АМС по следующим параметрам:

- параметры сбора данных Агентом;
- создаваемая Агентом нагрузка на ПК;
- создаваемая Агентом нагрузка на локальную сеть;
- создаваемая Агентом нагрузка на Веб-сервер;
- задержки, возникающие при передаче данных через сеть Интернет;
- и т. д.

Для проведения оценки параметров АМС (см. рис. 2) был реализован макет. Агент разработан на языке C# и реализует следующий функционал:

- сбор данных о запущенных процессах с интервалом 10 мс. Данный интервал выбран исходя из того, что в системе иногда запускаются процессы, время работы которых мало. Для реальной системы отслеживания запускаемых процессов это время опроса может быть недостаточно малым;
- сбор данных о нажатиях на кнопки компьютерной мыши. Данный сбор реализован в виде перехвата системного вызова;
- сбор данных об установленных подключениях с интервалом 1 с. Данный параметр выбран исходя из того, что соединения не закрываются сразу после завершения сессии обмена данными, если речь идет о TCP/IP. Они переходят в состояние "ожидание завершения". Для отслеживания передачи данных с помощью UDP необходимо реализовать иной метод наблюдения, например, наблюдение за сетевыми пакетами;
- отправка данных на Веб-сервер каждые 10 с. Если обработка БДС с помощью специального программного обеспечения занимает незначительное время, то можно утверждать, что выявление сбоя на элементе системы будет обнаружено максимум через 10 с. В реальной системе это время может варьироваться в зависимости от предполагаемого времени обнаружения сбоя.

На Веб-сервере разработана программа на языке PHP, которая выполняет следующие функции:

- журналирование обращений Агента к Веб-серверу;
 - прием данных от Агента и запись их в БДС.
- Для проведения эксперимента Агент был установлен и запущен на ПК пользователя. Через 14 дней сбора данных от Агента были получены следующие результаты:
- среднее число запущенных процессов за сутки — 1235;
 - среднее число нажатий клавиш — 4509;

- среднее число подключений — 13 197;
- среднее число нажатий на клавиши мыши — 6730;
- обращение агента за 1 сутки к БДС — 34 540;
- примерный объем БДС — 700 Мбайт.

Объем полученных данных, а также число обращений к БДС показывают, что сбор всех необходимых данных на макете приводит к большой нагрузке как на БДС, так и на сервер, принимающий данные от Агента (Веб-сервер). Следовательно, перед разработчиком такой системы встанет задача минимизации числа наблюдаемых параметров, а также уменьшения числа подключений для передачи данных. Например, для выявления отключения Агента от системы может быть достаточно реализовать "пульс", т. е. периодическое информирование БДС о том, что Агент функционирует. Для наблюдения за функционированием элемента ИС может быть достаточно наблюдения за одним процессом, выполняющим основной функционал ИС. Для предотвращения выхода из строя элемента необходимо наблюдать за параметрами оборудования, например, за состоянием жесткого диска. В таком случае потребуется информировать БДС только о событии ухудшения параметров, например, превышении допустимой температуры процессора или появлении ошибок на жестком диске. Эти решения приведут к снижению объема накопленных и передаваемых в БДС данных, а также позволят снизить число обращений к БДС.

Полученные в результате подобных экспериментов в реальных системах параметры могут быть использованы для создания прототипов рабочих АМС, оценок требуемой вычислительной мощности, требуемой пропускной способности линий связи, а также для проведения исследований по возможностям оптимизации.

Заключение

В данной статье описаны способы построения простейших АМС для сбора данных. Также проведено сравнение АМС по различным критериям: число элементов, стоимость элементов, сложность создания и обслуживания, надежность архитектуры, защищенность, скорость доставки данных. В рассмотренных АМС выявлены как преимущества, так и недостатки, что привело к необходимости определения параметров, определяющих выбор той или иной АМС. Поскольку выбор АМС — это сложная задача, которая зависит от большого

числа параметров, предложено использовать макетирование АМС. Цель макетирования — оценка параметров АМС в реальных условиях. Проведен эксперимент по макетированию АМС и сделан вывод о значимости получаемых экспериментальных данных.

Как и для любого проекта в области информационных технологий, реализация любой из АМС определена тремя базовыми параметрами: стоимость, качество, время. Уменьшение стоимости проекта вынуждает разработчиков использовать более дешевое оборудование и средства защиты информации, что приводит к снижению надежности и защищенности. Это снижает качество проекта по реализации АМС, но уменьшает время его реализации. Повышение качества приводит к увеличению стоимости и времени реализации, но при этом надежность, защищенность будет соответствовать необходимому уровню.

Оптимальное соотношение между временем реализации АМС, стоимостью реализации и качеством реализации АМС также может быть определено с помощью макетирования.

В данной статье не рассмотрены вопросы оптимизации сбора данных, так как не определен полный функционал системы сбора данных, но полученные в результате эксперимента данные показывают острую необходимость развития методов такой оптимизации.

Список литературы

1. **Grusho A. A., Grusho N. A., Zabezhaile M. I., Timonina E. E.** Data mining in ensuring information security // *Automatic Control and Computer Sciences*, 2016. Vol. 50. No. 8. P. 722—725.
2. **Shehory Onn.** Architectural Properties of Multi-Agent Systems. Pennsylvania: Carnegie Mellon University, 1998. URL: <https://pdfs.semanticscholar.org/c8f4/bd4f078fb2ac05650b-23b9ec81ebf1613e57.pdf> (дата обращения 11.02.19).
3. **Amit K. Chopra, Munindar P. Singh.** An Architecture for Multiagent Systems. An Approach Based on Commitments. URL: <https://pdfs.semanticscholar.org/03bb/63477b3d30be1dde04a2a7071365e3cf87d5.pdf> (дата обращения 11.02.19).
4. **Hyacinth S. Nwana.** Software agents: An overview // *The Knowledge Engineering Review*, 1996. Vol. 11, Iss. 3. P. 205—244.
5. **Jennings N. R., Sycara K., Wooldridge M. A.** Roadmap of Agent Research and Development // *Autonomous Agents and Multi-Agent Systems*, 1998. Vol. 1, Iss. 1. P. 7—38. URL: <https://doi.org/10.1023/A:1010090405266> (дата обращения 11.02.19).
6. **Balaji P. G., Srinivasan D.** An Introduction to Multi-Agent Systems // Srinivasan D., Jain L. C. (eds) *Innovations in Multi-Agent Systems and Applications — 1. Studies in Computational Intelligence*, 2010. Vol. 310. Springer, Berlin, Heidelberg. P. 1—27.
7. **Reida M., Shakshuki E. M.** Implementing a Multi-agent System for Recording and Transmitting Biometric Information of Elderly Citizens // *The 7th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH 2017)*, 2017. P. 334—343.

8. Грушо А. А., Забежайло М. И., Зацаринный А. А., Николаев А. В., Писковский В. О., Сенчило В. В., Судариков И. В., Тимонина Е. Е. Об анализе ошибочных состояний в распределенных вычислительных системах // Системы и средства информатики. 2018. Т. 28, № 1. С. 99–109.

9. Грушо А. А., Грушо Н. А., Тимонина Е. Е., Шоргин С. Я. Возможности построения безопасной архитектуры для динамически изменяющейся информационной системы // Системы и средства информатики. 2015. Т. 25, № 3. С. 78–93.

10. Грушо А. А., Грушо Н. А., Тимонина Е. Е., Шоргин С. Я. Архитектурные уязвимости распределенных информационно-вычислительных систем. Системы и средства информатики. 2016. Т. 26, № 3. С. 74–82.

11. Грушо А. А., Грушо Н. А., Левыкин М. В., Тимонина Е. Е. Безопасные архитектуры распределенных информационно-вычислительных систем на основе комплексной виртуализации // Проблемы информационной безопасности. Компьютерные системы, 2016. Вып. 4. С. 31–35.

N. A. Grusho, Candidate of Science (PhD) in physics and mathematics,
e-mail: info@itake.ru

E. E. Timonina, Doctor of Science (Tech), Professor, e-mail: eltimon@yandex.ru
Institute of Informatics Problems, Federal Research Center "Computer Sciences and Control"
of the Russian Academy of Sciences, Moscow

Comparison of Architectures of Multi-Agent Systems

Usage of the new or existing information systems is connected with need transfer of data acquisition about the incidents arising in them and also with transfer identification of the reasons of these incidents. If information systems have a set of hosts and also there are communications of information among them, then the logs accumulated for a small period can have a huge amount of records. It is natural that administrators of information systems are not able to carry out the full and detailed analysis of such information for a short period. In this paper architectures of multi-agent systems which are carrying out collection of data on incidents in information systems for the purpose of their further automatic analysis are considered.

Keywords: agents, multi-agent systems, architecture of multi-agent systems, information technologies

DOI: 10.17587/it.25.293-299

Acknowledgements: The work was carried out with the partial support of the Russian Foundation for Basic Research (grant 18-29-03081-microns).

References

1. Grusho A. A., Grusho N. A., Zabezhailo M. I., Timonina E. E. Data mining in ensuring information security, *Automatic Control and Computer Sciences*, 2016, vol. 50, no. 8, pp. 722–725.

2. Shehory Onn. Architectural Properties of Multi-Agent Systems. Pennsylvania: Carnegie Mellon University, 1998, available at: <https://pdfs.semanticscholar.org/c8f4/bd4f078fb2ac05650b-23b9ec81ebf1613e57.pdf> (date of access: 11.02.19).

3. Amit K. Chopra, Munindar P. Singh. An Architecture for Multiagent Systems. An Approach Based on Commitment, available at: <https://pdfs.semanticscholar.org/03bb/63477b3d30be1dde04a2a7071365e3cf87d5.pdf> (date of access: 11.02.19).

4. Hyacinth S. Nwana. Software agents: An overview, *The Knowledge Engineering Review*, 1996, vol. 11, iss. 3, pp. 205–244.

5. Jennings N. R., Sycara K., Wooldridge M. A. Roadmap of Agent Research and Development, *Autonomous Agents and Multi-Agent Systems*, 1998, vol. 1, iss. 1. P. 7–38, available at: <https://doi.org/10.1023/A:1010090405266> (date of access: 11.02.19).

6. Balaji P. G., Srinivasan D. An Introduction to Multi-Agent. Innovations in Multi-Agent Systems and Applications — 1. *Studies in Computational Intelligence*, 2010, vol. 310. Springer, Berlin, Heidelberg, pp. 1–27.

7. Reida M., Shakshuki E. M. Implementing a Multi-agent System for Recording and Transmitting Biometric Information of Elderly Citizens, *The 7th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH 2017)*, 2017. P. 334–343.

8. Grusho A. A., Zabezhailo M. I., Zatsarinnyy A. A., Nikolaev A. V., Piskovski V. O., Senchilo V. V., Sudarikov I. V., Timonina E. E. Ob analize oshibochnykh sostoyaniy v raspredelennykh vychislitel'nykh sistemakh (About the analysis of erratic statuses in the distributed computing systems), *Sistemy i Sredstva Informatiki*, 2018, vol. 28, no. 1, pp. 99–109 (in Russian).

9. Grusho A., Grusho N., Timonina E., Shorgin S. Vozmozhnosti postroeniya bezopasnoy arkhitektury dlya dinamicheskoi izmenyayushcheyasya informatsionnoy sistemy (Possibilities of Secure Architecture Creation for Dynamically Changing Information Systems), *Sistemy i Sredstva Informatiki*, 2015, vol. 25, no. 3, pp. 78–93 (in Russian).

10. Grusho A. A., Grusho N. A., Timonina E. E., Shorgin S. Ya. Arkhitekturnye uyazyimosti raspredelennykh informatsionno vychislitel'nykh sistem (Architectural vulnerabilities of the distributed information systems), *Sistemy i Sredstva Informatiki*, 2016, vol. 26, no. 3, pp. 74–82 (in Russian).

11. Grusho A. A., Grusho N. A., Levykin M. V., Timonina E. E. Bezopasnye arkhitektury raspredelennykh informatsionno vychislitel'nykh sistem na osnove kompleksnoy virtualizatsii (Secure architecture of distributed information systems on the basis of integrated virtualization), *Problemy Informatsionnoy Bezopasnosti. Komp'yuternye Sistemy*, 2016, no. 4, pp. 31–35 (in Russian).

Н. Т. Абдуллаев¹, канд. техн. наук, зав. кафедрой, e-mail: a.namik46@mail.ru,
О. А. Дышин², канд. физ.-мат. наук, доц., **И. Д. Ибрагимова**¹, ассистент, e-mail: irada432@gmail.com,
Х. Р. Ахмедова¹, канд. физ.-мат. наук, e-mail: yubaba66@hotmail.com,
¹Азербайджанский технический университет,
²Азербайджанский государственный университет нефти и промышленности

Сегментирование нестационарных физиологических сигналов с фрактальными свойствами

Для сегментирования нестационарного фрактального временного ряда сначала с помощью вейвлет-разложения определяется уровень разрешения, определяющий интервал самоподобия (фрактальности) исходного ряда и находится оценка трендовой составляющей. С использованием регрессионной модели временного ряда с волатильностью, представляющей детрендированный ряд, находим точки изменения волатильности, которые разделяют исходный ряд на квазистационарные участки (сегменты), соответствующие скачкам функции волатильности. Для оценивания точек изменения волатильности разработана вычислительная процедура, обобщающая итеративный метод центрированных кумулятивных сумм квадратов (ICSS).

Ключевые слова: фракталы, вейвлет-разложение, нестационарность, временной ряд, волатильность

Введение

Основной проблемой современной физиологии является нестационарность временных рядов, образуемая под воздействием внешних условий. Поэтому многие исследователи концентрируют внимание на присущих физиологическим сигналам эффектах нестационарности [1, 2]. Отметим, что нестационарность проявляется не только под влиянием окружающей среды, но также является важным естественным свойством физиологических временных рядов, которое может привести к патологическим последствиям.

Нестационарность временного ряда проявляется в том, что его статистические свойства изменяются с течением времени. В результате порождается неоднородность данных в том смысле, что среднее значение, стандартное отклонение и более высокие моменты (асимметрия, эксцесс) становятся зависимыми от интервала времени, на котором они вычисляются. Это приводит к тому, что анализ таких рядов усложняется, поскольку справедливость многих статистических процедур основана на предположении о стационарности исследуемых данных.

Кроме того, необходимо иметь в виду, что для многих процессов различной физической природы неотъемлемым является свойство статистической самоподобности, или фрактальности, выражающееся в масштабной инвариантности по отношению к тем или иным статистическим характеристикам. Так, например, наличие в динамическом ряде долговременных фрактальных корреляций, которые известны как корреляции типа $1/f$, приводит к неоднородностям на всех масштабах в форме самоподобия [3–5], затрудняя тем самым процедуру сегментирования исходного ряда, т. е. разбиения его на непересекающиеся смежные фрагменты, которые будут статистически однородны или, по крайней мере, будут обладать этим свойством в большей степени, чем исходные данные. Эта проблема хорошо изучена и известна как задача нахождения точек изменения (change-point problem) [6].

