

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 25

2019

№ 1

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 В. В. Карганова, А. А. Шевченко, Б. Ю. Малышева

«СПОСОБ ПОВЫШЕНИЯ РАБОТОСПОСОБНОСТИ ИНФОРМАЦИОННО-ВЫЧИСЛИТЕЛЬНОЙ СЕТИ ЗА СЧЕТ АДАПТИВНОГО УПРАВЛЕНИЯ ЗАЩИТОЙ»

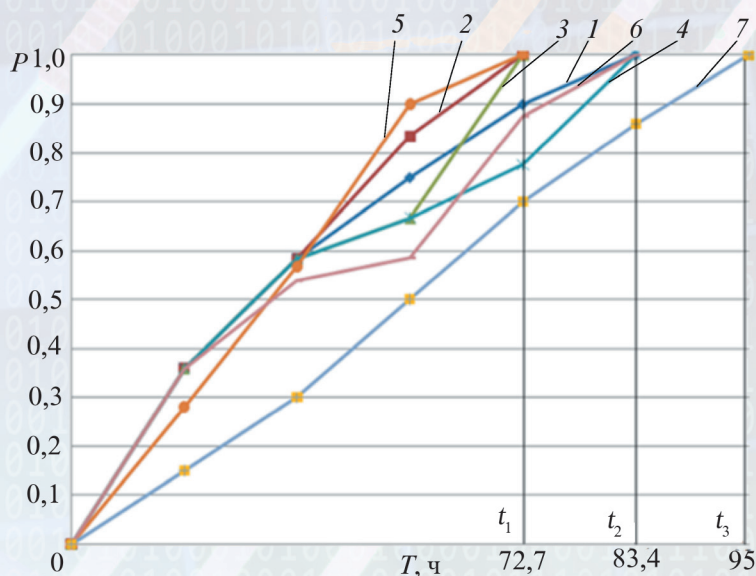


Рис. 5. График зависимости вероятности реализации угроз от времени реализации угроз:

- 1 – при реализации угрозы по пути S_{x1} ; 2 – при реализации угрозы по пути S_{x2} ;
- 3 – при реализации угрозы по пути S_{x3} ; 4 – при реализации угрозы по пути S_{x4} ;
- 5 – при реализации угрозы по пути S_{x5} ; 6 – при реализации угрозы по пути S_{z1} ;
- 7 – при реализации угрозы по пути S_{z2}

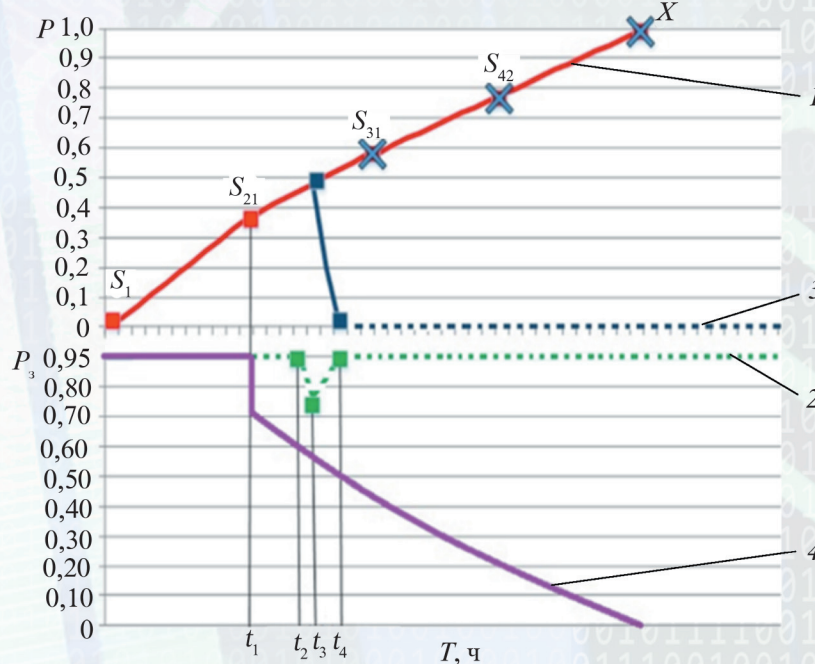


Рис. 6. Зависимости вероятности реализации угрозы P и вероятности защищенности P3 ИВС от времени при реализации предлагаемого метода с анализом динамики действий нарушителя:

- 1 – последовательность действий нарушителя для реализации угрозы X;
- 2 – уровень ИБ ИВС; t_1, t_2, t_3, t_4 – время; 3 – график зависимости защищенности ИВС от вероятности реализации угрозы; 4 – зависимость действий нарушителя от принятых мер защиты

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 25
2019
№ 1

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

DOI 10.17587/issn.1684-6400

УЧРЕДИТЕЛЬ

Издательство "Новые технологии"

СОДЕРЖАНИЕ

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

Карганов В. В., Шевченко А. А., Малышев Б. Ю. Способ повышения работоспособности информационно-вычислительной сети за счет адаптивного управления защитой 3

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

Бершадский А. М., Бождай А. С., Евсева Ю. И., Гудков А. А. Концепция рефлексивной самоадаптации прикладных программных систем 11

Сарамуд М. В., Ковалев И. В., Лосев В. В., Петросян М. О., Калинин А. О. Сравнение отказоустойчивых моделей программного обеспечения в имитационной среде исполнения 20

КОДИРОВАНИЕ И ОБРАБОТКА СИГНАЛОВ

Ефанов Д. В. Двухмодульные коды с суммированием — эффективный класс разделимых кодов для синтеза диагностического обеспечения устройств и систем управления 26

Дворников С. В., Пшеничников А. В., Манаенко С. С. Статистические характеристики помехозащищенных радиоприемных устройств с управлением частотным ресурсом 35

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИОМЕДИЦИНСКИХ СИСТЕМАХ

Исмайлова К. Ш. Применение различных методов оптимизации при расчете погрешности нейронной сети для диагностирования нервно-мышечных заболеваний 41

КОМПЬЮТЕРНАЯ ГРАФИКА

Конопацкий Е. В. Аппроксимация геометрических объектов с помощью дуг кривых, проходящих через наперед заданные точки 46

ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ

Полищук Ю. В. Способ хранения электронных документов с квазиструктурированным контентом 53

Перемитина Т. О., Ященко И. Г. Алгоритм комплексного анализа многомерных данных об экологическом состоянии окружающей среды 58

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.

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.
ПОЛЕЩУК 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

INFORMACIONNYYE TEHNOLOGII

Vol. 25
2019
No. 1

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

ISSN 1684-6400

CONTENTS

COMPUTING SYSTEMS AND NETWORKS

- Karganov V. V., Shevchenko A. A., Malyshev B. Y.** Method of Increase in Operability of an Information Network Due to the Adaptive Information Security Management 3

SOFTWARE ENGINEERING

- Bershadsky A. M., Bozhday A. S., Evseeva Yu. I., Gudkov A. A.** The Conception of Reflexive Self-Adaptation of Applied Software Systems 11
- Saramud M. V., Kovalev I. V., Losev V. V., Petrosyan M. O., Kalinin A. O.** Comparison of methodologies for increasing software fault tolerance in a simulation execution environment 20

CODING AND SIGNAL PROCESSING

- Efanov D. V.** Two-Modulus Codes with Summation — an Effective Class of Separable Codes for the Synthesis of Diagnostic Support for Devices and Control Systems 26
- Dvornikov S. V., Pshenichnicov A. V., Manaenko S. S.** Statistical Characteristics of Noise-Free Radio Links with Frequency Resource Management 35

INFORMATION TECHNOLOGIES IN BIOMEDICAL SYSTEMS

- Ismaylova K. Sh.** Application of Various Optimization Methods for Calculating the Neural Network Error for the Diagnosis of Neuromuscular Diseases 41

COMPUTER GRAPHICS

- Konopatskiy E. V.** Approximation of Geometric Objects Using Arcs of Curves Passing through to Advance Given Points 46

APPLICATION INFORMATION SYSTEMS

- Polishuk Yu. V.** The Method of Storing Electronic Documents with Semistructured Content 53
- Peremitina T. O., Yashchenko I. G.** Algorithm for Complex Analysis of Environment Ecological State 58

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"

В. В. Карганов, ст. науч. сотр., канд. техн. наук, доц., e-mail: vitalik210277@mail.ru,

А. А. Шевченко, мл. науч. сотр., e-mail: alex_pavel1991@mail.ru,

Б. Ю. Малышев, оператор научной роты, e-mail: bogdan160596@bk.ru,

Военная академия связи имени Маршала Советского Союза С. М. Буденного, Санкт-Петербург

Способ повышения работоспособности информационно-вычислительной сети за счет адаптивного управления защитой

Рассмотрен способ повышения работоспособности информационно-вычислительной сети, который аккумулирует знания о состоянии и развитии системы управления, методах и инструментах его применения в системных объектах. Инструментом решения является метод адаптивного управления защитой информационно-вычислительных сетей, отличающийся от известных тем, что предлагается применять результаты анализа динамики действий нарушителя. Представлены результаты расчетов.

Ключевые слова: информационная безопасность, информационно-вычислительная сеть, контейнерная виртуализация, система обнаружения вторжений, угроза, нарушитель

Введение

В связи с быстрым развитием компьютерных технологий, в том числе появлением сети Интернет, объединяющей разнородные сети, и переходом к информационному обществу проблема обеспечения информационной безопасности (ИБ) и построения автоматизированных систем менеджмента организации интегрированной структуры стала одной из наиболее актуальных проблем [1]. Кроме того, в ходе проведения анализа соответствующих источников в данной предметной области исследования [2–5] было выявлено, что к средствам защиты в настоящее время предъявляются более жесткие требования.

В статье [6] рассматривается способ обеспечения ИБ информационно-вычислительной сети (ИВС) путем реализации ложной сети на основе выделенного сервера с контейнерной виртуализацией [4, 7]. Однако при управлении ИВС не используются данные анализа динамики действий нарушителя. В статье [8] рассматривается способ контроля уязвимостей при масштабировании ИВС без учета динамики действий нарушителя.

При исследовании данных способов защиты ИВС недостаточно внимания уделено анализу динамики действий нарушителя, которые включают сценарии внешних и внутренних вторжений. Возникает противоречие между эффективными новыми средствами информационного вторжения и существующими способами защиты ИВС. Поэтому задача защиты ИВС от вторжений со стороны нарушителей является актуальной.

1. Цель, постановка задачи, условия и ограничения

Целью данной работы является повышение ИБ и работоспособности ИВС за счет анализа динамики действий нарушителя.

Постановка задачи: разработать метод адаптивного управления защитой ИВС на основе анализа динамики действий нарушителя путем контроля ситуационных параметров во взаимной противоборствующей обстановке при стохастической неопределенности [9, 10].

Условия и ограничения: метод адаптивного управления защитой ИВС должен включать

в себя мониторинг обстановки, оперативный контроль, распознавание последовательности действий нарушителя, моделирование стратегии воздействия нарушителя, процесс определения ситуационных параметров во взаимной противоборствующей обстановке с достоверным прогнозом стратегии вторжений.

2. Предлагаемое решение

Рассмотрена возможная структура ИВС, представленная на рис. 1.

Данная ИВС имеет топологию "Звезда" и включает в себя межсетевой экран, систему обнаружения вторжений, выделенный сервер с контейнерной виртуализацией, центр обработки данных, коммутаторы и ЭВМ пользователей. Для решения задач защиты и мониторинга ИВС необходимо не только обнаруживать и блокировать действия нарушителей, но также анализировать атаки и отвлекать нарушителей путем заманивая нарушителей на ложные информационные системы и проводить сбор информации о тактике нарушителя, осуществлять идентификацию и нейтрализацию.

В результате сочетания достоверного анализа и прогнозирования динамики действий предлагается адаптировать защиту ИВС, за счет чего должно обеспечиваться повышение оперативности отслеживания фаз развития кризисных ситуаций. На основании анализа деятельности нарушителя определяются слабые стороны системы защиты информации в ИВС.

Метод решения задачи заключается в анализе динамики действий нарушителя, обработке,

определении уязвимостей системы защиты информации при использовании выделенного сервера с контейнерной виртуализацией, прогнозировании возможных вторжений, представлении данных для выбора оптимального решения по повышению вероятности защищенности ИВС аналогично тому, как изложено в источниках [6, 11]. Учитывается динамический характер модели нарушителя, поэтому поиск и устранение уязвимостей в защите также являются динамически изменяющимися во времени процессами.

Цифровой поток, входящий и исходящий из сети Интернет, вначале проходит предварительную фильтрацию межсетевым экраном, после чего он поступает в систему обнаружения вторжений и анализируется с точки зрения наличия атак [4]. В случае, когда внутренний нарушитель пытается получить несанкционированный доступ к ресурсам ИВС, происходит анализ запросов, и если критический параметр больше допустимого уровня, то цифровой поток перенаправляется на компоненты ложной информационной системы, легитимные же запросы, удовлетворяющие требованиям политик безопасности системы обнаружения вторжений, перенаправляются на истинную информационную систему. Если же системе обнаружения вторжений не удалось обнаружить атаку на сетевом уровне, но при этом действия нарушителя были выявлены после их реализации на определенных хостах информационной системы, осуществляется перенаправление последующего цифрового потока нарушителя на компоненты выделенного сервера с контейнерной виртуализацией.

Результаты анализа работ [3–5, 7, 12] позволяют описать процесс мониторинга обстановки. Цифровой поток поступает в систему обнаружения вторжений, после чего происходит сканирование по заданным параметрам, в результате из всего цифрового потока выделяется только тот, который попадает под определенные критерии. После этого происходит выделение признаков и дальнейший анализ. Анализ осуществляется за счет уже имеющихся баз данных угроз. В случае соответствия цифрового потока критериям угроз проводится анализ моделей угроз, после чего принимается решение по защите ИВС.

Когда не удастся однозначно определить, какого рода цифровой

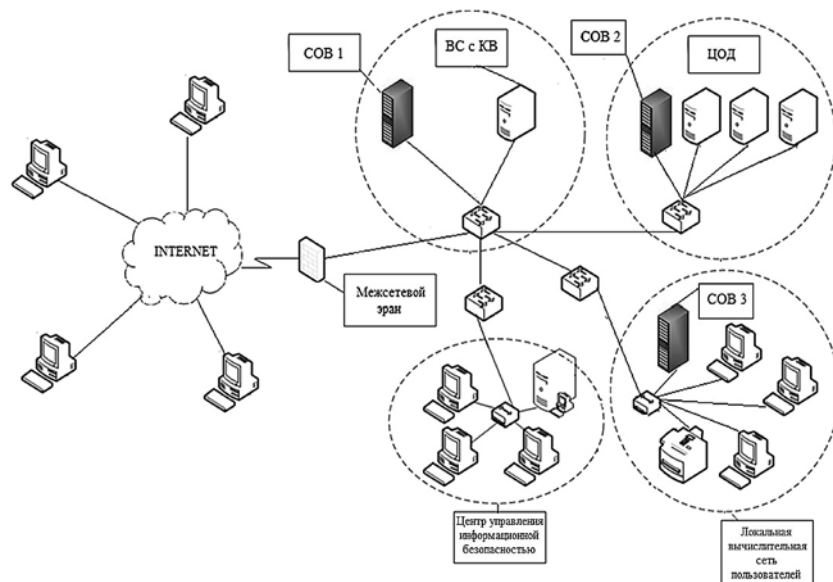


Рис. 1. Структура информационно-вычислительной сети (вариант)

поток, он отсеивается (блокируется) и перенаправляется на развернутую вычислительную сеть, и далее запросы такого рода анализируются, после чего принимаются меры по разрешению или запрету доступа. Далее выполняется обновление баз данных с учетом обнаруженных инцидентов.

На рис. 2 изображен алгоритм построения и функционирования рассматриваемой ИВС.

Данный алгоритм включает в себя два параллельных процесса:

1. Тестирование ИВС и выявление уязвимостей. Данное действие представлено в статьях [8, 13] и реализовано в работе [14].

2. Анализ цифрового потока с выявлением аномалий и последующим анализом динамики действий нарушителя. На основании динамики действий нарушителя строится модель угроз и принимаются меры по защите. Данный метод помогает защитить реальную информационную систему от компьютерных атак за счет анализа действий, выполняемых нарушителями, и принятия рациональных мер по защите реальных и возможных уязвимостей в данной сети [15].

Обобщенный алгоритм анализа действий нарушителя для данного вида атак представлен на рис. 3.

В соответствии с тем, обнаружены ли атаки на граничном хосте, их можно разделить на два типа [16–20]: обнаруживаемые и не обнаруживаемые атаки.

Атаки первой группы блокируются граничным хостом, не достигая рабочих серверов. При обнаружении такого рода атак система обнаружения вторжения (СОВ) должна изменять свою конфигурацию, чтобы последующие действия нарушителя перенаправлялись на ложные ИС. К атакам второго вида относят атаки, параметры которых не известны, а также атаки внутренних нарушителей через терминалы пользователей.

На основе возможных реализаций уязвимостей ИВС [21] составим модель нарушителя. Действия нарушителя возможно выявить при анализе общего журнала регистрации событий (сообщений от системы контроля целостности файлов; изменений настроек устройств

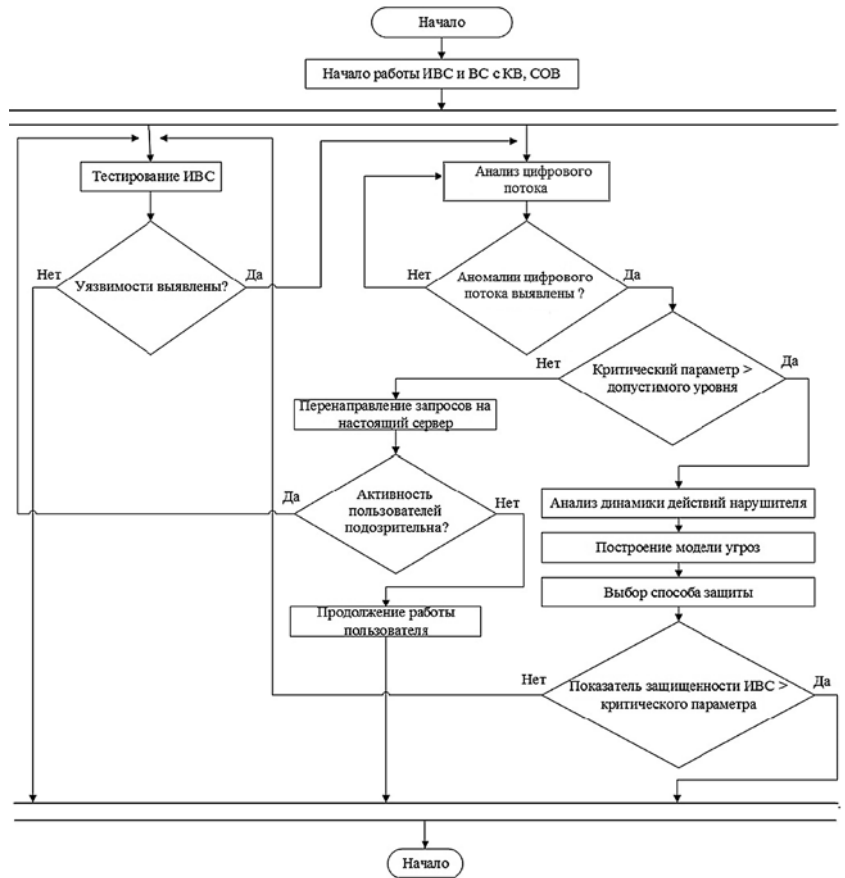


Рис. 2. Обобщенный алгоритм функционирования ИВС (ВС с КВ – выделенный сервер с контейнерной виртуализацией; СОВ – система обнаружения вторжений)



Рис. 3. Общий алгоритм анализа действий нарушителя

в ИВС). В ходе выявления в журнале регистрации этих событий действия нарушителя блокируются с оповещением администратора. На рис. 4 представлен вариант графа событий действий нарушителя.

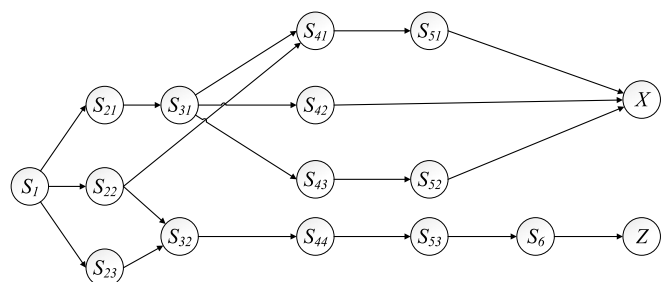


Рис. 4. Граф событий действий нарушителя (вариант)

В качестве одной из возможных моделей можно использовать представление действий нарушителя как систему с переменной структурой, поведение которой на случайных интервалах времени характеризуется различными структурами и описывается вероятностными законами [22, 23]. При этом переход одной структуры в другую происходит в случайный момент времени в зависимости от значения фазовых координат системы.

- событие " S_1 " соответствует началу действий нарушителя;
- событие " S_{21} " соответствует событию, в котором происходит измерение характеристик ИВС путем внедрения анализатора трафика;
- событие " S_{22} " соответствует стадии, в которой проводится тестирование состояния ИВС путем анализа запросов;
- событие " S_{23} " соответствует событию анализа "эхо-запросов";
- событие " S_{31} " соответствует событию анализа исходящего цифрового потока;
- событие " S_{32} " соответствует событию выявления хостов;
- событие " S_{41} " соответствует событию выявления паролей;
- событие " S_{42} " соответствует событию дешифрования информации;
- событие " S_{43} " соответствует событию, при котором несанкционированно используется авторизованный IP-адрес в сети;
- событие " S_{44} " соответствует событию, при котором происходит сканирование портов;
- событие " S_{51} " соответствует событию подмены пользователя в сети;
- событие " S_{52} " соответствует событию, при котором изменяются целостность, доступность и конфиденциальность информации;
- событие " S_{53} " соответствует событию, при котором происходит анализ характеристик приложений;
- событие " S_6 " соответствует режиму осуществления DDoS атак;
- событие " X " соответствует реализации угрозы хищения информации;
- событие " Z " соответствует реализации отказа в обслуживании.

Для разделения вариантов возможных сценариев развития событий используется схема "дерева вероятностей". Каждая ветвь представляет собой отдельный сценарий развития [21, 24]. На рис. 4 представлены семь путей для реализации угрозы хищения информации и отказа в обслуживании:

$$\begin{aligned} S_{x1} &= \{S_1, S_{21}, S_{31}, S_{41}, S_{51}, X\}; \\ S_{x2} &= \{S_1, S_{21}, S_{31}, S_{42}, X\}; \\ S_{x3} &= \{S_1, S_{21}, S_{31}, S_{43}, X\}; \\ S_{x4} &= \{S_1, S_{21}, S_{31}, S_{43}, S_{52}, X\}; \\ S_{x5} &= \{S_1, S_{22}, S_{41}, S_{51}, X\}; \\ S_{z1} &= \{S_1, S_{22}, S_{32}, S_{44}, S_{53}, S_6, Z\}; \\ S_{z2} &= \{S_1, S_{23}, S_{32}, S_{44}, S_{53}, S_6, Z\}. \end{aligned}$$

Определим вероятности реализуемости событий (элементов графа) для определения динамики действий нарушителя, воспользовавшись методикой работы [25]:

$$P = \frac{K_1 + K_2}{20}, \quad (1)$$

где K_1 — коэффициент исходной защищенности; K_2 — коэффициент реализации угрозы.

Время перехода из одного события в другое зависит от коэффициента реализуемости события:

$$T_{ij} = T_{\max j} - H_i T_{\text{исх}ij}, \quad (2)$$

где $T_{\max j}$ — максимальное время реализации j -го события ($T_{\max j} = 24$ ч); H_i — коэффициент реализуемости S_i -го события; $T_{\text{исх}ij}$ — исходное время перехода из i -го события в j -е событие ($T_{\text{исх}ij}$ от 0 до 24 ч).

Результаты вычислительного эксперимента, основанного на методике [25] и формулах (1) и (2), представлены в табл. 1.

Для определения наиболее вероятного пути реализации угроз необходимо рассчитать вероятности наступления каждого события из графа. Для расчета вероятности наступления события X и Z воспользуемся формулой расчета сложения вероятностей:

$$P_p = P_{(i)} + P_{(i+1)} - P_{(i)}P_{(i+1)},$$

где $P_{(i)}$, $P_{(i+1)}$ — вероятность наступления двух последующих событий.

Результаты расчета вероятности наступления события реализации угрозы хищения информации по возможному пути $S_{x1} = \{S_1, S_{21}, S_{31}, S_{41}, S_{51}, X\}$:

$$\begin{aligned} P_1 &= P_{S_1} + P_{S_{21}} - (P_{S_1} \times P_{S_{21}}) = 0,36; \\ P_2 &= P_1 + P_{S_{31}} - (P_1 \times P_{S_{31}}) = 0,584; \\ P_3 &= P_2 + P_{S_{41}} - (P_2 \times P_{S_{41}}) = 0,75; \\ P_4 &= P_3 + P_{S_{51}} - (P_3 \times P_{S_{51}}) = 0,9; \\ P_{\text{общ}} &= P_4 + P_X - (P_4 \times P_X) = 0,99. \end{aligned}$$

Таблица 1

**Коэффициенты реализуемости возможных событий
и время перехода из одного события в другое**

Путь S_{x1}	События	S_1	S_{21}	S_{31}	S_{41}	S_{51}	X	
	K_1	9	9	9	9	9	9	
	K_2	7	9	9	7	9	1	
	Время перехода одного события в другое $T, ч$	14,4	12,48	12,77	13,79	11,6	18,2	
Путь S_{x2}	События	S_1	S_{21}	S_{31}	S_{42}	X		
	K_1	9	9	9	9	9		
	Y_2	7	9	9	3	1		
	Время перехода одного события в другое $T, ч$	14,4	12,48	12,77	16,34	15,8		
Путь S_{x3}	События	S_1	S_{21}	S_{31}	S_{43}	X		
	K_1	9	9	9	9	9		
	K_2	7	9	9	1	1		
	Время перехода одного события в другое $T, ч$	14,4	12,48	12,77	17,62	15,19		
Путь S_{x4}	События	S_1	S_{21}	S_{31}	S_{43}	S_{52}	X	
	K_1	9	9	9	9	9	9	
	K_2	7	9	9	1	1	1	
	Время перехода одного события в другое $T, ч$	14,4	12,48	12,77	17,62	15,19	16,41	
Путь S_{x5}	События	S_1	S_{22}	S_{41}	S_{51}	X		
	K_1	9	9	9	9	9		
	K_2	7	7	7	9	1		
	Время перехода одного события в другое $T, ч$	14,4	12,48	14,02	11,39	18,31		
Путь S_{z1}	События	S_1	S_{21}	S_{32}	S_{44}	P_{53}	P_6	Z
	K_1	9	9	9	9	9	9	9
	K_2	7	9	4	9	9	7	4
	Время перехода одного события в другое $T, ч$	14,4	11,04	16,82	8,86	16,03	11,18	16,73
Путь S_{z2}	События	S_1	S_{23}	S_{32}	S_{44}	S_{53}	S_6	Z
	K_1	9	9	9	9	9	9	9
	K_2	7	4	4	9	9	7	4
	Время перехода одного события в другое $T, ч$	14,4	14,64	14,48	10,97	14,13	12,69	15,75

По аналогии рассчитывается вероятность наступления событий по всем путям графа. Результаты расчета зависимости вероятности реализации угроз от времени представлены на рис. 5 (см. вторую сторону обложки).

На рис. 5 $S_{x1}, S_{x2}, S_{x3}, S_{x4}, S_{x5}, S_{z1}, S_{z2}$ — пути реализации угроз X и Z ; t_1, t_2, t_3 — время, которое понадобится нарушителю для реализации угроз с учетом вариаций возможных действий нарушителя. Если произойдет событие S_1 , то для достижения угрозы хищения информации нарушитель выберет путь реализации S_{x2}, S_{x5} , а для достижения угрозы отказа в обслуживании он выберет путь S_{z2} .

Рассмотрим предлагаемый метод для анализа динамики действий нарушителя на примере защиты от угрозы хищения информации по пути реализации S_{x2} . На рис. 6 (см. вторую сторону обложки) представлена зависимость вероятности реализации угрозы и вероятности защищенности ИВС от времени реализации метода адаптивного управления защитой ИВС на основе анализа динамики действий нарушителя.

В промежуток времени t_1 происходит внедрение анализатора трафика. В момент времени t_2 происходит обнаружение системой обнаружения вторжений данного воздействия, при этом уровень защищенности падает. В дальнейшем происходит построение модели угроз. Затем принимаются меры по нейтрализации угрозы, которая была обнаружена, с принятием актуальных мер защиты объектов ИВС, которые будут атакованы нарушителем в ближайшее время согласно графу действий нарушителя.

После принятия в момент t_4 мер защиты реализация следующего воздействия нарушителя V_{31} уже невозможна, в силу снижения вероятности реализации угрозы к нулю защищенность вернется на уровень в 95 %.

Процесс проактивного обнаружения вторжений основывается на анализе запросов и удовлетворении их критериям, при этом сравнение проводится не только по идеальным моделям и критериям раннее обнаруженных угроз, но также и с помощью построения путей реализации угрозы, позволяющих определить динамику действий нарушителя.

Таблица 2

Сравнение адаптивного и традиционного методов управления защитой ИВС

Событие	Коэффициент реализуемости события		Время перехода одного события в другое, ч	
	Адаптивный метод управления защитой ИВС	Традиционный метод управления защитой ИВС	Адаптивный метод управления защитой ИВС	Традиционный метод управления защитой ИВС
S_1	0,8	0,8	14,4	14,4
S_{21}	0,8	0,9	12,48	11,04
S_{31}	0,9	0,9	12,77	14,06
S_{41}	0,8	0,8	13,79	12,75
S_{51}	0,9	0,9	11,6	12,52
X	0,5	0,9	18,2	12,73
Время реализации угрозы			82,64	77,5

Результаты сравнительного анализа адаптивного и традиционного методов управления защитой ИВС представлены в табл. 2. Анализ проводили на основе вычислительного эксперимента в предложенном варианте ИВС с помощью методики [25].

При использовании адаптивного метода управления защитой ИВС нарушитель потратит на 7 % больше времени на реализацию угрозы хищения информации, чем при использовании традиционного метода, что и является положительным эффектом предлагаемого метода защиты ИВС на основе анализа динамики действий нарушителя.

Заключение

Разработан способ повышения работоспособности ИВС за счет адаптивного управления защитой ИВС, отличающийся от известных методов, основанных на использовании специальных мер защиты, тем, что предложено применять результаты анализа динамики действий нарушителя. Предусмотрен алгоритм контроля ситуационных параметров во взаимной противоборствующей обстановке при стохастической неопределенности. Представлена архитектура прототипа этой системы, а также сценарии экспериментов, проводимых с прототипом. Рассмотрены текущее состояние и процедура анализа динамики действий нарушителя. Этот подход можно реализовать на программной эмуляции компонентов информационной системы введения нарушителя в заблуждение:

1) сегмента сети, где осуществляется эмуляция работы выделенного сервера с контейнерной виртуализацией (дубликат сети с рабочими серверами);

2) дубликата хоста рабочих серверов (хост-приманка);

3) дубликат сервисов и приложений — программы, которые копируют работу сервисов и приложений.

Управление защитой ИВС на основе результатов анализа динамики действий нарушителя ведет администратор безопасности. Выделенный сервер с контейнерной виртуализацией и система обнаружения вторжений могут поддерживаться стандартными операционными системами, могут включаться как дополнительные средства в действующие системы безопасности, повышая вероятность защищенности ИВС.