Методы оценки самоподобности динамических рядов приведены в работе [7]. В значительной степени свободным от ограничений является методика оценки самоподобности, основанная на вейвлет-анализе динамического ряда [5]. Наличие самоподобности в некото-

рых компонентах исследуемого динамического ряда означает неинформативность этих компонентов для оценки происходящих в изучаемой системе изменений. В рамках мультиразрешающего анализа [8], который проводится последовательным применением дискретного вейвлет-преобразования к исходной реализации $x(t)$, приближение $\tilde{x}_j(t_i)$ к сигналу $x(t_i)$ на уровне разрешения j разбивается на две составляющие — грубую (аппроксимирующую) $\tilde{x}_j^g(t_i)$ и уточненную (детализирующую) высокочастотную составляющую $\tilde{x}_j^d(t_i)$ с последующим их уточнением итерационным методом. Используя логарифмическую характеристику самоподобности [7], по линейному возрастающему участку этой характеристики с показателем Херста $0,5 \leq H \leq 1$ можно определить масштабный диапазон, в котором проявляется самоподобность изучаемой реализации. Вейвлет-анализ дает возможность установить обрыв линейного участка и характерный спад вышеуказанной логарифмической характеристики, определяющей предельную глубину j_f масштабной инвариантности. В ходе обработки реальных электрокардиограмм (ЭКГ) удалось установить [5], что $j_f = 4$ характеризует предельную глубину масштабной инвариантности сердечного ритма. Алгоритм быстрого вейвлет-преобразования позволяет вычислять коэффициенты вейвлет-разложения итерационным методом [9].

Многочисленное применение прямого и обратного вейвлет-разложения при $j > j_f$ позволяет получить оценку $\hat{\mu}(t)$ трендовой (регулярной) составляющей $\mu(t)$ процесса $x(t)$ и выделить нерегулярную статистическую составляющую $z(t) = x(t) - \hat{\mu}(t)$, изменение статистических характеристик (средней и/или дисперсии) которой позволяет сегментировать исходный временной ряд.

Сегментирование нестационарных временных рядов, представляемых ЭКГ, позволяет установить границы, разделяющие качественно различные по своим статистическим свойствам составляющие ритмограмм, что согласуется с физиологическими представлениями, на которых основан классический анализ variability сердечного ритма [10].

Для нахождения точек изменения (change-points) временного ряда нами использован метод непараметрического оценивания точек структурного изменения в моделях временных рядов с волатильностью [11], являющийся обобщением итеративного алгоритма кумулятивных сумм квадратов (iterated cumulative sums of squares — ICSS-algorithm), предложен-

ного в работе [12]. Для временного ряда нами использована модель [11] с неизвестной регрессивной функцией (условное среднее) и неизвестной функцией волатильности (условная вариация) с оценкой каждой из этих функций с помощью ядерных функций и последующего непараметрического оценивания точек структурного изменения волатильности. Для отыскания таких точек в работе [11] разработан алгоритм, основанный на статистике V_k^v , $0 \leq v \leq 1/2$, определяемый по k ($k = 1, \dots, N$) значениям временного ряда $\{y_i\}$, $i = 1, \dots, N$ (N — общая длина ряда). При $v = 0$ данный алгоритм является модификацией, обобщающей ICSS-algorithm [12].

1. Оценка самоподобности случайного процесса

Самоподобность, или фрактальность, для детерминированных фракталов означает масштабную инвариантность (неизменяемость) их геометрической конфигурации в определенном диапазоне масштабов (интервале самоподобия). В отличие от детерминированных фракталов, для которых возможно точное воспроизведение свойств при масштабировании в интервале самоподобия, для случайных фрактальных процессов можно говорить о статистической самоподобности, выражающейся в масштабной инвариантности по отношению к тем или иным характеристическим свойствам.

По определению [7], случайный процесс называется фрактальным, если некоторые из его важных статистических характеристик проявляют свойство масштабированной инвариантности. Так как такое свойство масштабирования математически выражается степенными зависимостями, часто степенной характер статистических характеристик случайного процесса является признаком принадлежности его к классу фрактальных процессов.

Одним из определяющих свойств фрактальных процессов является самоподобность [13]. Случайный процесс X_t называется статистически самоподобным, если сам этот процесс и процесс $a^{-H}X_{at}$, полученный из X_t с учетом измененного временного масштаба at , имеют одинаковые конечномерные плотности распределения вероятностей для всех положительных целых n :

$$W\{X_1, X_2, \dots, X_n\} = W\{a^{-H}X_{1a}, a^{-H}X_{2a}, \dots, a^{-H}X_{na}\}, \quad (1)$$

где a — коэффициент расширения; $0,5 \leq H \leq 1$ — показатель Херста [7].

При исследовании процессов по конечным выборкам в большинстве случаев распределение вероятностей неизвестно, поэтому на практике чаще всего используют понятие самоподобности N -го порядка, когда процессы X_t и $a^{-H}X_{at}$ имеют одинаковые характеристики, по крайней мере, до N -го порядка включительно. Также используется понятие асимптотически подобных процессов, когда свойство (1) выполняется при $a \rightarrow \infty$.

К основным признакам самоподобных процессов второго порядка ($N = 2$) относятся:

1) гиперболический вид корреляционной функции

$$R(k) \cong k^{(2H-2)}L(t_k), k \rightarrow \infty, \quad (2)$$

где k — номер отсчета в дискретном временном ряде t_k ; $L(t)$ — медленно меняющаяся функция при $t \rightarrow \infty$: $\lim_{k \rightarrow \infty} [L(tx)/L(t)] = 1, \forall x > 0$;

2) медленно затухающая дисперсия

$$D[X_m] \sim m^{(2H-2)}, m \rightarrow \infty, \quad (3)$$

где D — символ выборочной дисперсии; X_m — временная последовательность, полученная усреднением исходной последовательности $X_k = X(t_k)$ по непересекающимся последовательным блокам с размером m ;

3) степенной характер спектральной плотности мощности (СПМ) $S(\omega)$ вблизи нуля: $S(\omega) \sim \omega^{-\gamma}L_2(\omega), \omega \rightarrow \infty, 0 < \gamma < 1, L_2(\omega)$ — медленно меняющаяся функция частоты [7].

Из соотношения (2) видно, что $\sum_k R(k) \rightarrow \infty$ при $0,5 \leq H \leq 1$, что характеризует еще одно важное свойство многих фрактальных процессов — наличие долговременной корреляционной зависимости или, в данном случае, ее линейной составляющей. Однако подобный вид корреляционной функции не является необходимым условием фрактальности процесса.

Для многих процессов с фрактальной структурой характерным признаком является наличие распределения величин с "тяжелыми хвостами" [7]. Говорят, что случайная величина X имеет распределение $F(x)$ с "тяжелыми хвостами", если вероятность

$$P\{X > x\} = 1 - F(x) = x^{-\alpha}L(x), x \rightarrow \infty, \quad (4)$$

где $0 < \alpha < 2$ — индекс хвоста распределения, или параметр формы.

Как видно из соотношения (4), к распределениям с "тяжелыми" хвостами относятся распределения вероятностей, затухающие при $x \rightarrow \infty$ медленнее экспоненты. В то же время данное свойство также не является необходимым условием наличия фрактальной структуры процесса. Оценка параметра формы распределения α может быть выполнена на основании имеющейся в распоряжении реализации изучаемого процесса, но требует чрезвычайно большого объема статистических данных. Тем не менее, существует ряд тестов, позволяющих провести оценку параметра α по реализации в несколько тысяч отсчетов. Среди них следует отметить оценку Хилла, оценку по модифицированному QQ-графику и оценку момента DeNaan [7].

Как следует из определения (1), показатель Херста H является основной величиной, характеризующей самоподобность случайного процесса. При этом следует иметь в виду, что вывод о самоподобном характере случайного процесса на основании оценки, проведенной по одной реализации, может быть сделан, строго говоря, только при установленном свойстве его эргодичности, когда среднее по времени, установленное по одной реализации процесса, равно среднему по ансамблю всех реализаций (траекторий) процесса. В других случаях следует говорить о наличии самоподобной структуры процесса в заданном масштабном диапазоне для заданного набора данных.

К основным методам оценки показателя Херста для временных рядов относятся: анализ нормированного размаха, предложенный самим Херстом; анализ изменения выборочного значения дисперсии в зависимости от объема выборки; анализ СПМ с применением оценки Виттла и вейвлет-анализ [7].

В значительной мере свободной от ограничений и независимой от формы распределения (т. е. непараметрической) является методика оценки самоподобности на основе вейвлет-анализа динамического ряда, которая будет использована нами в расчетах. В основу данной методики положен мультиразрешающий анализ [8], который может быть проведен последовательным применением дискретного преобразования к исходной реализации. Любую функцию $x(t)$ из $L^2(R)$ ($L^2(R)$ -пространство квадратично интегрируемых функций на евклидовом пространстве $R = (-\infty, \infty)$) можно разложить на некотором заданном уровне разложения J в дискретный неизбыточный вейвлет ряд вида [15]

$$\tilde{x}(t) = \sum_{k \in Z} a_{j,k} \varphi_{j,k}(t) + \sum_{j=1}^{n_0} \sum_{k \in Z} d_{j,k} \psi_{j,k}(t), \quad (5)$$

где функции $\varphi_{j,k}(t) = 2^{-j/2} \varphi_0(2^{-j}t - k)$ и $\psi_{j,k}(t) = 2^{-j/2} \psi_0(2^{-j}t - k)$; $\varphi_R = \{\varphi_{j,k}\}$, $\psi_k = \{\psi_{j,k}\}$, $k \in Z$ (Z — множество всех целых чисел), образуют ортонормированный базис в $L^2(R)$ из сдвинутых и расширенных реализаций масштабирующей функции $\varphi_0(t)$ и вейвлет-функции $\psi_0(t)$, называемой иногда "материнским вейвлетом".

Дискретное вейвлет-преобразование состоит в отыскании коэффициентов $a_{j,k}$ и $d_{j,k}$, определяемых как

$$a_{j,k} = \int x(t) \varphi_{j,k}(t) dt, \quad d_{j,k} = \int x(t) \psi_{j,k}(t) dt. \quad (6)$$

Итерационные формулы быстрого вейвлет-преобразования (БВП) имеют вид:

$$a_{j+1,k} = \sum_m h_m a_{j,2k+m}, \quad d_{j+1,k} = \sum_m g_m a_{j,2k+m} \quad (7)$$

$$\text{с } a_{0k} = \int f(t) \varphi_0(t - k) dt. \quad (8)$$

Коэффициенты g_m выражения (7) связаны через h_m соотношением

$$g_k = (-1)^k h_{2M-1-k}; \quad h_k = \sqrt{2} \int \varphi(t) \varphi(2t - k) dk;$$

целое число M определяет число ненулевых h_k и длину области задания вейвлета.

Основополагающая идея вейвлет-представления сигнала $x(t)$ заключается в разбиении приближения $\tilde{x}_j(t_i)$ к сигналу $x(t_i)$ на две составляющие — грубую (аппроксимирующую $\tilde{x}_{j-1}(t_i)$) и уточненную (детализирующую) $\tilde{x}_{j-1}^d(t_i)$ с последующим их уточнением итерационным методом с помощью формулы БВП (7):

$$\begin{aligned} \tilde{x}_j(t_i) &= \tilde{x}_{j-1}(t_i) + \tilde{x}_{j-1}^d(t_i) = \\ &= \sum_{k \in Z} a_{j-1,k} \varphi_{j-1,k}(t_i) + \sum_{k \in Z} d_{j-1,k} \psi_{j-1,k}(t_i), \end{aligned} \quad (9)$$

где j — уровень разрешения; $a_j = \{a_{j-1,k}\}$, $d_j = \{d_{j-1,k}\}$ — наборы аппроксимирующих и детализирующих коэффициентов разложения $(j-1)$ -го уровня разрешения. Обычно в качестве $a_0 = \{a_{0,k}\}$ выбирается массив значений сигнала $x(t)$, $a_{0,i} = x(t_i)$.

Повторяя процедуру (9) m раз, $m = 1, \dots, m_0$, разлагая каждый раз сглаженную функцию $\tilde{x}_{j-m}(t_i)$ на еще более сглаженную часть $\tilde{x}_{j-m-1}(t_i)$ и детализирующую часть $\tilde{x}_{j-m-1}^d(t_i)$, получаем вейвлет-разложение аппроксимации j -го уровня разрешения $\tilde{x}(t)$ для глубины разложения m .

С учетом соотношения [7]

$$D \left[\sum_k w_{j,k} \psi_{j,k}(t) \right] = \sum_k w_{j,k}^2 \sim 2^{j(2H-2)} \quad (10)$$

можно провести оценку самоподобности, выделив линейный участок на графике зависимости логарифмической характеристики

$$\log_2[(2^j/n_0)] \sum_k d_{j,k}^2 \text{ от } \log_2 j. \quad (11)$$

Наклон этого графика определяется как $2\hat{H} - 1$ (\hat{H} — оценка показателя Херста H); n_0 — объем данных.

Аналитически доказано [14], что полученная таким образом оценка является несмещенной в достаточно произвольных условиях, а также эффективной при предположении гауссовской структуры данных.

В соответствии с выражением (10) линейно возрастающий участок зависимости (11) от $\log_2 j$ с показателем Херста $0,5 \leq H \leq 1$ соответствует масштабному диапазону, в котором проявляется самоподобность изучаемой реализации. На основании этого в работе [5] предложено использовать результаты вейвлет-анализа для выделения фрактального компонента исследуемой реализации как суммы восстановленных из вейвлет-разложения компонентов исходного динамического ряда для уровней разрешения j , соответствующих линейному участку зависимости (11). Обрыв линейного участка логарифмической характеристики (11) и характерный ее спад при некотором значении $j = j_f$ характеризуют предельную глубину самоподобности и связанной с ней масштабной инвариантности. В ходе обработки реальных данных ЭКГ с помощью вейвлет-анализа установлено, что для variability сердечного ритма $j_f = 4$, что, по-видимому, является характерным для данного класса физиологических процессов.

Для повышения достоверности указанной выше процедуры выделения фрактального компонента рекомендуется при наличии достаточного объема исходных данных проводить оценку на различных пространственных масштабах k . Глубина самоподобности для любого исследуемого случайного процесса может быть установлена путем проведения подобной процедуры для ряда реализаций (теоретически по всему ансамблю реализаций), хотя для многих практических приложений можно ограничиться несколькими достаточно длинными реализациями при вариации всей совокупности воздействующих факторов.

Итерационная процедура (9) позволяет получить оценку (5) трендовой составляющей $\mu(t)$ при $J > j_f$ (обозначим ее $\hat{\mu}(t)$) и выделить нерегулярную стохастическую составляющую $z(t) = x(t) - \hat{\mu}(t)$, обладающую свойством стационарности (в широком смысле), т. е. с постоянной средней m_z и ковариационной функцией $K_z(t_1, t_2) = E[z(t_1)z(t_2)]$, зависящей только от разности аргументов:

$$K_z(t_1, t_2) = K_z(t_1 - t_2) = K_z(\tau).$$

В этом случае и корреляционная функция $R_z(t_1, t_2) = E[z(t_1) - m_z](z(t_2) - m_z)$ зависит только от разности $\tau = t_2 - t_1$, $R_z = R_z(\tau)$; E — знак математического ожидания.