Список литературы

1. Андрианов В. И., Красов А. В., Липатников В. А. Инновационное управление рисками информационной безопасности. СПб.: Изд. Федеральное агентство связи, Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича, 2012. С. 396.
2. Карганов В. В., Левченко Г. Н., Драчев В. О., Котышчев И. А. К вопросу о существующих методах защиты информации в информационных системах // Матер. конф. ГНИИ "Нацразвитие". 2017. С. 108—117.
3. Карганов В. В. Концептуальные подходы качества обработки информации в информационной системе // Матер. конф. ГНИИ "Нацразвитие". 2017. С. 100—107.
4. ИСО/МЭК 27001. Информационные технологии. Методы защиты. Системы менеджмента защиты информации. Требования. Международный стандарт ISO. URL: <http://www.novsu.ru/file/1020711> (дата обращения: 15.07.2018).
5. ГОСТ Р ИСО/МЭК 13335-1-2006. Информационная технология. Методы и средства обеспечения безопасности. Часть 1. Концепция и модели менеджмента безопасности информационных и телекоммуникационных технологий // ГОСТ-эксперт. URL: <http://gostexpert.ru/gost/gost-13335-1-2006> (дата обращения: 15.07.2018).
6. Липатников В. А., Шевченко А. А., Яцкин А. Д., Семенова Е. Г. Управление информационной безопасностью организации интегрированной структуры на основе выделенного сервера с контейнерной виртуализацией // Информационно-управляющие системы. 2017. № 4 (89). С. 67—76. Doi:10.15217/issn1684-8853.2017.4.67.
7. Лукацкий А. Обнаружение атак. СПб.: БХВ-Петербург, 2008. 304 с.
8. Липатников В. А., Шевченко А. А. Способ контроля уязвимостей при масштабировании автоматизированной системы менеджмента предприятия интегрированной структуры // Информационные системы и технологии. 2016. № 2(94). С. 128—140.
9. Batista I. Model predictive control for stochastic systems by randomized algorithms. Eindhoven: Technische Universiteit Eindhoven, 2004.
10. Byres E., Lowe J. The myths and facts behind cyber security risk for industrial control systems // In ISA Process Control Conference, 2003.
11. Вандич А. П., Яцкин М. А., Карганов В. В., Привалов А. А., Скуднева Е. В. К вопросу об организации инфор-

мационного обмена для повышения защищенности сети передачи данных от технической компьютерной разведки // Труды ЦНИИС. Санкт-Петербургский филиал. 2017. Т. 1, № 4. С. 72—78.

12. **Карганов В. В., Драчев В. О., Левченко Г. Н.** Формирование модели предметной области для информационной системы // Инновационные технологии и технические средства специального назначения. Тр. десятой общерос. науч.-практ. конф. 2018. С. 264—268.

13. **Карганов В. В., Пилявец О. Г., Шевченко А. А.** К вопросу предупреждения и обеспечения требуемого уровня информационной безопасности информационно-вычислительной сети специального назначения от несанкционированных воздействий // Вопросы оборонной техники. Сер. 16. Технические средства противодействия терроризму. 2018. № 1—2 (115—116). С. 78—85.

14. **Патент 2635256.** Российская Федерация, МПК G06F 12/14. Способ защиты информационно-вычислительной сети от несанкционированных воздействий / Карганов В. В., Костарев С. В., Липатников В. А., Лобашев А. И., Шевченко А. А.; заявитель и патентообладатель Федеральное государственное казенное военное образовательное учреждение высшего образования "Военная академия связи имени Маршала Советского Союза С. М. Буденного" Министерства обороны Российской Федерации. № 2016117662; заявл. 04.05.2016; опубл. 09.11.2017, Бюл. № 31. 2 с.

15. **Кузнецов И. А., Липатников В. А., Шевченко А. А.** Способ многофакторного управления безопасностью информационно-телекоммуникационной сети системы менеджмента качества предприятий интегрированных структур // Вопросы радиоэлектроники. 2016. № 6. С. 23—28.

16. **Baddar S. A.-H., Merlo A., Migliardi M.** Anomaly Detection in Computer Networks: A State-of-the-Art Review //

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications. 2014. Vol. 5, N. 4. P. 29—64.

17. **Brindasri S., Saravanan K.** Evaluation Of Network Intrusion Detection Using Markov Chain // International Journal on Cybernetics & Informatics (IJCI). 2014. Vol. 3, N. 2. P. 11—20.

18. **Mazurek M., Dymora P.** Network anomaly detection based on the statistical selfsimilarity factor for HTTP protocol // Przegląd elektrotechniczny. 2014. P. 127—130.

19. **Ranjan R., Sahoo G.** A new clustering approach for anomaly intrusion detection // International Journal of Data Mining & Knowledge Management Process (IJDKP). 2014. Vol. 4, N. 2. P. 29—38.

20. **Sheth H., Shah B., Yagnik S.** A survey on RBF Neural Network for Intrusion Detection System // Int. Journal of Engineering Research and Applications. 2014. Vol. 4. P. 17—22.

21. **Браницкий А. А., Котенко И. В.** Анализ и классификация методов обнаружения сетевых атак // Тр. СПИИРАН. 2016. Вып. 2(45). С. 207—243.

22. **Pawar S. N.** Intrusion Detection in Computer Network using Genetic Algorithm Approach: A Survey // International Journal of Advances in Engineering & Technology. 2013. Vol. 6, Iss. 2. P. 730—736.

23. **Dave M. H., Sharma S. D.** Improved Algorithm for Intrusion Detection Using Genetic Algorithm and SNORT // International Journal of Emerging Technology and Advanced Engineering. 2014. P. 273—276.

24. **Ryan J., Lin M.-J.** Intrusion Detection with Neural Networks // Advances in Neural Information Processing Systems. 1998. P. 943—949.

25. **ФСТЭК.** Методика определения актуальных угроз безопасности персональных данных при их обработке в информационных системах. URL: <https://fstec.ru/component/attachments/download/290> (дата обращения: 15.07.2018).

V. V. Karganov, Senior Research Assistant, Candidate Of Technical Sciences, Associate Professor, e-mail: vitalik210277@mail.ru,

A. A. Shevchenko, Junior Research Assistant, e-mail: alex_pavell1991@mail.ru,

B. Y. Malyshev, Operator Of Scientist Company, e-mail: bogdan160596@bk.ru,

S. M. Budyonny Military Academy of Communications, Saint Petersburg, Russia

Method of Increase in Operability of an Information Network Due to the Adaptive Information Security Management

In article describes adaptive management, which is a separate type of management, namely, flexible and innovative. It accumulates knowledge about condition and development of the control system, methods and tools of its application in system objects. The solution tool is a method of adaptive control of informatively — computer network security (IAS), which differs from the known ones, in that the analysis results of the dynamics violator's actions are applied. The method contains: monitoring of the situation, operational control, recognition of the sequence of actions of the violator, modeling the strategy of influence of the violator, the process of determining the situational parameters with a reliable forecast of the invasion strategy. In the process of analysis, the network administrator receives information about the priority goals of the offender, the means used by him and the vulnerabilities of various elements, which makes it possible to quickly take measures to improve the security of the network and avoid compromising it. The situational parameters control algorithm in the mutual opposing situation at stochastic uncertainty is provided. The architecture of the prototype of this system is presented, as well as scenarios of experiments conducted with the prototype. The current state and the procedure for analyzing the dynamics of the violator's actions are considered. The results of calculations in tabular form are presented, namely: probabilities of realizability of possible events time of transition from one event to another; the timing of the realization of the threat of information theft X on the way P_{xj} using methods of adaptive management and the traditional protection of the IVS. Presented the calculation results of dependence probability of threats: from time to time, the probability of protection IVS for implementation of the proposed method based on the analysis of the dynamics offender's actions. The conclusions, the essence of which lies in the fact that the use of the method allows to maintain the efficiency of IVS at the required level with the dynamics of changing the threats set, taking into account the scaling in the planning and making changes to it in the conditions of information confrontation.

Keywords: the automated organization management system of integrated structure, data-processing network, information security, computer attacks, information security, risk assessment, container virtualization, proactive management, scaling, protectability index, system of detection of invasions, threat, violator

DOI: 10.17587/it.25.3-10

References

1. **Andrianov V. I., Krasov A. V., Lipatnikov V. A.** *Innovacionnoe upravlenie riskami informacionnoj bezopasnosti* (Innovation information security risk management), SPb., Federal'noe agentstvo svyazi, Sankt-Peterburgskij gosudarstvennyj universitet telekommunikacij im. prof. M. A. Bonch-Bruevicha, 2012, pp. 396 (in Russian).
2. **Karganov V. V., Levchenko G. N., Drachev V. O., Kotyashichev I. A.** *K voprosu o sushchestvuyushchih metodah zashchity informacii v informacionnyh sistemah* (Revisited actual methods of guarding in information systems), *Materialy konferencii GNII "NACRAZVITIE"*, 2017, pp. 108–117 (in Russian).
3. **Karganov V. V.** *Konceptual'nye podhody kachestva obrabotki informacii v informacionnoj sisteme* (Conceptual accesses quality of information handling in data system), *Materialy konferencij GNII "NACRAZVITIE"*, 2017, pp. 100–107 (in Russian).
4. **ISO/MEHK 27001.** *Informacionnye tekhnologii. Metody zashchity. Sistemy menedzhmenta zashchity informacii. Trebovaniya* (Information technology. Security techniques. Information security management system. Requirements), Mezhdunarodnyi standart ISO, available at: <http://www.novsu.ru/file/1020711> (date of access: 15.07.2018) (in Russian).
5. **GOST R ISO/MEHK 13335-1-2006.** *Informacionnaya tekhnologiya. Metody i sredstva obespecheniya bezopasnosti. CHast' 1. Koncepciya i modeli menedzhmenta bezopasnosti informacionnyh i telekommunikacionnyh tekhnologii* (Information technology. Security techniques. Part 1. Concepts and models for information and communications technology security management), GOST-ehkspert, available at: <http://gostexpert.ru/gost/gost-13335-1-2006> (date of access: 15.07.2018) (in Russian).
6. **Lipatnikov V. A., Shevchenko A. A., Yackin A. D., Semenova E. G.** *Upravlenie informacionnoj bezopasnost'yu organizacii integrirovannoj struktury na osnove vydelenogo servera s kontejnernoj virtualizaciej* (Information security management of the integrated structure organization based on a dedicated server with a container virtualization), *Informacionno-Upravlyayushchie Sistemy*, 2017, no. 4 (89), pp. 67–76, doi:10.15217/issn1684-8853.2017.4.67 (in Russian).
7. **Lukackii A.** *Obnaruzhenie atak* (Detection of computer attacks), SPb., BHV-Peterburg, 2008, 304 p. (in Russian).
8. **Lipatnikov V. A., Shevchenko A. A.** *Sposob kontrolya uyazvimostej pri masshtabirovanii avtomatizirovannoj sistemy menedzhmenta predpriyatiya integrirovannoj struktury* (The vulnerability control method applying while automated integrated structure organization management system scaling), *Informacionnye Sistemy i Tekhnologii*, 2016, no.2(94), pp. 128–140 (in Russian).
9. **Ivo Batina.** *Model predictive control for stochastic systems by randomized algorithms*, Eindhoven, Technische Universiteit Eindhoven, 2004.
10. **Byres E., Lowe J.** The myths and facts behind cyber security risk for industrial control systems, *In ISA Process Control Conference*, 2003.
11. **Vandich A. P., Yaichkin M. A., Karganov V. V., Privolov A. A., Skudneva E. V.** *K voprosu ob organizacii informacionnogo obmena dlya povysheniya zashchishchennosti seti peredachi dannyh ot tekhnicheskoy komp'yuternoj razvedki* (Revisited administration of data exchange for rising immunity of communications network from technical computer scouting), *Trudy CNIIS*, 2017, vol. 1, no. 4, pp. 72–78 (in Russian).
12. **Karganov V. V., Drachev V. O., Levchenko G. N.** *Formirovaniye modeli predmetnoj oblasti dlya informacionnoj sistemy* (Model formation of application domain for data system), *Innovacionnye Tekhnologii I Tekhnicheskie Sredstva Special'nogo Naznacheniya*, *Trudy desyatoy obshcherossijskoj nauchno-prakticheskoy konferencii*, 2018, pp. 264–268 (in Russian).
13. **Karganov V. V., Pilyavec O. G., Shevchenko A. A.** *K voprosu preduprezhdeniya i obespecheniya trebuemogo urovnya informacionnoj bezopasnosti informacionno-vychislitel'noj seti special'nogo naznacheniya ot nesankcionirovannyh vozdeystvij* (To the issue of prevention and ensuring the required level of information security of information network of A special purpose from unauthorized influences), *Voprosy oboronnoj tekhniki. Seriya 16: Tekhnicheskie sredstva protivodeystviya terrorizmu*, 2018, no. 1–2 (115–116), pp. 78–85 (in Russian).
14. **Patent 2635256.** *Rossiyskaya Federaciya, MPK G06F 12/14. Sposob zashchity informacionno-vychislitel'noj seti ot nesankcionirovannyh vozdeystvij* (Method of defense of information-calculation network from illegal actions), Karganov V. V., Kostarev S. V., Lipatnikov V. A., Lobashev A. I., Shevchenko A. A.; zayavitel' i patentoobladatel' Federal'noe gosudarstvennoe kazennoe voennoe obrazovatel'noe uchrezhdenie vysshego obrazovaniya "Voennaya akademiya svyazi imeni Marshala Sovetskogo Soyuza S. M. Budennogo" Ministerstva oborony Rossijskoj Federacii. — № 2016117662; zayavl. 04.05.2016; opubl. 09.11.2017, Byul. 31, 2 p. (in Russian).
15. **Kuznecov I. A., Lipatnikov V. A., Shevchenko A. A.** *Sposob mnogofaktornogo upravleniya bezopasnost'yu informacionno-telekommunikacionnoj seti sistemy menedzhmenta kachestva predpriyatij integrirovannyh struktur* (Multivariable control technique of information-telecommunication network security of integrated structure organization quality management system), *Voprosy radioelektroniki*, 2016, no. 6, pp. 23–28 (in Russian).
16. **Baddar S. A.-H., Merlo A., Migliardi M.** Anomaly Detection in Computer Networks: A State-of-the-Art Review, *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 2014, vol. 5, no. 4, pp. 29–64.
17. **Brindasri S., Saravanan K.** Evaluation Of Network Intrusion Detection Using Markov Chain, *International Journal on Cybernetics & Informatics (IJCI)*, 2014, vol. 3, no. 2, pp. 11–20.
18. **Mazurek M., Dymora P.** Network anomaly detection based on the statistical selfsimilarity factor for HTTP protocol, *Przeglad Elektrotechniczny*, 2014, pp. 127–130.
19. **Ranjan R., Sahoo G.** A new clustering approach for anomaly intrusion detection, *International Journal of Data Mining & Knowledge Management Process (IJDKP)*, 2014, vol. 4, no. 2, pp. 29–38.
20. **Sheth H., Shah B., Yagnik S.** A survey on RBF Neural Network for Intrusion Detection System, *Int. Journal of Engineering Research and Applications*, 2014, vol. 4, pp. 17–22.
21. **Branickij A. A., Kotenko I. V.** *Analiz i klassifikaciya metodov obnaruzheniya setevyh atak* (Analysis and classification of methods for network attack detection), *Trudy SPIIRAN*, 2016, iss. 2(45), pp. 207–243 (in Russian).
22. **Pawar S. N.** Intrusion Detection in Computer Network using Genetic Algorithm Approach: A Survey, *International Journal of Advances in Engineering & Technology*, 2013, vol. 6, iss. 2, pp. 730–736.
23. **Dave M. H., Sharma S. D.** Improved Algorithm for Intrusion Detection Using Genetic Algorithm and SNORT, *International Journal of Emerging Technology and Advanced Engineering*, 2014, pp. 273–276.
24. **Ryan J., Lin M.-J.** Intrusion Detection with Neural Networks, *Advances in Neural Information Processing Systems*, 1998, pp. 943–949.
25. **FSTEHK.** *Metodika opredeleniya aktual'nyh ugroz bezopasnosti personal'nyh dannyh pri ih obrabotke v informacionnyh sistemah* (Method of determination actual security risks in personal data while their processing in informational systems), available at: <https://fstec.ru/component/attachments/download/290> (date of access: 15.07.2018) (in Russian).

А. М. Бершадский, д-р техн. наук, проф., e-mail: bam@pnzgu.ru,
А. С. Бождай, д-р техн. наук, проф., e-mail: bozhday@yandex.ru,
Ю. И. Евсева, канд. техн. наук, доц., e-mail: shymoda@mail.ru,
А. А. Гудков, канд. техн. наук, доц., e-mail: alexei-ag@yandex.ru,
Пензенский государственный университет

Концепция рефлексивной самоадаптации прикладных программных систем¹

Предложен новый подход к решению важной проблемы современной IT-индустрии — самоадаптации программных компонентов на основе концепции поведенческой рефлексии. Рассматриваются базовые принципы рефлексивной адаптации, а также идея использования инженерии линеек программных продуктов для создания адаптивных программ. Приводится обобщенное архитектурное решение для построения рефлексивных самоадаптивных систем, способных менять свои поведенческие характеристики непосредственно в ходе исполнения без перекомпиляции исходного кода. Отдельное внимание уделено вопросам формализованного представления и использования моделей изменчивости при реализации вариантов адаптивного поведения прикладных программ.

Ключевые слова: адаптивные программные системы, рефлексивная самоадаптация, модель изменчивости, диаграмма характеристик, линейки программных продуктов, программная инженерия

Введение

Обеспечение информационных потребностей современного общества зависит от использования огромного арсенала программно-аппаратных комплексов и систем. Интенсивность и динамика информационных процессов предъявляют жесткие требования к поведенческим свойствам программного обеспечения (ПО). Суть этих требований сводится к тому, что прикладная программа должна функционировать адекватно специфике и содержанию информационных потоков своей предметной области, т. е. своевременно адаптироваться к непрерывным изменениям внешней среды. Существует целый ряд прикладных задач, для которых остановка обслуживающего ПО (для внесения обновлений в математические и информационные модели, алгоритмы, исходный код, интерфейсные оболочки и т. п.) сопряжена с огромными трудностями или вовсе невозможна. И единственным выходом в этом случае является самоадаптация программы в режиме реального времени, без остановок на

перепроектирование и перекомпиляцию. Такие программные компоненты должны уметь самостоятельно отслеживать критические изменения предметной области, адекватно приспосабливаясь к ним в фоновом режиме, не прекращая при этом выполнять свои основные полезные функции.

Таким образом, самоадаптация прикладного ПО — это одна из принципиально важных задач в сфере современной программной инженерии, имеющая как теоретическую важность, так и огромное практическое значение. Особую актуальность она имеет и для российского IT-сектора, поскольку в ближайшие годы следует ожидать повышенного интереса со стороны правительственных структур, бизнеса, науки к вопросам интеграции социальных и программно-кибернетических процессов. Это связано с приоритетной программой перехода страны к моделям цифровой экономики. По прогнозам экспертов, наибольшего развития к 2025 г. должны достичь сферы киберфизических и информационных систем, работающих с большими данными (BigData).

На фоне новых вызовов не следует забывать и о традиционных проблемах IT-индустрии. Прежде всего, это возрастающая структурная

¹ Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 18-07-00408.

сложность и даже громоздкость современных программных систем прикладного назначения и, как следствие, увеличение времени и стоимости разработки, трудности сопровождения, сокращение длительности жизненного цикла, снижение отказоустойчивости, надежности, гибкости. Неспособность к быстрому самовосстановлению ПО в аварийных ситуациях может привести к потерям больших объемов данных, а в условиях динамично изменяющегося современного мира многие информационные системы (например, системы, обеспечивающие работу крупных предприятий, ERP-системы), лишённые способности саморазвития и оперативной адаптации, быстро устаревают.

Создание фундаментальных научных основ, практических приемов и универсальных механизмов программной самоадаптации — перспективный путь для решения большинства перечисленных трудностей и задач. Многие проблемы, связанные с вопросами адаптивного поведения искусственных систем (прежде всего технических), успешно решались в рамках теории автоматического управления и регулирования. Базовые понятия и принципы этой научной дисциплины можно и в настоящее время считать актуальными для большинства объектов и явлений окружающего мира, обладающих системными свойствами. Программные системы не являются исключением. Однако практический опыт последних лет показал, что механизмы обратной связи, адаптации и системной рефлексии в программном обеспечении до сих пор плохо формализованы, а методы реализации адаптивного поведения в технических системах, предлагаемые классической теорией управления, не актуальны для программных систем в силу их информационной природы. Большинство существующих работ, посвященных вопросам адаптивного поведения в программных системах, сводятся к созданию различных архитектурных и программно-технических решений, преимущественно частного или специфического характера [1—6]. Однако эта проблема имеет гораздо более общий характер и требует разработки фундаментальных принципов, которые будут положены в основу универсальных механизмов самоадаптации для широкого класса прикладного ПО.

Проблематика системного поведения на основе обратных связей, в том числе и вопросы адаптации, является традиционным предметом изучения кибернетики. За более чем полувековую историю развития этой науки накоплен богатый опыт концептуальных, математических, информационных и технических решений по-

добных задач. Поэтому не случайно, что в начале 2000-х годов появились работы, использующие традиционные кибернетические подходы для описания программного поведения [7]. Термин "программная кибернетика" (software cybernetics) был введен известным специалистом в сфере программной инженерии К. Саи в работе [8] и первоначально обозначал попытку применить методы классической кибернетики и теории управления к программным системам для оптимизации длительности жизненного цикла и сокращения затрат на разработку сложных программных систем.

За 15 лет существования программная кибернетика претерпела существенные изменения и в настоящее время включает в себя не только попытки реализации базовых кибернетических принципов применительно к структуре ПО, но и достаточно оригинальные концепции, связанные с вопросами распределенных и облачных вычислений, создания и эксплуатации киберфизических систем (в частности, сетевых систем и интернета вещей), процессов разработки, тестирования и сопровождения программ [9—11]. Но несмотря на столь высокий интерес к этому направлению, тема создания универсальных (иными словами, применимых для широкого класса прикладных программных систем) механизмов адаптации и самоадаптации практически не изучена. Существующие подходы, изложенные, в частности, в работах [12, 13], не позволяют решить весь класс проблем, связанных с адаптацией. Системы, построенные с их использованием, не являются достаточно гибкими и надежными [14].

Особенно перспективными нам видятся идеи самоадаптации программных систем, выстроенные в близкой аналогии с принципами адаптации высокоорганизованных биологических существ. Речь идет о рефлексивных типах адаптивного поведения, которые связаны с сознательной деятельностью живого организма: самоанализом, способностью ориентироваться в сложных ситуациях и планировать собственные действия, обучением посредством наблюдения за собственными состояниями и объектами внешнего мира. Подобный выбор связан с тем, что современные программные системы прикладного назначения, обрабатывающие большие объемы сложноструктурированной и неструктурированной информации, имеющие крайне сложную архитектуру, должны иметь соответствующую поведенческую сложность и разнообразие без нарушения заданной функциональной компактности.

1. Концепция рефлексии

Рефлексия — термин, появившийся изначально в философии и впоследствии ставший популярным в других областях знаний, в частности в психологии. Впервые феномен рефлексии был основательно исследован в работах английского философа Дж. Локка, который считал, что существуют два источника человеческих знаний: объекты внешнего мира и деятельность собственного ума. К деятельности ума Локк причислял мышление, рассуждения, сомнение и считал, что все это познается с помощью специального внутреннего чувства — рефлексии, "наблюдения, которому ум подвергает свою деятельность" [15].

В настоящее время рефлексия принято определять как мыслительный процесс, направленный на самоосознание, самопознание, анализ своих эмоций и чувств, состояний, способностей, целей, стратегий поведения. Данный процесс свойственен исключительно человеческому разуму и является неотъемлемым компонентом социально-психологической и психофизиологической адаптации личности. Без рефлексии невозможно полноценное развитие индивидуума [16].

Современные программные системы и соответствующие им предметные области таковы, что невозможно заранее, на этапе проектирования, учесть все возможные обстоятельства функционирования. Многие вещи становятся очевидными лишь в процессе эксплуатации, а некоторые так и остаются скрытыми. Это вынуждает разработчиков проявлять повышенное внимание к процессу сопровождения системы, регулярно выпускать обновления, не содержащие кардинально новой функциональности.

Решением проблемы недостаточности знаний может стать надделение систем способностью к самостоятельному анализу собственного состояния. В этом случае разработчику достаточно заложить в программу лишь базовые механизмы анализа и поиска закономерностей (тем самым предполагая, что какие-то моменты на этапе разработки остались неучтенными), а выявлением недостающих знаний система будет заниматься самостоятельно на протяжении своего жизненного цикла. Такой подход позволит не только снизить затраты на сопровождение системы, но и выявить те сведения о ее предметной области, которые могут быть неизвестны даже экспертам.

В этом отношении процесс самоанализа системы будет схож с процессом рефлексии человеческого разума. Объектом данного процесса

будет собственная поведенческая продукция системы, а результатом — новые обратные связи, которые позволят системе в дальнейшем более эффективно функционировать. Как и в случае с человеческой рефлексией, программная рефлексия будет одним из основных компонентов высокоуровневой самоадаптации.

Перечислим базовые принципы концепции рефлексивной адаптации и требования к ее внедрению в сфере программной инженерии:

- инвариантность механизма самоадаптации к предметной области, типу и назначению программной системы;
- масштабируемость. В процедуре рефлексии должен учитываться тот факт, что уровень организации программной системы может быть различным: от простейшей утилиты до программного комплекса на основе нескольких систем. Также необходимо учесть, что адаптироваться может как система в целом, так и отдельные ее компоненты, при этом должны использоваться одни и те же методы адаптации;
- интеллектуальный анализ ретроспективных данных. Новые поведенческие обратные связи формируются путем выявления зависимостей и закономерностей в информации, отражающей работу программной системы на некотором временном промежутке в прошлом.

Рефлексия программных систем, безусловно, не является полноценным аналогом человеческой рефлексии, поскольку программы не обладают сознанием. Однако даже в упрощенной форме дополнительные рефлексивные контуры в структуре ПО позволяют решить многие задачи, связанные с недостаточностью знаний о предметной области системы и неполнотой процессов отладки и тестирования.

2. Архитектура адаптивной программной системы на основе рефлексивных компонентов

Рассмотренный выше принцип масштабируемости накладывает определенные требования на архитектурную организацию программной системы. Адаптируемая архитектура должна легко модифицироваться, причем речь идет не только о перестройке программы по результатам рефлексивного анализа, но и о модификации программы извне (например, самими разработчиками) и последующем учете изменений в процессе эксплуатации. Так, новый модуль, подключенный к системе, должен быть сразу же включен в общий процесс рефлексии, в резуль-

тате которого будет выявлена его связь с другими модулями, их взаимное влияние друг на друга. Помимо этого, определенные закономерности должны быть выявлены в рамках функционирования самого модуля.

Однако динамика современного мира такова, что модификацию программной системы коллективом (или коллективами) разработчиков часто не удается провести своевременно. Это может повлечь за собой возникновение аварийных или нестабильных ситуаций, сопровождаемых серьезными финансовыми потерями, техногенными и социально-экономическими катастрофами. Современный бизнес требует оперативных реакций как от людей и организаций, так и от программных продуктов, обеспечивающих их работу. Без сомнения, самоадаптивные системные компоненты окажутся незаменимым инструментом при решении этой непростой задачи.

Обобщая существующий опыт разработки систем с модульной архитектурой, можно сформулировать несколько важных принципов построения программных систем с рефлексивной самоадаптацией функционала:

- минимальными компонентами программной системы, подверженными адаптации, являются блоки изменчивости. Каждый блок изменчивости, входящий в состав системы, отвечает за возможную реализацию какого-либо атомарного компонента программы (функции, класса, модели объекта предметной области и т. п.);
- совокупность логически связанных блоков изменчивости, относящихся к одной предметной области и предназначенных для решения одной задачи, называется рефлексивным компонентом;
- рефлексивная программная система может состоять из одного или нескольких рефлексивных компонентов. Их число будет определять адаптивное разнообразие и рефлексивную сложность программы;
- рефлексивные компоненты объединены в единую систему посредством общего интерфейса и изначально могут не иметь данных друг о друге. Однако в процессе рефлексивного анализа будет происходить выявление степени и характера влияния одних компонентов на другие. Для этого необходимо предусмотреть каналы информационного обмена между компонентами.

Важно понимать, что рефлексивный компонент — это не обязательно программный модуль в классическом понимании. В ряде случаев выгодно иметь компоненты, реализованные

в форме сервисов в рамках концепции сервис-ориентированной архитектуры (service-oriented architecture [17]). К преимуществам сервис-ориентированного рефлексивного компонента можно отнести:

- низкую связность. Сервисы, входящие в состав системы, могут быть реализованы независимо от других ее служб, а также могут входить в состав других систем. Единственное требование — знание соответствующих протоколов взаимодействия;
- принципиальную допустимость использования функций одного сервиса сторонними приложениями и системами. Если возникнет необходимость в модификации определенной функциональности, то достаточно будет изменить ее только в одном сервисе, а не в каждой использующей его системе;
- открытые стандарты взаимодействия, значительно уменьшающие время подключения нового сервиса к существующей системе.

Подводя итог, можно выделить три уровня рефлексии адаптивной программной системы на основе предложенной архитектуры:

- *уровень внутрикомпонентный*. Анализ поведенческой продукции отдельного компонента. Поиск зависимостей, формирование обратных связей;
- *уровень межкомпонентный*. Анализ поведенческой продукции нескольких компонентов, поиск зависимостей, оценка влияния компонентов друг на друга. Формирование обратных связей в нескольких компонентах;
- *уровень межсистемный*. Поиск зависимостей и оценка влияния компонентов одной системы на компоненты другой. Формирование межсистемных обратных связей.

3. Использование концепции инженерии линеек программных продуктов для создания адаптивных программных систем

Инженерия линеек программных продуктов (Software Product Lines Engineering, SPLE) — это концепция повторного использования компонентов программного обеспечения, позволяющая разрабатывать семейства (линейки) продуктов с сокращением времени выхода на рынок и повышением качества [18].

Центральным понятием концепции SPLE выступает понятие модели изменчивости (variability model). Модель изменчивости — это некоторое формализованное описание множества возможных конфигураций программной системы [19]. Наиболее близким по смыслу

понятием в технических науках является понятие морфологического множества [20]. По сути модель изменчивости и представляет собой морфологическое множество программной системы, дополненное некоторыми ограничениями и правилами. Данные ограничения и правила касаются вопросов совместимости отдельных компонентов потенциальной системы друг с другом, однако могут иметь и иной смысл (в зависимости от типа используемой модели изменчивости).

В SPLE используются следующие типы моделей изменчивости [21]:

- *модели характеристик* — модели, которые наиболее часто используются на практике. Графически отображаются в форме модифицированного И/ИЛИ-дерева, называемого диаграммой характеристик (рис. 1), могут иметь различное формализованное представление (гиперграф, пропозициональная формула, алгебраическая нотация и др.);
- *ортогональные модели изменчивости*. Схожи с моделями характеристик, графически также отображаются в форме диаграмм. Основное отличие заключается в том, что ортогональные модели показывают только наличие изменчивости в рассматриваемой программной системе, в то время как модели характеристик предоставляют более конкретное описание как предметной области, так и точек изменчивости;
- *модели решений*. Модель решений включает в себя следующие компоненты [22]: вопросы из предметной области, на которые нужно получить ответы в процессе разработки программного продукта; множества возможных ответов на вопросы; ссылки на используемые компоненты (активы) или другие решения; описания последствий принятия решения (ответа на определенный вопрос или выбор определенного актива).

Обобщенная архитектура адаптивной программной системы, основанной на моделях изменчивости и уровнях рефлексии, приведенных в предыдущем разделе, представлена на рис. 2. Самоадаптивная программная система охватывается тремя уровнями рефлексивной обратной связи: уровнем отдельных компонентов, уровнем групп компонентов и общесистемным уровнем. Каждый контур рефлексии обеспечивается своей моделью изменчивости (МИ), отвечающей за выбор текущих конфигураций компонентного и системного уровней. Общая координация разноуровневой совокупности МИ и потоков поведенческих данных осуществляется модулем анализа поведенче-

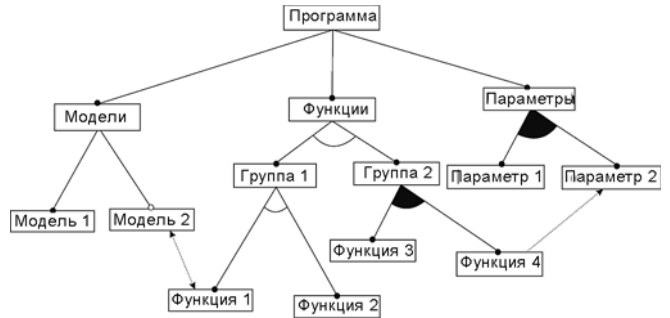


Рис. 1. Пример диаграммы характеристик

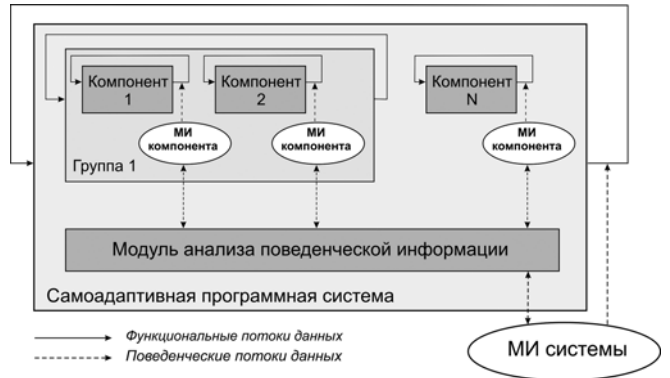


Рис. 2. Обобщенная архитектура самоадаптивной программной системы, основанной на моделях изменчивости

ской информации. Этот модуль не всегда является подсистемой системного уровня (как изображено на рисунке). Возможны и более гибкие архитектурные вариации, когда каждый отдельный компонент имеет свою собственную подсистему поведенческого анализа.

4. Пути формализации адаптивного поведения рефлексивной программной системы

Как уже было сказано выше, в рамках концепции SPLE модели изменчивости представляют собой аналог моделей морфологических множеств. Процесс создания оптимальной конфигурации программного продукта по модели изменчивости схож с процессом структурно-параметрического синтеза технических систем по морфологическому множеству, но гораздо менее формализован.

Концепция DSPLE [23] развивает идею SPLE: модели изменчивости динамических линеек программных продуктов (DSPL) используются для формирования оптимальной конфигурации программной системы в процессе ее выполнения. DSPL-системы можно считать адаптивными.

С нашей точки зрения, одним из перспективных подходов к реализации механизма программной рефлексии является интеграция

принципов DSPLE и технологии Data Mining [24]. При этом первым шагом к созданию универсальной технологии синтеза адаптивных систем должна стать разработка математической модели изменчивости, задающей формальные правила структурных и функциональных изменений для широкого класса ПО. Для этих целей предлагается использовать хорошо известную модель характеристик (feature model) — она обладает хорошей визуальной наглядностью диаграммы (по этой причине в научном обиходе часто используются другие термины — диаграмма характеристик, модель характеристик, feature diagram) и легко формализуема [25].

В таблице представлены основные типы отношений в моделях характеристик. Каждая из характеристик служит абстракцией определенного программного компонента (модуля, класса, 3D-модели, мультимедийного ресурса и т. д.). Отношения между характеристиками определяют потенциальный набор конфигураций системы в целом. Кроме того, с помощью отношений будет задаваться степень совместимости различных компонентов в рамках одного программного продукта.

Отношения в модели характеристик

Обозначение	Наименование	Описание
	Отношение обязательной дочерней характеристики	Если родительская характеристика включена в конфигурацию диаграммы, то дочерняя характеристика тоже должна быть включена
	Отношение опциональной дочерней характеристики	Включение дочерней характеристики в конфигурацию не является обязательным
	ИЛИ	Если родительская характеристика включена в конфигурацию, то также должно быть включено определенное число дочерних
	Исключающее ИЛИ	Только одна из представленных дочерних характеристик должна присутствовать в конечной конфигурации
	Отношение включения	Включение в конфигурацию характеристики А требует включения характеристики В
	Отношение исключения	Включение в конфигурацию характеристики А требует исключения характеристики В

Для формального описания зададим модель характеристик в виде ориентированного прямого гиперграфа (или F -графа):

$$F = (N, E, \Delta, \Psi), \quad (1)$$

где $N = \{N_1, N_2, \dots, N_n\}$ — конечное множество вершин графа, каждая из которых представляет собой характеристику диаграммы; $E = \{E_1, E_2, \dots, E_m\}$ — конечное множество гиперребер (отношений модели характеристик); $\Delta \in N$ — вершина, представляющая корневую характеристику диаграммы; $\Psi: E \rightarrow M, M \subset N \times N$ — функция маркировки, которая присваивает значение мощности $mv(E_i) = (\min, \max) \in M$, где M — конечное множество значений мощности модели характеристик; N — конечное множество вершин гиперграфа, каждому ориентированному гиперребру E_i , так что $\min, \max \in Z \wedge \min \geq 0, \max > 0, \min \leq \max \wedge \max \leq q_i$, где Z — множество целых чисел.

Значение мощности — это пара из минимального и максимального числа вершин головного множества гиперребра, которые могут быть включены в конфигурацию диаграммы характеристик, которой соответствует гиперграф.

С помощью отношений опциональной дочерней характеристики, исключаящего ИЛИ и множественного ИЛИ реализуются блоки изменчивости — структурные единицы программной системы, затрагиваемые адаптивными процессами. Блок изменчивости может использоваться для определения множества состояний отдельного компонента программной системы, а также правила выбора того или иного состояния в каждом конкретном случае. Таким образом, можно описать, к примеру, ситуацию выбора оптимального алгоритма генерации изображения в зависимости от конфигурации аппаратной платформы или же настроить параметры какого-либо сервиса либо совокупности сервисов (системы).

В соответствии с принципом масштабируемости блок изменчивости может задавать стратегии адаптивного поведения отдельного компонента сервиса, всего сервиса или даже всей системы в целом. Отсюда следует, что блок изменчивости не атомарен, он может включать в себя другие блоки, может иметь иерархическую структуру.

Блок изменчивости описывается диаграммой характеристик:

$$VB = (SubF, BlockParams, BlockStates, Rule), \quad (2)$$

где $SubF$ — гиперграфовое (формула (1)) представление модели характеристик блока измен-

чивости; $BlockParams = \{BlockParam_1, BlockParam_2, \dots, BlockParam_k\}$ — множество параметров, влияющих на состояние блока; $BlockStates = \{BlockState_1, BlockState_2, \dots, BlockStates_z\}$ — множество состояний блока, при этом $BlockState_i \subseteq \subseteq SubF \forall i = 1, \dots, z$; $Rule: BlockParams' \rightarrow \rightarrow BlockStates$ — функция, ставящая в соответствие каждому элементу множества $BlockParams'$ некоторый элемент множества $BlockStates$. Множество $BlockParams'$ представляет собой множество подмножеств $BlockParams$. Корневая вершина такой диаграммы называется точкой вариации.

Множество влияющих параметров $BlockParams$ можно представить следующим образом:

$$BlockParams = ExtParams \cup InnerParams \cup TargetParams, \quad (3)$$

где $ExtParams = \{ExtParam_1, ExtParam_2, \dots, ExtParam_{k1}\}$ — множество внешних по отношению к рассматриваемому блоку параметров;

$InnerParams = \{InnerParam_1, InnerParam_2, \dots, InnerParam_{k2}\}$ — множество внутренних, варьируемых параметров блока;

$TargetParams = \{TargetParam_1, TargetParam_2, \dots, TargetParam_{k3}\}$ — множество целевых параметров. Элементы множества $TargetParam$, как правило, связаны с некоторыми целевыми функциями и определяются (в отличие от других подмножеств множества $Param$) экспертным путем.

Рассмотрим простейший пример. Пусть имеется некоторое мобильное приложение, которое должно полноценно работать в автономном режиме (без подключения к сети Интернет). Основное назначение приложения — проведение теста на оценку пространственного мышления пользователя. Для этого приложение использует несколько алгоритмов генерации изображений: алгоритм на основе глубоких нейросетей, генетический алгоритм и алгоритм, основанный на случайном переборе и комбинации фрагментов изображения. Выбор алгоритма должен быть определен исходя из аппаратной конфигурации устройства. При этом предполагается, что в основном программном коде не заложено никаких сведений о конкретных конфигурациях мобильных устройств и не заданы условные операторы выбора. После инсталляции и запуска приложение должно самостоятельно установить нужные зависимости и провести выбор алгоритма.

Диаграмма характеристик блока изменчивости, отвечающего за выбор необходимого алгоритма, изображена на рис. 3. Вершина с именем "Algorithms" является точкой вариации, верши-

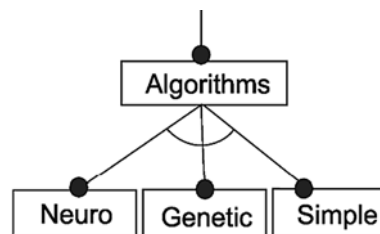


Рис. 3. Диаграмма характеристик блока изменчивости, отвечающего за выбор алгоритма генерации изображений

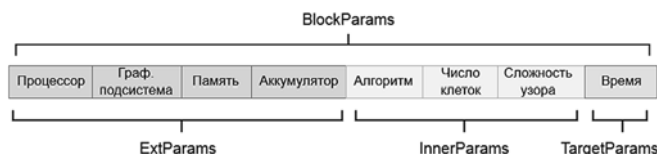


Рис. 4. Параметры блока изменчивости

ны "Neuro", "Genetic" и "Simple" представляют собой соответствующие алгоритмы.

На рис. 4 представлены параметры блока изменчивости $BlockParams$. Как видно, к внешним параметрам $ExtParams$ относятся основные характеристики мобильного устройства. Следует отметить, что не все из перечисленных характеристик будут оказывать влияние на производительность устройства при выборе определенного алгоритма. Выявление того, является ли определенная характеристика влияющей на производительность, также является одной из задач рефлексии. К внутренним варьируемым параметрам ($InnerParams$) относится как сам алгоритм (нейросетевой, генетический или простой), так и его параметры: число клеток (фрагментов) изображения и его сложность. Целевые параметры ($TargetParams$) представляет время визуализации изображения. Оно должно быть минимально возможным.

Множество $BlockStates$ в данном случае будет содержать в себе различные конфигурации диаграммы, представленной на рис. 3, т.е. пары из корневой характеристики ("Algorithms") и одного из возможных алгоритмов ("Neuro", "Genetic", "Simple").

Задача процедуры рефлексивной адаптации заключается в формировании функции $Rule$, устанавливающей соответствие между группой влияющих параметров блока изменчивости и элементами множества $BlockStates$. Методы, позволяющие достичь данной цели, будут рассмотрены в одной из ближайших статей.

Заключение

Проводимое исследование имеет ярко выраженный междисциплинарный характер. Про-

водятся аналогии между прикладными программными системами и сложноорганизованными живыми организмами с точки зрения их возможной самоадаптации к изменяющимся внешним условиям. Принципы работы аппарата самоадаптации ПО могут быть заимствованы от механизмов адаптации высших биологических существ, связанных с процессами самосознания и психики: рефлексии, планирования поведения в реальном времени и саморазвития на основе наблюдения за внешним миром. Наделение прикладной программной системы подобными адаптивными свойствами позволит ей осуществлять модификацию своей структуры на качественно ином уровне, что, в свою очередь, повысит ее надежность, отказоустойчивость и гибкость, снизит стоимость сопровождения и продлит срок эксплуатации. Такая система будет способна расширять класс решаемых задач на протяжении всего жизненного цикла, а также возьмет на себя выполнение тех операций, которые в настоящее время считаются частью обязанностей определенных специалистов (например, сможет выполнять функции самоадминистрирования).

Список литературы

1. **Carvalho M. B.** et al. The journey: a service-based adaptive serious game on probability // *Serious games analytics* / edited by C. S. Loh, Y. Sheng, D. Ifenthaler. Cham: Springer International Publishing, 2015. P. 97–106.
2. **Bernon C.** et al. Engineering self-organizing systems // *Self-organizing software: from natural to artificial adaptation* / edited by G. Di Marzo Serugendo, M. P. Gleizes, A. Karageorgos. Heidelberg: Springer Berlin Heidelberg, 2011. P. 283–312.
3. **Wagnier G., Le Meur A. F., Duchien L.** A model-based framework to design and debug safe component-based autonomic systems // *Architecture for adaptive software systems* / edited by R. Mirandola, I. Gorton, C. Hofmeister. Heidelberg: Springer Berlin Heidelberg, 2009. P. 1–17.
4. **Torbeyns J., Lehitinen E., Elen J.** Describing and studying domain-specific serious games. Cham: Springer International Publishing, 2015. 250 p.
5. **Colledanchise M., Ogren P.** How behavior trees modularize robustness and safety in hybrid systems // 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. Chicago. 2014. P. 53–62.
6. **Berinstein P.** et al. Game development tool essentials. Heidelberg: Springer Berlin Heidelberg, 2014. 467 p.
7. **Kenett R. S.** Future directions of software cybernetics: A position paper // 35th IEEE Annual Computer Software and Applications Conference Workshops. Washington: IEEE Computer Society, 2003. P. 43–44.
8. **Cai K. Y., Chen T. Y., Tse T. H.** Towards research on software cybernetics // 7th IEEE International Symposium on High Assurance Systems Engineering (HASE'02). Washington: IEEE Computer Society, 2002. P. 240.
9. **Ravindran K., Rabby M.** Software cybernetics to infuse adaptation intelligence in networked systems // IEEE International

Conference on the Network of the Future (NOF). Washington: IEEE Computer Society, 2013. P. 1–6.

10. **Park J. S.** Essence-based, goal-driven adaptive software engineering // *EEE/ACM 4th SEMAT Workshop on General Theory of Software Engineering (GTSE)*. Washington: IEEE Computer Society, 2015. P. 33–38.
11. **Liu C., Jiang C., Hu H., Cai K. Y., Huang D., Yau S. S.** Control-based approach to balance services performance and security for adaptive service based systems (ASBS) // 33rd Annual IEEE International Computer Software and Applications Conference (COMPSAC'09). Washington: IEEE Computer Society, 2009. P. 473–478.
12. **Ravindran K., Rabby M.** Software cybernetics to infuse adaptation intelligence in networked systems // Fourth IEEE International Conference on the Network of the Future (NOF). Washington: IEEE Computer Society, 2013. P. 1–6.
13. **Choi, T., Chan, H., Yue, X.** Recent development in big data analytics for business operations and risk management // *IEEE Transactions on Cybernetics*. Washington: IEEE Computer Society, 2016. P. 1–12.
14. **Mayer P., Klarl A., Hennicker R.** The autonomic cloud: a vision of voluntary, peer-2-peer cloud computing // 7th IEEE International Conference on Self-Adaptation and Self-Organizing Systems Workshops (SASOW). Washington: IEEE Computer Society, 2013. P. 89–94.
15. **Гиппенрейтер Ю. Б.** Введение в общую психологию: Курс лекций. М.: Юрайт, 2002. С. 89.
16. **Столярченко Л. Д.** Психология: учебник для вузов. СПб.: Питер, 2016. С. 124.
17. **Liu C., Jiang C., Hu H., Cai K. Y., Huang D., Yau S. S.** Control-based approach to balance services performance and security for adaptive service based systems (ASBS) // 33rd Annual IEEE International Computer Software and Applications Conference (COMPSAC'09). Washington: IEEE Computer Society, 2009. P. 473–478.
18. **Schobbens P. E., Heymans P., Trigaux J. C.** Feature diagrams: a survey and formal semantics // In 14th IEEE International Requirements Engineering Conference (RE'06). Washington: IEEE Computer Society, 2011. P. 139–148.
19. **Kang K. C.** et al. Feature-oriented domain analysis (FODA): feasibility study. Pittsburgh: Software Engineering Institute, 1990. 161 p.
20. **Sinnema M., Deelstra S.** Classifying variability modeling techniques // *Information and software technology*. 2007. N. 7. P. 42–54.
21. **Liaskos S., Dinkelaker T.** et al. On goal-based variability acquisition and analysis // In 14th IEEE International Requirements Engineering Conference (RE'06). Washington: IEEE Computer Society, 2010. P. 77–85.
22. **Berger T.** Variability modeling in the real: an empirical journey from software product lines to software ecosystems. Leipzig: University of Leipzig, 2012. 225 p.
23. **Younis O., Ghouli S., Alomari M. H.** Systems variability modeling: a textual model mixing class and feature concepts // *International Journal of Computer Science & Information Technology*. 2013. N. 5. P. 127–139.
24. **Webber D. L., Gomaa H.** Modeling variability in software product lines with the variation point model // *Science of Computer Programming*. 2004. N. 3. P. 305–331.
25. **Losif-Lazar A. F., Schaef I., Wasowski A.** A Core Language for Separate Variability Modeling // *Leveraging Applications of Formal Methods, Verification and Validation. Technologies for Mastering Change* / edited by T. Margaria, B. Steffen. Heidelberg: Springer Berlin Heidelberg, 2014. P. 257–272.

A. M. Bershinsky, Dr. of Tech. Sciences, prof., e-mail: bam@pnzgu.ru,
A. S. Bozhday, Dr. Tech. Sciences, prof., e-mail: bozhday@yandex.ru,
Yu. I. Evseeva, Cand. of Tech. Sciences, Assoc. prof., e-mail: shymoda@mail.ru,
A. A. Gudkov, Cand. of Tech. Sciences, Assoc. prof., e-mail: alexei-ag@yandex.ru,
Penza State University

The Conception of Reflexive Self-Adaptation of Applied Software Systems

The article suggests a new approach to solving the important problem of the modern IT industry — self-adaptation of software components based on the concept of behavioral reflection. The basic principles of reflexive adaptation are considered, as well as the idea of using the engineering of software product lines to create adaptive programs. A generalized architectural solution for constructing reflexive self-adaptive systems capable of changing their behavioral characteristics directly during execution without recompiling the source code is given. Special attention is paid to the issues of formalized presentation and use of variability models for implementing adaptive behavior of application programs. The study was carried out with the financial support of the Russian Foundation for Basic Research in the framework of the scientific project No. 18-07-00408.

Keywords: adaptive software systems, reflexive self-adaptation, variability model, characteristics diagram, software product lines, software engineering

DOI: 10.17587/it.25.11-19

References

1. **Carvalho M. B.** et al. The journey: a service-based adaptive serious game on probability, *Serious games analytics*, edited by C. S. Loh, Y. Sheng, D. Ifenthaler, Cham, Springer International Publishing, 2015, pp. 97–106.
2. **Bernon C.** et al. Engineering self-organizing systems, *Self-organizing software: from natural to artificial adaptation*, edited by G. Di Marzo Serugendo, M. P. Gleizes, A. Karageorgos, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 283–312.
3. **Wagnier G., Le Meur A. F., Duchien L.** A model-based framework to design and debug safe component-based autonomous systems, *Architecture for adaptive software systems*, edited by R. Mirandola, I. Gorton, C. Hofmeister, Heidelberg, Springer Berlin Heidelberg, 2009, pp. 1–17.
4. **Torbeyns J., Lehtinen E., Elen J.** Describing and studying domain-specific serious games, Cham, Springer International Publishing, 2015, 250 p.
5. **Colledanchise M., Ogren P.** How behavior trees modularize robustness and safety in hybrid systems, *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Chicago, IEEE, 2014, pp. 53–62.
6. **Berinstein P.** et al. Game development tool essentials, Heidelberg, Springer Berlin Heidelberg, 2014, 467 p.
7. **Kenett R. S.** Future directions of software cybernetics: A position paper, *35th IEEE Annual Computer Software and Applications Conference Workshops*, Washington, IEEE Computer Society, 2003, pp. 43–44.
8. **Cai K. Y., Chen T. Y., Tse T. H.** Towards research on software cybernetics, *7th IEEE International Symposium on High Assurance Systems Engineering (HASE'02)*, Washington, IEEE Computer Society, 2002, pp. 240.
9. **Ravindran K., Rabby M.** Software cybernetics to infuse adaptation intelligence in networked systems, *IEEE International Conference on the Network of the Future (NOF)*, Washington, IEEE Computer Society, 2013, pp. 1–6.
10. **Park J. S.** Essence-based, goal-driven adaptive software engineering, *EEE/ACM 4th SEMAT Workshop on General Theory of Software Engineering (GTSE)*, Washington, IEEE Computer Society, 2015, pp. 33–38.
11. **Liu C., Jiang C., Hu H., Cai K. Y., Huang D., Yau S. S.** Control-based approach to balance services performance and security for adaptive service based systems (ASBS), *33rd Annual IEEE International Computer Software and Applications Conference (COMPSAC'09)*, Washington, IEEE Computer Society, 2009, pp. 473–478.
12. **Ravindran K., Rabby M.** Software cybernetics to infuse adaptation intelligence in networked systems, *Fourth IEEE International Conference on the Network of the Future (NOF)*, Washington, IEEE Computer Society, 2013, pp. 1–6.
13. **Choi T., Chan H., Yue X.** Recent development in big data analytics for business operations and risk management, *IEEE Transactions on Cybernetics*, Washington, IEEE Computer Society, 2016, pp. 1–12.
14. **Mayer P., Klarl A., Hennicker R.** The autonomic cloud: a vision of voluntary, peer-2-peer cloud computing, *7th IEEE International Conference on Self-Adaptation and Self-Organizing Systems Workshops (SASOW)*, Washington, IEEE Computer Society, 2013, pp. 89–94.
15. **Gippenreiter Yu. B.** Introduction to general psychology. Lecture course, Moscow, Yurayt, 2002, pp. 89 (in Russian).
16. **Stolyarenko L. D.** Psychology: a textbook for high schools, St. Petersburg, Peter, 2016, pp. 124.
17. **Liu C., Jiang C., Hu H., Cai K. Y., Huang D., Yau S. S.** Control-based approach to balance services performance and security for adaptive service based systems (ASBS), *33rd Annual IEEE International Computer Software and Applications Conference (COMPSAC'09)*, Washington, IEEE Computer Society, 2009, pp. 473–478.
18. **Schobbens P. E., Heymans P., Trigaux J. C.** Feature diagrams: a survey and formal semantics, *In 14th IEEE International Requirements Engineering Conference (RE'06)*, Washington, IEEE Computer Society, 2011, pp. 139–148.
19. **Kang K. C.** et al. Feature-oriented domain analysis (FODA): feasibility study, Pittsburgh, Software Engineering Institute, 1990, 161 p.
20. **Sinnema M., Deelstra S.** Classifying variability modeling techniques, *Information and software technology*, 2007, no. 7, pp. 42–54.
21. **Liaskos S., Dinkelaker T.** et al. On goal-based variability acquisition and analysis, *In 14th IEEE International Requirements Engineering Conference (RE'06)*, Washington, IEEE Computer Society, 2010, pp. 77–85.
22. **Berger T.** Variability modeling in the real: an empirical journey from software product lines to software ecosystems, Leipzig, University of Leipzig, 2012, 225 p.
23. **Younis O., Ghouli S., Alomari M. H.** Systems variability modeling: a textual model mixing class and feature concepts, *International Journal of Computer Science & Information Technology*, 2013, no. 5, pp. 127–139.
24. **Webber D. L., Gomaa H.** Modeling variability in software product lines with the variation point model, *Science of Computer Programming*, 2004, no. 3, pp. 305–331.
25. **Losif-Lazar A. F., Schaefer I., Wasowski A.** A Core Language for Separate Variability Modeling, Leveraging Applications of Formal Methods, Verification and Validation. Technologies for Mastering Change, edited by T. Margaria, B. Steffen, Heidelberg, Springer Berlin Heidelberg, 2014, no. 257–272.

М. В. Сарамуд, инженер, e-mail: msaramud@gmail.com,
И. В. Ковалев, д-р техн. наук, проф., e-mail: kovalev.fsu@mail.ru,
В. В. Лосев, канд. техн. наук, доц., e-mail: basilos@mail.ru,
М. О. Петросян, аспирант, e-mail: mopetrosyan@gmail.com,
А. О. Калинин, аспирант, e-mail: andrey.kalinin@phkp.ru,

Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева

Сравнение отказоустойчивых моделей программного обеспечения в имитационной среде исполнения

Рассматривается методика, позволяющая сравнить основные отказоустойчивые модели программного обеспечения с программной избыточностью на основе методологии: N-версионного программирования (с четкими и нечеткими алгоритмами голосования), N-версионного программирования с самопроверкой, восстанавливаемых блоков, согласованных восстанавливаемых блоков и $t/(n-1)$ -версионного программирования. Для реализации методики создана имитационная среда исполнения отказоустойчивых программных моделей. Проанализированы результаты моделирования в имитационной среде.

Ключевые слова: мультиверсионное программирование, программная избыточность, надежность, блоки восстановления, среда исполнения

Введение

В последние годы активно развиваются отрасли, требующие надежных, отказоустойчивых систем управления. К их числу относятся высокотехнологичные производства [1], использующие композитные и опасные материалы, автономные беспилотные объекты — от мультироторных систем до автомобилей с функцией автопилота и моторизованных кресел с голосовым управлением для людей с ограниченными возможностями. Все более актуальным становится вопрос научно-обоснованной разработки отказоустойчивых систем управления. Поскольку целевыми устройствами нередко становятся коммерческие устройства массовой доступности, актуальным также является вопрос минимизации временных и материальных затрат на создание программного обеспечения (ПО) для них. Одним из наиболее эффективных подходов к повышению надежности ПО является введение избыточности. Известны следующие отказоустойчивые модели ПО на основе методологии: N-версионного программирования (NVP); N-версионного программирования с самопроверкой (NSCP); восстанавливаемых блоков (RB); согласованных восстанавливаемых блоков (CRB); $t/(n-1)$ -версионного программирования [2]. Каждая из упомянутых моделей обладает своими преимуществами и недостатками. Выбор одной из них не является очевидным. Необходимо также определить число программных версий и требования по надежности как к этим версиям, так и к блоку

принятия решения, к аппаратному обеспечению [3]. Для решения задачи сравнения и выбора той или иной отказоустойчивой модели ПО предложена имитационная среда исполнения мультиверсионного ПО, представленная в настоящей статье.

Методика сравнения отказоустойчивых моделей ПО

Для большинства прикладных систем управления возможна реализация всех основных моделей повышения отказоустойчивости путем введения программной избыточности [4]. Возникает задача выбора соответствующей модели либо по соотношению характеристик и затрат на разработку, либо по целевой характеристике, если не все модели позволяют достичь требуемого значения целевой характеристики. В любом случае необходим инструментарий, позволяющий оценить характеристики надежности ПО или его компонента при реализации всех основных отказоустойчивых моделей при заданных характеристиках программных модулей и блока принятия решения.

Предлагаемая методика позволяет оценивать все модели на одинаковом наборе ответов версий программ на каждой итерации, что дает возможность сравнить результаты работы всех моделей в одинаковых условиях, соответствующих заданным характеристикам системы. Это обстоятельство позволяет определить, какая из моделей будет лучше работать имен-

но в представленных условиях, а именно: при определенной надежности версий на каждом из изменяющихся потоков данных; с учетом того, с какой вероятностью будут возникать межверсионные ошибки или ошибки округления и с какой вероятностью будут проявляться сбои в приемочных тестах и т. д.

Модель $t/(n - 1)$ -версионного программирования

Отдельно следует остановиться на модели $t/(n - 1)$ версионного программирования, поскольку алгоритм ее работы не описан в русскоязычной литературе, а сама модель представляет интерес. Процедура принятия решения в данной модели мультиверсионного ПО, предложенная Цзе Сюй (Jie Xu) из университета Ньюкастла (University of Newcastle upon Tyne), основана на $t/(n - 1)$ диагностируемости [5]. Для простоты будем называть его $t/(n - 1)$ -алгоритмом принятия решений. Суть алгоритма состоит в том, что сравниваются не все выходы версий, а лишь некоторые из них, достаточные для принятия решения.

Рассмотрим пример системы с числом версий $N = 5$ и максимальным числом ошибок $t = 2$, т. е. рассмотрим вариант $2/(5 - 1)$ -алгоритма. При числе ошибок, не превышающем t , алгоритм гарантирует выбор правильного варианта из N выходов версий. Однако и при превышении числа неправильных выходов версий система не обязательно выберет неверный суммарный выход. Может произойти и выбор правильного выхода, однако такой результат не гарантируется [5].

Рассмотрим алгоритм подробнее на данном примере. Сравниваются попарно выходы четырех версий: 1 — 2; 2 — 3; 3 — 4. Получаем три результата сравнений ω_{12} , ω_{23} , ω_{34} , равные 0, если выходы совпадают, и равные 1, если различаются. На основании только этих трех результатов сравнения алгоритм принимает решение о переключении выхода между выходами 1-й, 4-й и 5-й версий, т. е. версии 2 и 3 используются только для сравнения, значения их выходов никогда не используется в качестве выхода системы. Более наглядно схема работы представлена на рис. 1.

Как видно из рис. 1, схема принятия решения в $t/(n - 1)$ -алгоритме относительно несложна. В случае пяти мультиверсий для принятия решения (правильного управления переключателем выходов) необходимы только результаты трех парных сравнений выходов четырех версий, значение выхода пятой версии для принятия ре-

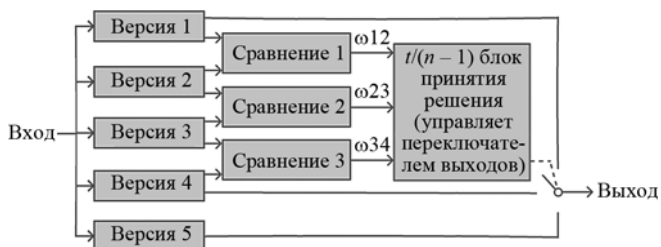


Рис. 1. Схема работы $t/(n - 1)$ алгоритма при $n = 5$ и $t = 2$

шения не используется. Логика управления переключателем выходов на основании результатов сравнений при $n = 5$ представлена в табл. 1.

Изучив табл. 1 и рис. 1, можно прийти к выводу, что при относительно надежных версиях в большей части случаев компараторы будут возвращать $(0;0;0)$, и на выход будет подаваться значение выполнения первой версии. Можно также сделать вывод об отсутствии необходимости каждый раз исполнять пятую версию, а делать это только в случае соответствующих значений результатов сравнений, когда на выход необходимо подать именно результат пятой версии $((0;1;0), (1;0;1), (1;1;1))$. Этот факт снижает среднюю нагрузку, требуемую среде исполнения мультиверсионного ПО для работы (в подавляющем большинстве случаев будут рассчитаны четыре из пяти версий).