Для проверки процесса $z(t)$ на стационарность строится график выборочной автокорреляционной функции-коррелограммы [15]:

$$r_k = \frac{\sum_{t=k+1}^n (z(t) - m_z)(z(t-k) - m_z)}{\sum_{t=1}^n (z_t - m_z)^2}.$$

Коррелограмма стационарного временного ряда быстро убывает с ростом k .

2. Оценка точки изменения волатильности временного ряда

Рассмотрим модель временного ряда

$$Y_i = \mu(t_i) + \sigma(t_i)\varepsilon_i, \quad i = 1, 2, \dots, T, \quad (12)$$

где $\{Y_i\}$ — последовательность случайных величин и $\{\varepsilon_i\}$ — последовательность стационарных ошибок наблюдений величины Y в моменты времени t_i с $E(\varepsilon_i) = 0$ и $\text{Var}(\varepsilon_i) = 1$; $\mu(t)$ и $\sigma(t)$ — регрессионная функция (условное среднее) и функция волатильности (условная вариация), соответственно. Модель (12) является частным случаем непараметрической модели [11]

$$Y_i = \mu(X_i) + \sigma(X_i)\varepsilon_i, \quad (13)$$

когда $X = t$ и, следовательно, $X_i = t_i$.

Следуя процедуре нахождения одной точки изменения (single change-point) волатильности [12], положим

$$\sigma(t) = \begin{cases} \tau_1\sigma_0(t), & \text{если } t \leq k_0 \\ \tau_2\sigma_0(t), & \text{если } t > k_0 \end{cases}, \quad (14)$$

где τ_1 , τ_2 и k_0 — неизвестные параметры. Для упрощения положим $k_0 = [T\theta_0]$, $0 < \theta_0 < 1$, где $[a]$ означает наибольшее целое число, не превосходящее a .

Пусть гипотеза H_0 означает, что волатильность не имеет точек изменения. При выполнении этой гипотезы из (12) следует, что

$$E(Y_i - \mu(t_i))^2 = \sigma^2(t_i).$$

Введем обозначения

$$Z_i = (Y_i - \mu(t_i))/\sigma_0(t_i), \\ S_T = \sum_{i=1}^T Z_i^2, S_k = \sum_{i=1}^k Z_i^2, S_{T-k} = \sum_{i=k+1}^T Z_i^2 \quad (15)$$

и определим

$$V_k = \left(\left(\frac{k(T-k)}{T^2} \right) \right)^{1/2} \left(\frac{1}{T-k} S_{T-k} - \frac{1}{k} S_k \right). \quad (16)$$

При известных функциях $\mu(t)$ и $\sigma_0(t)$ методом наименьших квадратов (МНК) для точки скачка волатильности k_0 можно получить следующую оценку [11]:

$$\hat{k} = \arg \max |V_k|. \quad (17)$$

При выполнении гипотезы H_0 выражения $S_{T-k}\sigma_0(t)/(T-k)$ и $S_k\sigma_0(t)/k$ будут несмещенными оценками для общей волатильности. Разность $(1/(T-k))S_{T-k} - (1/k)S_k$ (и, стало быть, V_k) будет близка к 0 при выполнении гипотезы H_0 и отлична от нуля, если волатильность изменяется. Простые вычисления приводят к равенству

$$V_k = \left(\frac{(T-k)k}{T^2} \right)^{1/2} \left(\frac{1}{T-k} S_{T-k} - \frac{1}{k} S_k \right) = \\ = \left(\frac{1}{k(T-k)} \right)^{1/2} S_T D_R, \quad (18)$$

где

$$D_k = \frac{k}{T} - \frac{S_k}{S_T}. \quad (19)$$

Из соотношения (18) следует, что D_k также можно использовать в качестве оценки точки изменения k_0 :

$$\hat{k} = \arg \max_k |D_k| = \arg \max_k (k(T-k))^{1/2} V_k. \quad (20)$$

Вообще говоря, можно ввести следующую статистику обнаружения изменения волатильности [11]:

$$V_k^v = \left(\frac{k}{T} \left(1 - \frac{k}{T} \right) \right)^{\frac{1}{2}-v} V_k, \quad 0 \leq v \leq 1/2. \quad (21)$$

Статистика D_k , определяемая формулой (19), лишь по знаку отличается от статистики

$$D_k = \frac{S_k}{S_T} - \frac{k}{T}, \quad \text{используемой в работе [12].}$$

Из выражения (21) можно получить оценку для точки изменения k_0 :

$$\hat{k}(v) = \arg \max_k |V_k^v|. \quad (22)$$

При проверке гипотезы $H_0: \lambda_T = 0$ против гипотезы H_1 , где $\lambda_T = \tau_2^2 - \tau_1^2$, множитель $\left(\frac{k}{T} \left(1 - \frac{k}{T} \right) \right)^{\frac{1}{2}}$ при $0 \leq v \leq \frac{1}{2}$ обеспечивает явное преимущество статистике V_k , поскольку она становится очень близкой к смежной альтернативе $\lambda_T \sim T^{1/2}$ из работы [16].

Задавая $\hat{k}(v)$, можно получить следующие оценки величин τ_1^2 и τ_2^2 :

$$\hat{\tau}_1^2(v) = \frac{1}{\hat{k}(v)} S_{\hat{k}(v)}, \quad \hat{\tau}_2^2(v) = \frac{1}{T - \hat{k}(v)} S_{T - \hat{k}(v)}.$$

Для анализа асимптотических свойств оценок точек изменения волатильности вводятся следующие условия [11].

Условие J_1 . Скачок функции условной вариации $\lambda_T = \tau_2^2 - \tau_1^2$ есть постоянная величина.

Условие J_2 . Значение скачка стремится к нулю при неограниченном росте размера выборки, т. е. $\lambda_T \rightarrow 0$ и $T\lambda_T^2 / \log T \rightarrow \infty$ при $T \rightarrow \infty$.

Условие (AS.1). Последовательность $\{t_i, \varepsilon_i\}$ удовлетворяет одному из двух альтернативных условий:

а) пусть $\mathcal{F}_i = \{t_1, t_2, \dots, t_i, \varepsilon_1, \varepsilon_2, \dots, \varepsilon_{i-1}\}$. Предположим, что $\{\varepsilon_i\}$ — последовательность мартингал-разностей (с дискретным временем) относительно $\{\mathcal{F}_i\}$, т. е. $E|\varepsilon_i| < \infty$ для всех $i \geq 0$ и $E(\varepsilon_{i+1} / \mathcal{F}_i) = 0$ и $\sup_i E|\varepsilon_i|^{4+\delta} < \infty$ для некоторого $\delta > 0$.

б) $\{\varepsilon_i\}$ — строго стационарная последовательность и удовлетворяет условию перемешивания с коэффициентами перемешивания $\alpha(n)$, для которых $\sum_{n=1}^{\infty} (\alpha(n))^{\delta/2+\delta} < \infty$, $\delta > 0$. К тому же, $E|\varepsilon_i|^{4+\delta} < \infty$.

Условие перемешивания для стационарных последовательностей определяется следующим образом [17].

Пусть $\{X_n, n \geq 1\}$ — стационарная последовательность случайных величин на вероятностном пространстве (с координатами Ω, \mathcal{F}, P). Определим для любых множеств $A, B \subset \mathcal{F}$ величины

$$\rho(A, B) = \sup \{ |\text{corr}(X, Y)|, X \in L_2(A), Y \in L_2(B) \}; \quad (23)$$

$$\rho(n) = \sup_k \rho(X_i 1 \leq i \leq k), \sigma(X_i, i \geq k+n).$$

Последовательность $\{X_n, n \geq 1\}$ называется ρ -перемешанной, если $\{\rho(n) \rightarrow 0\}$ при $n \rightarrow \infty$.

При условиях J_1, J_2 и (AS1) верно асимптотическое предельное соотношение по вероятности [11]:

$$\hat{k}(v) - k_0 = O_P(1/\lambda_T^2) \quad (24)$$

для $0 \leq v \leq 1/2, k_0 = [\theta_0, T]$ для некоторого $0 < \theta_0 < 1$. Равенство (24) означает что при $T \rightarrow \infty \hat{k}(v)$ сходится к k_0 по вероятности, т. е. $P\{|\hat{k}(v) - k_0| > \varepsilon\} \rightarrow 0$ при $T \rightarrow \infty$ со скоростью $1/\lambda_T^2$. При $0 \leq v \leq 1/2$ условие J_2 может быть ослаблено так, что $\lambda_T \rightarrow 0$ и $T\lambda_T^2 \rightarrow \infty$ при $T \rightarrow \infty$. Оценка точки изменения $\theta_v = \hat{k}(v)/T$ для $0 \leq v \leq 1/2$ при тех же условиях, что и в (24), сходится по вероятности к точной точке изменения $\theta_0 = k_0/T$ со скоростью $(T\lambda_T^2)^{-1}$, т. е.

$$|\theta_v - \theta_0| = O_P((T\lambda_T^2)^{-1}). \quad (25)$$

Для установления асимптотического распределения статистики V_k^v при известных функциях регрессии $\mu(t)$ и условной вариации $\sigma(t)$ вводится обозначение

$$V_T^v(t) = \begin{cases} 0, & \text{если } 0 \leq t < 1/(T+1); \\ V_{[(T+1)t]}^v, & \text{если } 1/(T+1) \leq t \leq T/(T+1); \\ 0, & \text{если } T/(T+1) \leq t \leq 1. \end{cases} \quad (26)$$

Если $\tau_1 = \tau_2 = 1$, т. е. верна гипотеза H_0 , то при условии (AS.1) выполняется предельное соотношение [11]

$$\lim_{T \rightarrow \infty} \frac{\sqrt{T} \sigma}{\sigma_w} \sup_{\delta < t < 1-\delta} V_T^v(t) \stackrel{d}{\rightarrow} \sup_{\delta < t < 1-\delta} (t(1-t))^{-v} |B(t)|, \quad (27)$$

где \xrightarrow{d} означает сходимость по распределению; δ — любое число из интервала $(0,1)$; $\{B(t), 0 \leq t \leq 1\}$ — броуновское движение на $[0,1]$; $\sigma = \lim_{T \rightarrow \infty} S_T/T$ и

$$\sigma_w^2 = E(Z_1^2 - EZ_1^2)^2 + 2 \sum_{i=2}^T E((Z_1^2 - EZ_1^2)(Z_i^2 - EZ_i^2)). \quad (28)$$

Аппроксимация распределения случайной величины $\left(\frac{1}{x^4}\right)$

$$\sup_{\delta < t < 1-\delta} (t(1-t))^{-1/2} |B(t)|$$

из правой части (27) с $v = 1/2$ получается с помощью формулы [16]

$$P \left\{ \sup_{h < t < 1-l} \left(\frac{B^2(t)}{t(1-t)} \right)^{1/2} \geq x \right\} = \frac{x \exp\{-x^2/2\}}{(2\pi)^{1/2}} \left\{ \log \frac{(1-h)(1-l)}{hl} - \frac{1}{x^2} \log \frac{(1-h)(1-l)}{hl} + \frac{4}{x^2} + 0 \left(\frac{1}{x^4} \right) \right\}, \quad (29)$$

если положить в ней $h = l = \delta$. Это дает возможность рассчитать асимптотические критические значения правой части (27).

В случае наличия скачка волатильной функции, т. е. в случае гетероскедастичной регрессионной модели с $\sigma^2(t) \neq \text{const}$ (при $\sigma^2(t) = \text{const} = \sigma^2$ регрессионная модель называется гомоскедастичной), оценка точки изменения определяется как [11]

$$\hat{k} = \arg \max_k |\tilde{V}_k^v|, \quad (30)$$

где

$$\tilde{V}_k^v = \left(\frac{k(T-k)}{T^2} \right)^{1-v} \left(\frac{1}{T-k} \tilde{R}_{T-k} - \frac{1}{k} \tilde{R}_k \right), \quad (31)$$

$$\tilde{R}_k = \sum_{i=1}^k \tilde{W}_i^2, \quad \tilde{R}_{T-k} = \sum_{i=k+1}^T \tilde{W}_i^2, \quad \tilde{R}_T = \sum_{i=1}^T \tilde{W}_i^2; \quad (32)$$

$$\tilde{W}_i = Y_i - \hat{\mu}(t_i), \quad (33)$$

$\hat{\mu}(t)$ — оценка регрессионной функции $\mu(t)$. Предельные соотношения (24), (25) и (27) остаются справедливыми при замене $\hat{k}(v)$ и \tilde{V}_k на $\hat{k}(v)$ и \tilde{V}_k , соответственно, если выполнены

условия J_1, J_2 , а вместо (AS.1) — следующие условия (AS.2)—(AS8) из работы [11]:

(AS.2): $\mu(t)$ и $\sigma_0(t)$ непрерывны на $[a_1, a_2]$ вместе с производными до третьего порядка включительно;

(AS.3): плотность распределения $f(t)$ величин $\{t_i\}$ ограничена с $M < f(t) < M'$ (M, M' — положительные числа) и непрерывна на $[a_1, a_2]$ вместе с производными до второго порядка включительно;

(AS.4): функции условных плотностей $f_{t|y_i}$ и $f_{(t_i, t_j)(y_i, y_j)}(t_i, t_j | (Y_1, Y_e))$ ограничены при всех $l > 0$;

(AS.5): $E|t|^l < \infty$ и $E|Y|^l < \infty$ при достаточно большом $l = 0$;

(AS.6): ядерная функция $K(\cdot)$, используемая для оценки $\sigma_0^2(x)$, имеет симметрическую функцию плотности с носителем из интервала $[-c_e, c_0]$ с ограниченной производной, и преобразование Фурье-функций $K(\cdot)$ абсолютно интегрируемо;

(AS.7): ширина b_n полос ядерной функции $K(\cdot)$ удовлетворяет неравенству

$$c_1 n^{-1/5} \leq b_n \leq c_2 n^{-1/5},$$

где c_1, c_2 — некоторые положительные постоянные (в расчетах нами принято $b_n = n^{-1/5}$);

(AS.8): $(t_i, \varepsilon_i), i = 1, 2, \dots$ — строго стационарная последовательность и удовлетворяет условию перемешивания с коэффициентом $\alpha(n) = O(c^n)$ для некоторого $0 < c < 1$.

Условие (AS.8) более сильное, чем условие (AS.1) Поскольку последовательность времен $\{t_i\}$ состоит из независимых и одинаковых распределенных (с равномерным распределением) величин, условия (AS.3) и (AS.4) выполняются автоматически.

При наличии точки изменения k_0 у волатильности $\sigma(t)$ оценка для $\sigma_0(t)$ при условии $E(\varepsilon^2|t) = 1$ определяется обычно формулой [20]

$$\hat{\sigma}_s^2(t) = \frac{\sum_{i=1}^n \mathcal{K}_{n,b}(t)(t_i - t)(Y_i - \hat{\mu}(t_i))^2}{\sum_{i=1}^n \mathcal{K}_{n,b}(t_i - t)}, \quad (34)$$

где $1 \leq n \leq k_0; n \rightarrow \infty$ при $T \rightarrow \infty$ в случае выполнения альтернативной гипотезы H_1 ($\sigma(t)$ имеет на $[0, T]$ точку изменения k_0) и $n = T$ в случае выполнения нулевой гипотезы H_0 ($\sigma(t)$ не имеет на $[0, T]$ точку изменения).

В формуле (34)

$$\mathcal{K}_{n,b}(t) = \frac{1}{b} K\left(\frac{t_i - t}{b}\right), \quad (35)$$

$K(\cdot)$ — ядерная функция и $b = b_n$ — последовательность ширин полос. Выбор ядерной функции $K(\cdot)$ и последовательности b_n обсуждается ниже.

Как известно, сглаживание полинома второго порядка $Y^2(t)$ или $(Y(t) - \hat{\mu}(t))^2$ чувствительно к выбросам. В частности, оценка (34) имеет очень большое смещение, когда имеется некоторое число выбросов или функция распределения наблюдений имеет "тяжелые хвосты". В связи с этим целесообразно использовать для оценки $\sigma_0(t)$ оценку абсолютного отклонения [21]:

$$\hat{\sigma}_d^2(t) = \frac{\sum_{i=1}^n y \mathcal{K}_{i,n,b}(t_i - t) |Y_i - \hat{\mu}(t_i)|}{\sum_{i=1}^n \mathcal{K}_{i,n,b}(t_i - t)}. \quad (36)$$

В качестве оценки величины σ в (27) принимается $\hat{\sigma} = s_T/T$. Состоятельную оценку для величины σ_w^2 , являющейся дисперсией случайной величины $Z_t = (Y_t - \mu(t))/\sigma_0(t) = \varepsilon_t$, получим на основе анализа [17] выборочной стационарной последовательности $\{\hat{Z}_i, i = 1, \dots, T\}$, $\hat{Z}_i = (Y_i - \hat{\mu}(t_i))/\hat{\sigma}_0(t_i)$ с $\hat{\sigma}_0(t_i) = \hat{\sigma}_d(t_i)$, где $\hat{\sigma}_d^2(t)$ определяется формулой (36), а $\hat{\mu}(t)$ — оценка $\bar{x}(t)$, получаемая вейвлет-разложением (5) с $J > j_f$.

Будем полагать, что $\hat{Z}_n, n \geq 1$, есть стационарная последовательность случайных величин, обладающая свойством ρ -перемешивания при $n \rightarrow \infty$, т. е. согласно (23)

$$\rho(n) = \sup_k \rho(\sigma(\hat{Z}_i, 1 \leq i \leq k), \sigma(\hat{Z}_i, 1 \geq k + n)).$$

Пусть

$$E\hat{Z}_1 = \mu_1, E\hat{Z}_1^2 < \infty, S_n = S(n) = \sum_{i=1}^n \hat{Z}_i. \quad (37)$$

Предположим, что $\sigma_n^2 = \text{Var}\hat{S}_n \rightarrow \infty$ и $\sum_{n=1}^{\infty} \rho(2^n) < \infty$. Тогда

$$\sigma_n^2/n \rightarrow \sigma^2 \text{ при } n \rightarrow \infty, \quad (38)$$

где σ — некоторое положительное число; Var — вариабельность значения [18].

Пусть далее $\{l_n, n \geq 1\}$ — последовательность целых положительных чисел с $1 \leq l_n \leq n$. Обозначим

$$S_j(k) = \sum_{i=j+1}^{j+k} \hat{Z}_i, \bar{Z}_n = \frac{1}{n} \sum_{i=1}^n \hat{Z}_i, \quad (39)$$

$$C_p = 2^{-p/2} \sqrt{\pi} \Gamma\left(\frac{p+1}{2}\right), p \geq 1,$$

где Γ — гамма-функция.

Введем следующие статистики:

$$B_{n,p} = \left\{ \frac{C_p}{n - l_n} \sum_{j=0}^{n-l_n} \left(\frac{|S_j(l_n) - l_n \bar{Z}_n|}{\sqrt{l_n}} \right)^p \right\}^{1/p}; \quad (40)$$

$$\hat{B}_{n,p} = \left\{ \frac{C_p}{n - \hat{l}_n + 1} \sum_{j=0}^{n-\hat{l}_n} \left(\frac{|S_j(l_n) - l_n \mu_1|}{\sqrt{l_n}} \right)^p \right\}^{1/p}. \quad (41)$$

Если $l_n \rightarrow \infty$ и $l_n/n \rightarrow 0$ при $n \rightarrow \infty$, тогда при $n \rightarrow \infty$

$$B_{n,p} \rightarrow \sigma; \quad (42)$$

$$\frac{\hat{S}_n - n\mu_1}{B_{n,p}\sqrt{n}} \xrightarrow{d} N(0,1), \quad (43)$$

где σ — то же, что и в выражении (38), при этом (43) остается справедливым при замене $B_{n,p}$ на $\hat{B}_{n,p}$; \xrightarrow{d} означает сходимость по распределению, что эквивалентно сходимости функции распределения левой части $F_n(x)$ к функции распределения правой части $F(x)$ в каждой точке x , где $F(x)$ непрерывна [19].

Таким образом, при вышеуказанных условиях на последовательность $\{\hat{Z}_n, n \geq 1\}$ в качестве оценки величины σ_w можно принять $B_{n,p}$ или $\hat{B}_{n,p}$ (в расчетах принималось $p = 1$ с $C_1 = \Gamma(1)$ и $l_n = \ln(n)$).

3. Выбор ядерной оценки и ширин полосы в оценках волатильности

В случае гладкой функции волатильности $\sigma(x)$ (т. е. функция $\sigma(x)$ непрерывна вместе с производными до некоторого конечного порядка $s \geq 1$) в ядерных оценках (34), (36), где $\mathcal{K}_{n,b}(t)$ представляется в виде (35), в качестве функции $\mathcal{K}(t)$ можно принять стандартизованное ядро Епанечникова

$$K(t) = 3(1 - t^2/5) | (t^2 \leq 5) / (4\sqrt{5}) \quad (44)$$

или ядро Гаусса

$$K(t) = \frac{1}{\sqrt{2\pi}} \exp\{-t^2/2\}. \quad (45)$$

Для оценки ширины полос b_n в выражении (35) воспользуемся методом, использующим χ^2 -статистику $R^2(y, \hat{\mu}, \hat{\sigma}_{b_n})$, которая определяется как [22]

$$R^2(y, \hat{\mu}, \hat{\sigma}_{b_n}) = \sum_{i=1}^n \frac{(Y_i - \hat{\mu}(t_i))^2}{\hat{\sigma}_{b_n}^2(t_i)}, \quad (46)$$

где $y = (Y_1, \dots, Y_n)$; $\hat{\sigma}_{b_n}^2$ — оценка $\hat{\sigma}_d^2(\cdot)$, определяемая формулой (36) с шириной полосы $b = b_n$; ER^2 — ожидаемое значение χ^2 -статистики Пирсона (46), приближенно равное числу степеней свободы n [22]. Так как R^2 зависит от оцениваемой (условной) вариации, то равенство (46) можно использовать для оптимального выбора ширины полосы b_n следующим образом:

$$\hat{b}_{n,\text{opt}} = \arg \min_{b_n} |R^2(y, \hat{\mu}, \hat{\sigma}_{b_n}) - n|. \quad (47)$$

4. Обнаружение нескольких точек изменения функции волатильности

Для обнаружения точек изменения (multiple change points) функции волатильности в работе [12] был предложен итеративный метод центрированных кумулятивных сумм квадратов (iterative cumulative sums of squares, ICSS). Суть метода заключается в следующем.

Рассмотрим стационарный временной ряд $\{a_t\}$ с нулевым средним значением и дисперсиями $\sigma_t^2, t = 1, \dots, T$. Пусть $C_k = \sum_{i=1}^k a_i$ — кумулятивная (накопленная) сумма ряда $\{a_t\}$ и

$$D_k = \frac{C_k}{C_T} - \frac{k}{T}, k = 1, \dots, T, \quad D_0 = D_T = 0 \quad (48)$$

— центрированная (и нормализованная) кумулятивная сумма квадратов.

Для постоянной дисперсии $\sigma_t = \sigma$ последовательность $\{D_k\}$ осциллирует относительно нуля. Если возможны внезапные изменения дисперсии σ_t , то график зависимости D_k от k с высокой вероятностью будет заключен в некоторых границах. Эти границы вычисляются из анализа асимптотического распределения D_k относительно некоторой постоянной дисперсии.

Пусть

$$k^* = \arg \max_k |D_k|. \quad (49)$$

Если этот максимум превышает упомянутые выше граничные уровни $|D_k|$ с уровнем значимости α , то k^* можно принять за оценку точки изменения дисперсии.

В работе [11] предложен модифицированный алгоритм ICSS (назовем его MICSS), в котором статистика D_k заменяется на V_k^v . Критическое значение $c_\alpha^v(T)$ при $v = 0$ заданы в работе [12], а при $v = 1/2$ — в работе [16]. Значения $c_\alpha^v(T)$ при $v = 1/2$ нетрудно подсчитать, как будет показано ниже, с помощью неравенства (29). Обычно за уровень значимости применяется $\alpha = 0,05$.

В методе MICSS на шаге 1

$$M(t_1:t_2) = \max_{t_1 < k < t_2} \sqrt{(t_2 - t_1 + 1)/2} |D_k(a[t_1:t_2])| \quad (50)$$

заменяется статистикой

$$M(t_1:t_2) = \max_{t_1 < k < t_2} \frac{\hat{\sigma} \sqrt{t_2 - t_1 + 1}}{\hat{\sigma}_w} |\tilde{V}_k^v(a[t_1:t_2])|, \quad (51)$$

где $a[t_1:t_2]$ означает, что выборка значений a_t берется при $t_1 \leq t \leq t_2$, где $t_1 < t_2$, а $\tilde{V}_k^v(a[t_1:t_2])$ — статистика (31), определенная на интервале $[t_1, t_2]$.

В равенстве (51) оценка $\hat{\sigma}$ определяется по формуле

$$\hat{\sigma} = \frac{1}{T} \tilde{R}_T = \frac{1}{T} \sum_{i=1}^T (Y_i - \hat{\mu}(t_i)), \quad (52)$$

а $\hat{\sigma}_w$ — по формуле (40) с $p = 1$, где

$$\hat{Z}_i = \frac{(Y_i - \hat{\mu}(t_i))}{\hat{\sigma}_0(t_i)}, \hat{S}_j(l_n) = \sum_{i=j+1}^{j+l_n} \hat{Z}_i, \bar{\hat{Z}}_i = \sum_{i=j+1}^n \hat{Z}_i,$$

а $\hat{\sigma}_0(t_i) = \hat{\sigma}_d(t_i)$, где $\hat{\sigma}_d(t)$ определяется в (36).

Оценки $\hat{\sigma}$ и $\hat{\sigma}_0(t_i)$, зависящие от n ($n = T$), являются состоятельными в том смысле, что при $n \rightarrow \infty$ они сходятся по вероятности к своим предельным значениям, т. е. вероятность того, что их отклонения от предельных значений (σ и $\sigma_0(t_i)$ соответственно) превышают любое заданное $\varepsilon > 0$, стремится к нулю при $n \rightarrow \infty$ [19].

Численная реализация алгоритма обнаружения изменения волатильности нестационарного временного ряда проводится в два этапа. На первом этапе по заданным значениям

Y_1, \dots, Y_{n_0} временного ряда вычисляются коэффициенты $w_{j,k}$ $k \in Z, j = 1, \dots, J, (J \gg 4)$ в вейвлет-разложении (5) с использованием пакета Wavelet Toolbox MATLAB [15]. Строим график зависимости логарифмической характеристики (11) от $\log_2 j$. Наклон линейно возрастающего участка этой зависимости определяет показатель Херста H , а его спад — границу j_f интервала самоподобности временного ряда. Задаваясь интервалом дискретизации Δt по времени t , вычислим вейвлет-разложение (5) при уровне разрешения $J > j_f$ и примем это разложение за оценку $\hat{\mu}(t)$ тренда $\mu(t)$ исходного ряда. Рассчитаем значения $\mu(t_i)$ во всех точках интервала наблюдения $[1, T]$ с шагом Δt (в расчетах принималось $\Delta t = 0,01$).

На втором этапе для вычисления статистики \tilde{V}_k^v , определяемой формулой (31), по формулам (32) проводится расчет величин $\tilde{W}_i, \tilde{R}_k, \tilde{R}_{T-k}, \tilde{R}_T$, по формулам (36) — величины $\hat{\sigma}_d^2(t_i)$, по которым получают оценки $\hat{\sigma}_0(t_i) = \hat{\sigma}_{b_n}(t_i)$, и оптимальная ширина l_n полосы b_n оценки $\hat{\sigma}_d(t_i) = \hat{\sigma}_{b_n}(t_i)$ — по формуле (47). Оценка $\hat{\sigma}_w$ принимается равной величине $B_{n,p}$ с $p = 1$, вычисляемой по формуле (40). За ядерную функцию $K(t)$ при расчете $\hat{\sigma}_d(t_i)$ по формуле (46) принимается функция Епанечникова (44). Величина V_k^v рассчитывается при $v = 0$ и $v = 1/2$ по формуле (31). Для обнаружения точек изменения волатильности $\sigma(t)$ модели временного ряда (12) будем использовать алгоритм MICSS.

Критические значения статистики (50) при различных $T^* = t_2 - t_1 + 1$ определены в таблице для разных уровней значимости α ($\alpha = 1 - p$).

T^*	100	200	300	400	500	∞
p	SE	SE	SE	SE	SE	
0,05	0,003	0,003	0,003	0,003	0,003	0,520
0,10	0,003	0,003	0,003	0,003	0,003	0,571
0,25	0,004	0,003	0,003	0,003	0,003	0,677
0,50	0,004	0,003	0,003	0,003	0,003	0,828
0,75	0,004	0,004	0,004	0,004	0,004	10,019
0,90	0,006	0,006	0,007	0,006	0,006	1,224
0,95	0,009	0,004	0,008	0,010	0,009	1,358
0,99	0,004	0,012	0,028	0,020	0,018	1,628

Стандартные ошибки в этой таблице получены как $SE = \sqrt{p(1-p)/10,000}$, где p — доля

серий с $\max_k \sqrt{T/2} |D_k| < 1,358$. При $\alpha = 0,05$ (именно такой уровень значимости принимался нами в расчетах) асимптотические значения (при больших T) определяются как $c_{\alpha,0}(T^*) = 1,27; 1,30; 1,31; 1,31; 1,33; 1,358$ при $T^* = 100; 200; 300; 400; 500; \infty$, соответственно, будем предполагать, что

$$c_{\alpha,0}(T^*) = c_{\alpha,0}(100i) + \left(\frac{T^*}{100} - i \right) \times (c_{\alpha,0}(100(i+1)) - c_{\alpha,0}(100i)),$$

если

$$100i < T^* < 100(i+1) \text{ при } i = 1, 2, 4 \\ \text{и } c_{\alpha,0}(T^*) = 1,31 \text{ при } 300 \leq T^* \leq 400.$$

Из соотношений (18), (22) следует

$$|V_k^0| = \left(\frac{k}{T} \left(1 - \frac{k}{T} \right) \right)^{1/2} \frac{1}{(k(T-k))^{1/2}} S_T |D_k| = \frac{S_T}{T} |D_k|.$$

С учетом оценки $\hat{\sigma} = S_T/T$ статистику (51) при $v = 1/2$ можно записать в виде

$$M(t_1:t_2) = \max_k \frac{\hat{\sigma}^2 \sqrt{t_2 - t_1 + 1}}{\hat{\sigma}_w} |D_k|. \quad (53)$$

Следовательно, критические значения статистики (51) при $v = 1/2$ (обозначим их $\tilde{c}_{\alpha,0}(T^*)$)

отличаются от $c_{\alpha,0}(T^*)$ множителем $\frac{\hat{\sigma} \sqrt{2}}{\hat{\sigma}_w}$, т. е. $\hat{c}_{\alpha,0}(T^*) = \frac{\hat{\sigma} \sqrt{2}}{\hat{\sigma}_w} c_{\alpha,0}(T^*)$, где $T^* = t_2 - t_1 + 1$.