Следует отметить, что и сам алгоритм принятия решения также является в значительной мере менее ресурсоемким по сравнению с голосованием, особенно со взвешенными его модификациями, где при каждом голосовании исполняются все версии, создаются классы и рассчитываются веса для каждого из них. Для $t/(n - 1)$ при $n = 5$ необходимо только три простые операции сравнения с бинарным выходом. Далее осуществляется однозначный, априорно заданный выбор выхода для одного из восьми возможных сочетаний значений выходов компараторов (табл. 1).

Таблица 1

Возможные варианты выбора на основе выходов компараторов для $n = 5$

ω_{12}	ω_{23}	ω_{34}	Предположительно правильные версии
0	0	0	1, 2, 3, 4
0	0	1	1, 2, 3
0	1	0	5
0	1	1	1, 2
1	0	0	2, 3, 4
1	0	1	5
1	1	0	3, 4
1	1	1	5

Программная реализация имитационной среды исполнения

Среда исполнения мультиверсионного ПО реализована в виде программного комплекса, который позволяет исполнять программные модули не только с помощью методологии N -версионного программирования (NVP), а также с использованием остальных пространственных мультиверсионных моделей, к числу которых относятся: модель восстанавливающихся блоков (RB), модель согласованных восстанавливающихся блоков (CRB); модель $t/(n - 1)$ -версионного программирования и модель мультиверсионного программирования с самопроверкой (NSCP).

В данной среде в качестве программных модулей выступают симуляции версий, которые работают в соответствии со следующими заданными в интерактивной форме характеристиками:

1) общая надежность версии в каждом потоке данных. В системе реализовано три последовательно изменяющих потока данных для исследования реакции системы на изменение работы программных модулей при различных входных данных;

2) вероятности возникновения межверсионной ошибки, "неточности". В частности, параметр допуска E для нечетких алгоритмов голосования используется симуляциями как значение разброса "неточностей".

В рассматриваемой среде исполнения симуляции версий дают следующие три типа ошибок: случайную ошибку, симулирующую сбой в модуле, межверсионную ошибку и неточность — ответ, близкий к правильному, т. е. удаленный от него на величину, не превышающую допуск, но не равный ему. Этот тип ошибки симулирует "неточность" — ошибки округления при нехватке разрядности, неточности оцифровки выходов аналоговых датчиков и т. д. Это ситуация, когда алгоритмически версия сработала верно, но дала неточный ответ из-за ошибок округления, оцифровки, нехватки разрядности, большой разницы в порядке величин при операциях с плавающей запятой [6].

Так как в среде исполнения ошибка проявляется с заданной для каждой версии и каждого потока данных вероятностью, то при возникновении ошибки выполняется ряд проверок. Если ошибка является не первой в текущем голосовании, то с заданной вероятностью генерируется межверсионная ошибка, т. е. возвращается значение, совпадающее со значением предыдущей ошибки. Этот тип ошибки симулирует связную ошибку — допущенный алгоритмический просчет, одинаковый в не-

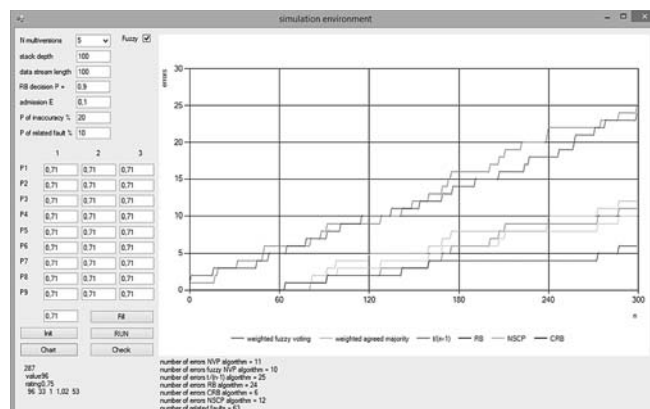


Рис. 2. Интерфейс программы с результатами моделирования: число ошибок в каждой модели и график их распределения по времени

скольких версиях, которые дадут одинаковую ошибку при одинаковом входе [7].

"Неточность" также генерируется с заданной вероятностью. При этом возвращается выход, не равный правильному, но удаленный от него не более чем на заданное отклонение E . Если предыдущие вероятности не срабатывают, то возвращается случайная ошибка, симулирующая сбой в текущей версии модуля.

На рис. 2 представлен результат моделирования на основе 300 итераций мультиверсионного голосования с заданными в программе характеристиками модулей для варианта исполнения с симуляциями версий. Пользователем задается: число версий, глубина стека для взвешенных алгоритмов с забыванием, длина потока данных, вероятность ошибки приемочного теста алгоритмов с восстанавливающимися блоками, допуск вхождения для нечетких алгоритмов, вероятности возникновения неточностей и межверсионных ошибок, надежности каждой версии в каждом из трех потоков данных.

В предложенной программной реализации имеется возможность задавать значения надежности для всех версий, проводить инициализацию значений параметров системы, осуществлять моделирование с заданными параметрами, формирование графиков и проверку. Интерфейс программы имеет следующие области: графическая область, область вывода информации о последней возникшей ошибке и результатов моделирования, а именно общего числа накопленных ошибок за все итерации и числа межверсионных ошибок. Графики отображают распределение возникновения ошибок по каждой модели во времени.

Для практической проверки работоспособности предложенного инструментария в имитационной среде разработаны и используются в качестве версий пять различных алгоритмов

Результат моделирования при различной надежности версий

Число ошибок	Надежность версий						
	0,7	0,75	0,8	0,85	0,9	0,95	0,99
Число ошибок в NVP с четким алгоритмом	10	6	3	1	0	0	0
Число ошибок в NVP с нечетким алгоритмом	12	9	4	1	0	1	0
Число ошибок в $i/(n-1)$ -версионном программировании	28	14	7	3	1	0	0
Число ошибок в восстанавливаемых блоках (RB)	23	15	20	18	12	6	1
Число ошибок в согласованных RB	6	6	3	2	0	0	0
Число ошибок в NVP с самопроверкой	13	9	2	6	1	1	0
Число межверсионных ошибок	57	42	32	19	14	2	0

оптимизации: 1 — метод деления отрезка пополам; 2 — метод золотого сечения; 3 — метод дихотомии; 4 — метод квадратичной аппроксимации и 5 — метод Фибоначчи. В каждом алгоритме наложено ограничение на число итераций $K = 10$, поскольку в системах реального времени важно гарантированно получить ответ за заданный промежуток времени [8], т. е. нельзя допускать слишком длительного исполнения циклов. Это позволит дополнительно сравнить точность алгоритмов, которую они смогут достигнуть при введенном (одинаковом для всех версий) ограничении на число итераций.

В имитационной среде обеспечена поддержка принятия решения на основе любой из рассмотренных ранее моделей, однако в качестве примера будем рассматривать выходы модели N -версионного программирования с нечетким взвешенным голосованием согласованным большинством с забыванием. Отметим, что выбор нечеткого голосования обусловлен характером выходов версий — ожидаются ответы, близкие к реальным минимумам функций, но не обязательно равные им. Целесообразно использовать алгоритмы с элементом нечеткой логики [9], чтобы близкие ответы могли увеличивать веса мультиверсий, повышая устойчивость алгоритмов и модели в целом.

Включение в состав среды алгоритма взвешенного голосования позволяет нам численно сравнить алгоритмы оптимизации, так как по результатам прогонов мы будем накапливать их веса, рассчитывая оценку вероятности правильного ответа, основанную на статистике работы каждой версии. Чем чаще версия будет ошибаться, тем меньше будет ее вес. Оптимальной в нашем случае будет версия с наибольшей суммой весов, соответственно, оптимальным будет алгоритм, который реализуется данной версией.

В имитационной среде допускается возможность задавать глубину стека для взвешенного алгоритма голосования, значение допуска для элемента нечеткой логики и общее число прогонов. С учетом заданных значений указанных параметров имитационная среда осуществляет мультиверсионное исполнение всех алгоритмов и принятие решения, причем на каждой итерации изменяются значения коэффициентов функций, используемых в оптимизационных алгоритмах.

Более того, через каждые 100 итераций изменяется тип функции (в имитационной среде реализовано пять различных функций). Это позволяет исследовать и качество работы алгоритмов оптимизации, поскольку на различных типах функций используемые алгоритмы оптимизации показывают различные результаты. Итоговая оценка работы алгоритмов представ-

ляет собой сумму результатов работы по всем пяти функциям при 100 различных коэффициентах для каждой из них, что является достаточно объективным показателем и позволяет отсеять алгоритмы, работающие хорошо только в узком диапазоне вводимых параметров.

Анализ результатов моделирования в имитационной среде

При анализе результатов моделирования рассматривается зависимость числа ошибок для каждой отказоустойчивой модели ПО от надежности версий (программных модулей).

Установим следующие значения параметров системы: число мультиверсий = 5; глубина стека = 100; длина потока данных = 100; надежность приемочного теста = 0,9; допуск для нечетких алгоритмов = 0,1; вероятность возникновения неточности = 0,2; вероятность возникновения межверсионной ошибки = 0,1. Важно отметить, что в программе задаются не абсолютные вероятности конкретного типа ошибок, а условные, т. е. вероятность возникновения события ошибки. Например, общая вероятность возникновения неточности при работе программного модуля с надежностью $P = 0,95$ будет не 0,2, а $(1 - 0,95) \cdot 0,2 = 0,01$.

Итак, зафиксировав указанные параметры, мы можем менять значения надежности программных модулей (для простоты будем использовать равные значения надежности для всех версий и для всех трех потоков данных), используя кнопку "fill" (рис. 2). При ее нажатии все значения надежности версий приравниваются к одному значению, введенному в основном

окне программы, слева от кнопки. Проведем моделирование с различными значениями надежности версий от 0,7 до 0,95 с шагом 0,05 и 0,99.

Из результатов моделирования, приведенных в табл. 2, видно, как растет число межверсионных ошибок при снижении надежности версий. Виден рост числа допущенных ошибок, отличающийся для различных отказоустойчивых моделей ПО. Большое число ошибок зарегистрировано в модели восстанавливаемых блоков, отчасти это обусловлено относительно низкой заданной надежностью приемочного теста (0,9). Наилучшие результаты при невысокой надежности версий показывает модель согласованных восстанавливаемых блоков, поскольку она сочетает в себе механизмы как восстанавливаемых блоков, так и N -версионного программирования.

На рис. 3–5 (см. третью сторону обложки) и рис. 6 приведены результаты работы имитационной среды исполнения отказоустойчивого ПО с различными алгоритмами оптимизации при различных допусках в алгоритмах голосования.

Рассмотрим результаты работы имитационной среды с алгоритмами оптимизации на рис. 3 (см. третью сторону обложки). При допуске $E = 0,045$ есть однозначно лучший алгоритм, это метод 2 (метод золотого сечения). Он всегда голосует за выигравший класс, т. е. значения функции не являются равными, но отличаются не более чем на 0,045 в любую сторону. Таким образом, не обязательно иметь максимально близкий к правильному ответ, так как в соответствии с логикой работы взвешенного алгоритма голосования I в стек весов получают все версии с коэффициентом вхождения больше нуля, вне зависимости от его конкретного значения [10].

Очевидно, что значение допуска E будет существенно влиять на работу имитационной среды исполнения отказоустойчивых моделей ПО, поскольку определяет отклонение, которое может "допустить" версия и быть признан-

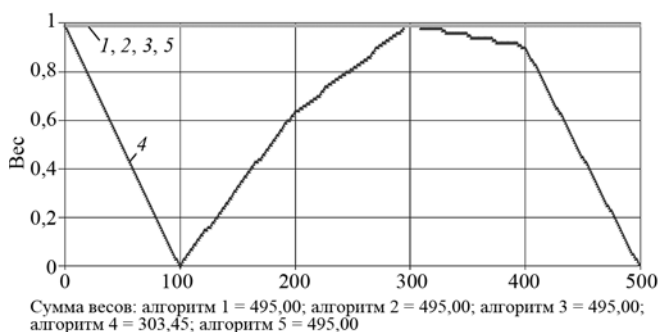


Рис. 6. Результат работы среды при $E = 10$:

1 — алгоритм 1; 2 — алгоритм 2; 3 — алгоритм 3; 4 — алгоритм 4; 5 — алгоритм 5

ной верно ответившей. Рассмотрим случаи уменьшения допуска E до 0,001 на рис. 4 (см. третью сторону обложки) и его увеличения до значений 3 и 10 на рис. 5 (см. третью сторону обложки) и рис. 6 соответственно.

Из представленных результатов видно, что при уменьшении допуска суммы весов резко падают, поскольку алгоритм начинает все "строже" принимать решение о корректности выхода версии, помещая в стек весов "0" за отклонение, превышающее 0,001. При увеличении допуска наблюдается обратная картина, суммы весов растут, достигая максимального показателя в 495 уже для нескольких версий, поскольку даже алгоритмы, дающие ответы с существенным отклонением, признаются ответившими верно. В таком случае система уже не может однозначно выбрать наилучший алгоритм и выдает список субоптимальных алгоритмов, т. е. получивших максимальные суммы весов.

Заключение

Разработанная имитационная среда исполнения отказоустойчивых моделей ПО позволяет сравнить программные реализации алгоритмов, основанные на современных методологиях повышения отказоустойчивости с введением программной избыточности, в одинаковых условиях. Задавая характеристики имитационной среды и вероятности корректной работы всех программных компонентов, мы получаем результаты моделирования, которые упрощают выбор отказоустойчивой модели для реализации разрабатываемого ПО с избыточностью.

Исполнение в имитационной среде программных реализаций оптимизационных алгоритмов позволяет выбрать лучший алгоритм на заданном наборе функций с учетом допустимого отклонения, что показывает работоспособность предложенного инструментария на реальной прикладной задаче, когда заведомо правильный выход не известен, так как среда работает в реальном ПО, а не с симуляциями программных версий.

Список литературы

1. Лосев В. В., Ковалев И. В. Реинжиниринг информационного обеспечения интегрированных систем управления производством // Приборы. 2010. № 3 (117). С. 31–36.
2. Jie Xu, Randell B. Software fault tolerance: $t/(n-1)$ -variant programming, Software fault tolerance: $t/(n-1)$ -variant programming // IEEE Transactions on Reliability. 1997. Vol. 46, Iss. 1. P. 60–68.
3. Ковалев И. В., Котенок А. В. К проблеме выбора алгоритма принятия решения в мультиверсионных системах // Информационные технологии. 2006. № 9. С. 39–44.

4. Ковалев И. В., Лосев В. В., Сарамуд М. В., Ковалев Д. И., Петросян М. О. К вопросу реализации мультиверсионной среды исполнения бортового программного обеспечения автономных беспилотных объектов средствами операционной системы реального времени // Вестник СибГАУ. Красноярск, 2017. Т. 18, № 1. С. 58—61.

5. Xu J. The $t(n-1)$ -diagnosability and its applications to fault tolerance // Digest of Papers. Fault-Tolerant Computing: The Twenty-First International Symposium, 1991. P. 496—503.

6. Kovalev I., Voroshilova A., Losev V., Saramud M., Chuvashova M., Medvedev A. Comparative Tests of Decision Making Algorithms for a Multiversion Execution Environment of the Fault Tolerance Software // Proc. of 2017 European Conference on Electrical Engineering and Computer Science (EECS 2017). 2017.

7. Kovalev I., Losev V., Saramud M., Petrosyan M. Model implementation of the simulation environment of voting algo-

rithms, as a dynamic system for increasing the reliability of the control complex of autonomous unmanned objects // MATEC Web of Conferences 132, 04011 (2017).

8. Стельмах В. О., Ковалев И. В. Построение отказоустойчивых систем управления на основе мультиверсионного подхода // Информационно-телекоммуникационные системы и технологии (ИТСИТ-2012): Материалы Всерос. молодеж. конф. 2012. С. 172—173.

9. Ковалев И. В. и др. Оценка надежности АСУ с блокирующими модулями защиты // Приборы. 2013. № 6. С. 20—23.

10. Kovalev I. V., Zelenkov P. V., Losev V. V., Kovalev D. I., Ivleva N. V., Saramud M. V. Multiversion environment creation for control algorithm implementation by autonomous unpiloted objects // IOP Conf. Series: Materials Science and Engineering 173 (2017) 012025. doi:10.1088/1757-899X/173/1/012025.

M. V. Saramud, Engineer, e-mail: msaramud@gmail.com, I. V. Kovalev, Professor, e-mail: kovalev.fsu@mail.ru,

V. V. Losev, Assistant Professor, e-mail: basilos@mail.ru,

M. O. Petrosyan, Postgraduate Student, e-mail: mopetrosyan@gmail.com,

A. O. Kalinin, Postgraduate Student, e-mail: andrey.kalinin@phkp.ru,

Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, 660014, Russian Federation

Comparison of Methodologies for Increasing Software Fault Tolerance in a Simulation Execution Environment

In the article considered technique which makes it possible to compare the main methodologies for increasing fault tolerance with the introduction of software redundancy: N -version programming (with majority and fuzzy voting algorithms), N self-checking programming, recovery blocks, consensus recovery blocks, and $t/(n-1)$ -version programming. The software implementation of this technique allowed to compare them under the same conditions, with the specified system characteristics and the probabilities of correct operation of all components, to obtain system characteristics using these methodologies based on the simulation results. The results of modeling in the proposed environment are analyzed. A non-trivial methodology of $t/(n-1)$ -Variant Programming based on $t/(n-1)$ -diagnosability is considered in detail. The results of it's work are compared with classical methodologies. This simplifies the choice of methodology for implementing the software under development. A software implementation with real algorithms allows you to select the best optimization algorithm for a given set of functions and the amount of allowable deviation. Also it shows the working capacity of the proposed toolkit on a real applied task, where the system no longer knows the correct output, but works in real conditions, rather than with simulation versions.

Keywords: multi-version programming, software redundancy, reliability, recovery blocks, execution environment, related software fault, $t/(n-1)$ -diagnosability, consensus recovery blocks, optimization, simulation

DOI: 10.17587/it.25.20-25

References

1. Losev V. V., Kovalev I. V. Reengineering information support of integrated systems of production management, *The Devices*, 2010, no. 3 (117), pp. 31—36 (in Russian).

2. Jie Xu, Randell B. Software fault tolerance: $t/(n-1)$ -variant programming, *IEEE Transactions on Reliability*, 1997, vol. 46, iss. 1, pp. 60—68.

3. Kovalev I. V., Kotenok A. V. Choice Problems the Decision Algorithm for Multi-Version Systems, *Informacionnye Tekhnologii*, 2006, no. 9, pp. 39—44 (in Russian).

4. Kovalev I. V., Losev V. V., Saramud M. V., Kovalev D. I., Petrosyan M. O. To the question of implementation of multiversion execution environment software of onboard autonomous pilotless objects by means of real-time operating system, *Vestnik SibGAU*, 2017, vol. 18, no. 1, pp. 58—61 (in Russian).

5. Xu J. The $t(n-1)$ -diagnosability and its applications to fault tolerance, *Digest of Papers. Fault-Tolerant Computing: The Twenty-First International Symposium*, 1991, pp. 496—503.

6. Kovalev I., Voroshilova A., Losev V., Saramud M., Chuvashova M., Medvedev A. Comparative Tests of Decision

Making Algorithms for a Multiversion Execution Environment of the Fault Tolerance Software, *Proceedings of 2017 European Conference on Electrical Engineering and Computer Science (EECS 2017)*, 2017.

7. Kovalev I., Losev V., Saramud M., Petrosyan M. Model implementation of the simulation environment of voting algorithms, as a dynamic system for increasing the reliability of the control complex of autonomous unmanned objects, *MATEC Web of Conferences* 132, 04011 (2017).

8. Stel'makh V. O., Kovalev I. V. Building fault-tolerant control systems based on the multi-version approach, *Materialy vserossiiskoi molodezhnoi konferentsii "Informatsionno-telekommunikatsionnye sistemy i tekhnologii (ITSIT-2012)"*, 2012, pp. 172—173 (in Russian).

9. Kovalev I. Evaluation of the reliability of ACS with blocking protection modules, *The Devices*, 2013, no. 6, pp. 20—23 (in Russian).

10. Kovalev I. V., Zelenkov P. V., Losev V. V., Kovalev D. I., Ivleva N. V., Saramud M. V. Multiversion environment creation for control algorithm implementation by autonomous unpiloted objects, *IOP Conf. Series: Materials Science and Engineering* 173 (2017) 012025, doi:10.1088/1757-899X/173/1/012025.

КОДИРОВАНИЕ И ОБРАБОТКА СИГНАЛОВ

CODING AND SIGNAL PROCESSING

УДК 004.052.32 + 681.518.5

DOI: 10.17587/it.25.26-34

Д. В. Ефанов, д-р техн. наук, доц., e-mail: TrES-4b@yandex.ru,
ООО "ЛокоТех-Сигнал", Российский университет транспорта (МИИТ), Москва

Двухмодульные коды с суммированием — эффективный класс разделимых кодов для синтеза диагностического обеспечения устройств и систем управления

Обсуждаются результаты исследования эффективных способов модификации классических кодов с суммированием. Проведен анализ способов построения кодов с суммированием, описаны принципы формирования значений разрядов контрольных векторов, дана классификация кодов с суммированием единичных информационных разрядов. Подробно описаны двухмодульные коды с суммированием единичных информационных разрядов, в том числе установлены некоторые новые свойства данного класса кодов, а также отмечены преимущества и недостатки данных кодов с позиции обнаружения ошибок различных видов и кратностей в информационных векторах.

Ключевые слова: код с суммированием единичных разрядов, код Бергера, модульный код с суммированием, код Боуза—Лина, двухмодульный код с суммированием, характеристики обнаружения ошибок, свойства кода, классификация кодов

Введение

Современной тенденцией в области развития техники и систем автоматического управления является внедрение компьютерных технологий — от использования простейших микроэлектронных и микропроцессорных компонентов до создания сложных киберфизических систем [1–5]. Системы управления становятся более сложными с позиции технической реализации, расширяется их функционал, а также растет степень автоматизации процессов управления. Это все происходит на фоне уменьшения габаритных размеров устройств, увеличения плотности интеграции элементов, централизации функций и оборудования в отдельных узлах систем и т. д.

Развитие техники и вычислительных технологий требует от разработчика все большего внимания к проблеме организации надежной и безопасной передачи данных, своевременного тестирования компонентов и парирования опасных отказов [6–12]. Важным вопросом в связи с этим является использование избыточного кодирования для поддержания высокого уровня надежности и отказоустойчивости аппаратно-программных средств. Кодирование данных в системах управления используется повсеместно [13], а применяемые коды —

крайне разнообразны: от простейших кодов с обнаружением ошибок (кодов с контролем четности комбинаций (кодов паритета) и кодов с суммированием) до более сложных кодов с корректирующими возможностями (кодов Хэмминга, Рида — Соломона, БЧХ-кодов и т. д.) [14].

Ряд критически важных приложений теории кодирования, к слову, не требует корректировки возникающих искажений, а ориентирован только на фиксацию возникшего дефекта и блокировку неверно функционирующего блока системы управления [15]. Подобное обстоятельство говорит о том, что применение весьма избыточных кодов для решения задачи обнаружения ошибок (кодов, ориентированных на исправление ошибок) стало бы невыгодным с позиции аппаратных затрат на реализацию дополнительного оборудования обнаружения некорректно функционирующих блоков, а также создавало бы условия для маскирования неисправностей и их накопления. Таким образом, отдельным направлением развития в теории кодирования является изучение простейших кодов, ориентированных на обнаружение ошибок и имеющих малую избыточность [16].

Данная статья посвящена описанию результатов исследований автора в области получе-

ния эффективных модификаций классических кодов с суммированием единичных информационных разрядов, или (m, k) -кодов (m и k — длины информационных и контрольных векторов соответственно).

1. Принципы построения кодов с суммированием единичных информационных разрядов

Принципы построения (m, k) -кодов основаны на применении операции суммирования в кольце вычетов по заранее установленному модулю. При этом если речь идет о классических кодах Бергера [17], то при определении значений разрядов контрольного вектора предварительно суммируется число единичных информационных разрядов в кольце вычетов по модулю $M = t + 1$, а затем полученное число представляется в бинарной форме и записывается в разряды контрольного вектора. Число разрядов в контрольных векторах кодов Бергера определяется величиной $k = \lceil \log_2(m + 1) \rceil$. Уменьшение значения модуля и выбор его из множества $M \in \{2; 3; \dots; M\}$ дает возможность построения так называемых модульных кодов с суммированием с уменьшенной избыточностью по сравнению с кодами Бергера (остаточных кодов с суммированием) [18]. При модификации классических кодов с суммированием выделяются две и более групп информационных разрядов, объединяемые в векторы, для которых отдельно определяются значения сумм единичных разрядов в кольце вычетов по заранее установленному модулю, а затем полученные числа представляются в бинарной форме и записываются в разные разряды контрольных векторов [19].

Алгоритм 1. *Обобщенный алгоритм построения кодов с суммированием единичных информационных разрядов:*

1. Множество разрядов информационных векторов разбивается на t подмножеств D_1, D_2, \dots, D_t .
2. Устанавливаются значения модулей M_1, M_2, \dots, M_t ($M_i \in \{2; 3; \dots; t_i + 1\}$, где t_i — число разрядов в i -м подмножестве).
3. Определяются наименьшие неотрицательные вычеты сумм единичных разрядов в каждом подмножестве D_i по соответствующему модулю M_i .
4. Полученные числа представляются в бинарной форме и записываются последовательно в разряды контрольного вектора.

На рис. 1 иллюстрируется работа алгоритма при построении одного из так называемых *трехмодульных кодов с суммированием единичных информационных разрядов*. В представлен-

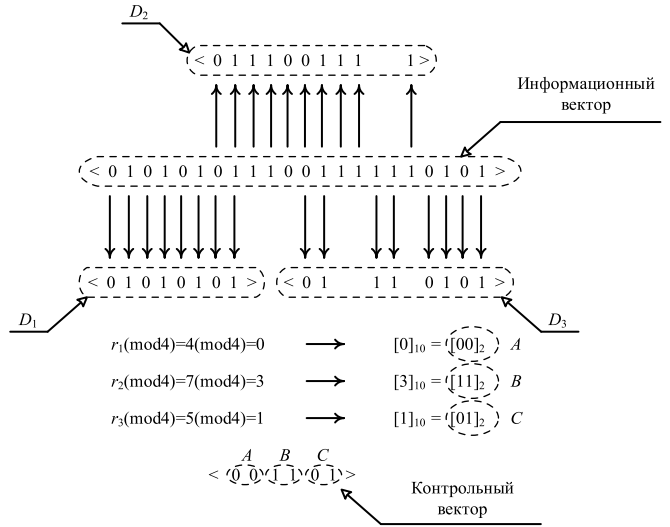


Рис. 1. Получение значений разрядов контрольных векторов для трехмодульного кода

ном коде выделено три подмножества информационных разрядов — D_1, D_2 и D_3 . Установлены следующие значения модулей подсчета суммарного числа единичных разрядов: $M_1 = M_2 = M_3 = 4$. Для каждого подмножества определяется наименьший неотрицательный вычет суммы показанных на рисунке разрядов — чисел $r_1(\text{mod}4), r_2(\text{mod}4)$ и $r_3(\text{mod}4)$. Значения вычетов в бинарной форме записываются последовательно в разряды контрольного вектора.

Как следует из алгоритма и из представленного примера, существует громадное число способов построения кодов с суммированием. Оно определяется не только числом подмножеств и разрядов в каждом из них, но и различными сочетаниями конкретных разрядов информационного вектора и значений допустимых модулей.

Остановимся на рассмотрении первой модификации классических кодов с суммированием — кодах, при построении которых выделяются только два подмножества информационных разрядов [20]. Такие коды назовем *двухмодульными кодами с суммированием единичных информационных разрядов* и обозначим $TM(m, k)$ -коды. Двухмодульные коды являются более простыми, чем иные модификации с большим числом подмножеств разрядов информационных векторов. Кроме того, даже при малых значениях t могут быть построены двухмодульные коды, сравнимые по избыточности с классическими и модульными кодами с суммированием [21].

2. Классификация кодов с суммированием

Все многообразие кодов с суммированием единичных информационных разрядов, или

(m,k) -кодов, может быть классифицировано согласно принципу выделения подмножеств разрядов, для которых отдельно считается значение наименьшего неотрицательного вычета веса по заранее установленному модулю, а также согласно числу разрядов в контрольных векторах.

Целесообразно рассматривать такие коды с суммированием, у которых избыточность не превосходит избыточности классических кодов Бергера ($k \leq \lceil \log_2(m+1) \rceil$). Естественно, существуют коды и с большим числом контрольных разрядов, но во многих случаях они имеют более сложные устройства кодирования и декодирования, чем коды с обозначенным ограничением избыточности. В зависимости от значения m могут быть построены одно-модульные и многомодульные коды.

Под *одно-модульными* кодами понимаются такие коды, для которых подсчитываются наименьшие неотрицательные вычеты по заранее установленному модулю по всем разрядам информационного вектора. Такие коды известны как классические коды Бергера и модульные коды с суммированием (остаточные коды или коды Боуза — Лина [22, 23]), достаточно глубоко исследованы и широко внедрены в прак-

тику. Данные коды обозначены как $S(m,k)$ - и $SM(m,k)$ -коды. Вопросам исследования характеристик классических и модульных кодов с суммированием посвящено множество работ, в том числе работы [24, 25].

Многомодульные коды — это коды, для которых подсчитываются сразу же несколько наименьших неотрицательных вычетов веса части информационных разрядов по заранее установленным модулям. Для кодов с суммированием единичных разрядов данный принцип применен в работе [26], где построено семейство $RS(m,k)$ -кодов. В статье [27] этот же принцип распространен на модификацию модульных кодов с суммированием и на получение $RSM(m,k)$ -кодов. Для малых значений длин информационных векторов ($m \leq 20$) эффективной модификацией кодов с суммированием единичных разрядов оказывается модификация, связанная с выделением двух частей информационного вектора для контроля и построение класса *двухмодульных* кодов, или описываемых здесь $TM(m,k)$ -кодов [20]. Двухмодульными кодами также являются и $RS(m,k)$ - и $RSM(m,k)$ -коды. Также возможно построение кодов с тремя и большим числом подмножеств

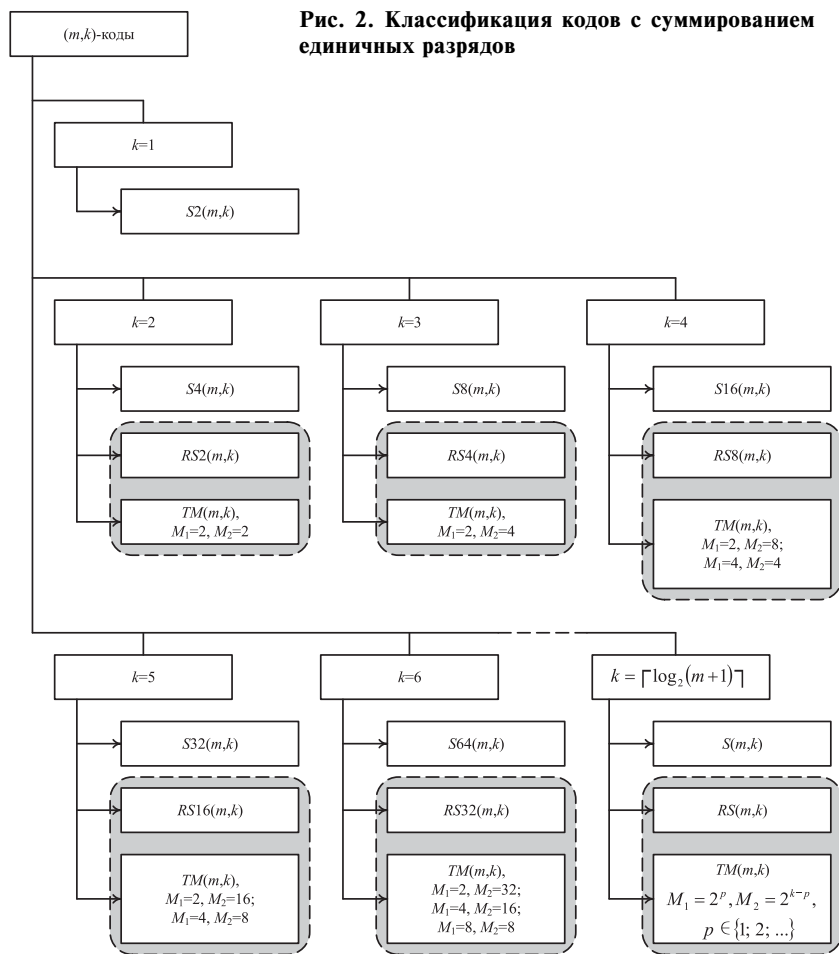
контролируемых групп разрядов информационных векторов [19].