Для определения критических значений статистики (51) при $v = C$ (обозначим их $\tilde{c}_{\alpha,1/2}(T^*)$) можно воспользоваться асимптотическим соотношением (27). С этой целью осуществим замену $\tilde{t} = t/T$, при которой $t = t_1 \rightarrow \tau = 1/T$ и $t = t_2 \rightarrow \tau = t_2/T$, при этом интервал $t_1 \leq t \leq t_2$ отображается в интервал $\tilde{t}_1 \leq \tilde{t} \leq \tilde{t}_2$.

Пусть $\delta = \frac{1}{T(T+1)}$. Тогда отрезки $[1, T]$ и $\left[\frac{1}{T+1}, \frac{T}{T+1} \right]$ оси t после замены $\tilde{t} = t/T$ отображаются в отрезки $\left[\frac{1}{T}, 1 \right]$ и $\left[\frac{1}{T(T+1)}, \frac{1}{T+1} \right]$ оси \tilde{t} соответственно, при этом отрезок $\frac{1}{T(T+1)} \leq \tilde{t} \leq \frac{1}{T+1}$ находится внутри отрезка

$\delta \leq \tilde{t} \leq 1 - \delta$. Поэтому равенство (26) на оси \tilde{t} запишется в виде $V_T^v(\tilde{t}) = V_{\left[\frac{T+1}{T}\right]}^v$, а соотношение (27) можно записать на оси как

$$\lim_{T \rightarrow \infty} \frac{\sqrt{T} \hat{\sigma}}{\hat{\sigma}_w} \sup_{\tilde{\delta} < \tilde{t} < 1 - \tilde{\delta}} |\tilde{V}_T^v(\tilde{t})|^d \rightarrow \sup_{\tilde{\delta} < \tilde{t} < 1 - \tilde{\delta}} (\tilde{t}(1 - \tilde{t}))^{-\nu} |\tilde{B}(\tilde{t})|, \quad (54)$$

где $\tilde{\delta} = \delta/T$, $\tilde{B}(\tilde{t}) = B(t)|_{t=T\tilde{t}}$.

Обозначим α правую часть уравнения (29) без слагаемого $O\left(\frac{1}{x^4}\right)$. Решим относительно x нелинейное уравнение

$$x \exp\{-x^2/2\} / \sqrt{2\pi} \times \left\{ 2 \log\left(\frac{1}{\delta} - 1\right) \left(1 - \frac{1}{x^2}\right) + \frac{4}{x^2} \right\} = \alpha, \quad (55)$$

полученное из правой части (29) при $h = l = \tilde{\delta}$. При начальном условии $x = x_0$ (например, $x_0 = 100$) найдем решение x^* уравнения (55) при фиксированном α в системе MATLAB.

Тогда величину $\tilde{c}_{\alpha, |2}(T^*) = \sqrt{\frac{T^*}{T}} x^*$ можно принять за критическое значение статистики (51) при $\nu = \frac{1}{2}$.

Заключение

1. Под сегментированием обычно понимается задача разделения имеющегося временного ряда на сегменты (периоды) с разной динамикой.

2. Если представить поведение сложной системы с помощью адекватной модели, то переходному процессу будет соответствовать переход параметров модели в пространстве состояний от одной устойчивой фазовой траектории к другой.

3. В случае достаточно большого числа отсчетов в выборке и числа реализаций динамических рядов процедура сегментации сигнала на стационарные фрагменты должна давать однозначные результаты. Если сравнивать между собой распределения и дисперсии в двух последовательных фрагментах временного ряда, то в точке их стыка можно констатировать нарушение стационарности процесса в узком смысле этого слова.

4. Оценка статистических свойств физиологического сигнала возможна, однако, только

на некотором конечном временном интервале. При этом сократить оцениваемый интервал путем увеличения числа отсчетов в единицу времени за счет большей частоты оцифровки изучаемого непрерывного физиологического процесса можно только до определенного предела: до того момента, когда соседние отсчеты не станут сильно коррелированными. Именно этим обстоятельством ограничивается возможность распространения теоретического определения стационарности на реальные физические процессы, поэтому было введено в практику понятие "квазистационарность", выделяющее понимание стационарности в таком ограничительно оценочном аспекте.

5. Наличие самоподобности, т. е. масштабной инвариантности по отношению к основным статистическим характеристикам, приводит к непрерывному повторению в интервале самоподобия свойств некоторых компонентов исследуемого динамического ряда, означая неинформативность этих компонентов для оценки происходящих изменений.

6. Наиболее свободной от различных ограничений является методика оценки самоподобности, основанная на дискретном вейвлет-преобразовании динамического ряда в рамках мультиразрешающего анализа. С помощью суммы восстановленных из вейвлет-разложения компонентов исходного динамического ряда выделяется фрактальная компонента с установлением глубины самоподобности изучаемого случайного процесса и находится оценка его трендовой составляющей.

7. С использованием регрессионной модели временного ряда с волатильностью, представляемой динамическим рядом после режектирования (удаления) его трендовой составляющей, можно определить точки изменения (change-points) волатильности, которые разбивают исходный ряд на квазистационарные сегменты, соответствующие скачкам функции волатильности. С этой целью применяется специальная вычислительная процедура, обобщающая итеративный метод центрированных кумулятивных сумм квадратов.

Список литературы

1. **Kantz H., Schreiber T.** Nonlinear Time series Analysis. Cambridge, Cambridge Univ. Press, 2003. 388 p.
2. **Hegger R., Kantz H., Matassini L.** Coping with Nonstationarity by Oerembedding // Physical Review Letters. 2000. Vol. 84. P. 3197.
3. **Theory and Application of long-range dependence** / Edited by P. Donkhan, G. Oppenheim, M. S. Taqqu. Boston: Birkhauser, 2002. P. 355–367.

4. **Ivanov P. Ch., Amaral L. A. N., Goldberger A. L., Stanley H. E.** Stochastic feedback and regulation of biological rhythms // *Europhys. Lett.* 1998, 43(4). P. 363–368
5. **Богачев М. И.** Исследования влияния фрактальных свойств динамических рядов на оценку параметров их нестационарных фрагментов // *Изв. вузов России. Радио-электроника.* 2006. Вып. 3. С. 3–12.
6. **Change-point problems.** Lecture notes and monograph series / Edited by E. Carlstein, H. G. Muller, D. Siegmund. Hayward, CA: Institute of Mathematical Statistics, 1994. Vol. 23. 385 p.
7. **Peintgen H. O., Jurgens H., Saupe D.** Chaos and fractals, New frontiers of science. New York, Springer-Verlag, 1992. 984 p.
8. **Малла С.** Вейвлеты в обработке сигналов. М.: Мир, 2005. 671 с.
9. **Дремин И. М., Иванов О. В., Нечитайло В. А.** Вейвлеты и их наименования // *УФН.* 2001. Т. 171, № 5. С. 465–501.
10. **Marek M.** Heart Rate Variability: Standards of measurement, Physiological interpretation and clinical use // *Circulation.* 1996. Vol. 93(5). P. 1043–1065.
11. **Chen G., Choi Y., Zhou Y.** Nonparametric estimation of structural change points in volatility models for time series // *Journal of Econometrics.* 2005. Vol. 12. P. 79–114.
12. **Inclan C., Tiao G.** Use of Cumulative Sums of square for retrospective detection of changes of variance // *Journal of American Statistical Association.* 1994. Vol. 89, N. 427. P. 913–923.
13. **Mandelbrot B. B., Van Ness J. W.** Fractional Brownian motions, fractional noises and applications // *SIAM Rev.* 1968. Vol. 10. P. 422–437.
14. **Abry P., Veitch D.** Wavelet analysis of long range dependence traffic // *IEEE Trans. Of Inf. Th.* 1998. Vol. 44, N. 1. P. 2–15.
15. **Смоленцев Н. К.** Основы теории вейвлетов. Вейвлеты в МАТЛАБ. М.: ДМК пресс, 2008. 448 с.
16. **Csorgo M., Horvath L.** Limit Theorems in Change-point analysis. New York: Wiley, 1997. 438 p.
17. **Poligrad M., Shao Q. M.** Estimation of the Variance of partial Sums for ρ Mixing random variables // *Journal of multivariate analysis.* 1995. Vol. 52. P. 140–157.
18. **Poligrad M., Shao Q. M.** Self-normalizing central limit theorem for sums of weakly dependent random variables // *J. Theoret. Probab.* 1994. Vol. 7. P. 309–338.
19. **Ширяев А. Н.** Вероятность. М.: Наука, 1989. 640 с.
20. **Fan J., Yao Q.** Efficient estimation of conditional variance functions in stochastic regression // *Biometrika.* 1998. Vol. 85. P. 645–660.
21. **Xia Y., Tong H., Li W. K.** Abundance deviation estimation of volatility in nonparametric models // Research report No. 177. Department of Statistics and actuarial science. The University of Hong Kong, Hong Kong, 1988. P. 133–173.
22. **Choi J. M., Muller H. G.** Nonparametric quasilielihood // *The annual of statistics.* 1999. Vol. 27. P. 36–64.

N. T. Abdullaev¹, Ph. D., Head of Department, e-mail: a.namik46@mail.ru,
O. A. Dyshin², Ph. D., Associate Professor, **I. D. Ibrahimova**¹, Assistant, e-mail: irada432@gmail.com,
Kh. R. Ahmadova¹, Ph. D., e-mail: yubaba66@hotmail.com,
¹Azerbaijan Technical University,
²Azerbaijan State Oil and Industrial University

Segmentation of Non-Stationary Physiological Signals with Fractal Properties

Segmentation is usually understood as the task of separation an available time series into segments (periods) with different dynamics.

If we present the behavior of a complex system with an adequate model, then the transition process will correspond to the transition of model parameters in the state space from one stable phase trajectory to another.

In the case of a sufficiently large number of countdown in the sample and the number of realizations of the dynamic series, the procedure of segmentation of a signal into stationary fragments should give unambiguous results. If compare the distributions and dispersions in two successive fragments of a time series, then at the point of their junction, can state a violation of the stationarity of the process in the narrow sense.

Evaluation of the statistical properties of the physiological signal is possible, however, only on a certain finite time interval. At the same time, the estimated interval can be reduced by increasing the number of samples per unit of time due to the higher frequency of digitization of the studied continuous physiological process only up to a certain limit: until the moment when neighboring counts become strongly correlated. It is this circumstance that limits the possibility of distribution the theoretical definition of stationarity to real physical processes, therefore the concept of "quasi-stationarity" was introduced into practice, highlighting the understanding of stationarity in such a restrictively evaluative aspect.

Presence of self-similarity, i.e. scale invariance in relation to the main statistical characteristics leads to continuous repetition in the self-similarity interval of the properties of some components of the studied dynamic series, meaning that these components are not informative for assessing the changes.

The most free from various restrictions is the method of self-similarity assessment, based on discrete wavelet transform of the dynamic series in the framework of multiresolution analysis. Using the sum of the components of the original time series reconstructed from the wavelet decomposition, the fractal component is selected with the determination of the self-similarity depth of the random process under study and an estimate of its trend component is found.

Using the regression model of a time series with volatility represented by a dynamic series after rejection (removal) of its trend component, it is possible to determine change-points of volatility, which divide the initial series into quasistationary segments corresponding to the volatility function jumps. For this purpose, a special computational procedure is applied, which generalizes the iterative method of centered cumulative sums of squares.

Keywords: fractals, wavelet expansion, nonstationarity, time series, volatility

References

1. **Kantz H., Schreiber T.** Nonlinear Time series Analysis, Cambridge, Cambridge Univ. Press, 2003, 388 p.
2. **Hegger R., Kantz H. and Matassini L.** Coping with Nonstationarity by Oerembedding, *Physical Review Letters*, 2000, vol. 84, p. 3197.
3. **Donkhan P., Oppenheim G., Taqqu M. S.** ed. Theory and Application of long-range dependence, Boston, Birkhauser, 2002, pp. 355—367.
4. **Ivanov P. Ch., Amaral L. A. N., Goldberger A. L., Stanley H. E.** Stochastic feedback and regulation of biological rhythms, *Europhys. Lett.*, 1998, 43(4), pp. 363—368.
5. **Bogachev M. I.** *Issledovaniya vliyatel'nykh fraktal'nykh svoystv dinamicheskikh ryadov na otsenku parametrov ikh nestatsionarnykh fragmentov* (Studies of the effects of fractal properties of dynamical series on the estimation of the parameters of their non-stationary fragments), *Izv. Vuzov Rossii. Radioelektronika*, 2006, iss. 3, pp. 3—12 (in Russian).
6. **Carlstein E., Muller H. G., Siegmund D.** ed. Change-point problems. Lecture notes and monograph series, Hayward, CA, Institute of Mathematical Statistics, 1994, VD 23, 385 p.
7. **Peitgen H. O., Jurgens H., Saupe D.** Chaos and fractals, New frontiers of science, New York, Springer-Verlag, 1992, 984 p.
8. **Malla S.** *Veyvlety v obrabotke signalov* (Wavelets in signal processing), Moscow, Mir, 2005, 671 p. (in Russian).
9. **Dremin I. M., Ivanov O. V., Nechitaylo V. A.** *Veyvlety i ikh naimenovaniya* (Wavelets and their names), *UFN*, 2001, vol. 171, no. 5, pp. 465—501 (in Russian).
10. **Marek M.** Heart Rate Variability: Standards of measurement, Physiological interpretation and clinical use, *Circulation*, 1996, vol. 93(5), pp. 1043—1065.
11. **Chen G., Choi Y., Zhou Y.** Nonparametric estimation of structural change points in volatility models for time series, *Journal of Econometrics*, 2005, vol. 12, pp. 79—114.
12. **Inclan C., Tiao G.** Use of Cumulative Sums of square for retrospective detection of changes of variance, *Journal of American Statistical Association*, 1994, vol. 89, no. 427, pp. 913—923.
13. **Mandelbrot B. B., Van Ness J. W.** Fractional Brownian motions, fractional noises and applications, *SIAM Rev.*, 1968, vol. 10, pp. 422—437.
14. **Abry P., Veitch D.** Wavelet analysis of long range dependence traffic, *IEEE Trans. Of Inf. Th.*, 1998, vol. 44, no. 1, pp. 2—15.
15. **Smolentsev N. K.** *Osnovy teorii veyvletov* (Fundamentals of the theory of wavelets), *Veyvlet in MATLAB*, Moscow, DMKA press, 2008, 448 p. (in Russian).
16. **Csorgo M., Horvath L.** Limit Theorems in Change-point analysis, New York, Wiley, 1997, 438 p.
17. **Poligrad M., Shao Q. M.** Estimation of the Variance of partil Sums for ρ Mixing random variables, *Journal of Multivariate Analysis*, 1995, vol. 52, pp. 140—157.
18. **Poligrad M., Shao Q. M.** Self-normalizing central limit theorem for suns of weakly dependent random variables, *J. Theoret. Probab.*, 1994, vol. 7, pp. 309—338.
19. **Shiryayev A. N.** *Veroyatnost'* (Probability), Moscow, Science, 1989, 640 p. (in Russian).
20. **Fan J., Yao Q.** Effiant estimation of conditional variance functions in stoachastic regression, *Biometrika*, 1998, vol. 85, pp. 645—660.
21. **Xia Y., Tong H., Li W. K.** Abstitute deviation estimation of volatility in nonparametric models, Research report No. 177. Department of Statistics and actuarial science, The University of Hong Kong, Hong Kong, 1988, pp. 133—173.
22. **Choin J. M., Muller H. G.** Nonparametrics quasilikelihood, *The annual of statistics*, 1999, vol. 27, pp. 36—64.



XV НАУЧНО-ПРАКТИЧЕСКАЯ КОНФЕРЕНЦИЯ
SECR 2019 «РАЗРАБОТКА ПО»

ОТКРЫТ ПРИЕМ ДОКЛАДОВ

Главная тема - разработка
программного обеспечения

14-15 ноября, Санкт-Петербург

От технологий программирования до
образования и ведения бизнеса в ИТ

Принимаются заявки на доклады,
мастер-классы, научные статьи с
презентацией.

Срок подачи: 20 августа 2019

Преимущества для спикеров:

- Бесплатное участие
- Премия за лучшую исследовательскую работу
- Планируется публикация научных статей в электронной библиотеке ACM

www.secrus.org contact@secrus.org

Л. В. Савченко, канд. техн. наук, e-mail: lsavchenko@hse.ru,
Национальный исследовательский университет Высшая школа экономики — Нижний Новгород

Система постановки произношения на основе сверточных нейронных сетей и информационной теории восприятия речи

Рассматривается задача постановки произношения на основе применения методов глубокого обучения совместно с информационной теорией восприятия речи. Для повышения эффективности тестирования качества произношения предложено проводить дообучение сверточной нейронной сети с использованием наилучших эталонов пользователя. Экспериментально показано, что предложенный подход характеризуется высокой точностью и скоростью распознавания для различных акустических моделей по сравнению с известными аналогами.

Ключевые слова: постановка произношения, распознавание речи, сверточные нейронные сети, глубокое обучение, информационное рассогласование Кульбака — Лейблера

Введение

Задача постановки произношения в последнее время вызывает интерес из-за обширной сферы практического применения [1, 2], например, в информационных системах обучения людей с нарушением слуха [3, 4] и особенностями речи (акцент, заикание), обучения иностранным языкам [5, 6] и в других областях, где существует необходимость постановки произношения, близкого к произношению некоторого эталонного диктора [7, 8]. Кроме того, качественное произношение слов является неотъемлемой частью задачи распознавания речи и способно существенно повысить точность распознавания [9].

Как известно, для различных носителей национального языка возникает проблема вариативности устной речи и тесно связанная с ней проблема самостоятельной оценки обучающимся качества своего произношения [10]. В самой постановке задачи содержится очевидное противоречие: обучаемый с недостаточной на данный момент языковой подготовкой и ограниченными возможностями в процессе самообучения должен приблизиться по своему произношению к некоторому эталону, который он слабо себе представляет. Указанное противоречие с успехом преодолевается с использованием предложенного в работе [11] подхода на основе информационной теории восприятия

речи (ИТВР). В этом подходе достижимость эталонного произношения обеспечивается использованием лучших образцов произнесений от одного или даже группы дикторов. Система обучения на основе ИТВР способна запоминать лучшее произношение диктором слова и оценивать качество последующего произнесения того же слова по отношению к этим наилучшим для диктора словам, а не только по отношению к используемым по умолчанию эталонам, введенным идеальным диктором. Для оценки качества произношения используется тестирование различимости слов, которое может быть осуществлено с помощью одного из известных методов автоматического распознавания речи [12]. В указанных выше работах использовалась только реализация принципа минимума информационного рассогласования Кульбака—Лейблера [13] для сопоставления оценок спектральной плотности мощности речевых сигналов. В то же время сейчас, как известно [14], наибольшей точностью распознавания характеризуются подходы на основе технологий глубокого обучения и, в частности, сверточных нейронных сетей (СНС) [15]. Таким образом, представляет несомненный интерес комбинирование ИТВР и современных методов глубокого обучения для повышения эффективности систем постановки произношения. Исследованиям в этом актуальном направлении и посвящена настоящая работа.

Постановка произношения на основе информационной теории восприятия речи

Пусть задана база данных в виде множества из $R > 1$ эталонных слов $X_r = \{x_{r,j}\}$, где $r = \overline{1, R}$ — номер слова, $\{x_{r,j}\}$, $j = \overline{1, J_r}$ — вектор отсчетов речевого сигнала, представляющего j -ю реализацию r -го слова; J_r — число эталонных реализаций r -го слова. В традиционных системах обучения речи каждое слово, чаще всего, представляется одним словом ($J_r = 1$), произнесенным эталонным диктором [16]. Далее в процессе постановки произношения пользователь обычно последовательно обучается произносить каждое r -е слово до тех пор, пока вектор отсчетов \mathbf{x} не будет достаточно близок к одному из эталонов:

$$\min_{j=\overline{1, J_r}} \rho(\mathbf{x}, \mathbf{x}_{r,j}) < \rho_0, \quad (1)$$

где $\rho(\mathbf{x}, \mathbf{x}_{r,j})$ — некоторая мера близости между сигналами \mathbf{x} и $\mathbf{x}_{r,j}$, а ρ_0 — подобранное экспериментально максимальное допустимое отклонение одноименных реализаций слов.

В подходе на основе ИТВР [11] достижимость эталонного произношения обеспечивается использованием не одного, а нескольких "эталонов", включающих в себя и лучшие образцы произнесений от одного или даже группы учащихся лиц, успешно прошедших обучение ранее. Такая система способна запоминать лучшее произношение диктором слова и оценивать качество последующего произнесения того же слова по отношению к этим наилучшим для диктора словам, а не только по отношению к используемым по умолчанию эталонам, введенным идеальным диктором.

Согласно ИТВР [11] одноименные реализации в сознании человека группируются в соответствующие классы вокруг некоторого центра — эталонной метки. Для определения понятия "центр кластера" в информационной теории восприятия речи используется кластерная модель слов [17]: эталон $\mathbf{x}_r^* \in X_r$ образует информационный центр r -го класса, если в пределах множества X_r он характеризуется минимальным разбросом — средней суммой информационных рассогласований Кульбака—Лейблера $\rho(\mathbf{x}_{r,k}, \mathbf{x}_{r,j})$ [18] относительно всех других его меток-реализаций $\mathbf{x}_{r,j}$, $j = \overline{1, J_r}$, в пределах каждого r -го класса:

$$\mathbf{x}_r^* = \arg \min_{\mathbf{x}_{r,k}, k \in \{1, \dots, J_r\}} \delta_{r,k}, \quad (2)$$

где $\delta_{r,k}$ — средний разброс эталонных реализаций r -го слова относительно k -го эталона:

$$\delta_{r,k} = \frac{1}{J_r} \sum_{j=1}^{J_r} \rho(\mathbf{x}_{r,k}, \mathbf{x}_{r,j}). \quad (3)$$

После того как пользователь обучился произносить каждое слово, проводят оценку качества произношения всех слов с помощью их распознавания на основе принципа минимума информационного рассогласования [12]. Тем самым, каждое произнесенное слово будет характеризоваться не только близостью к эталону, но и его различимостью относительно других эталонов. В случае неудовлетворительного результата (плохая различимость определенных слов, их удаленность друг от друга) обучение произношению следует повторить.

Заметим, что эффективность обучения во многом определяется качеством алгоритмов распознавания. В следующем разделе рассмотрим современные подходы к распознаванию изолированных слов.

Нейросетевые методы распознавания изолированных слов

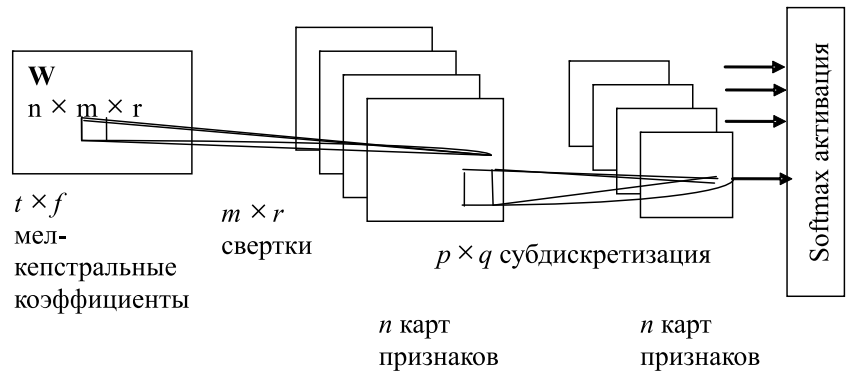
Задача распознавания изолированных слов состоит в том, чтобы вновь поступающему на вход речевому сигналу поставить в соответствие одно из R эталонных слов. Одним из наиболее популярных способов решения задачи в настоящее время являются глубокие нейронные сети, которые предполагают моделирование акустических сигналов с помощью многоуровневых последовательно соединенных слоев нелинейных функций. Например, такие известные приложения голосового поиска, как Google и Apple Siri, реализованы с помощью глубоких сетей. Отметим, что при распознавании речи с помощью глубокой нейронной сети проводится множество матричных вычислений, что приводит к существенным затратам вычислительных ресурсов (хранение акустической модели) на этапе распознавания. При этом глубокие сети оказываются чувствительными к репрезентативности обучающего множества (речевого корпуса), поэтому их применение может привести к неудовлетворительным результатам для распознавания нестандартной речи (при наличии помех, дефектов произношения и т. п.) [19].

При работе с аудиосигналами в настоящее время, например в системах DeepSpeech [20],

вместо обычных многослойных нейронных сетей прямого распространения используются рекуррентные нейронные сети [21]. Такие сети имеют циклические связи и позволяют хранить информацию о предыдущем состоянии. Рекуррентные сети можно представить себе как множество копий одной и той же сети, причем каждая копия передает сообщение следующей копии. В задаче распознавания речи такие рекуррентные архитектуры позволяют автоматически настроить систему для эффективного различения набора распознаваемых слов [22].

К сожалению, вычислительная сложность распознавания речи с помощью глубоких рекуррентных сетей является чрезвычайно высокой, поэтому на практике для их реализаций обычно используются удаленные мощные вычислительные серверы. В связи с этим в последнее время приобретают популярность глубокие СНС, которые не только характеризуются намного более высокой скоростью принятия решений, но и показали более точные результаты распознавания на больших и маленьких словарях в ряде работ [22, 23]. Используемая в них свертка издавна применялась в цифровой обработке речевых сигналов в разнообразных методах линейной фильтрации. Особенность СНС заключается в том, что в ней нейроны первых уровней упорядочены в особую структуру — на первых слоях нейроны разбиты на карты признаков определенного размера, и разные карты внутри одного слоя соответствуют нейронам разного типа, которые реагируют на разные особенности спектра речевого сигнала.

Архитектура типичной СНС показана на рисунке. На вход сети подается речевой сигнал $\mathbf{V} \in \mathcal{R}^{t \times f}$, где t и f — число фреймов и число характерных признаков. Например, в качестве входного сигнала могут использоваться мел-кепстральные коэффициенты размером $t \times f = 32 \times 40$. Затем трехмерный тензор весов $\mathbf{W} \in \mathcal{R}^{(m \times r) \times n}$ сворачивается с входным сигналом \mathbf{V} . Выход сверточного слоя затем передается на следующий сверточный слой. В библиотеке TensorFlow используется сверточная нейронная сеть типа `sn-n-trad-fpool3`, которая, как показано в статье [24], позволяет повысить точность распознавания изолированных слов по сравнению с глубокими нейронными сетями на 27 %. На выходе СНС получают оценки апостериорной вероятности принадлежно-



Архитектура сверточной нейронной сети

сти к каждому r -му эталонному классу слов с помощью softmax активации [25]:

$$P(r|\mathbf{x}) = \frac{\exp(z_r)}{\sum_{j=1}^R \exp(z_j)}, \quad r = 1, 2, \dots, R. \quad (4)$$

Здесь z_r — выход r -го нейрона предпоследнего слоя СНС, на вход которой подан сигнал \mathbf{x} .

Предлагаемый в настоящей работе алгоритм постановки произношения на основе ИТВР и СНС представлен ниже.

Алгоритм постановки произношения на основе информационной теории восприятия речи и сверточных нейронных сетей

Входные данные: база данных эталонных слов \mathbf{x}_r^* .

Выходные данные: оценка точности и времени распознавания слов.

Параметры: порог на слова ρ_0 , необходимое число добавляемых эталонов m .

1. Для каждого r -го класса, $r = \overline{1, R}$:
 - 1.1. Записать речевой сигнал \mathbf{x} .
 - 1.2. Пока пользователь не обучится стабильному произношению, повторить (для каждого r -го класса)
 - 1.2.1. Для каждого произнесенного слова \mathbf{x} с помощью предобученной СНС оценить апостериорную вероятность эталона r -го класса согласно (5).
 - 1.3. Пока число добавленных слов меньше m ,
 - 1.3.1. Если максимальная апостериорная вероятность соответствует r -му классу, то
 - 1.3.1.1. Добавить слово \mathbf{x} в базу данных эталонных слов.
2. Провести дообучение сверточной нейронной сети на основе добавленных слов.
3. Для каждого r -го класса слов повторить
 - 3.1. Определить информационный центр-эталон согласно (2).
 - 3.2. Записать речевой сигнал \mathbf{x} .
 - 3.3. Оценить время, правильность распознавания слова \mathbf{x} и средний разброс всех эталонов внутри класса согласно (3).
4. Вернуть среднюю вероятность правильного распознавания всех слов и время распознавания слова.