На рис. 2 приведена классификация кодов с суммированием единичных информационных разрядов, охватывающая все виды известных кодов с суммированием. В классификации коды систематизированы по числу разрядов в контрольных векторах, а также по типу — одно- или двухмодульные коды.

3. Характеристики обнаружения ошибок двухмодульными кодами

Как отмечается в работе [20], для каждого значения m может быть построено целое семейство двухмодульных кодов с различными подмножествами разрядов информационных векторов, для которых вычисляются суммы в кольце вычетов по модулям M_1 и M_2 . Для описания свойств кодов далее будем рассматривать двухмодульный код, получаемый по следующему алгоритму (частный случай).

Алгоритм 2. Алгоритм построения двухмодульных кодов с суммированием единичных информацион-



ных разрядов с "половинным" разбиением разрядов на группы:

1. Множество разрядов информационных векторов разбивается на два подмножества D_1 и D_2 , причем в подмножество D_1 включается $\left\lfloor \frac{m}{2} \right\rfloor$ младших разрядов информационного вектора, а в подмножество $D_2 - \left\lceil \frac{m}{2} \right\rceil$ старших разрядов информационного вектора.

2. Устанавливаются значения модулей M_1 и M_2 , выбираемые из множества $M_i \in \{2; 4; 8; \dots; 2^{\lceil \log_2(m_i+1) \rceil}\}$, где m_i — число разрядов в i -м подмножестве.

3. Определяются наименьшие неотрицательные вычеты сумм единичных разрядов в каждом из подмножеств D_1 и D_2 по соответствующим модулям M_1 и M_2 .

4. Полученные числа представляются в бинарной форме и записываются последовательно в разряды контрольного вектора.

Утверждение. Двухмодульный код, построенный по алгоритму 2, будет обнаруживать любые ошибки нечетных кратностей.

Справедливость формулировки утверждения вытекает из особенностей построения $TM(m,k)$ -кода. Согласно алгоритму 2 $D_1 \cap D_2 = \emptyset$. Таким образом, каждому из разрядов информационного вектора принадлежит только одна часть контрольного вектора, соответствующая либо подмножеству D_1 , либо подмножеству D_2 . Так как модуль для каждого подмножества является степенью числа 2, младшие разряды в каждом из подвекторов контрольного вектора, соответствующих подмножествам D_1 и D_2 , будут описываться функцией сложения по модулю два значений всех разрядов каждого из подмножеств. Отсюда как раз и следует, что построенный таким образом код будет обнаруживать любые ошибки с нечетными кратностями.

Рассмотрим три семейства $TM(m,k)$ -кодов:

- 1) $TM(m,k)$ -коды со значениями $M_1 = 4$, $M_2 = 2^{\lceil \log_2(m+1) \rceil - 2}$;
- 2) $TM(m,k)$ -коды со значениями $M_1 = M_2 = 4$;
- 3) $TM(m,k)$ -коды со значениями $M_1 = 4$, $M_2 = 2$.

Коды с меньшим числом контрольных разрядов рассматривать не имеет смысла, так как они относятся к классическим и модифицированным кодам паритета.

Сравним между собой $TM(m,k)$ -коды и известные классические коды Бергера ($S(m,k)$ -коды), модульные коды ($SM(m,k)$ -коды), модифицированные классические и модульные коды ($RS(m,k)$ - и $RSM(m,k)$ -коды). При этом будем анализировать характеристики кодов с одинаковой избыточностью:

1) $TM(m,k)$ -коды со значениями $M_1 = 4$, $M_2 = 2^{\lceil \log_2(m+1) \rceil - 2}$ сравним с $S(m,k)$ - и $RS(m,k)$ -кодами ($k = \lceil \log_2(m+1) \rceil$);

2) $TM(m,k)$ -коды со значениями $M_1 = M_2 = 4$ — с $S16(m,k)$ - и $RS8(m,k)$ -кодами ($k = 4$);

3) $TM(m,k)$ -коды со значениями $M_1 = 4$, $M_2 = 2$ — с $S8(m,k)$ - и $RS4(m,k)$ -кодами ($k = 3$).

С позиции задач технической диагностики для любого (m,k) -кода важными являются показатели обнаружения ошибок в целом (общее число необнаруживаемых ошибок $N_{m,k}$), а также по видам и кратностям (наличие и общее число необнаруживаемых монотонных, симметричных и асимметричных ошибок различных кратностей [28]).

Для оценки эффективности обнаружения ошибок в информационных векторах (m,k) -кодов в общем (без выделения различных видов ошибок и кратностей) удобно использовать коэффициент $\xi_{m,k}$, показывающий, насколько близок рассматриваемый (m,k) -код по своим характеристикам к такому разделимому коду, который имеет минимальное общее число необнаруживаемых ошибок в информационных векторах ($N_{m,k}^{\min}$) [26]. Минимальное общее число необнаруживаемых (m,k) -кодом ошибок определяется по следующей формуле:

$$N_{m,k}^{\min} = 2^m(2^{m-k} - 1). \quad (1)$$

Исходя из формулы (1), можно заключить, что для каждой пары значений m и k существует свое число $N_{m,k}^{\min}$, по отношению к которому можно вычислить коэффициент $\xi_{m,k}$. Значение коэффициента $\xi_{m,k}$ вычисляется как отношение:

$$\xi_{m,k} = \frac{N_{m,k}^{\min}}{N_{m,k}} \cdot 100 \%. \quad (2)$$

Для кодов с одинаковыми значениями m и k сравнимы и соответствующие коэффициенты $\xi_{m,k}$. Значение $\xi_{m,k}$ лежит в диапазоне от 0 до 100 %, и чем ближе оно к 100 %, тем эффективнее рассматриваемый (m,k) -код использует свои контрольные разряды с позиции обнаружения общего числа ошибок в информационных векторах.

На рис. 3 представлены графики зависимости значений $\xi_{m,k}$ от числа разрядов в информационных векторах для классических и модифицированных кодов Бергера. $TM(m,k)$ -коды в диапазоне значений $m = 4..7$ обнаруживают большее число ошибок, чем классические $S(m,k)$ -коды, но меньшее, чем их модификация $RS(m,k)$ -коды. Такое обстоятельство объясняется следующими фактами. У $RS(m,k)$ - и $TM(m,k)$ -кодов распределение информаци-

онных векторов между всеми контрольными векторами является более сбалансированным, чем у $S(m,k)$ -кодов. У $S(m,k)$ -кодов распределение информационных векторов между контрольными векторами крайне неравномерно, что определяется принципами построения кода [24]. Неравномерность распределения информационных векторов между контрольными в $RS(m,k)$ - и $TM(m,k)$ -кодах сглаживается за счет определения двух групп наименьших неотрицательных вычетов весов информационных векторов и отдельного их контроля в рядах контрольных векторов. Преимущество же $RS(m,k)$ -кодов перед $TM(m,k)$ -кодами в обозначенном диапазоне длин информационных векторов связано с тем, что сами $RS(m,k)$ -коды являются частным случаем $TM(m,k)$ -кодов, а контроль по модулю $M = 4$ в первых осуществляется по всем разрядам информационных векторов, а во вторых — только по половине.

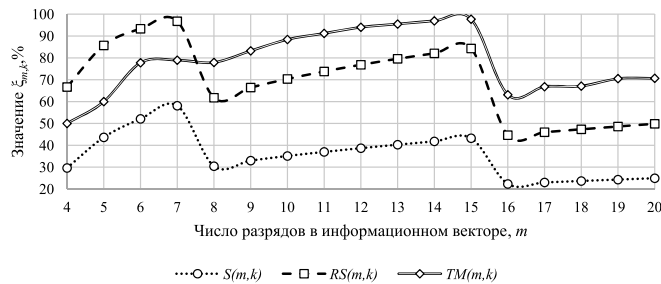


Рис. 3. Значения коэффициентов $\xi_{m,k}$ для кодов с числом контрольных разрядов $k = \lceil \log_2(m+1) \rceil$

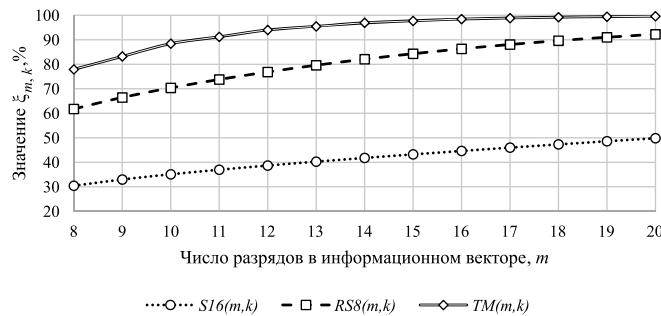


Рис. 4. Значения коэффициентов $\xi_{m,k}$ для кодов с числом контрольных разрядов $k = 4$

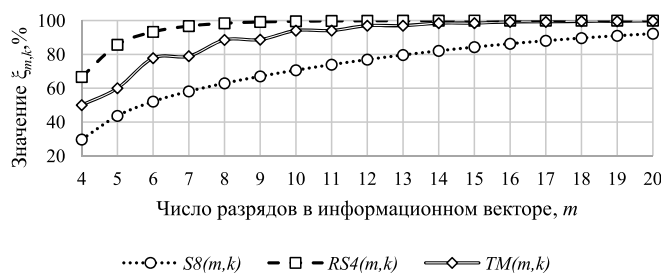


Рис. 5. Значения коэффициентов $\xi_{m,k}$ для кодов с числом контрольных разрядов $k = 3$

При значениях $m \geq 8$ $TM(m,k)$ -коды обнаруживают большее число ошибок, чем $S(m,k)$ - и $RS(m,k)$ -коды, причем двухмодульные коды в данном диапазоне значений m более чем вдвое эффективнее классических кодов Бергера. Подобное преимущество $TM(m,k)$ -кодов связано с изменениями значений модулей — M_1 остается равным четырем, тогда как M_2 становится равным четырем, а при достижении значений $m = 16, 32, \dots$ принимает значения $M_2 = 8, 16, \dots$. С каждым достижением значения $m = 2^a$, $a \in \{2; 3; \dots\}$ значения коэффициента $\xi_{m,k}$ падают для всех трех кодов по отношению к значениям при $m = 2^a - 1$. Это обусловливается появлением нового контрольного разряда у всех кодов. С дальнейшим увеличением значения m до ближайшей степени числа 2 наблюдается постепенное увеличение значений $\xi_{m,k}$ для всех трех кодов.

На рис. 4 приводятся зависимости значения коэффициента $\xi_{m,k}$ от значения m для кодов с числом контрольных разрядов $k = 4$. Здесь на всех значениях m преимущество в обнаружении ошибок принадлежит $TM(m,k)$ -кодам. С увеличением длины информационного вектора все коды с разной скоростью приближаются к коду с минимальным общим числом необнаруживаемых ошибок.

На рис. 5 даны зависимости для кодов с числом контрольных разрядов $k = 3$. Для сравниваемых кодов наихудшим с позиции эффективности обнаружения общего числа ошибок является модульный код $S8(m,k)$, тогда как наиболее эффективным является код $RS4(m,k)$. С увеличением длины информационного вектора по показателю $\xi_{m,k}$ $TM(m,k)$ -код приближается к $RS4(m,k)$ -коду гораздо быстрее, чем $S8(m,k)$ -код. Следует отметить, что двухмодульный код и $RS4(m,k)$ — это коды из одного семейства, в котором $M_1 = 4$, а $M_2 = 2$.

Показатель $\xi_{m,k}$ является, однако, только общим коэффициентом, никак не характеризующим (m,k) -код по обнаружению им ошибок конкретной кратностью d и конкретных видов (монотонных, симметричных или асимметричных [28]).

Так как рассмотренные выше $TM(m,k)$ -коды со значениями $M_1 = 4$, $M_2 = 2$ являются известными модификациями кода $RS(m,k)$, а также явно не обнаруживают большее число ошибок в информационных векторах, чем последние (рис. 5), далее их рассматривать не будем. Также не будем рассматривать и $TM(m,k)$ -коды со значениями $M_1 = M_2 = 4$, так как в диапазоне длин информационных векторов $m = 8 \dots 15$ это как раз и есть двухмодульные коды с $M_1 = 4$, $M_2 = 2^{\lceil \log_2(m+1) \rceil - 2}$. Таким образом, приведем

Показатели обнаружения ошибок в информационных векторах $TM(m,k)$ -кодов

m	k	Число необнаруживаемых ошибок					
		Всего	Монотонных	Симметричных	Асимметричных	Двукратных	Четырехкратных
4	3	32	8	20	4	24	8
5	3	160	48	88	24	112	48
6	3	576	96	336	144	288	288
7	3	2432	400	1272	760	960	1280
8	4	4928	66	4646	216	1536	2560
9	4	19 072	404	17 148	1520	4096	9216
10	4	72 960	1480	62 680	8800	10 240	30 720
11	4	285184	6320	232 120	46 744	25 600	97 280
12	4	1 111 040	22 560	858 392	230 088	61 440	291 840
13	4	4 384 768	84 800	3 211 024	1 088 944	147 456	849 920
14	4	17 289 216	300 160	12 027 232	4 961 824	344 064	2 379 776
15	4	68 657 152	1 058 096	45 462 032	22 137 024	802 816	6 537 216
16	5	212 476 928	576 706	165 571 366	46 328 856	1 835 008	16 859 136
17	5	803 276 800	1 197 476	625 608 364	176 470 960	4 194 304	43 810 816
18	5	3 200 913 408	4 311 432	2 363 642 904	832 959 072	9 437 184	113 836 032
19	5	12 179 943 424	9 936 720	8 982 319 272	3 187 687 432	21 233 664	286 654 464
20	5	48 626 032 640	33 165 600	34 133 803 160	14 459 063 880	47 185 920	723 517 440

здесь более подробные характеристики сравнимого с классическим и модифицированным кодами Бергера $TM(m,k)$ -кода с $M_1 = 4$, $M_2 = 2^{\lceil \log_2(m+1) \rceil - 2}$.

В табл. 1 приводятся рассчитанные значения числа необнаруживаемых $TM(m,k)$ -кодами ошибок в информационных векторах. При этом выделено общее число необнаруживаемых ошибок по видам, а также дано число необнаруживаемых ошибок малой кратностью ($d = 2$ и $d = 4$) — наиболее вероятные искажения.

Необнаруживаемые $TM(m,k)$ -кодами ошибки распределяются по видам так, как это показано на рис. 6. Наименьший процент в общем числе необнаруживаемых ошибок имеют монотонные ошибки (показатель $\nu_{m,k}$), несколько больший — асимметричные ошибки (показатель $\alpha_{m,k}$), наибольшую долю необнаруживаемых ошибок составляют симметричные ошибки (показатель $\sigma_{m,k}$). До величины $m = 8$ в классе необнаруживаемых монотонные ошибки имеют существенную долю — около 20...30 %, что сказывается и на доле симметричных необнаруживаемых ошибок — она составляет примерно 50...60 %. Для $TM(m,k)$ -кодов с числом разрядов в информационных векторах $m = 4...7$ значения модулей $M_1 = 4$, $M_2 = 2$. Именно значение второго модуля и оказывает столь существенное влияние на наличие монотонных необнаруживаемых ошибок в информационных векторах. По достижении значе-

ния $m = 8$ значение второго модуля изменяется ($M_2 = 4$). Это приводит к резкому уменьшению доли монотонных необнаруживаемых ошибок и к соответствующему увеличению доли симметричных необнаруживаемых ошибок. С увеличением значения m от величины $m = 8$ до $m = 15$ наблюдается постепенное уменьшение значения величины $\sigma_{m,k}$, тогда как величина $\alpha_{m,k}$, наоборот, возрастает. При этом значение $\nu_{m,k}$ незначительно изменяется, оставаясь в пределах 1,3...2,2 %. По достижении значения $m = 16$ увеличивается значение второго модуля ($M_2 = 8$). Это обуславливает небольшое изменение в распределении необнаруживаемых ошибок по видам: вновь увеличивается доля необнаруживаемых симметричных ошибок и снижаются доли необнаруживаемых монотон-

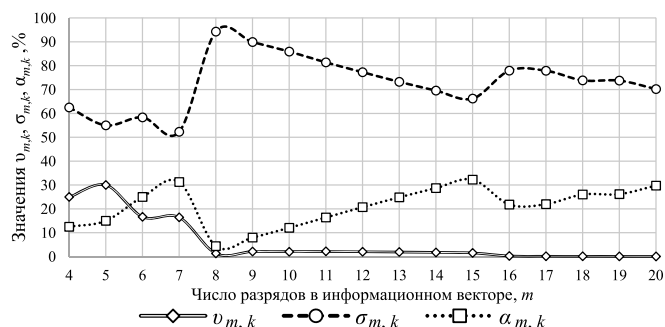


Рис. 6. Распределение необнаруживаемых $TM(m,k)$ -кодами ошибок по видам

Показатели обнаружения ошибок $TM(m,k)$ -кодами по кратностям

m	k	$\gamma_{m,k}$	Значения величин $\beta_{m,d}$, %																	
			2	4	6	8	10	12	14	16	18	20								
4	3	13,333	25	50																
5	3	16,129	35	30																
6	3	14,286	30	30	0															
7	3	14,961	35,714	28,571	21,429															
8	4	7,549	21,429	14,286	10,714	25														
9	4	7,29	22,222	14,286	11,905	13,889														
10	4	6,965	22,222	14,286	11,905	13,889	0													
11	4	6,803	22,727	14,394	12,284	12,879	11,364													
12	4	6,624	22,727	14,394	12,284	12,879	11,364	25												
13	4	6,535	23,077	14,51	12,471	12,646	12,238	13,462												
14	4	6,441	23,077	14,51	12,471	12,646	12,238	13,462	0											
15	4	6,394	23,333	14,615	12,587	12,572	12,413	12,692	11,667											
16	5	4,947	23,333	14,135	10,992	9,62	8,55	7,993	6,927	14,063										
17	5	4,676	23,529	14,044	10,761	9,364	8,354	7,752	7,142	7,445										
18	5	4,658	23,529	14,191	11,058	9,626	8,604	7,943	7,335	7,445	0									
19	5	4,431	23,684	14,106	10,846	9,4	8,412	7,765	7,255	7,029	6,291									
20	5	4,423	23,684	14,241	11,107	9,64	8,634	7,945	7,412	7,13	6,476	13,281								

ных и асимметричных ошибок. С увеличением значения m сохраняется тенденция для диапазона $m = 8...15$.

В табл. 2 приводятся рассчитанные значения долей необнаруживаемых $TM(m,k)$ -кодами ошибок от общего их числа (показатель $\gamma_{m,k}$) и долей необнаруживаемых ошибок кратностью d от общего числа ошибок данной кратностью (показатель $\beta_{m,d}$). В диапазоне длин информационных векторов $m = 4...7$ значение $\gamma_{m,k}$ превышает 10 %, а значения величин $\beta_{m,d}$ находятся в пределах 20...50 %. В данном диапазоне длин информационных векторов $TM(m,k)$ -коды не обнаруживают значительное число ошибок, однако меньшее, чем классические коды Бергера [24]. Для $TM(m,k)$ -кодов со значениями длин информационных векторов $m \geq 8$ наблюдается улучшение показателей обнаружения ошибок: значение $\gamma_{m,k}$ становится меньше 10 % и с увеличением m уменьшается. Доли необнаруживаемых ошибок по кратностям также уменьшаются: в диапазоне $m = 8...20$ значения $\beta_{m,2} = 21...24$ %, $\beta_{m,4} = 14...15$ %, $\beta_{m,6} = 10...13$ % и т. д.

На рис. 7 и рис. 8 зависимости для $\gamma_{m,k}$ и $\beta_{m,2}$ от значения m иллюстрируются графиками, а также дается сравнение с известными $S(m,k)$ и $RS(m,k)$ кодами. $TM(m,k)$ -коды имеют явное преимущество перед известными кодами с суммированием единичных информационных разрядов по показателям обнаружения

ошибок в общем в диапазоне $m = 8...20$. Стоит отметить, что в данном диапазоне значения величин $\beta_{m,2}$ для $TM(m,k)$ - и $RS(m,k)$ -кодов одинаковы.

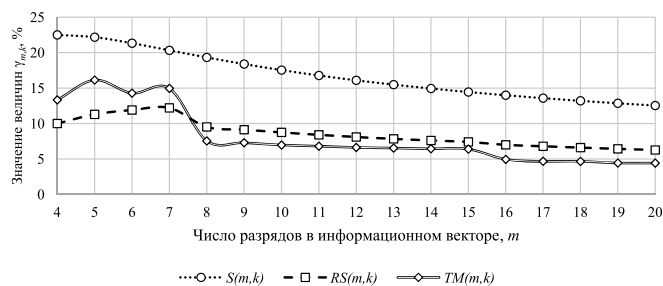


Рис. 7. Значения показателей $\gamma_{m,k}$ для различных кодов с суммированием

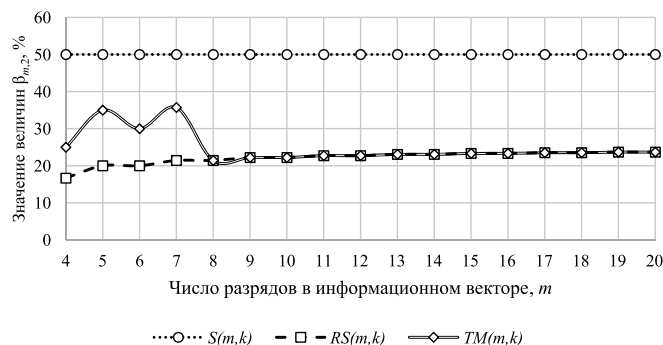


Рис. 8. Значения показателей $\beta_{m,2}$ для различных кодов с суммированием

Недостатком $TM(m,k)$ -кодов является присутствие в классе необнаруживаемых монотонных ошибок с кратностями $d = 2, 4, \dots$ при $m = 4\dots 7$ и монотонных ошибок с кратностями $d = 4, 8, \dots$ при $m = 8\dots 20$. Например, $S(m,k)$ -коды обнаруживают все монотонные ошибки, а $RS(m,k)$ -коды — в диапазоне $m = 4\dots 7$ не обнаруживают часть монотонных ошибок кратностью $d = 4$, в диапазоне $m = 8\dots 15$ — $d = 8$, в диапазоне $m = 15\dots 20$ — кратностью $d = 16$ [19]. Эти особенности в распределениях необнаруживаемых ошибок $TM(m,k)$ -кодами следует учитывать при их использовании для решения задач синтеза систем с обнаружением неисправностей [29].

Заключение

Представленные в статье двухмодульные коды с суммированием единичных информационных разрядов — это расширенный класс известных ранее модификаций классического кода Бергера, обладающий улучшенными характеристиками обнаружения ошибок как в целом, так и по кратностям. Важным преимуществом перед другими кодами с суммированием является то, что двухмодульный код обнаруживает гораздо большее число симметричных ошибок в информационных векторах.

Следует, однако, учитывать тот факт, что распределение необнаруживаемых ошибок по видам в двухмодульных кодах определяется значениями устанавливаемых модулей при построении кода. В зависимости от этих значений находится и тот факт, будет ли двухмодульный код обнаруживать все монотонные ошибки конкретной кратностью d или нет (аналогичное обстоятельство имеет место для асимметричных ошибок). Это, в свою очередь, определяет принципы использования двухмодульных кодов при решении задач синтеза систем с обнаружением неисправностей.

Список литературы

1. Ubar R., Raik J., Vierhaus H.-T. Design and Test Technology for Dependable Systems-on-Chip (Premier Reference Source). — Information Science Reference, Hershey — New York, IGI Global, 2011, 578 p.
2. Dikmen M., Burns C. Trust in Autonomous Vehicles: The Case of Tesla Autopilot and Summon // IEEE International Conference on Systems, Man, and Cybernetics (SMC), 5–8 October 2017, Banff, AB, Canada, pp. 1093–1098.
3. Kharchenko V., Kondratenko Yu., Kasprzyk J. Green IT Engineering: Concepts, Models, Complex Systems Architectures // Springer Book series "Studies in Systems, Decision and Control". 2017. Vol. 74. 305 p.
4. Brogi A., Forti S. QoS-Aware Deployment of IoT Applications Through the Fog // IEEE Internet of Things Journal, 2017, Vol. 4, Issue 5, pp. 1185–1192.

5. Naharov V. Cyber Physical Computing for IoT-driven Services. — New York, Springer International Publishing AG, 2018, 279 p.
6. Стемковский А. Л., Тельпухов Д. В., Соловьев Р. А., Мячиков М. В., Тельпухова Н. В. Разработка технологически независимых метрик для оценки маскирующих свойств логических схем // Вычислительные технологии. 2016. Т. 21, № 2. С. 53–62.
7. Tshagharyan G., Harutyunyan G., Shoukourian S., Zorian Y. Securing Test Infrastructure of System-on-Chips // Proceedings of 14th IEEE East-West Design & Test Symposium (EWDTS'2016), Yerevan, Armenia, October 14–17, 2016, p. 29–32.
8. Степченко Ю. А., Каменских А. Н., Тюрин С. Ф., Рождественский Ю. В. Модели отказоустойчивых самосинхронных схем // Системы и средства информатики. 2016. Т. 26, № 4. С. 19–30.
9. Стемковский А. Л., Тельпухов Д. В., Соловьев Р. А., Тельпухова Н. В. Исследование вероятностных методов оценки логической уязвимости комбинационных схем // Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС), Россия, М.: ИППМ РАН, 2016. № 4. С. 121–126.
10. Sklyar V., Kharchenko V., Bardis N. G. Assurance Case for Green IT Applications: Proof of Compliance with Power Consumption Claims // 4th International Conference on Mathematics and Computers in Sciences and in Industry (MCSI), 24-27 August 2017, Corfu, Greece, pp. 124–127.
11. Matrosova A., Mitrofanov E., Ostanin S., Kirienko I. Trojan Circuits Preventing and Masking in Sequential Circuits // Proceedings of 23rd IEEE On-Line Testing and Robust System Design (IOLTS'2017), Thessaloniki, Greece, 3-5 July 2017, pp. 213–214.
12. Dautov R., Mosin S. A Technique to Aggregate Classes of Analog Fault Diagnostic Data Based on Association Rule Mining // 19th International Symposium on Quality Electronic Design (ISQED), 13-14 March 2018, Santa Clara, CA, USA, pp. 238–243.
13. Fujiwara E. Code Design for Dependable Systems: Theory and Practical Applications. — John Wiley & Sons, 2006, 720 p.
14. Ryan W. E., Lin S. Channel Codes: Classical and Modern. Cambridge University Press, 2009, 708 p.
15. Дрозд А. В., Харченко В. С., Антошук С. Г., Дрозд Ю. В., Дрозд М. А., Сулима Ю. Ю. Рабочее диагностирование безопасных информационно-управляющих систем / Под ред. А. В. Дрозда и В. С. Харченко. Харьков: Национальный аэрокосмический университет им. Н. Е. Жуковского "ХАИ", 2012, 614 с.
16. Piestrak S. J. Design of Self-Testing Checkers for Unidirectional Error Detecting Codes. — Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, 1995, 111 p.
17. Berger J. M. A Note on Error Detecting Codes for Asymmetric Channels // Information and Control. 1961. Vol. 4. Iss. 1. P. 68–73.
18. Sapozhnikov V., Sapozhnikov V., Efanov D. Modular Sum Code in Building Testable Discrete Systems // Proceedings of 13th IEEE East-West Design & Test Symposium (EWDTS'2015), Batumi, Georgia, September 26–29, 2015, pp. 181–187.
19. Efanov D., Sapozhnikov V., Sapozhnikov V. Generalized Algorithm of Building Summation Codes for the Tasks of Technical Diagnostics of Discrete Systems // Proceedings of 15th IEEE East-West Design & Test Symposium (EWDTS'2017), Novi Sad, Serbia, September 29 — October 2, 2017, pp. 365–371.
20. Efanov D. V., Sapozhnikov V. V., Sapozhnikov V. I. V. Two-Modulus Codes with Summation of One-Data Bits for Technical Diagnostics of Discrete Systems // Automatic Control and Computer Sciences. 2018. Vol. 52. Iss. 1. P. 1–12.
21. Ефанов Д. В., Сапожников В. В., Сапожников В. В. Исследование двухмодульных кодов с суммированием единичных разрядов со счетом по модулю четыре // Электронное моделирование. 2018. Т. 40, № 4. С. 29–54.
22. Согомонян Е. С., Слабаков Е. В. Самопроверяемые устройства и отказоустойчивые системы. М.: Радио и связь, 1989. 208 с.
23. Das D., Toubia N. A. Synthesis of Circuits with Low-Cost Concurrent Error Detection Based on Bose-Lin Codes // Journal

of Electronic Testing: Theory and Applications. 1999. Vol. 15. Iss. 1—2. P. 145—155.

24. Ефанов Д. В., Сапожников В. В., Сапожников Вл. В. О свойствах кода с суммированием в схемах функционального контроля // Автоматика и телемеханика. 2010. № 6. С. 155—162.

25. Ефанов Д. В., Сапожников В. В., Сапожников Вл. В. Применение модульных кодов с суммированием для построения систем функционального контроля комбинационных логических схем // Автоматика и телемеханика. 2015. № 10. С. 152—169.

26. Блюдов А. А., Ефанов Д. В., Сапожников В. В., Сапожников Вл. В. Построение модифицированного кода Бергера с минимальным числом необнаруживаемых ошибок инфор-

мационных разрядов // Электронное моделирование. 2012. Т. 34. № 6. С. 17—29.

27. Блюдов А. А., Ефанов Д. В., Сапожников В. В., Сапожников Вл. В. О кодах с суммированием единичных разрядов в системах функционального контроля // Автоматика и телемеханика. 2014. № 8. С. 131—145.

28. Сапожников В. В., Сапожников Вл. В., Ефанов Д. В. Классификация ошибок в информационных векторах систематических кодов // Известия вузов. Приборостроение. 2015. Т. 58, № 5. С. 333—343.

29. Ефанов Д. В. Синтез самопроверяемых комбинационных устройств на основе кодов с эффективным обнаружением симметричных ошибок // Труды СПИИРАН. 2018. № 4. С. 62—91.

D. V. Efanov, DSc, Associate Professor, Head of the Direction of monitoring and diagnosis systems department of "LocoTech-Signal" LLC, Professor of Russian university of transport, Moscow

Two-Modulus Codes with Summation — an Effective Class of Separable Codes for the Synthesis of Diagnostic Support for Devices and Control Systems

This article is devoted to the presentation of the results of the research of effective methods of classical sum code modification. The article analyzed the methods for sum code constructing, described the principles of the formation of number of bits in check vectors, codes with summation of ones data bits classification is given. Two-modulus codes with summation of ones data bits are described in detail, in particular, some new properties of this class of codes are identified, and also advantages and disadvantages of these codes are noted from the positions of any type and multiplicity error detection in data vectors.