Здесь на первом этапе пользователь обучается стабильному произношению слов, т. е. в базу данных эталонов добавляются только те слова, для которых максимальная оценка апостериорной вероятности на выходе СНС (4) превышает некоторый наперед заданный порог p_0 :

$$\max_{r=1,R} P(r|\mathbf{x}) > p_0. \quad (5)$$

Далее проводится дообучение СНС на основе наилучших пользовательских эталонов. После этого оценивается качество постановки произношения, т. е. вновь произнесенные слова распознаются с помощью дообученной СНС. Кроме того, для проверки стабильности произнесения оценивается степень близости произнесенных пользователем слов внутри r -го класса. Для этого в каждом r -м классе ищется центр-эталон (2) и вычисляется средний разброс ("радиус" в терминологии ИТВР) каждого класса — минимальная средняя удаленность от центра (3). Небольшой радиус соответствует качественному и стабильному произнесению слова. Если разброс оказывается больше некоторого наперед заданного порога, то обучение для r -го слова следует пройти еще раз.

Результаты экспериментальных исследований

В экспериментальной части статьи рассматривается задача оценки качества произношения десяти слов (команд) английского языка ("down", "go", "left", "no", "off", "on", "right", "stop", "up", "yes"). Дообучение СНС проводили с помощью набора скриптов Simple Audio Recognition из библиотеки TensorFlow. Использовали несколько встроенных акустических моделей. Модель conv базируется на топологии cnn-trad-fpool3 [24], модель low_latency_conv ис-

пользует нейронную сеть cnn-one-fstride4 [24], модель low_latency_svdf использует топологию rank-constrained [26] (сжатие нейронных сетей), модель tiny_conv состоит из односверточного нейронного слоя (была разработана для работы на устройствах с небольшим объемом оперативной памяти). Экспериментально были оценены точность и время распознавания для всех доступных видов акустических моделей.

Для записи сигнала применялся встроенный в ноутбук микрофон. Частота дискретизации F установлена равной 16 кГц. Тестовое множество содержало 100 элементов (по 10 реализаций каждой команды). В табл. 1 представлены результаты сравнительного анализа точности и времени распознавания слов английского языка для всех доступных акустических моделей с обучением и без обучения (наилучшие результаты выделены полужирным шрифтом). Здесь в первой части эксперимента распознавание проводилось сначала для базовых акустических моделей на словах эталонного диктора (столбцы "Без дообучения" табл. 1). Во второй части эксперимента проводилось дообучение сверточной нейронной сети на 100 реализациях эталонных команд (по 10 реализаций каждой команды). В качестве эталонов были добавлены все 100 произносимых слов пользователя (столбцы "Все эталоны" в табл. 1). В заключение эксперимента дообучение сети проводилось на основе только лучших реализаций пользователя, выбранных с помощью критерия (5). Результаты приведены в столбцах "Лучшие эталоны" в табл. 1.

Из табл. 1 видно, что акустическая модель conv превосходит по точности распознавания другие доступные акустические модели, если не проводить обучение сети (72%), в то же время наименьшее время распознавания команды имеет модель tiny_conv (4...5 мс). Дообучение

Таблица 1

Сравнительный анализ точности и времени распознавания для различных акустических моделей

Акустические модели	Точность и время распознавания					
	Без дообучения		С обучением			
			Все эталоны		Лучшие эталоны	
	Точность, %	Время, мс	Точность, %	Время, мс	Точность, %	Время, мс
Conv	72	7,5	91	6,5	94	6,5
low_latency_conv	27	7	92	6	96	6
low_latency_svdf	46	11	55	10.5	60	10.5
tiny_conv	20	5	50	4	65	4

Таблица 2

Разброс (3) эталонов для каждого слова

Слово	Радиус до обучения	Радиус после обучения
Down	1,9	0,17
Go	0,85	0,23
Left	0,55	0,3
No	1,14	0,25
Off	0,17	0,17
On	1,56	0,17
Right	0,54	0,3
Stop	0,28	0,28
Up	0,17	0,17

сети позволяет существенно повысить точность распознавания команд (на 19...65 %). В то же время обучение на лучших образцах пользователя позволяет повысить точность распознавания еще на 3...15 %. Таким образом, акустические модели conv и low_latency_conv позволяют достичь наивысшей точности распознавания, а акустическая модель tiny_conv имеет наименьшее время обработки речевого сигнала.

В качестве примера практического применения подхода к тестированию стабильности произнесения слов на заключительных шагах работы (см. *Алгоритм*) рассмотрим результаты оценки разбросов (3) внутри каждого класса (табл. 2). Первоначальная постановка произношения (второй столбец табл. 2) прошла недостаточно качественно (оценка радиуса (3) для слов "down", "no", "on" оказалась намного выше среднего значения). Поэтому, согласно предложенному подходу, пользователь повторно прошел обучение произнесению этих слов (табл. 2). Результаты (оценки среднего разброса (3)) представлены в третьем столбце табл. 2. Здесь качество постановки произношения всех слов можно считать приемлемым, так как разброс расстояний оказался значительно ниже по сравнению с первоначальными оценками радиусов (3). Таким образом, предлагаемый подход на основе ИТВР и глубоких СНС позволяет повысить точность распознавания слов за счет дообучения нейронной сети и постановки качественного произношения.

Заключение

В настоящей работе предложен новый подход для оценки качества произношения слов,

основанный на комбинировании информационной теории восприятия речи и глубоких СНС, позволяющий увеличить точность распознавания слов за счет дообучения нейронной сети на наиболее качественно произнесенных словах пользователя. Результаты экспериментальных исследований показывают, что добавление в базу данных эталонных команд пользователя способно увеличить точность распознавания на 19...69 % (см. табл. 1). В дальнейших исследованиях интерес представляет применение предложенного подхода при распознавании команд при наличии у обучающегося пользователя явно выраженных дефектов речи (таких, например, как заикание, наличие акцента, отсутствие в речи некоторых звуков).

Список литературы

1. Савченко В. В., Акатьев Д. Ю. Обучение звуковому строю языка глухонемых и слабослышащих на основе информационной теории восприятия речи // Информационные технологии. 2010. № 2. С. 60—66.
2. Krasnova E., Bulgakova E. The use of speech technology in computer assisted language learning systems // Lecture Notes in Computer Science. 2014. Т. 8773. Р. 459—466.
3. Денисова И. А. Игровые технологии как условие повышения качества произношения учащихся с нарушениями слуха младшего школьного возраста // Череповецкие научные чтения. 2015. С. 52—54.
4. Schuller B., Steidl S., Batliner A., Burkhardt F., Devillers L., Muller C., Narayanan S. Paralinguistics in speech and language-State-of-the-art and the challenge // Computer Speech and Language. 2013. Vol. 27, N. 1. Р. 4—39.
5. Ахмедова М. М., Рахимова М. И., Отамуродова Ф. Э. Методика обучения произношению иностранного языка // Наука и Мир. 2016. Т. 3, № 6 (34). С. 52—53.
6. Golonka E. Technologies for foreign language learning: a review of technology types and their effectiveness // Computer Assisted Language Learning. 2014. Т. 27, N. 1. Р. 70—105.
7. Miller C. Computational Approaches to Exploring Persian-Accented English // Research in Language. 2015. Т. 13, N. 1. Р. 51—60.
8. Hu W., Qian Y., Soong F. A new DNN-based high quality pronunciation evaluation for computer-aided language learning // INTERSPEECH. 2013. Р. 1886—1890.
9. Савченко Л. В. Оценка качества произношения на основе метода нечеткого фонетического кодирования // Телекоммуникации. 2017. № 5. С. 42—48.
10. Попова М. И. Обучение студентов языкового вуза иноязычной письменной речи на основе информационных технологий // Наука и образование. 2011. № 4. С. 85—89.
11. Савченко В. В. Информационная теория обучения речи // Изв. вузов России. Радиоэлектроника. 2009. № 3. С. 3—12.
12. Benesty J., Sondh M., Huang Y. Springer handbook of speech recognition. New York: Springer, 2008, 1176 p.
13. Kullback S. Information theory and statistics. New York: Dover Pub, 1997. 408 p.
14. LeCun Y., Bengio Y., Hinton G. Deep learning // Nature. 2015. Т. 521, N. 7553. 436 p.
15. Zhang Y., Chan W., Jaitly N. Very deep convolutional networks for end-to-end speech recognition // Acoustics, Speech

and Signal Processing (ICASSP), 2017. IEEE International Conference on. IEEE, 2017. P. 4845–4849.

16. **Кнеллер Э. Г., Караульных Д. В.** Устройство для обучения разговорной (устной) речи с визуальной обратной связью: пат. на полезную модель № WO2016053141 A1, Роспатент: по заявке РСТ/RU2015/000583 от 17.09.2015

17. **Савченко Л. В.** Автоматическое распознавание изолированных слов на основе теории нечетких множеств и кластерной модели минимальных речевых единиц // Информационные технологии. 2014. № 2. С. 9–13.

18. **Савченко В. В.** Исследование стационарности случайных временных рядов с использованием принципа минимума информационного рассогласования // Известия высших учебных заведений. Радиофизика. 2017. Т. 60, № 1. С. 89–96.

19. **Васильев Е. М., Меренков В. В.** Система распознавания фонетических образов на основе нейросетевой модели восприятия речи // Вестник Воронежского государственного технического университета. 2009. № 10. С. 130–134.

20. **Amodei D., Ananthanarayanan S., Anubhai R., Bai J., Battenberg E., Case C., Chen J.** Deep speech 2: End-to-end speech recognition in english and mandarin // International Conference on Machine Learning. 2016. С. 173–182.

21. **Graves A., Mohamed A., Hinton G.** Speech recognition with deep recurrent neural networks // Acoustics, speech and signal processing (ICASSP). 2013. IEEE, 2013. P. 6645–6649.

22. **Toth L.** Combining Time- and Frequency-Domain Convolution in Convolutional Neural Network-Based Phone Recognition // Proceedings of the Acoustics, speech and signal processing (ICASSP). 2014. P. 190–194.

23. **Sainath T. N., Mohamed A., Kingsbury B., Ramabhadran B.** Deep Convolutional Neural Networks for LVCSR // Proceedings of the Acoustics, speech and signal processing (ICASSP). 2013. P. 8614–8618.

24. **Sainath T. N., Parada C.** Convolutional neural networks for small-footprint keyword spotting // Sixteenth Annual Conference of the International Speech Communication Association (ICASSP). 2015. P. 1478–1482.

25. **Zhang Y., Pezeshki M., Brakel P., Zhang S., Bengio C., Courville A.** Towards end-to-end speech recognition with deep convolutional neural networks // arXiv preprint arXiv:1701.02720, 2017.

26. **Nakkiran P., Alvarez R., Prabhavalkar R., Parada C.** Compressing deep neural networks using a rank-constrained topology // Sixteenth Annual Conference of the International Speech Communication Association (ICASSP). 2015. P. 1473–1477.

L. V. Savchenko, Ph. D., e-mail: lsavchenko@hse.ru,
National Research University Higher School of Economics — N. Novgorod

Computer-Assisted Language Learning Based on Convolutional Neural Networks and Information Theory of Speech Perception

In this paper we consider a problem of computer assisted language and pronunciation learning based on the deep neural networks and the information theory of speech perception. At first, a user learns the stable pronunciation of words. The best utterances from the user with high posterior probability estimated by the pre-trained convolutional neural network are added to the training set. Next, this training set is used to fine-tune this convolutional neural network. If new utterances are successfully recognized with the resulted neural network, it is concluded that pronunciation of all words is distinguishable. In this case in order to additionally verify the stability of pronunciation of each class (word), the closeness of the user pronunciations is estimated by computing the average Kullback-Leibler information discrimination between each signal and the centroid reference of the class. If this mean discrimination for particular word is greater than a certain threshold, then the training for this word should be repeated. The experimental results for learning of English words proved that the proposed approach is characterized by higher accuracy and speed for existing acoustic models when compared to conventional techniques.

Keywords: computer-assisted learning system, speech recognition, convolutional neural network, deep learning, Kullback-Leibler information discrimination

DOI: 10.17587/it.25.313-318

References

1. **Savchenko V. V., Akatjev D. Yu.** Learning the sound structure of the language of deaf and hard of hearing on the basis of information theory of speech perception, *Informacionnye Tekhnologii*, 2010, no. 2, pp. 60–66 (in Russian).

2. **Krasnova E., Bulgakova E.** The use of speech technology in computer assisted language learning systems, *Lecture Notes in Computer Science*, 2014, vol. 8773, pp. 459–466.

3. **Denisova I. A.** Game technologies as a condition of improving the quality of pronunciation of students with hearing impairments of primary school age, *Cherepovets Scientific Readings*, 2015, pp. 52–54 (in Russian).

4. **Schuller B., Steidl S., Batliner A., Burkhardt F., Devillers L., Muller C., Narayanan S.** Paralinguistics in speech and language—State-of-the-art and the challenge, *Computer Speech and Language*, 2013, vol. 27, no. 1, pp. 4–39.

5. **Ahmedova M. M., Rahimova M. I., Otamurodova F. Je.** Methods of teaching foreign language pronunciation, *Science and World*, 2016, vol. 3, no. 6 (34), pp. 52–53 (In Russian).

6. **Golonka E.** Technologies for foreign language learning: a review of technology types and their effectiveness, *Computer Assisted Language Learning*, 2014, vol. 27, no. 1, pp. 70–105.

7. **Miller C.** Computational Approaches to Exploring Persian-Accented English, *Research in Language*, 2015, vol. 13, no. 1, pp. 51–60.