Keywords: code with summation ones bits; Berger code; two-modulus sum code; error detection properties; code properties; code classification

DOI: 10.17587/it.25.26-34

References

1. Ubar R., Raik J., Vierhaus H.-T. Design and Test Technology for Dependable Systems-on-Chip (Premier Reference Source), Information Science Reference, Hershey — New York, IGI Global, 2011, 578 p.

2. Dikmen M., Burns C. Trust in Autonomous Vehicles: The Case of Tesla Autopilot and Summon, *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 5—8 October 2017, Banff, AB, Canada, pp. 1093—1098.

3. Kharchenko V., Kondratenko Yu., Kacprzyk J. Green IT Engineering: Concepts, Models, Complex Systems Architectures, *Springer Book series "Studies in Systems, Decision and Control"*, 2017, vol. 74, 305 p.

4. Brogi A., Forti S. QoS-Aware Deployment of IoT Applications Through the Fog, *IEEE Internet of Things Journal*, 2017, vol. 4, iss. 5, pp. 1185—1192.

5. Hahanov V. Cyber Physical Computing for IoT-driven Services, New York, Springer International Publishing AG, 2018, 279 p.

6. Stempkovsky A. L., Telpukhov D. V., Solovjov R. A., Myachikov M. V., Telpukhova N. V. *Razrabotka tekhnologicheskoi nezavisimyh metrik dlya ocenki maskiruyushchih svoystv logicheskikh skhem* (The development of technology-independent metrics for evaluation of the masking properties of logic), *Vychislitelnyye Tekhnologii*, 2016, vol. 21, iss. 2, pp. 53—62 (in Russian).

7. Tshagharyan G., Harutyunyan G., Shoukourian S., Zorian Y. Securing Test Infrastructure of System-on-Chips, *Proceedings of 14th IEEE East-West Design & Test Symposium (EWDTS'2016)*, Yerevan, Armenia, October 14—17, 2016, p. 29—32.

8. Stepchenkov Yu. A., Kamenskikh A. N., Tyurin S. F., Rozhdestvenskiy Yu. V. *Modeli otkazoustojchivyyh samosinhronnykh*

skhem (Models of fault-tolerant self-timed circuits), *Sistemy i Sredstva Informatiki*, 2016, vol. 26, iss. 4, pp. 19—30 (in Russian).

9. Stempkovsky A. L., Telpukhov D. V., Solovjov R. A., Telpukhova N. V. *Issledovanie veroyatnostnykh metodov ocenki logicheskoy uyazvimosti kombinacionnykh skhem* (Probabilistic methods for reliability evaluation of combination circuits), *Problemy razrabotki perspektivnykh mikro- i nanoehlektronnykh sistem (MES)*, Moscow, Russia, IPPM RAS, 2016, iss. 4, pp. 121—126 (in Russian).

10. Sklyar V., Kharchenko V., Bardis N. G. Assurance Case for Green IT Applications: Proof of Compliance with Power Consumption Claims, *4th International Conference on Mathematics and Computers in Sciences and in Industry (MCSI)*, 24—27 August 2017, Corfu, Greece, pp. 124—127.

11. Matrosova A., Mitrofanov E., Ostanin S., Kirienko I. Trojan Circuits Preventing and Masking in Sequential Circuits, *Proceedings of 23rd IEEE On-Line Testing and Robust System Design (IOLTS'2017)*, Thessaloniki, Greece, 3—5 July 2017, pp. 213—214.

12. Dautov R., Mosin S. A Technique to Aggregate Classes of Analog Fault Diagnostic Data Based on Association Rule Mining, *19th International Symposium on Quality Electronic Design (ISQED)*, 13—14 March 2018, Santa Clara, CA, USA, pp. 238—243.

13. Fujiwara E. Code Design for Dependable Systems: Theory and Practical Applications, John Wiley & Sons, 2006, 720 p.

14. Ryan W. E., Lin S. Channel Codes: Classical and Modern, Cambridge University Press, 2009, 708 p.

15. Drozd A. V., Kharchenko V. S., Antoshchuk S. G., Drozd Yu. V., Drozd M. A., Sulima Yu. Yu. *Rabochee diagnostirovanie bezopasnykh informatsionno-upravlyayustchikh sistem* (Objects and Methods of On-Line Testing for Safe Instrumentation and Control Systems), Kharkov, National Aerospace University "KhAI", 2012, 614 p. (in Russian).

16. **Piestrak S. J.** Design of Self-Testing Checkers for Unidirectional Error Detecting Codes, Wrocław, Oficyna Wydawnicza Politechniki Wrocławskiej, 1995, 111 p.
17. **Berger J. M.** A Note on Error Detecting Codes for Asymmetric Channels, *Information and Control*, 1961, vol. 4, iss. 1, pp. 68–73.
18. **Sapozhnikov V., Sapozhnikov VI., Efanov D.** Modular Sum Code in Building Testable Discrete Systems, *Proceedings of 13th IEEE East-West Design & Test Symposium (EWDTS'2015)*, Batumi, Georgia, September 26–29, 2015, pp. 181–187.
19. **Efanov D., Sapozhnikov V., Sapozhnikov VI.** Generalized Algorithm of Building Summation Codes for the Tasks of Technical Diagnostics of Discrete Systems, *Proceedings of 15th IEEE East-West Design & Test Symposium (EWDTS'2017)*, Novi Sad, Serbia, September 29 – October 2, 2017, pp. 365–371.
20. **Efanov D. V., Sapozhnikov V. V., Sapozhnikov VI. V.** Two-Modulus Codes with Summation of One-Data Bits for Technical Diagnostics of Discrete Systems, *Automatic Control and Computer Sciences*, 2018, vol. 52, iss. 1, pp. 1–12.
21. **Efanov D. V., Sapozhnikov V. V., Sapozhnikov VI. V.** The research of two-modulus codes with summation of ones in data bits with calculation by modulo "four", *Electronic Modeling*, 2018, vol. 40, iss. 4, pp. 29–54 (in Russian).
22. **Sogomonyan E. S., Slabakov E. V.** *Samoproverjaemye ustrojstva i otkazoustojchivyye sistemy* (Self-checking devices and fault-tolerant systems), Radio & Svjaz', Moscow, 208 p. (in Russian).
23. **Das D., Touba N. A.** Synthesis of Circuits with Low-Cost Concurrent Error Detection Based on Bose-Lin Codes, *Journal of Electronic Testing: Theory and Applications*, 1999, vol. 15, iss. 1–2, pp. 145–155.
24. **Efanov D. V., Sapozhnikov V. V., Sapozhnikov VI. V.** *O svoystvakh koda s summirovaniem v skhemakh funkcional'nogo kontrol'ya* (On Sum Code Properties in Concurrent Error Detection Systems), *Avtomatika i telemekhanika*, 2010, iss. 6, pp. 155–162 (in Russian).
25. **Efanov D. V., Sapozhnikov V. V., Sapozhnikov VI. V.** *Primenenie modul'nykh kodov s summirovaniem dlya postroeniya sistem funkcional'nogo kontrol'ya kombinatsionnykh logicheskikh skhem* (Application of Modular Summation Codes to Concurrent Error Detection Systems for Combinational Boolean Circuits), *Avtomatika i telemekhanika*, 2015, iss. 10, pp. 152–169 (in Russian).
26. **Blyudov A. A., Efanov D. V., Sapozhnikov V. V., Sapozhnikov VI. V.** *Postroenie modifitsirovannogo koda Bergera s minimal'nykh chislom neobnaruzhivaemykh oshibok informatsionnykh razryadov* (Formation of the Berger Modified Code with Minimum Number of Undetectable Errors of Data Bits), *Electronnoye Modelirovanje*, 2012, vol. 34, iss. 6, pp. 17–29 (in Russian).
27. **Blyudov A. A., Efanov D. V., Sapozhnikov V. V., Sapozhnikov VI. V.** *O kodah s summirovaniem edinichnykh razryadov v sistemah funkcional'nogo kontrol'ya* (On Codes with Summation of Data Bits in Concurrent Error Detection Systems), *Avtomatika i telemekhanika*, 2014, no. 8, pp. 131–145. (in Russian).
28. **Sapozhnikov V. V., Sapozhnikov VI. V., Efanov D. V.** *Klassifikatsiya oshibok v informatsionnykh vektorakh sistematicheskikh kodov* (Errors Classification in Information Vectors of Systematic Codes), *Izvestiya Vysshikh Uchebnykh Zavedeniy. Priborostroenie*, 2015, vol. 58, iss. 5, pp. 333–343 (in Russian).
29. **Efanov D. V.** *Sintez samoproveryaemykh kombinatsionnykh ustrojstv na osnove kodov s ehffektivnym obnaruzheniem simmetrichnykh oshibok* (The Synthesis of Self-Checking Combinational Devices on the Basis of Codes with the Effective Symmetrical Error Detection), *Trudy SPIIRAN*, 2018, iss. 4, pp. 62–91 (in Russian).

УДК 621.391

DOI: 10.17587/it.25.35-40

С. В. Дворников, д-р техн. наук, проф., e-mail: practicsdv@yandex.ru,
А. В. Пшеничников, канд. техн. наук, доц., e-mail: siracooz77@mail.ru,
С. С. Манаенко, канд. техн. наук, e-mail: manaenkoss@mail.ru,

Военная академия связи имени Маршала Советского Союза С. М. Буденного, г. Санкт-Петербург

Статистические характеристики помехозащищенных радиолиний с управлением частотным ресурсом

Приведены результаты исследования статистических характеристик помехозащищенных радиолиний, функционирующих в условиях неоднородности сигнальной и помеховой обстановки на рабочих частотах. Предлагается реализация принципов адаптивной помехозащищенной радиосвязи. Доказана гипотеза об изменении функции плотности распределения превышения уровня сигнала над уровнем помех в помехозащищенных радиолиниях с адаптивным управлением. Обоснована необходимость разработки методов оценки вероятностно-временных характеристик таких радиолиний.

Ключевые слова: помехозащищенные режимы функционирования, адаптивное управление, частотный ресурс, статистические характеристики

Введение

Существенное увеличение возможностей средств преднамеренного деструктивного воздействия на ресурсы радиолиний обуславливает поиск новых подходов к обеспечению помехозащищенности их функционирования.

В настоящее время большинство технических решений по реализации помехозащищенных режимов ограничивается применением алгоритмов медленной программной перестройки рабочей частоты (ППРЧ) в совокупности с помехоустойчивыми видами модуляционных форматов [1–7].

В общем случае такие решения изначально по своей сути противоречивы, поскольку основываются на снижении энергетического потенциала радиолинии вследствие реализации режима ППРЧ и применения многомерных сигнальных конструкций с достаточно низкими свойствами помехозащищенности. Для решения данного противоречия в работе [8] предложены модели помехозащищенных линий радиосвязи. Из анализа полученных в [8] решений следует, что наиболее эффективным видом управления ресурсами помехозащищенных радиолиний, обеспечивающими повышение их свойств помехоустойчивости, является реализация методов адаптивного управления. При технической реализации выбранных методов управления наибольшую актуальность приобретает задача оценки их вероятностно-временных характеристик.

Решение данной задачи нашло отражение в ряде работ [9–14]. Анализ предложенных в них решений показывает, что они получены в граничных условиях статистической однородности рабочих частот, что существенно сужает область их практического применения. Для разрешения данного противоречия в статье представлены результаты исследования статистических характеристик помехозащищенных радиолиний с управлением их ресурсами.

Модель помехозащищенной радиолинии с возможностью адаптивного управления частотным ресурсом

При проведении исследований полагалось, что канал связи является каналом прерывистой связи с быстрыми и медленными замираниями, а также неоднородными условиями сигнальной и помеховой обстановки на рабочих частотах. В качестве режима помехозащиты был выбран режим медленной ППРЧ. Предполагалось, что статистические характеристики линии радиосвязи с ППРЧ определяются характеристиками сигналов и помех на рабочих частотах.

В работе [15] приведены результаты исследований функций распределений огибающих сигналов и помех на рабочей частоте радиолинии. В частности показано, что они описываются законами Релея

$$\begin{aligned} W(U_c) &= \frac{2 U_c}{U_{c \text{эфф}}^2} \exp\left(-\frac{U_c^2}{U_{c \text{эфф}}^2}\right); \\ W(U_n) &= \frac{2 U_n}{U_{n \text{эфф}}^2} \exp\left(-\frac{U_n^2}{U_{n \text{эфф}}^2}\right), \end{aligned} \quad (1)$$

и Райса

$$W(U_c) = \frac{2 U_c}{U_{c \text{эфф}}^2} \exp\left(-\frac{U_c^2 + U_{c \text{ср}}^2}{U_{c \text{эфф}}^2}\right) I_0\left(\frac{2 U_c U_{c \text{ср}}}{U_{c \text{эфф}}^2}\right); \quad (2)$$

$$W(U_n) = \frac{2 U_n}{U_{n \text{эфф}}^2} \exp\left(-\frac{U_n^2 + U_{n \text{ср}}^2}{U_{n \text{эфф}}^2}\right) I_0\left(\frac{2 U_n U_{n \text{ср}}}{U_{n \text{эфф}}^2}\right), \quad (3)$$

где $U_{c \text{эфф}}$, $U_{n \text{эфф}}$ — эффективные напряжения флюктуирующей составляющей сигнала и помехи соответственно, являющиеся параметрами распределений; $U_{c \text{ср}}$, $U_{n \text{ср}}$ — амплитуды регулярной составляющей сигнала и помехи; I_0 — функция Бесселя нулевого порядка.

Флюктуация фазы сигнала и помехи φ характеризуется равномерным распределением в интервале от 0 до 2π :

$$W(\varphi) = \frac{1}{2\pi}. \quad (4)$$

Вместе с тем представленные распределения огибающих сигналов и помех справедливы на относительно коротких временных интервалах (до нескольких минут), на которых параметры распределения $U_{c \text{эфф}}$ ($U_{n \text{эфф}}$) можно считать постоянными. Заметно, что на более длительных временных интервалах параметры распределений сигналов и помех в законах Релея (Райса) являются случайной величиной с плотностью вероятности $W(U_{c \text{эфф}})$ и $W(U_{n \text{эфф}})$ соответственно.

Согласно работе [15] плотности вероятности $W(U_{c \text{эфф}})$, $W(U_{n \text{эфф}})$ описываются логарифмически нормальным законом, а выраженные в децибелах относительно 1 мкВ значения $U_{c \text{эфф}}$ и $U_{n \text{эфф}}$, характеризующие уровни сигналов и помех, являются случайными величинами y и x , плотности вероятности которых определяются нормальным законом:

$$W(y) = \frac{1}{\sqrt{2\pi}\sigma_y} \exp\left(-\frac{(y - \bar{y})^2}{2\sigma_y^2}\right); \quad (5)$$

$$W(x) = \frac{1}{\sqrt{2\pi}\sigma_x} \exp\left(-\frac{(x - \bar{x})^2}{2\sigma_x^2}\right), \quad (6)$$

где \bar{y} и \bar{x} , σ_y и σ_x — средние значения и среднеквадратические отклонения уровней сигналов, помех соответственно.

Для учета адаптивного управления частотным ресурсом радиолинии с ППРЧ введем понятие коэффициента использования частоты:

$$\alpha_i = \frac{\tau_{\Sigma i}}{T_a}, \quad (7)$$

где $\tau_{\Sigma i}$ — суммарный временной интервал работы радиолинии на i -й частоте ($i = 1, 2, \dots, m$); m — число рабочих частот; T_a — время анализа функционирования радиолинии с ППРЧ.

При проведении исследований введено ограничение: сумма коэффициентов использования частот в радиолинии с ППРЧ равна единице.

Коэффициенты использования частот определим в виде вектора использования рабочих частот $A = \|\alpha_1 \alpha_2 \dots \alpha_m\|$, характеризующего процесс управления частотно-временным ресурсом радиолинии с ППРЧ, под которым в работе понимаются частоты и относительное время использования каждой частоты на интервале анализа работы радиолинии с ППРЧ. Вектор использования рабочих частот определяет алгоритм управления частотным ресурсом радиолинии.

Предположим, что радиолиния функционирует на $m = 4$ частотах. Элементарный временной интервал работы радиолинии на одной частоте составляет величину τ_f , длительность анализа работы радиолинии $T_a = 17 \tau_f$, суммарные временные интервалы работы радиолинии на первой, второй, третьей и четвертой частотах составляют $\tau_{\Sigma 1} = 6\tau_f$, $\tau_{\Sigma 2} = 2\tau_f$, $\tau_{\Sigma 3} = 5\tau_f$, $\tau_{\Sigma 4} = 4\tau_f$ соответственно. Тогда в соответствии с выражением (7) коэффициенты использования частот в радиолинии с ППРЧ равны $\alpha_1 = \frac{6}{17}$, $\alpha_2 = \frac{2}{17}$, $\alpha_3 = \frac{5}{17}$, $\alpha_4 = \frac{4}{17}$.

Коэффициент использования частоты в радиолинии с ППРЧ может быть представлен различными способами. Учитывая, что при условии мгновенной перестройки радиолинии с частоты на частоту суммарное число перестроек в радиолинии $K_{\text{пер } \Sigma}$ определяется отношением длительности анализа работы радиолинии с ППРЧ к длительности элементарного временного интервала работы радиолинии на одной частоте

$$K_{\text{пер } \Sigma} = \frac{T_a}{\tau_f}, \quad (8)$$

а суммарное число перестроек радиолинии на i -ю частоту за время T_a определяется отношением суммарного временного интервала работы радиолинии на i -й частоте к величине τ_f :

$$K_{\text{пер } \Sigma i} = \frac{\tau_{\Sigma i}}{\tau_f}, \quad (9)$$

коэффициент использования i -й частоты, определяемый выражением (7), представим как

$$\alpha_i = \frac{K_{\text{пер } \Sigma i}}{K_{\text{пер } \Sigma}}. \quad (10)$$

Число перестроек радиолинии на i -ю частоту $K_{\text{пер } i}$ соответствует числу элементарных временных интервалов работы радиолинии на

одной частоте τ_f при передаче информации на i -й частоте

$$S_i = K_{\text{пер } i}, \quad i = 1, 2, \dots, m.$$

Поэтому другим представлением коэффициента использования частоты в радиолинии с ППРЧ является отношение числа временных интервалов S_i к общему числу элементарных временных интервалов работы S_0 , составляющих длительность анализа:

$$\alpha_i = \frac{S_i}{S_0}, \quad (11)$$

где $S_0 = \frac{T_a}{\tau_f}$.

В изложенной модели радиолинии с ППРЧ значения коэффициентов использования частот функционально определим величиной превышения уровня сигнала над уровнем помех

$$\alpha_i = f(z_i), \quad (12)$$

где $z_i = y_i - x_i$.

Таким образом, полученная модель помехозащищенной радиолинии полностью реализует возможность управления ее частотным ресурсом.

При этом для методов адаптивного управления справедливо неравенство

$$\frac{df(z_i)}{dz_i} > 0. \quad (13)$$

Выражения (5)—(13) определяют функциональную модель помехозащищенной радиолинии, обеспечивающую адаптивное управление ее частотным ресурсом в условиях неоднородной сигнальной и помеховой обстановки на рабочих частотах.

Разработанная модель явилась основой для получения статистических характеристик помехозащищенных радиолиний.

Статистические характеристики помехозащищенных радиолиний

Статистические характеристики радиолиний с ППРЧ определены функцией распределения превышения уровня сигнала над уровнем помех. Если в радиолинии с ППРЧ реализован равномерный закон использования частот, определяемый условием $\alpha_1 = \alpha_2 = \dots = \alpha_m = 1/m$, то функцию превышения уровня сигнала над уровнем помех в радиолинии с ППРЧ представим в виде

$$F_{\text{ППРЧ}}(z) = \int_{-\infty}^z W_f(s) ds. \quad (14)$$

Анализ выражения (14) позволяет выдвинуть гипотезу, которую сформулируем в следующей редакции:

равномерный закон использования рабочих частот в радиолинии с ППРЧ определяет идентичность функций распределения превышения уровня сигнала над уровнем помех в радиолинии и функции распределения превышения уровня сигнала над уровнем помех на сумме рабочих частот

$$F_{\text{ППРЧ равн}}(z) = F_f(z), \quad (15)$$

где $F_f(z) = \int_{-\infty}^z W_f(s) ds$.

Следовательно, при технической реализации в радиолинии закона использования частот, отличного от равномерного, функция распределения превышения уровня сигнала над уровнем помех в общем случае будет отличаться от функции распределения превышения уровня сигнала над уровнем помех на рабочих частотах.

Для доказательства данной гипотезы получен общий вид статистических плотностей вероятности превышения уровня сигнала над уровнем помех в радиолинии с ППРЧ $W_{\text{ППРЧ}}^*(z)$ при различных значениях коэффициентов использования рабочих частот $\alpha_1, \alpha_2, \dots, \alpha_m$, реализующих адаптивное управление частотным ресурсом радиолинии.

Для соблюдения необходимой точности анализируемых статистических плотностей $W_{\text{ППРЧ}}^*(z)$ предполагалось, что в линии радиосвязи программная перестройка осуществляется с использованием $m = 1000$ рабочих частот, плотность вероятности превышения уровня сигнала над уровнем помех на рабочих частотах описывается нормальным законом с математическим ожиданием $\bar{z} = 25$ дБ и среднеквадратическим отклонением $\sigma_z = 1$ дБ.

В целях обеспечения адаптивного управления в радиолинии с ППРЧ рабочие частоты представлены вариационным рядом, в котором частотам с большим индексом соответствует большее превышение уровня сигнала над уровнем помех

$$z_1 \leq z_2 \leq \dots \leq z_{1000}. \quad (16)$$

Гистограмма, характеризующая статистическую плотность вероятности превышения уровня сигнала над уровнем помех на 1000 рабочих частотах, приведена на рис. 1. На рис. 2 представлена гистограмма, характеризующая статистическую плотность вероятности превышения уровня сигнала над уровнем помех в радиолинии с ППРЧ при равномерном законе использования частот ($\alpha_1 = \dots = \alpha_{1000} = 0,001$).

На рис. 3 представлена гистограмма, характеризующая статистическую плотность

вероятности превышения уровня сигнала над уровнем помех при значениях коэффициентов использования частот $\alpha_1 = \dots = \alpha_{320} = 0, \alpha_{320} = \dots = \alpha_{1000} = 1,47 \cdot 10^{-3}$, то есть при реализации порогового метода управления.

На рис. 4 представлена гистограмма, характеризующая статистическую плотность веро-

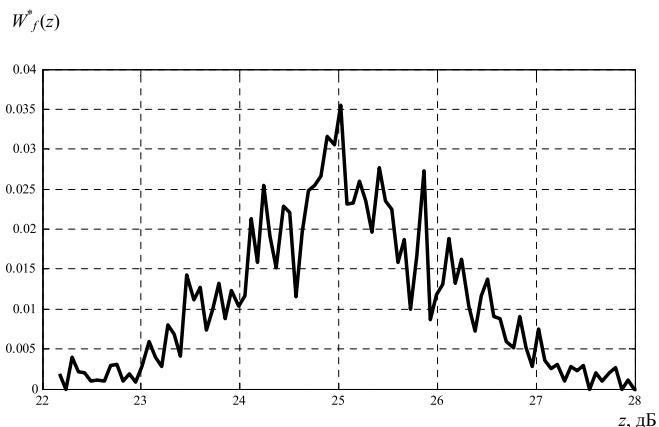


Рис. 1. Статистическая плотность вероятности превышения уровня сигнала над уровнем помех на рабочих частотах

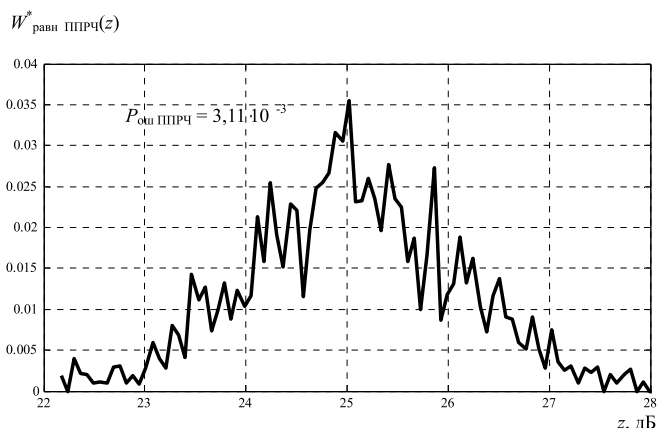


Рис. 2. Статистическая плотность вероятности превышения уровня сигнала над уровнем помех в радиолинии с ППРЧ при $\alpha_1 = \dots = \alpha_{1000} = 0,001$

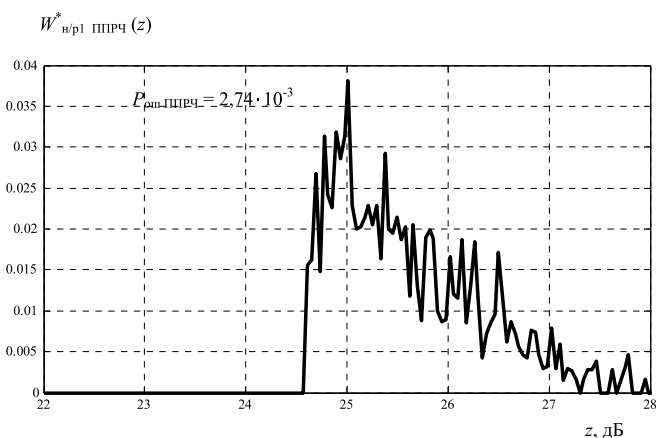


Рис. 3. Статистическая плотность вероятности превышения уровня сигнала над уровнем помех в радиолинии с ППРЧ при $\alpha_1 = \dots = \alpha_{320} = 0, \alpha_{320} = \dots = \alpha_{1000} = 1,47 \cdot 10^{-3}$

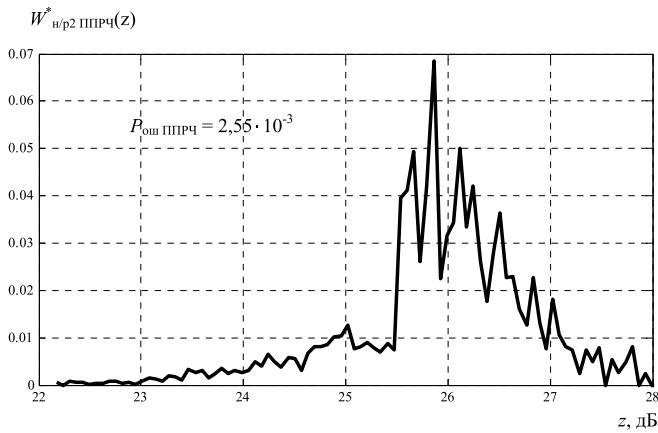


Рис. 4. Статистическая плотность вероятности превышения уровня сигнала над уровнем помех в радиолнии с ППРЧ при $\alpha_1 = \dots = \alpha_{320} = 2,5 \cdot 10^{-4}$, $\alpha_{320} = \dots = \alpha_{692} = 3,33 \cdot 10^{-4}$, $\alpha_{692} = \dots = \alpha_{1000} = 0,796$

ятности превышения уровня сигнала над уровнем помех радиолнии с ППРЧ при значении коэффициентов использования частот $\alpha_1 = \dots = \alpha_{320} = 2,5 \cdot 10^{-4}$, $\alpha_{320} = \dots = \alpha_{692} = 3,33 \cdot 10^{-4}$, $\alpha_{692} = \dots = \alpha_{1000} = 0,796$. Представленный в данном варианте закон использования частот в радиолнии с ППРЧ предусматривает преимущественную перестройку на частоты, на которых превышение уровня сигнала над уровнем помех имеет большие значения.

Результаты, представленные на рис. 1–4, получены методом имитационного моделирования. При этом первоначально, в соответствии с известными законами распределения уровней сигналов и помех на рабочих частотах $W^*f(z)$ моделировали сигнальную и помеховую обстановку на этих частотах. Далее осуществляли программную перестройку по законам использования частот, описываемых соответствующим вектором коэффициентов использования частот $A = \|\alpha_1 \alpha_1 \dots \alpha_m\|$. Затем собирали статистику превышений уровня сигнала над уровнем помех, после чего рассчитывали статистическую плотность вероятности. Вероятность ошибки в радиолнии с ППРЧ получена исходя из статистической плотности вероятности превышения уровня сигнала над уровнем помех в радиолнии

$$P_{\text{ош ППРЧ}} = \int_{-\infty}^{\infty} P_{\text{ош}}(z) W_{\text{ППРЧ}} d(z). \quad (16)$$

Заключение

Результаты проведенного исследования позволяют сделать следующие выводы.

1. При равномерном законе использования частот в помехозащищенных радиолниях

статистическая плотность вероятности превышения уровня сигнала над уровнем помех идентична статистической плотности вероятности превышения уровня сигнала над уровнем помех на рабочих частотах.

2. В случае равномерного закона использования рабочих частот в помехозащищенных радиолниях могут использоваться известные методы оценки их эффективности.

3. Применение адаптивного управления ресурсами радиолнии приводит к изменению их статистических характеристик, что обуславливает необходимость разработки дополнительных методов оценки их вероятностно-временных характеристик.

4. Применение управления ресурсами помехозащищенных радиолний позволяет повысить эффективность их функционирования.

5. Пороговый метод управления ресурсами помехозащищенных радиолний менее эффективен, чем адаптивное управление.

Направление дальнейшего исследования авторы связывают с разработкой методов оценки вероятностно-временных характеристик помехозащищенных радиолний с адаптивным управлением.

Список литературы

1. Дворников С. В., Манаенко С. С., Пшеничников А. В. Спектрально-эффективные сигналы с непрерывной фазой // Вестник Воронежского государственного технического университета. 2016. Т. 12, № 2. С. 87–93.
2. Дворников С. В., Пшеничников А. В., Манаенко С. С. Помехоустойчивая модель сигнала КАМ-16 с трансформированным созвездием // Информационные технологии. 2015. Т. 21, № 9. С. 685–689.
3. Дворников С. В., Манаенко С. С., Дворников С. С., Погорелов А. А. Синтез фазоманипулированных вейвлет-сигналов // Информационные технологии. 2015. Т. 21, № 2. С. 140–143.
4. Дворников С. В., Кузнецов Д. А., Кожевников Д. А., Пшеничников А. В., Манаенко С. С. Теоретическое обоснование синтеза ансамбля биортогональных сигналов с повышенной помехоустойчивостью // Вопросы радиоэлектроники. Серия: Техника телевидения. 2015. № 5 (22). С. 16–20.
5. Yen-Ming Chen, Yeong-Luh Ueng. Noncoherent Amplitude/Phase Modulated Transmission Schemes for Rayleigh Block Fading Channels // IEEE Trans. Com. — 2013. Vol. 61, N. 1. P. 128–135.
6. Pickholtz R. L., Shilling D. L., Milstein L. B. Theory of spread-spectrum communications // IEEE Trans. Com. 1982. Vol. COM-30, N. 5. P. 855–884.
7. Ray K. J., Wang B. Cognitive Radio Networking and Security: A Game-Theoretic View. Cambridge University Press, 2010. 618 p.
8. Пшеничников А. В. Интегральная модель радиолнии в конфликтной ситуации // Информация и космос. 2016. № 4. С. 39–45.
9. Скляр Б. Цифровая связь. Теоретические основы и практическое применение / Пер. с англ.; Под ред. А. В. Назаренко. М.: Вильямс, 2003. 1104 с.
10. Athanasios D., Kalivas G. SNR estimation for low bit rate OFDM systems in AWGN channel // Proceedings of the ICN/ICONS/MCL. 2006. 198 p.
11. Chandler G. Development and Evaluation of an LPI Figure of Merit for Direct-Sequence and Frequency-Hopping Systems // IEEE Milcom'89, Commun., Conf. Boston. 1989. Vol. 2. P. 20–23.