8. **Hu W., Qian Y., Soong F.** A new DNN-based high quality pronunciation evaluation for computer-aided language learning, *INTERSPEECH*, 2013, pp. 1886–1890.
9. **Savchenko V. V.** Pronunciation quality assessment based on the fuzzy phonetic coding method, *Telecommunications and Radio Engineering*, 2017, no. 5, pp. 42–48 (in Russian).
10. **Popova M. I.** Learning students of the language University of foreign language writing on the basis of information technology, *Science and Education*, 2011, no. 4, pp. 85–89 (in Russian).
11. **Savchenko V. V.** Information theory of speech learning, *Journal of the Russian Universities. Radioelectronics*, 2009, no. 3, pp. 3–12 (in Russian).
12. **Benesty J., Sondh M., Huang Y.** Springer handbook of speech recognition, New York, Springer, 2008, 1176 p.
13. **Kullback S.** Information theory and statistics, New York, Dover Pub, 1997, 408 p.
14. **LeCun Y., Bengio Y., Hinton G.** Deep learning, *Nature*, 2015, vol. 521, no. 7553, 436 p.
15. **Zhang Y., Chan W., Jaitly N.** Very deep convolutional networks for end-to-end speech recognition, *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, IEEE, 2017, pp. 4845–4849.
16. **Kneller E. G., Karaulnykh D. V.** A device for learning conversational (oral) speech with visual feedback: Patent RF № WO2016053141 A1, Rospatent: by request PCT/RU2015/000583 or 17.09.2015 (in Russian).
17. **Savchenko L. V.** Automatic recognition of isolated words on the basis of fuzzy set theory and cluster model of minimal speech units, *Informacionnye Tekhnologii*, 2014, no.2, pp. 9–13 (in Russian).
18. **Savchenko V. V.** Study of the stationarity of random time series using the principle of the information-divergence minimum, *Radiophysics and Quantum Electronics*, 2017, vol. 60, no. 1, pp. 81–87.
19. **Vasilyev E. M., Merenkov V. V.** Vowels recognition system on the basis of neuronet simulation of speech perception, *Herald of Voronezh state technical University*, 2009, no. 10, pp. 130–134 (in Russian).
20. **Amodei D., Ananthanarayanan S., Anubhai R., Bai J., Battenberg E., Case C., Chen J.** Deep speech 2: End-to-end speech recognition in English and Mandarin, International Conference on Machine Learning, 2016, pp. 173–182.
21. **Graves A., Mohamed A., Hinton G.** Speech recognition with deep recurrent neural networks, *Acoustics, speech and signal processing (ICASSP)*, 2013, IEEE.— 2013, pp. 6645–6649.
22. **Toth L.** Combining Time-and Frequency-Domain Convolution in Convolutional Neural Network-Based Phone Recognition, *Proceedings of the Acoustics, speech and signal processing (ICASSP)*, 2014, pp. 190–194.
23. **Sainath T. N., Mohamed A., Kingsbury B., Ramabhadran B.** Deep Convolutional Neural Networks for LVCSR, *Proceedings of the Acoustics, speech and signal processing (ICASSP)*, 2013, pp. 8614–8618.
24. **Sainath T. N., Parada C.** Convolutional neural networks for small-footprint keyword spotting, *Sixteenth Annual Conference of the International Speech Communication Association (ICASSP)*, 2015, pp. 1478–1482.
25. **Zhang Y., Pezeshki M., Brakel P., Zhang S., Bengio C., Courville A.** Towards end-to-end speech recognition with deep convolutional neural networks, arXiv preprint arXiv:1701.02720, 2017.
26. **Nakkiran P., Alvarez R., Prabhavalkar R., Parada C.** Compressing deep neural networks using a rank-constrained topology, *Sixteenth Annual Conference of the International Speech Communication Association (ICASSP)*, 2015, pp. 1473–1477.

Адрес редакции:

107076, Москва, Стромьинский пер., 4

Телефон редакции журнала (499) 269-5510

E-mail: it@novtex.ru

Технический редактор *Е. В. Конова.*

Корректор *Е. В. Комиссарова.*

Сдано в набор 12.03.2019. Подписано в печать 24.04.2019. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT519. Цена договорная.

Журнал зарегистрирован в Министерстве Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-15565 от 02 июня 2003 г.

Оригинал-макет ООО "Авансед солюшнз". Отпечатано в ООО "Авансед солюшнз".
119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.

“ЭКСТРЕМАЛЬНАЯ РОБОТОТЕХНИКА”

30^{-ая} Международная НАУЧНО-ТЕХНИЧЕСКАЯ КОНФЕРЕНЦИЯ

Конференция "Экстремальная робототехника - 2019" - МОСТ МЕЖДУ ЕВРОПОЙ И АЗИЕЙ

СИМПОЗИУМЫ



Космическая робототехника

- Орбитальные сервисные роботы и кластерная инфраструктура
- Планетоходы и робототехнические базы
- Роботизированные миссии на орбитальных станциях



Воздушная робототехника

- Групповое управление
- Привязные дроны и их группировки
- Оптическая связь



Наземная робототехника

- Мобильные платформы и манипуляторы
- Навигация и картографирование
- Связь и сети



Морская робототехника

- Подводные навигация и связь
- Обслуживание шельфового оборудования
- Роботы для геологоразведки

СЕМИНАРЫ



Робототехника для атомной отрасли

- Автоматизированный радиационный мониторинг
- Роботизированный вывод реактора из эксплуатации
- Оборудование для аварийных служб



Медицинская робототехника

- Ассистирование при хирургических операциях
- Роботизированные процедуры
- Экзоскелеты для восстановительной терапии



Образовательная робототехника

- Молодежные соревнования
- Формирование креативного мышления
- Робототехника в образовании



Коллаборативная робототехника

- Индустриальные коботы
- Взаимодействие человека и робота
- Мягкая робототехника и безопасность

Этапы и даты

Регистрация и предоставление материалов тезисов доклада; материалов доклада; экспертного заключения о возможности публикации: до 30 апреля 2019

Рассмотрение материалов Программным комитетом; уведомление участников о принятии докладов: до 15 мая 2019

Оформление участия

Для участия в конференции необходимо зарегистрироваться на сайте мероприятия: er.rtc.ru и направить материалы тезисов, докладов и экспертное заключение в Секретариат конференции на электронный адрес: mrspsb@rtc.ru

Стоимость участия и условия оплаты на сайте мероприятия.

Рабочий язык конференции: АНГЛИЙСКИЙ и РУССКИЙ

Секретариат конференции: тел.: +7 (812) 552-45-21; e-mail: mrspsb@rtc.ru

Рисунки к статье Р. М. Юсупова, В. И. Воробьева, М. Ю. Петрова
**«ПРЕДСТАВЛЕНИЕ ЭВОЛЮЦИИ СРЕДСТВ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ
В ЭКСПОЗИЦИИ МУЗЕЯ ИСТОРИИ СПИИРАН»**

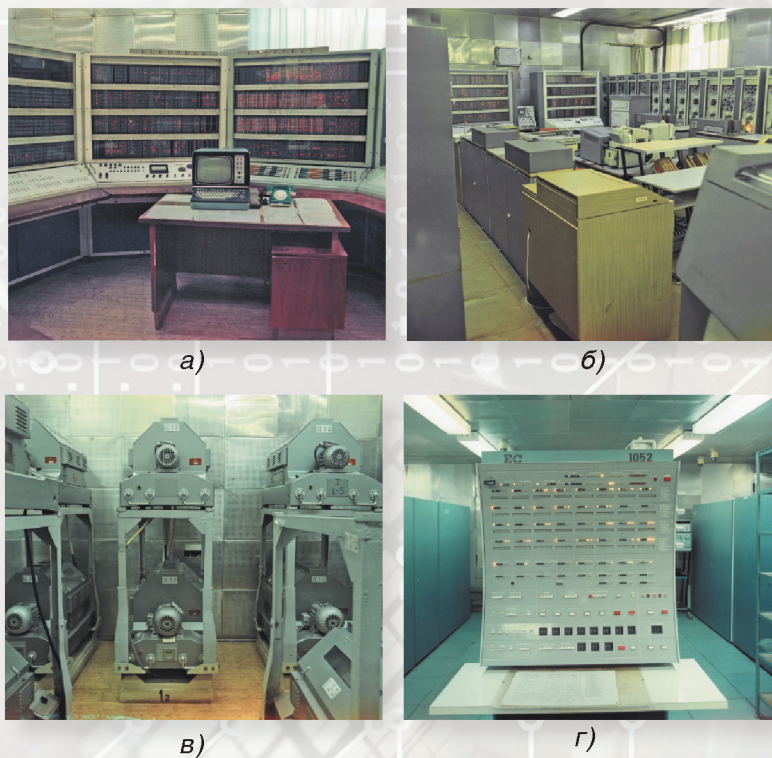
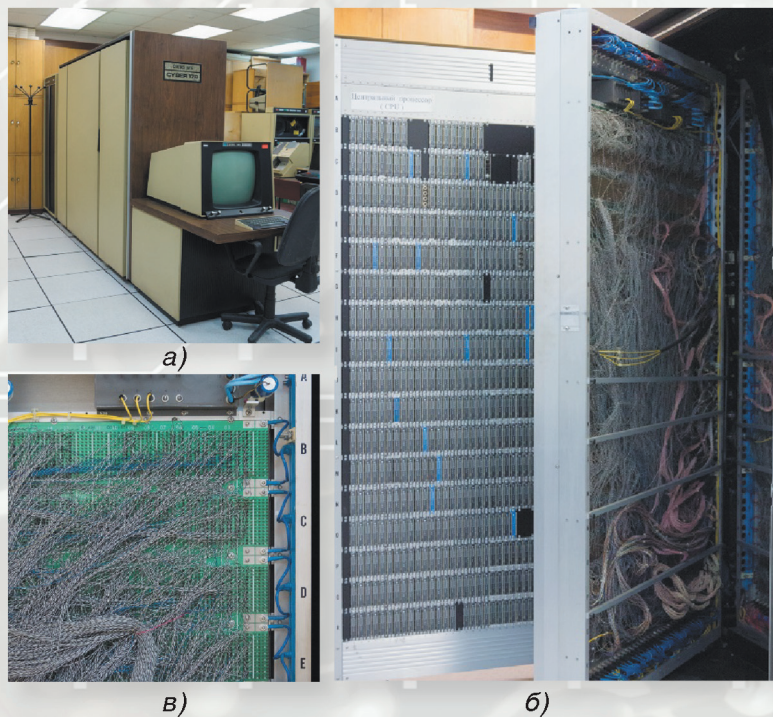


Рис. 1. ЭВМ БЭСМ-6 (а – в) и ЕС-1052 (г)



**Рис. 2. Вычислительный комплекс CYBER-172-6
и элементы центрального процессора**

Рисунки к статье Р. М. Юсупова, В. И. Воробьева, М. Ю. Петрова
«ПРЕДСТАВЛЕНИЕ ЭВОЛЮЦИИ СРЕДСТВ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ
В ЭКСПОЗИЦИИ МУЗЕЯ ИСТОРИИ СПИРАН»

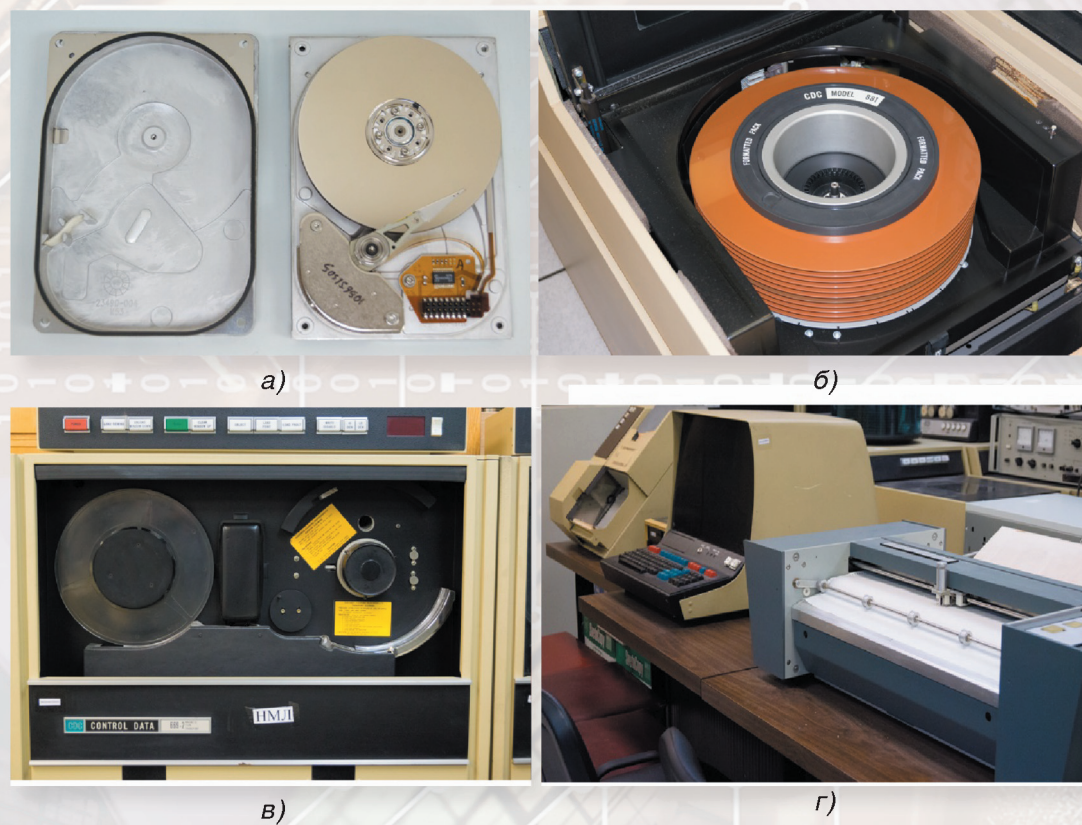


Рис. 3. Периферийные устройства

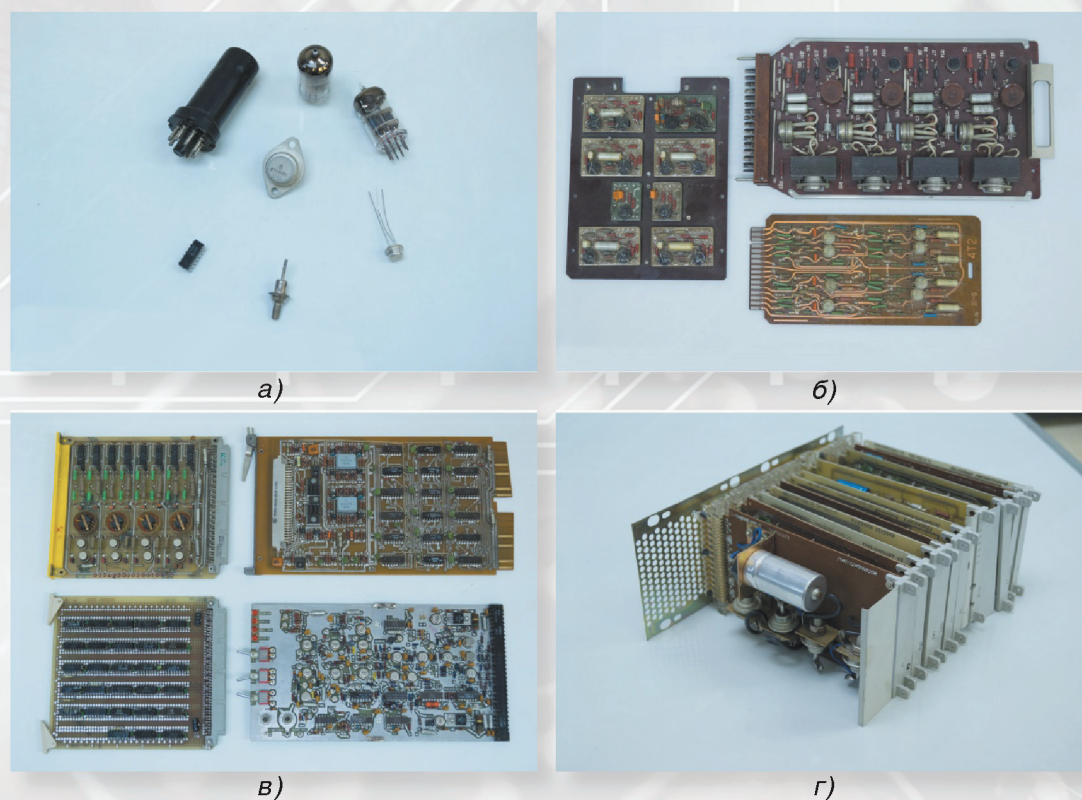


Рис. 5. Элементная база и узлы ЭВМ 1950-80-х годов