12. **Волинская А. В., Сапожников И. В.** Моделирование линейного мультипараметрического модема для каналов передачи информации с низким отношением сигнал/помеха // Информационные технологии. 2015. Т. 21, № 8. С. 590—594.

13. **Борисов В. И., Зинчук В. М., Лимарев А. Е.** и др. Помехозащищенность систем радиосвязи / Под ред. В. И. Борисова. М.: Радио и связь, 2000. 384 с.

14. **Дворников С. В., Пшеничников А. В., Манаенко С. С., Дворников С. С.** Метод формирования помехозащищенных сигнальных конструкций // Информационные технологии. 2017. Т. 23, № 9. С. 669—676.

15. **Комарович В. Ф., Сосунов В. Н.** Случайные помехи и надежность КВ связи. М.: Связь, 1977. 136 с.

S. V. Dvornikov, Professor, e-mail: practicsv@yandex.ru,
A. V. Pshenichnikov, Assistant Professor, e-mail: siracooz77@mail.ru,
S. S. Manaenko, Assistant Professor, e-mail: manaenkoss@mail.ru,
 Military Communications Academy, St. Petersburg

Statistical Characteristics of Noise-Free Radio Links with Frequency Resource Management

The article summarizes the results of a study of the statistical characteristics of noise-immune radiolines operating under conditions of heterogeneity of the signal and interference conditions at operating frequencies. As an estimation parameter, the amount of excess of the signal level above the interference level was chosen, which determines the probabilistic and temporal characteristics of the information transfer process. The notion of the frequency utilization factor determining the radio resource management strategy has been introduced. The dependence of the frequency utilization factor on the signal level exceeding the level of interference is determined. The notion of adaptive control of the frequency-time resource of an interference-free radio link is introduced. A hypothesis about the influence of control methods on the statistical characteristics of noise-protected radio links is formulated. For its justification, an imitation model of an interference-free radio link has been constructed, on the basis of which statistical functions for distributing the signal level over the level of interference have been obtained. The development of the simulation model is based on the probabilistic and temporal characteristics that determine the change in the energy parameters of radio signals and interference at the operating frequencies of the radio link. The peculiarity of the developed approach is the consideration of the methods of resource control of the radio link in the development of the stochastic model of the radio channel. It is substantiated that threshold management methods are less effective than adaptive control methods. It is shown that the implementation of the uniform law of the use of radio link resources does not change the functional dependence of the probabilistic-temporal characteristics, therefore, with the given control method, known algorithms for calculating the efficiency of the me-protected radio links are practically implemented. Control methods, other than uniform, determine the change in the distribution law characterizing the signal level exceeding the level of interference in the radio link; therefore, with such methods of determining the type of control, known methods for estimating the efficiency of the functional models of radio links are inadequate for the calculation conditions.

Keywords: anti-interference mode of operation, adaptive control, frequency resource, statistical characteristics

DOI: 10.17587/it.25.35-40

References

- Dvornikov S. V., Manaenko S. S., Pshenichnikov A. V.** *Spektral'no-jeffektivnye signaly s nepreryvnoj fazoj* (Spectral-effective signals with continuous phase), *Vestnik voronezhskogo gosudarstvennogo tekhnicheskogo universiteta*, 2016, vol. 12, no. 2, pp. 87—93 (in Russian).
- Dvornikov S. V., Pshenichnikov A. V., Manaenko S. S.** *Pomehoustojchivaja model' signala QAM-16 s transformirovannym sozvezdijem* (Interference-signal model QAM-16 with a transformed constellation), *Informacionnye Tehnologii*, 2015, vol. 21, no. 9, pp. 685—689 (in Russian).
- Dvornikov S. V., Manaenko S. S., Dvornikov S. S., Pogorelov A. A.** *Sintez fazomanipulirovannyh vejrlet-signalov* (Synthesis of phase-shift wavelet signals), *Informacionnye Tehnologii*, 2015, vol. 21, no. 2, pp. 140—143 (in Russian).
- Dvornikov S. V., Kuznecov D. A., Kozhevnikov D. A., Pshenichnikov A. V., Manaenko S. S.** *Teoreticheskoe obosnovanie sinteza ansambli biortogonal'-nyh signalov s povyshennoj pomehoustojchivost'ju* (Theoretical justification for the synthesis of an ensemble of biorthogonal signals with increased noise immunity), *Voprosy radioelektroniki. Serija: Tehnika televidenija*, 2015, no. 5 (22), pp. 16—20 (in Russian).
- Yen-Ming Chen, Yeong-Luh Ueng.** Noncoherent Amplitude/Phase Modulated Transmission Schemes for Rayleigh Block Fading Channels, *IEEE Trans. Com.*, 2013, vol. 61, no. 1, pp. 128—135.
- Pickholtz R. L., Shilling D. L., Milstein L. B.** Theory of spread-spectrum communications, *IEEE Trans. Com.*, 1982, vol. COM-30, no. 5, pp. 855—884.
- Ray K. J., Wang B.** Cognitive Radio Networking and Security: A Game-Theoretic View. Cambridge University Press, 2010, 618 p.
- Pshenichnikov A. V.** *Integral'naja model' radiolinii v konfliktnoj situacii* (Integral model of a radio link in a conflict situation), *Informacija i Kosmos*, 2016, no. 4, pp. 39—45 (in Russian).
- Sklyar B.** *Cifrovaja svjaz'. Teoreticheskie osnovy i prakticheskoe primenenie* (Digital communication. Theoretical foundations and practical application), Moscow, Vil'jams, 2003, 1104 p. (in Russian).
- Athanasios D., Kalivas G.** SNR estimation for low bit rate OFDM systems in AWGN channel, *Proceedings of the ICN/ICONS/MCL*, 2006, 198 p.
- Chandler G.** Development and Evaluation of an LPI Figure of Merit for Direct-Sequence and Frequency-Hopping Systems, *IEEE Milcom'89, Commun., Conf. Boston*, 1989, vol. 2, pp. 20—23.
- Volynskaya A. V., Sapozhnikov I. V.** *Modelirovanie linejnogo mul'tiparametricheskogo modema dlya kanalov pedachi informacii s nizkim otnošeniem signal/pomekha* (Modelling of the Linear Multiparametrical Modem for Channels of Information Transfer with the Low Relation a Signal/Noise), *Informacionnye Tehnologii*, 2015, vol. 21, no. 8, pp. 590—594 (in Russian).
- Borisov V. I., Zinchuk V. M., Limarev A. E.** i dr. *Pomehozashishhennost' sistem radiosvjazi* (Interference immunity of radio communication systems), Moscow, Radio i svjaz', 2000, 384 p. (in Russian).
- Dvornikov S. V., Pshenichnikov A. V., Manaenko S. S., Dvornikov S. S.** *Metod formirovanija pomehozashishhennyh signal'nyh konstrukcij* (A method of forming multiposition jamproof signal constructions), *Informacionnye Tehnologii*, 2017, vol. 23, no. 9, pp. 669—676 (in Russian).
- Komarovich V. F., Sosunov V. N.** *Sluchajnye pomehi i na-dezhnost' KV svjazi* (Random interference and reliability of short-wave coupling), Moscow, Svjaz', 1977, 136 p. (in Russian).

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИМЕДИЦИНСКИХ СИСТЕМАХ

INFORMATION TECHNOLOGIES IN BIOMEDICAL SYSTEMS

УДК 004.855.5

DOI: 10.17587/it.25.41-45

К. Ш. Исмаилова, канд. техн. наук, доц., e-mail: is_kamalya@yahoo.com,
Азербайджанский государственный университет нефти и промышленности, Баку

Применение различных методов оптимизации при расчете погрешности нейронной сети для диагностирования нервно-мышечных заболеваний

Проводится сравнение и выбор оптимального метода при расчете погрешности обучаемой нейронной сети для решения задачи диагностирования нервно-мышечных заболеваний. Выбрана совокупность значений электромиографических сигналов нормального и патологических состояний пациентов. Результаты тестирования оцениваются по абсолютным и относительным значениям ошибок на пяти интервалах для выявления лучшего метода оптимизации. Для поставленной задачи лучшие результаты получены при применении метода оптимизации BFGS.

Ключевые слова: нейронные сети, методы оптимизации, нервно-мышечные заболевания, абсолютная погрешность, относительная погрешность, градиентный спуск, модифицированный Par Tap, сопряженные градиенты, BFGS

Введение

Биомедицинские сигналы представляют собой электрические сигналы, генерируемые любым органом, отражающие функциональное состояние исследуемого органа. Электромиография (ЭМГ) является техникой для оценки и регистрации электрической деятельности скелетных мышц. Основные интересы исследований лежат в области клинической, а также биомедицинской инженерии. Достижения в области инженерии расширили возможности электромиографии за пределы традиционных диагностических приложений и позволяют использовать их в различных областях.

Традиционно нейрофизиолог может получить доступ к потенциалам мышечных единиц (ПМЕ) и оценивать информацию по полученным паттернам и параметрам сигналов. ПМЕ из разных моторных нейронов перекрывают друг друга, и это приводит к формированию интерференционной картины, что затрудняет точное определение отдельных паттернов. По этой причине были разработаны различные алгоритмы количественного компьютерного анализа ЭМГ. К числу таких алгоритмов относятся и алгоритмы искусственного интеллекта — нейронные, нейронечеткие, гибридные и т. д. Методы искусственного интеллекта эф-

фективно используются в медицинских диагностических инструментах для повышения точности диагностики и предоставления дополнительных знаний.

Искусственная нейронная сеть (ИНС) представляет собой систему соединенных и взаимодействующих между собой простых процессоров. Персептрон является одной из первых моделей нейронных сетей, способен обучаться и основывается при этом на статистических данных. Информация об образе распределяется по весовым коэффициентам (аналогично коэффициентам системы уравнений), которые в совокупности описывают те или иные фрагменты образа.

В информационных источниках существует много подходов для обработки ЭМГ сигналов с использованием этих алгоритмов. В работе [1] использованы различные виды методов обучения для классификации сигналов ЭМГ. Модель автоматически классифицирует сигналы ЭМГ как нормальную, миопатическую и нейропатическую. Чтобы извлечь полезную информацию из ЭМГ, используют различные способы экстракции признаков, такие как дискретное вейвлет-преобразование (DWT) и авторегрессивное моделирование (AR). Адаптивный нейронечеткий вывод системы (ANFIS) с гибридным алгоритмом обучения,

метод опорных векторов (SVM) и нечеткая модификация метода опорных векторов (FSVM) позволяют классифицировать ЭМГ-сигналы путем сравнения их по признаку точности.

В работе [2] предлагается классификация нервно-мышечных заболеваний из сигналов ЭМГ на основе сочетания различных методов извлечения объектов и типов классификаторов. Комбинация вейвлет-преобразования (WT) и метода опорных векторов улучшает точность классификации по сравнению с другими комбинациями.

Исследователи в работе [3] контролировали электрофизиологическую активность мышц, используя биологическую обратную связь, полученную из сигналов электромиограмм, собранных в соответствующих точках иннервации. Предлагается комплексный метод выявления нервно-мышечных заболеваний у субъекта и стратегия непрерывной терапевтической оценки с использованием матрицы оценки реабилитации. Инструмент принятия решений прошел обучение с использованием широкого спектра физиологических данных, включающих различную степень миопатии и невропатии от начальных стадий к острым. Статистические, спектральные и кепстральные признаки, извлеченные из ЭМГ, использовались для обучения классификатора нейронной сети с каскадной корреляцией для оценки заболеваний. Для оценки качества работы ИНС используются такие статистические показатели, как точность, специфичность и чувствительность. Диагностический выход классификатора составляет 91,2 % точности, 85,3 % специфичности и 91,35 % чувствительности. Стратегия также была расширена за счет включения изотонических сокращений в дополнение к статическим изометрическим сокращениям. Эта комплексная стратегия предлагается как помощь врачам при планировании процедуры лечения, чтобы максимально повысить терапевтическую ценность процесса реабилитации.

Методы вычислительной разведки (бесконтактный доступ к компьютерной информации, циркулирующей в информационной системе) и машинного обучения, такие как искусственные нейронные сети (ANN), служат мощным инструментом для анализа сигналов ЭМГ и создания оптимальных схем миоэлектрического контроля для протезов. В исследовании [4] анализируется эффективность четырех различных нейронных сетевых структур (прямое, повторяющееся, встречное распространение и самоорганизующаяся карта), которым было

поручено классифицировать скорость ходьбы при использовании входов ЭМГ из 14 различных мышц ног. Эксперименты свидетельствуют о том, что самоорганизующиеся карты нейронных сетей способны классифицировать скорость ходьбы с точностью более 99 %.

Метод опорных векторов — широко используемый метод машинного обучения со многими приложениями для классификации биомедицинских сигналов. В работе [5] предложена новая модель PSO-SVM, которая гибридизировала методы оптимизации роя частиц (PSO) и SVM для улучшения точности классификации сигналов ЭМГ. Этот механизм оптимизации включает настройку параметров ядра в процедуре обучения SVM, что существенно влияет на точность классификации. Эксперименты проводили на основе сигнала ЭМГ для классификации на нормальный, нейрогенный или миопатический. Полученные результаты, очевидно, подтверждают превосходство метода SVM по сравнению с обычными методами машинного обучения и предполагают, что предлагаемая система классификации PSO-SVM дает дополнительные существенные улучшения с точки зрения точности классификации.

В работе [6] представлена методология глобальной оптимизации нейронной сети. Целью является одновременная оптимизация многоуровневых весовых и архитектурных характеристик персептрона (MLP) для создания топологий с несколькими соединениями и высокой степенью классификации для любых наборов данных. Этот подход сочетает в себе преимущества имитационной модели и алгоритма обучения *backpropagation* для создания автоматического процесса проектирования сетей с высокой степенью классификации и низкой сложностью. Экспериментальные результаты, полученные для четырех задач классификации и одной задачи прогнозирования, оказались лучше, чем полученные с помощью наиболее часто используемых методов оптимизации.

В статье [7] дается обзор многочисленных методов, доступных для распознавания структур движения сигналов ЭМГ как для изотонических, так и для изометрических сокращений. Различные методы анализа сигналов сравниваются, результаты иллюстрируют их применимость в настройках реального времени. Это представляет интерес для исследователей, которые хотели бы выбрать наиболее подходящую методологию для классификации движений, особенно при различных типах сокращений мышц. Для извлечения признаков

в качестве основной принимается функция плотности вероятности (PDF) сигналов ЭМГ. После краткого описания различных методов предварительной обработки, извлечения признаков и классификации сигналов ЭМГ проведем их сравнение с точки зрения их производительности.

Метод расчета погрешностей нейронных сетей

Известно, что важным и самым доступным индикатором нейронно-сетевых систем являются погрешности вычислений. Самые распространенные виды погрешностей, характеризующиеся легкостью вычисления, — это абсолютные и относительные погрешности (ошибки), которые вычисляются по следующим формулам [8]:

- абсолютная погрешность

$$P_A = P_n - P_3;$$

- относительная погрешность

$$P_O = \frac{P_n - P_3}{P_3} \cdot 100,$$

где P_n — ответ (диагноз) сети; P_3 — истинный диагноз (входное значение).

Для эксперимента выбраны электромиографические сигналы в норме и в патологическом состоянии: при карпальном туннельном синдроме, кубитальном туннельном синдроме и демиелинизирующей нейропатии. В качестве входных параметров сети выбраны данные 16 точек измерения для диагностирования. Использованы 26 нормальных и 34 патологических сигнала для создания обучающей совокупности сети.

Выход сети OUTPUT содержит четыре элемента:

1 — норма; 2 — полиневропатия; 3 — карпальный туннельный синдром; 4 — кубитальный туннельный синдром.

Компьютерная реализация эксперимента

Компьютерная реализация эксперимента проводилась в программной среде NeuroPro 0.25, и для обучения сети были выбраны методы оптимизации: градиентный спуск, модифицированный *Par Tan*, сопряженные градиенты, BFGS.

Фрагменты результатов первого эксперимента и расчет погрешности даны в табл. 1–4.

Таблица 1

Фрагмент результата прогнозирования с помощью многослойного персептрона методом оптимизации "градиентный спуск"

Заданное значение	Полученное значение	Абсолютная ошибка	Относительная ошибка
1	1,2463	0,2463	24,6313
1	1,2538	0,2538	25,3780
1	1,1084	0,1084	10,8361
1	1,2290	0,2290	22,9043
2	1,7661	-0,2339	-11,6943
2	2,1642	0,1642	8,2084
2	2,2029	0,2029	10,1441
3	2,7355	-0,2645	-8,8180
3	2,7288	-0,2712	-9,0384
4	4,2662	0,2662	6,6550
4	3,8891	-0,1109	-2,7716
4	4,2564	0,2564	6,4100

Таблица 2

Фрагмент результата прогнозирования с помощью многослойного персептрона методом оптимизации "модифицированный *Par Tan*"

Заданное значение	Полученное значение	Абсолютная ошибка	Относительная ошибка
1	1,2988	0,2988	29,8753
1	1,2725	0,2725	27,2539
1	1,1259	0,1259	12,5930
1	1,2302	0,2302	23,0237
2	1,7203	-0,2797	-13,9873
2	2,1983	0,1983	9,9146
2	2,2777	0,2777	13,8863
3	2,7718	-0,2282	-7,6082
3	2,7404	-0,2596	-8,6537
4	3,9893	-0,0107	-0,2681
4	3,7240	-0,2760	-6,9007
4	3,9131	-0,0869	-2,1725

Таблица 3

Фрагмент результата прогнозирования с помощью многослойного персептрона методом оптимизации "сопряженные градиенты"

Заданное значение	Полученное значение	Абсолютная ошибка	Относительная ошибка
1	1,2193	0,2193	21,9344
1	1,2832	0,2832	28,3217
1	1,0870	0,0870	8,7026
1	1,1494	0,1494	14,9437
2	1,7178	-0,2822	-14,1088
2	2,1470	0,1470	7,3521
2	2,8329	0,8329	41,6473
3	2,8329	-0,1671	-5,5685
3	2,7245	-0,2755	-9,1838
4	4,2747	0,2747	6,8685
4	3,9931	-0,0069	-0,1729
4	4,2719	0,2719	6,7966

Таблица 4

Фрагмент результата прогнозирования с помощью многослойного персептрона методом оптимизации "BFGS"

Заданное значение	Полученное значение	Абсолютная ошибка	Относительная ошибка
1	1,2903	0,2903	29,0290
1	1,2729	0,2729	27,2865
1	1,1298	0,1298	12,9763
1	1,2034	0,2034	20,3440
2	1,7179	-0,2821	-14,1065
2	2,1506	0,1506	7,5296
2	2,2543	0,2543	12,7168
3	2,7295	-0,2705	-9,0155
3	2,7724	-0,2276	-7,5863
4	4,0351	0,0351	0,8785
4	4,0744	0,0744	1,8596
4	4,0764	0,0764	1,9106

Для сравнения методов оптимизации использовано число значений ошибок, попавших в выбранный интервал. Выбрано пять интервалов для абсолютных ошибок. Значение интервалов и число абсолютных ошибок, попавших в соответствующий интервал, приведены в табл. 5.

Для сравнения методов оптимизации с использованием относительных ошибок тоже выбрано пять интервалов. Значения интервалов и число относительных ошибок, попавших в соответствующие интервалы, даны в табл. 6.

Таблица 5

Сравнение абсолютных ошибок

№ эксперимента	Число абсолютных ошибок в соответствующем интервале				
	<0,05	0,05...0,09	0,1...0,19	0,2...0,29	>0,3
1	0	0	3	9	0
2	0	2	2	8	0
3	1	1	3	7	0
4	1	2	2	7	0

Таблица 6

Сравнение относительных ошибок

№ эксперимента	Число относительных ошибок в соответствующих интервалах				
	< 1	1...9	10...19	20...29	> 30
1	0	6	3	3	0
2	1	5	3	3	0
3	1	6	2	2	1
4	1	5	3	3	0

Сравнение проводили с точки зрения того, в каком из экспериментов было меньше ошибок с большими значениями. Так, результаты 1-го и 2-го экспериментов дают большое число ошибок с большими значениями (табл. 5). Несмотря на то, что число ошибок в результатах 3-го и 4-го экспериментов не очень различаются, выбран 4-й вариант с учетом малого числа ошибок в 3-м интервале.

Аналогичные рассуждения можно провести и для данных табл. 6. Здесь значения 2-го и 4-го экспериментов совпадают, но выбран 4-й вариант, так как в табл. 5 в этом эксперименте было получены наилучшие результаты.

Заключение

Сравнение методов оптимизации с помощью погрешностей сети дает возможность сделать вывод, что для поставленной задачи лучшие результаты получены при применении метода оптимизации BFGS.

Такой подход облегчает работу экспериментатора в выборе метода оптимизации при работе с нейронными сетями для задачи диагностирования нервно-мышечных заболеваний с точки зрения оценки погрешностей сети.

Список литературы

1. **Akhila Devi B. V., Priyadharsini S. S.** Diagnosis Of Neuromuscular Disorders Using Softcomputing Techniques // International Journal of Soft Computing and Engineering (IJSCE). 2013. Vol. 3, Iss. 5. P. 105–111.
2. **Kehri V., Ingle R., Awale R., Oimbe S.** Techniques of EMG signal analysis and classification of Neuromuscular diseases // Advances in Intelligent Systems Research. Vol. 137. P. 485–491.
3. **Sailesh Conjeti, Bijay Kumar Rout.** Strategy For Electromyography Based Diagnosis Of Neuromuscular Diseases For Assistive Rehabilitation // International Journal on Bioinformatics & Biosciences (IJBB). 2013. Vol. 3, N. 3. P. 25–39.
4. **Alayna Kennedy, Rory Lewis.** Optimization Of Neural Network Architecture For Biomechanic Classification Tasks With Electromyogram Inputs // International Journal of Artificial Intelligence and Applications (IJAIA). 2016. Vol. 7, N. 5. P. 1–16.
5. **Subasi A.** Classification of EMG Signals Using PSO optimized SVM for Diagnosis of Neuromuscular Disorders // Computers in Biology and Medicine. 2013. N. 43. P. 576–586.
6. **Ludermir T. B., Yamazaki A., Zanchettin C.** An Optimization Methodology for Neural Network Weights and Architectures // IEEE Transactions On Neural Networks. 2006. Vol. 17, N. 6. P. 1452–1459.
7. **Nurhazimah N.** et all. A Review of Classification Techniques of EMG Signals during Isotonic and Isometric Contractions // Sensors. 2016. N. 16. P. 1304–1332.
8. **Куликов Л. К., Быкова Н. М., Привалов Ю. А., Навтнович Н. А., Шабанова О. Г.** Диагностика гиперальдостеронизма Т больных с инциденталомиями надпочечников при помощи нейронных сетей // Фундаментальные исследования. 2010. № 3. С. 88–94. URL: <http://www.fundamental-research.ru/ru/article/view?id=6659> (дата обращения: 30.02.2018).

Application of Various Optimization Methods for Calculating the Neural Network Error for the Diagnosis of Neuromuscular Diseases

The proposed work is devoted to the calculation of the error of the neural network for comparison and the choice of the optimal method for teaching this problem. Analysis of the literature shows that the problem of finding the optimal number of neurons of the hidden layer does not have a unique solution due to the lack of an established methodology. There are four specific limitations that distinguish the training of a neurocomputer from common optimization tasks: the astronomical number of parameters, the need for high parallelism in learning, the multicriteria nature of the problems being solved, the need to find a sufficiently wide area in which the values of all minimized functions are close to minimal. It is known that the most important and most accessible indicator of such systems are the results of errors. The most common of the errors, and at the same time, the absolute and relative errors that are easy to calculate. The output of the OUTPUT network contains four elements: 1 is the norm; 2 — polyneuropathy; 3 — carpal tunnel syndrome; 4 — a cubital tunnel syndrome. Computer implementation of the experiment was carried out in the NeuroPro 0.25 software environment and optimization methods were chosen for network training: Gradient descent, Modified Par Tan, Conjugate gradients, BFGS. To compare the optimization methods, the number of error values within the selected interval is used. Five intervals for absolute errors are selected. To compare optimization methods using relative errors, five intervals are chosen. Comparison of optimization methods with the help of network errors makes it possible to conclude that for this task the best results were obtained using the BFGS optimization method.

Keywords: neural network errors, optimization methods, neuromuscular diseases, absolute error, relative error, gradient descent, modified Par Tan, conjugate gradients, BFGS

DOI: 10.17587/it.25.41-45

References

1. Akhila Devi B. V., Priyadharsini S. S. Diagnosis Of Neuromuscular Disorders Using Softcomputing Techniques, *International Journal of Soft Computing and Engineering (IJSC)*, 2013, vol. 3, iss. 5, pp. 105—111.
2. Kehri V., Ingle R., Awale R., Oimbe S. Techniques of EMG signal analysis and classification of Neuromuscular diseases, *Advances in Intelligent Systems Research*, vol. 137, pp. 485—491.
3. Sailesh Conjeti, Bijay Kumar Rout. Strategy For Electromyography Based Diagnosis Of Neuromuscular Diseases For Assistive Rehabilitation, *International Journal on Bioinformatics & Biosciences (IJBB)*, 2013, vol. 3, no.3, pp. 25—39.
4. Alayna Kennedy, Rory Lewis. Optimization Of Neural Network Architecture For Biomechanic Classification Tasks With Electromyogram Inputs. *International Journal of Artificial Intelligence and Applications (IJAA)*, 2016, vol. 7, no. 5, pp. 1—16
5. Subasi A. Classification of EMG Signals Using PSO optimized SVM for Diagnosis of Neuromuscular Disorders, *Computers in Biology and Medicine*, 2013, no. 43, pp. 576—586.
6. Ludermir T. B., Yamazaki A., Zanchettin C. An Optimization Methodology for Neural Network Weights and Architectures, *IEEE Transactions On Neural Networks*, 2006, vol. 17, no. 6, pp. 1452—1459.
7. Nurhazimah N. et al. A Review of Classification Techniques of EMG Signals during Isotonic and Isometric Contractions. *Sensors*, 2016, no. 16, pp. 1304—1332.
8. Kulikov L. K., Bykova N. M., Privalov Ju. A., Navtanovich N. A., Shabanova O. G. Diagnostika giperaldosteronizma T bolnyh s incidentalomami nadpochechnikov pri pomoshhi neyronnyh setey (Diagnosis of hyperaldosteronism in patients with adrenal incidents with neural networks), *Fundamentalniye issledovaniya*, 2010, vol. 3, pp. 88—94 (in Russian), available at: <http://www.fundamental-research.ru/ru/article/view?id=6659> (date of access: 30.02.2018) (in Russian).

Е. В. Конопацкий, канд. техн. наук, доц., e-mail: e.v.konopatskiy@mail.ru,
Донбасская национальная академия строительства и архитектуры, г. Макеевка

Аппроксимация геометрических объектов с помощью дуг кривых, проходящих через наперед заданные точки

Рассматривается способ аппроксимации геометрических объектов многомерного аффинного пространства с помощью дуг алгебраических кривых, проходящих через наперед заданные точки, который позволяет существенно уменьшить аналитическое описание многофакторных процессов и явлений. Приводится пример использования предложенного способа аппроксимации для аналитического описания геометрической модели распределения прочностных характеристик по всему объему бетонной колонны.

Ключевые слова: аппроксимация, геометрический объект, геометрическое моделирование, дуга кривой, наперед заданные точки, многомерная интерполяция, факторы влияния, функция отклика, точечное уравнение, точность аппроксимации, коэффициент детерминации

Введение

В инженерной практике и научной работе постоянно возникает необходимость моделирования и аналитического описания многофакторных процессов и явлений на основе экспериментальных и статистических данных для их анализа, прогнозирования и оптимизации результатов моделирования. Для решения этой задачи широко используются методы аппроксимации и интерполяции. Общая концепция моделирования геометрических объектов методом многомерной интерполяции применительно к решению задач моделирования и оптимизации многофакторных процессов и явления была изложена в работе [1]. Однако итоговый результат моделирования таких объектов, включающий в себя большое количество исходных данных, может иметь значительный размер, что с одной стороны не является неразрешимой проблемой, учитывая мощность современной вычислительной техники, но с другой стороны, это значительно затрудняет анализ и дальнейшее использование результатов моделирования. Одним из выходов из этой ситуации является использование аппроксимации геометрических объектов. При этом следует учитывать сложность визуального восприятия многомерного пространства,

что приводит к необходимости использования не зрительной, но логической наглядности, основанной на методах обобщения и аналогии.

В соответствии с определением [2] под аппроксимацией следует понимать замену одних математических объектов другими, в том или ином смысле близкими к исходным. В данном случае под аппроксимацией будем понимать замену одних геометрических объектов другими, имеющими одинаковое число текущих параметров с исходными объектами. К примеру, дуга кривой как однопараметрическое множество точек имеет один текущий параметр, и аппроксимирующий ее геометрический объект также должен быть однопараметрическим множеством точек. Аналогичным образом отсек поверхности как двухпараметрическое множество точек аппроксимируется другим отсеком поверхности и т. д.

На данный момент существует достаточно большое число методов аппроксимации, классифицированных в работе [3]. Существенным их недостатком является сложность обобщения на многомерное пространство, в котором можно моделировать многофакторные процессы и явления практически не имеющие ограничений по числу моделируемых факторов. С практической точки зрения широкое распространение получил метод наименьших ква-

дратов, на котором основаны другие методы (например, регрессионный анализ [4]). К недостаткам таких методов можно отнести слабую устойчивость по отношению к изменениям исходных данных. Среди алгоритмов машинной графики [5, 6] широкое распространение получили аппроксимационные алгоритмы на основе составных кривых, к которым относятся различные сплайны [7] и обводы [8–10]. Главным недостатком использования составных кривых для моделирования многофакторных процессов и явлений является использование равномерной сети точек и связанная с этим сложность использования в многомерном пространстве. Другим недостатком составных кривых является требование к порядку гладкости стыковки дуг между собой, хотя этот недостаток применительно к геометрическому моделированию процессов и явлений можно считать весьма условным. Также к недостаткам составных кривых следует отнести невозможность прогнозирования поведения функции отклика в случае, когда факторы влияния выходят за пределы исследуемой области. Резюмируя, можно сделать вывод о том, что любой метод моделирования имеет свои преимущества и недостатки. Наибольшее же значение имеет целесообразность применения того или иного метода для решения конкретной практической задачи.

В работе [1] для моделирования геометрических объектов методом многомерной интерполяции используются дуги алгебраических кривых, проходящих через наперед заданные точки, модифицированные на основе полиномов Бернштейна [11, 12]. Эти же дуги кривых могут быть эффективно использованы для аппроксимации геометрических объектов многомерного аффинного пространства. Тогда задача сводится к определению узловых точек аппроксимации. В некоторых случаях их можно выделить из имеющихся узлов интерполяции.

Модель распределения прочностных характеристик в бетонной колонне, полученная методом многомерной интерполяции

Рассмотрим в качестве примера моделирование распределения прочностных характеристик по всему объему бетонной колонны. С точки зрения геометрического моделирования этот процесс можно представить в виде гиперповерхности отклика, принадлежащей четырехмерному пространству, которая опре-

деляется тремя факторами, влияющими на процесс. В качестве факторов влияния используются координаты x , y и z , определяющие положение искомой точки по всему объему бетонной колонны, а в качестве функции отклика — показатели соотношения характеристик бетона к показателям характеристик стандартных образцов в процентах. Изначально постановка задачи и первый вариант ее решения были опубликованы в работах [11, 13]. Однако предложенное решение основывалось на том аппарате моделирования, который был развит в то время и имело ряд ограничений. Более прогрессивным и качественным оказалось решение с использованием метода многомерной интерполяции и дуг кривых, проходящих через наперед заданные точки, предложенное в работе [14], в соответствии с которым процесс моделирования был разбит на три этапа.

1 этап. Определение направляющих линий 1-го яруса (рис. 1) с помощью дуг кривых 4-го порядка, проходящих через пять наперед заданных точек:

$$M_{11} = M_{15} = af_1 + be_1 + cd_1 + de_1 + ef_1;$$

$$M_{12} = M_{14} = ae_1 + bc_1 + cb_1 + dc_1 + ee_1;$$

$$M_{13} = ad_1 + bb_1 + ca_1 + db_1 + ed_1,$$

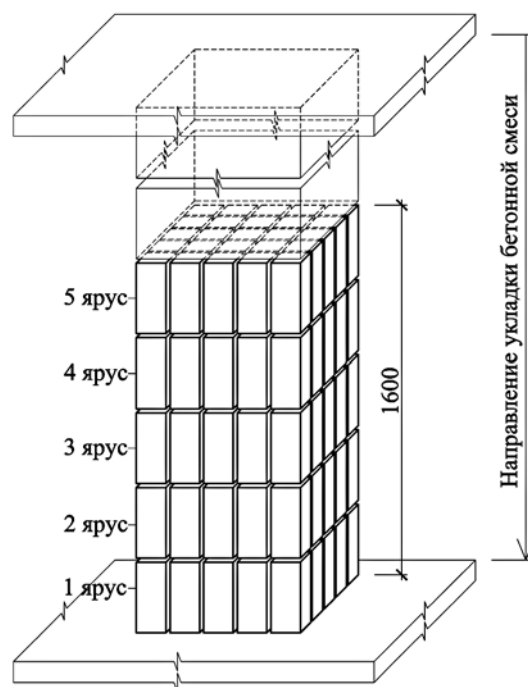


Рис. 1. Схема расположения вертикальных ярусов выпиленных образцов

где

$$a = \bar{u}^4 - \frac{13}{3}\bar{u}^3u + \frac{13}{3}\bar{u}^2u^2 - \bar{u}u^3;$$

$$b = 16\bar{u}^3u - \frac{64}{3}\bar{u}^2u^2 + \frac{16}{3}\bar{u}u^3;$$

$$c = -12\bar{u}^3u + 40\bar{u}^2u^2 - 12\bar{u}u^3;$$

$$d = \frac{16}{3}\bar{u}^3u - \frac{64}{3}\bar{u}^2u^2 + 16\bar{u}u^3;$$

$$e = -\bar{u}^3u + \frac{13}{3}\bar{u}^2u^2 - \frac{13}{3}\bar{u}u^3 + u^4;$$

u — текущий параметр дуги кривой 4-го порядка, проходящей через пять наперед заданных точек; $\bar{u} = 1 - u$ — дополнение параметра до 1; $0 \leq u \leq 1$; $a_i, b_i, c_i, d_i, f_i, e_i$ — значения прочностных характеристик (i — номер яруса, для первого яруса $i = 1$) (рис. 2).

Экспериментальные значения прочностных характеристик, которые являются функциями отклика, приводятся в работах [11, 13–14].

2 этап. Точечное уравнение образующей поверхности отклика для 1-го яруса:

$$M_1 = M_{11} \left(\bar{v}^4 - \frac{13}{3}\bar{v}^3v + \frac{13}{3}\bar{v}^2v^2 - \bar{v}v^3 \right) +$$

$$+ M_{12} \left(16\bar{v}^3v - \frac{64}{3}\bar{v}^2v^2 + \frac{16}{3}\bar{v}v^3 \right) +$$

$$+ M_{13} \left(-12\bar{v}^3v + 40\bar{v}^2v^2 - 12\bar{v}v^3 \right) +$$

$$+ M_{14} \left(\frac{16}{3}\bar{v}^3v - \frac{64}{3}\bar{v}^2v^2 + 16\bar{v}v^3 \right) +$$

$$+ M_{15} \left(-\bar{v}^3v + \frac{13}{3}\bar{v}^2v^2 - \frac{13}{3}\bar{v}v^3 + v^4 \right),$$

где $\bar{v} = 1 - v$; $0 \leq v \leq 1$.

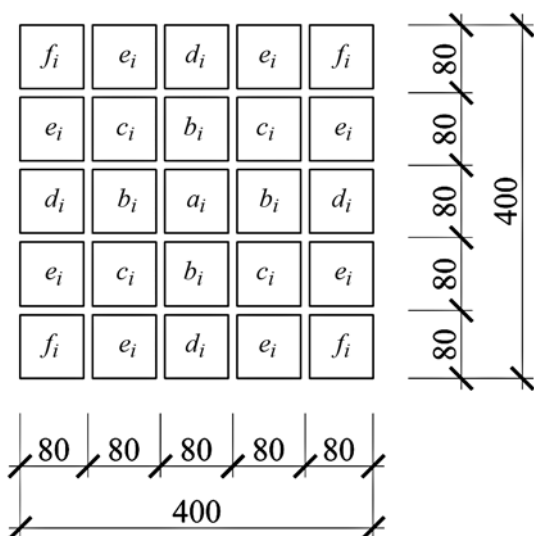


Рис. 2. Схема расположения, адресации и размеров выпиленных образцов на плане

Остальные точечные уравнения четырех поверхностей отклика, соответствующие 2, ..., 5 ярусам (см. рис. 1), определяются аналогичным образом. В уравнениях изменяются только индексы значений прочностных характеристик.

3 этап. Определение гиперповерхности отклика, для которой направляющими являются текущие точки образующих поверхностей отклика с первого по пятый ярус включительно:

$$M = M_1 \left(\bar{w}^4 - \frac{13}{3}\bar{w}^3w + \frac{13}{3}\bar{w}^2w^2 - \bar{w}w^3 \right) +$$

$$+ M_2 \left(16\bar{w}^3w - \frac{64}{3}\bar{w}^2w^2 + \frac{16}{3}\bar{w}w^3 \right) +$$

$$+ M_3 \left(-12\bar{w}^3w + 40\bar{w}^2w^2 - 12\bar{w}w^3 \right) +$$

$$+ M_4 \left(\frac{16}{3}\bar{w}^3w - \frac{64}{3}\bar{w}^2w^2 + 16\bar{w}w^3 \right) +$$

$$+ M_5 \left(-\bar{w}^3w + \frac{13}{3}\bar{w}^2w^2 - \frac{13}{3}\bar{w}w^3 + w^4 \right),$$

где $\bar{w} = 1 - w$; $0 \leq w \leq 1$.

Таким образом, получаем вычислительный точечный алгоритм моделирования отсека гиперповерхности отклика, проходящей через 125 наперед заданных точек, который определяется тремя параметрами: u, v и w , которые соответствуют трем факторам влияния. Итоговое точечное уравнение с учетом исходных экспериментальных данных в работе [14] не было приведено из-за большого его объема, а результат был представлен в виде вычислительного алгоритма геометрической модели. Следует также отметить, что все представленные выше точечные уравнения получены в рамках математического аппарата БН-исчисление (точечное исчисление Балюбы—Найдыша) [15–17]. Для перехода от точечных уравнений, которые являются символьной записью, к системе параметрических уравнений необходимо выполнить покоординатный расчет, отчего объем вычислений увеличится в четыре раза. Однако используя равномерное распределение параметра, заложенное при определении точечных уравнений дуг кривых, проходящих через наперед заданные точки, можно перейти от значений параметров u, v и w , которые изменяются от 0 до 1, к натуральным значениям факторов влияния x, y и z . Это позволит существенно сократить вычислительный алгоритм, а с учетом аппроксимации и вообще представить его в виде одного итогового уравнения.

Аппроксимация модели распределения прочностных характеристик в бетонной колонне

Учитывая симметричное расположение значений функции отклика в плане (рис. 2), воспользуемся для аппроксимации дугой кривой 2-го порядка, проходящей через три наперед заданные точки, которая описывается следующим точечным уравнением:

$$M = A_1\bar{u}(1 - 2u) + 4\bar{u}uA_2 + A_3u(2u - 1),$$

где A_i — исходные точки дуги кривой 2-го порядка.

В результате получим следующий вычислительный алгоритм, состоящий из трех этапов.

1 этап. Определение направляющих линий для 1-, 3- и 5-го ярусов (см. рис. 1):

$$M_{11} = M_{13} = 4d_1\bar{u}u + f_1(1 - 4\bar{u}u);$$

$$M_{12} = 4a_1\bar{u}u + d_1(1 - 4\bar{u}u);$$

$$M_{31} = M_{33} = 4d_3\bar{u}u + f_3(1 - 4\bar{u}u);$$

$$M_{32} = 4a_3\bar{u}u + d_3(1 - 4\bar{u}u);$$

$$M_{51} = M_{53} = 4d_5\bar{u}u + f_5(1 - 4\bar{u}u);$$

$$M_{52} = 4a_5\bar{u}u + d_5(1 - 4\bar{u}u).$$

2 этап. Точечные уравнения образующих поверхностей отклика для 1-, 3- и 5-го ярусов:

$$M_1 = M_{11}\bar{v}(1 - 2v) + 4\bar{v}vM_{12} + M_{13}v(2v - 1);$$

$$M_3 = M_{31}\bar{v}(1 - 2v) + 4\bar{v}vM_{32} + M_{33}v(2v - 1);$$

$$M_5 = M_{51}\bar{v}(1 - 2v) + 4\bar{v}vM_{52} + M_{53}v(2v - 1).$$

3 этап. Определение гиперповерхности отклика:

$$\frac{E_b^3}{E_b} = M_1\bar{w}(1 - 2w) + 4\bar{w}wM_3 + M_5w(2w - 1),$$

где $\frac{E_b^3}{E_b}$ — показатель соотношения характеристик бетона, отнесенный к показателю характеристик стандартных образцов [14].

После подстановок и преобразований с учетом экспериментальных данных из работы [14] получим геометрическую модель гиперповерхности отклика, принадлежащую четырехмерному пространству, которая определяется тремя параметрами: u , v и w :

$$\begin{aligned} \frac{E_b^3}{E_b} = & 123,5 - 92wu^2 + 92wu + 92wv - 92wv^2 + \\ & + 51,2w^2u^2 - 51,2w^2u - 51,2w^2v + 51,2w^2v^2 + \end{aligned}$$

$$\begin{aligned} & + 518,4w^2u^2v + 518,4w^2uv^2 - 745,6wuv^2 + \\ & + 745,6wu^2v^2 - 518,4w^2u^2v^2 - 745,6wuv^2 + \\ & + 745,6wuvv - 518,4w^2uv - 90,6w + 59,2w^2. \end{aligned}$$

Для перехода от значений параметров к натуральным значениям факторов с учетом равномерности распределения параметра были получены следующие линейные зависимости:

$$\begin{cases} x = 320u; \\ y = 320v; \\ z = 1280w. \end{cases}$$

Таким образом, получим зависимость

$$\frac{E_b^3}{E_b} = f(x, y, z). \text{ Следует отметить, что в дан-}$$

ном случае в качестве примера используется только один столбец экспериментальных данных, приведенных в работе [14]. Остальные две геометрических модели можно получить аналогичным образом, используя предложенный в статье вычислительный алгоритм и соответствующий столбец данных.

Оценка точности аппроксимации геометрической модели

Для оценки точности аппроксимации и проверки достоверности полученных моделей существует достаточно много различных критериев, к которым относятся коэффициент детерминации, стандартная ошибка модели, средняя относительная ошибка аппроксимации и т. п. Однако работ, посвященных оценке достоверности геометрических моделей с учетом их специфики, автору найти не удалось. Наиболее часто в литературе встречается точность аппроксимации применительно к методам математической статистики (например, к регрессионному анализу). В работе [18] предлагается для проверки адекватности и точности полученной модели процесса ниточного соединения использовать критерий Стьюдента и величину относительной погрешности. Однако эти критерии скорее учитывают специфику проверки достоверности самих экспериментальных данных на предмет случайных ошибок, чем специфику геометрических моделей. В сложившейся ситуации воспользуемся различными критериями математической статистики для оценки точности аппроксимации геометрической модели и сравним результаты.

В общем, точность любой модели характеризуется разностью между фактическими и расчетными значениями исследуемого показателя. Такой мерой точности является стандартная ошибка модели $S_{\text{мод}}$, которую можно вычислить по формуле:

$$S_{\text{мод}} = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n - 2}},$$

где $\sum_{i=1}^n (y_i - \hat{y}_i)^2$ — сумма квадратов регрессионных остатков, которая включает фактические y_i и расчетные \hat{y}_i значения исследуемой переменной.

Однако наиболее удобно оценивать точность модели с помощью средней относительной ошибки аппроксимации $E_{\text{отн}}$, которая показывает, на сколько процентов в среднем модельные значения \hat{y}_i отличаются от фактических y_i :

$$E_{\text{отн}} \approx 0,8 \frac{S_{\text{мод}}}{\bar{y}} 100 \%,$$

где \bar{y} — выборочное среднее.

Если $E_{\text{отн}} \leq 5 \%$, то считается, что модель имеет достаточно высокую точность. В нашем случае $E_{\text{отн}} = 2,16 \%$.

Кроме того, для оценки адекватности полученной модели был использован коэффициент детерминации R^2 , определяемый по формуле

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2},$$

где $\sum_{i=1}^n (y_i - \bar{y})^2$ — общая дисперсия.

Коэффициент детерминации для модели принимает значения от 0 до 1. Чем ближе значение коэффициента к 1, тем сильнее зависимость. При оценке геометрических моделей будем это интерпретировать как соответствие полученной модели исходным экспериментальным (или статистическим) данным. Для приемлемых моделей предполагается, что коэффициент детерминации должен быть хотя бы не меньше 50 %. Модели с коэффициентом детерминации выше 80 % можно признать достаточно хорошими. Равенство коэффициента

детерминации 1 означает, что было достигнуто полное соответствие между геометрической моделью и исходными данными. В нашем случае коэффициент детерминации $R^2 = 0,94$.

Таким образом, вне зависимости от выбора критериев адекватности геометрической модели, заимствованных из математической статистики, в данном случае была достигнута очень хорошая сходимость между геометрической моделью и исходными экспериментальными данными.

Заключение

В работе предложен способ аппроксимации геометрических объектов многомерного аффинного пространства с помощью дуг алгебраических кривых, проходящих через наперед заданные точки. В качестве примера использовалась геометрическая модель распределения прочностных характеристик по всему объему бетонной колонны. Также были исследованы способы оценки точности аппроксимации геометрических моделей. Для этого были использованы методы математической статистики. В результате была достигнута очень хорошая сходимость между геометрической моделью и исходными экспериментальными данными (средняя относительная ошибка аппроксимации $E_{\text{отн}} = 2,16 \%$ составила при коэффициенте детерминации $R^2 = 0,94$). К преимуществам использования дуг кривых, проходящих через наперед заданные точки, для аппроксимации геометрических объектов многомерного пространства следует отнести то, что часть исходных точек участвует в процессе аппроксимации в качестве узловых. Это способствует достижению высоких показателей точности аппроксимации. Вместе с тем к недостаткам предложенного способа можно отнести отсутствие методики выбора узловых точек аппроксимации из массива исходных данных. Это является перспективой дальнейших исследований.

Список литературы

1. Конопаккий Е. В. Геометрическое моделирование и оптимизация многофакторных процессов и явлений методом многомерной интерполяции // Тр. Междунар. науч. конф. по физико-технической информатике СРТ2018, 28—31 мая 2018 г. Москва-Протвино, 2018. С. 299—306.
2. Математика: Энциклопедия / Под ред. Ю. В. Прохорова. М.: Большая Российская энциклопедия, 2003. 845 с.
3. Голубинский А. Н. Методы аппроксимации экспериментальных данных и построения моделей // Вестник ВИ МВД России. 2007. № 2. URL: <https://cyberleninka.ru/article/n/>

metody-approksimatsii-eksperimentalnyh-dannyh-i-postroeniya-modeley (дата обращения: 22.07.2018).

4. **Шашков В. Б.** Прикладной регрессионный анализ. Многофакторная регрессия: Учеб. пособ. Оренбург: ГОУ ВПО ОГУ, 2003. 363 с.

5. **Прэртт У.** Цифровая обработка изображений / Пер. с англ. М.: Мир, 1982. Кн. 2. 480 с.

6. **Роджерс Д., Адамс Дж.** Математические основы машинной графики. М.: Мир, 2001. 605 с.

7. **Квасов Б. И.** Методы изогеометрической аппроксимации сплайнами. М.: ФИЗМАТЛИТ, 2006. 360 с.

8. **Иванов Г. С.** Конструирование одномерных обводов, принадлежащих поверхностям, путем их отображения на плоскость // Геометрия и графика. 2018. Т. 6, № 1. С. 3—9. DOI: 10.12737/article_5ad07ed61bc114.52669586.

9. **Крысько А. А., Конопацкий Е. В., Чураков А. Я.** Геометрические основы конструирования одномерного обвода через k наперед заданных точек в БН-исчислении // Сучасні проблеми моделювання: зб. наук. праць. Мелітополь: МДПУ ім. Б. Хмельницького, 2015. Вып. 4. С. 76—81.

10. **Крысько А. А.** Геометрическое и компьютерное моделирование эксплуатируемых конструкций тонкостенных оболочек инженерных сооружений с учетом несовершенств геометрической формы: дис. ... канд. техн. наук: 05.23.01, 05.01.01. Макеевка, 2016. 191 с.

11. **Конопацкий Е. В.** Геометричне моделювання алгебраїчних кривих та їх використання при конструюванні поверхонь у точковому численні Балюби-Найдиша: дис. ... канд. техн. наук: 05.01.01. Мелітополь, 2012. 164 с.

12. **Бумага А. И.** Геометрическое моделирование физико-механических свойств композиционных строительных материалов в БН-исчислении: дис. ... канд. техн. наук: 05.23.05, 05.01.01. Макеевка, 2016. 164 с.

13. **Югов А. М., Булавицкий М. С., Конопацкий Е. В.** Визначення міцнісних та деформативних властивостей важкого бетону по об'єму монолітних колон при їх обстеженні // Дороги і мости: Збірник наукових праць. К.: ДерждорНДІ, 2009. Вип. 11. С. 372—379.

14. **Конопацкий Е. В., Воронова О. С.** Геометрическая модель процесса распределения прочностных характеристик в бетонной колонне // Прикладная математика и вопросы управления. Пермь: ПНИПУ, 2017. № 1. С. 37—44.

15. **Балюба И. Г.** Конструктивная геометрия многообразий в точечном исчислении: дис. ... докт. техн. наук: 05.01.01. Макеевка, 1995. 227 с.

16. **Найдыш В. М., Балюба И. Г., Верещага В. М.** Алгебра БН-исчисления // Прикладна геометрія та інженерна графіка: Міжвідомчий науково-технічний збірник. К.: КНУБА, 2012. Вип. 90. С. 210—215.

17. **Балюба И. Г., Найдыш В. М.** Точечное исчисление: Учеб. пособие; под ред. В. М. Верещаги. Мелітополь: МГПУ ім. Б. Хмельницького, 2015. 236 с.

18. **Чижик М. А.** Методология параметрического проектирования технологических процессов швейного производства на основе многомерного геометрического моделирования: дис. ... докт. техн. наук: 05.19.04. СПб., 2018. 272 с.

E. V. Konopatskiy, Assistant Professor, e-mail: e.v.konopatskiy@mail.ru
Donbas National Academy of Civil Engineering and Architecture, Makeevka

Approximation of Geometric Objects Using Arcs of Curves Passing through to Advance Given Points

The paper proposes a method for approximating geometric objects of multidimensional affine space using arcs of algebraic curves passing through to advance given points. As an example, we used a geometric model of the distribution by strength characteristics over the entire volume of a concrete column. The estimation methods of accuracy the approximation of geometrical models were also investigated. Methods of mathematical statistics were used for this purpose. As a result, a very good convergence between the geometric model and the initial experimental data was achieved (the average relative approximation error was $E_{real} = 2,16\%$ at the determination coefficient $R^2 = 0,94$). The advantages of using the arcs curves passing through the predetermined points for the approximation the geometric objects of multidimensional space include the fact that part of the initial points is involved in the process of approximation as nodal. This contributes to the achievement of high approximation accuracy. On the other hand, the disadvantages of the proposed method include the lack of a technique for selecting nodal approximation points from the array of initial data. This is a prospect for further research.

Keywords: approximation, geometric object, geometric modeling, the arc of the curve, passing through to advance given points, multidimensional interpolation, factors of influence, the response function, point equation, approximation accuracy, the coefficient of determination

DOI: 10.17587/it.25.46-51

References

1. **Конопатский Е. В.** Geometricheskoe modelirovanie i optimizatsiya mnogofaktornyx processov i yavleniy metodom mnogomernoy interpolatsii (Geometric modeling and optimization of multifactorial processes and phenomena by the method of multidimensional interpolation), *Trudy Mezhdunarodnoy nauchnoy konferentsii po*

fiziko-texnicheskoy informatike CPT2018, Moskva-Protvino, 2018, pp. 299—306 (in Russian).

2. **Proxorov Yu. V.** ed. Matematika: E'nciklopediya (Mathematics: Encyclopedia), Moscow, Bol'shaya Rossiyskaya e'nciklopediya, 2003, 845 p. (in Russian).

3. **Golubinskij A. N.** *Metody approksimatsii e'ksperimental'nykh dannykh i postroeniya modelej* (Approximation methods of experi-

mental data and construction of models), *Vestnik VI MVD Rossii*, 2007, no. 2, available at: <https://cyberleninka.ru/article/n/metody-aproksimatsii-eksperimentalnyh-dannyh-i-postroeniya-modeley> (date of access: 22.07.2018) (in Russian).

4. **Shashkov V. B.** *Prikladnoj regressionnyj analiz. Mnogofaktornaya regressiya* (Applied regression analysis. Multivariate regression), Orenburg, GOU VPO OGU, 2003, 363 p. (in Russian).

5. **Pre'tt U.** *Cifrovaya obrabotka izobrazhenij* (Digital image processing, Moscow, Mir, 1982, vol. 2, 480 p.

6. **Rodzher D., Adams Dzh.** *Matematicheskie osnovy mashinnoj grafiki* (Mathematical foundations of computer graphics), Moscow, Mir, 2001, 605 p. (in Russian).

7. **Kvasov B. I.** *Metody izogeometricheskoj approksimacii splajnami* (Isogeometric spline approximation methods), Moscow, FIZMATLIT, 2006, 360 p. (in Russian).

8. **Ivanov G. S.** *Konstruirovaniye odnomernyx obvodov, prinadlezhashhix poverxnostyam, putem ix otobrazheniya na ploskost'* (Construction of Belonging to Surfaces One-Dimensional Contours by Mapping Them to a Plane), *Geometriya i grafika*, vol. 6, no. 1, pp. 3–9 (in Russian), DOI: 10.12737/article_5ad07ed61bc114.52669586 (in Russian).

9. **Krys'ko A. A., Konopackij E. V., Churakov A. Ya.** *Geometricheskie osnovy konstruirovaniya odnomernogo obroda cherez k napered zadannyx toček v BN-ischislenii* (Geometrical bases of construction of one-dimensional circle through k in advance of the given points in BN-calculation), *Suchasni problemi modelyuvannya: zb. nauk. prac'*, Melitopol', MDPU im. B. Xmel'nic'kogo, 2015, vol. 4, pp. 76–81 (in Ukrainian).

10. **Krys'ko A. A.** *Geometricheskoe i komp'yuternoe modelirovaniye e'kspluatiruemyx konstrukcij tonkostennyx obolochek inzhenernyx sooruzhenij s uchyotom nesovershenstv geometricheskoj formy. Kand, diss.* [Geometric and computer modeling of the operated structures of thin-walled shells of engineering structures taking into account the imperfections of the geometric shape. Cand. Diss.] Makeevka, 2016, 191 p. (in Russian).

11. **Konopac'kij E. V.** *Geometrichne modelyuvannya algebraychnix krivix ta ix vikoristannya pri konstruyuvanni poverxon' u*

tochkovomu chislenni Balyubi-Najdisha. Kand, Diss. [Geometric modeling of algebraic curves and their use in the construction of surfaces in point calculus of Balyubi-Naidish. Cand. Diss.], Melitopol, 2012, 164 p. (in Ukrainian).

12. **Bumaga A. I.** *Geometricheskoe modelirovaniye fizikomexanicheskix svojstv kompozicionnyx stroitel'nyx materialov v BN-ischislenii. Kand, Diss.* [Geometric modeling the physical and mechanical properties of composite building materials in BN-calculation. Cand. Diss.], Makeevka, 2016, 164 p. (in Russian).

13. **Yugov A. M., Bulavic'kij M. S., Konopac'kij E. V.** *Viznachennyya micnisnix ta deformativnix vlastivostej vazhkoغو betonu po ob'emnu monolitnix kolon pri ix obstezhenii* (Determination of strength and deformation properties of heavy concrete by volume of monolithic columns during their inspection), *Dorogi i mosti, Zbirnik naukovix prac'*, K., DerzhdorNDI, 2009, vol. 11, pp. 372–379 (in Ukrainian).

14. **Konopackij E. V., Voronova O. S.** *Geometricheskaya model' processa raspredeleniya prochnostnyx xarakteristik v betonnoj kolonne. Prikladnaya matematika i voprosy upravleniya, Perm'*, PNIPU, 2017, no. 1, pp. 37–44 (in Russian).

15. **Balyuba I. G.** *Konstruktivnaya geometriya mnogoobrazij v tochechnom ischislenii. Dokt, Diss.* [Constructive geometry of manifolds in a point calculation. Doct. Diss.], Makeevka, 1995, 227 p. (in Russian).

16. **Najdysh V. M., Balyuba I. G., Vereshhaga V. M.** *Algebra of BN-calculation, Prikladnaya geometriya ta inzhenerna grafika*, 2012, vol. 90, pp. 210–215 (in Ukrainian).

17. **Balyuba I. G., Najdysh V. M.** *Tochechnoe ischislenie* [Point calculation]. Melitopol, 2015, 236 p. (in Russian).

18. **Chizhik M. A.** *Metodologiya parametricheskogo proektirovaniya texnologicheskix processov shvejnogo proizvodstva na osnove mnogomernogo geometricheskogo modelirovaniya. Dokt, Diss.* [Methodology the parametric design of technological processes sewing production on the base of multidimensional geometric modeling. Doct. Diss.], SPb., 2018, 272 p. (in Russian).

Международная конференция "Информационные технологии в бизнесе и производстве"

Международная конференция "Информационные технологии в бизнесе и производстве — 2019" пройдет на базе Новосибирского государственного технического университета (Новосибирск, Россия)
13–15 февраля 2019 г.

К участию в конференции допускаются законченные научно-исследовательские работы, написанные на хорошем английском языке и соответствующие заявленным требованиям. Предоставляемые материалы должны быть оригинальными и соответствовать направлениям секций конференции.

Секции конференции:

1. Микропроцессорные и телекоммуникационные устройства вычислительных сетей.
2. Математическое моделирование и компьютерный анализ данных.
3. Робототехника и управление в технических системах.
4. Автоматизация и компьютеризация проектирования.
5. Информационные технологии автоматизированного машиностроения.

Контакты: E-mail: ITBI2016@mail.ru

Ю. В. Полищук, канд. техн. наук, доц., e-mail: youga_polishuk@bk.ru,
Оренбургский государственный университет

Способ хранения электронных документов с квазиструктурированным контентом

Рассмотрен способ хранения электронных документов с квазиструктурированным контентом, который реализует минимизацию их объема хранения за счет выделения единого шаблона оформления документов, извлечения из документов фактографических данных с последующим сжатием шаблона и фактографических данных.

Ключевые слова: квазиструктурированная информация, сжатие информации, обработка электронных документов

Введение

В процесс работы предприятия формируется сопутствующая эксплуатационная документация, которая, как правило, представлена коллекцией документов. Среди документов единого вида можно выделить общую структуру, но оформление и порядок размещения информации в них будет различен. Последнее обусловлено тем, что документация формируется на основе внутренних стандартов предприятия или в соответствии с требованиями ГОСТ, в которых определены требования к содержанию документов.

Контент документов данного вида квазиструктурирован, т. е. представлен квазиструктурированной информацией. Под квазиструктурированной информацией понимают информацию, в которой можно выделить некую структуру, однако структура эта заранее целиком или частично не известна, либо может меняться с течением времени [1].

При длительном периоде работы предприятия скапливается большое количество электронной документации с квазиструктурированным контентом, как правило, представленной документами формата MS Word, и минимизация объема ее хранения является актуальной задачей.

подавляющее большинство исследований в области обработки квазиструктурированной информации ориентировано на хранение не-

посредственно фактографических данных, а не электронных документов. В данной работе рассматривается способ хранения электронных документов с квазиструктурированным информационным наполнением. Электронный документ представляет собой квазиструктурированный информационный контент, обладающий структурой и визуальным оформлением.

Таким образом, способ хранения электронных документов с квазиструктурированным информационным наполнением должен реализовать хранение не только фактографических данных, но и их структуры и визуального оформления в документе.

Исследования в данном направлении выполнены в работе [2], а на предлагаемый способ преобразования слабоформализуемых документов для минимизации их объема при хранении получен патент [3].

К недостаткам данного способа следует отнести отсутствие математической модели информационного наполнения обрабатываемых документов, отсутствие возможности описания лексикологическим деревом фактографического контента (неунифицированной информации) для документов.

Исследования, проведенные в настоящей работе, соответствуют направлениям Федеральной целевой программы "Информационное общество" (2011–2020)", утвержденной постановлением Правительства РФ от 20.10.2010 г. № 1815-р, и критическим технологиям РФ

(технологии информационных, управляющих и навигационных систем), утвержденными Президентом РФ (Пр-899 от 7.07.2011 г.).

Постановка задачи

Для минимизации объема, требуемого для хранения электронных документов с квазиструктурированным контентом, необходимо отделить фактографические данные документа от их визуального представления в документе. Так как визуальное представление будет идентичным для всей коллекции документов, то его можно хранить в виде шаблона формы электронного документа, в котором будут храниться все визуальные особенности представления фактографической информации в контенте документов.

Таким образом, алгоритм сохранения электронных документов предложенным способом будет иметь вид, представленный на рис. 1.

Для восстановления электронного документа необходимо к фактографическому контенту выбранного документа применить шаблон формы электронного документа (рис. 2).

Практическая реализация

В качестве примера рассмотрим пример хранения электронной документации газоконденсатного месторождения. Применим предлагаемый способ для хранения коллекции документов вида "Информационная карта скважины".

Для обработки контента документов данного вида необходима разработка квазиструктурированной модели фактографического контента информационного наполнения документа, которая с учетом спецификации Xml Schema Definition (XSD) может быть записана следующим образом [4]:

$$S = \left\langle \begin{matrix} root, sObj, LObj, minOccurs, \\ maxOccurs, sMet, Obj_smet \end{matrix} \right\rangle, \quad (1)$$

где *root* — корневой объект; $root \in sObj$; *sObj* — конечное множество объектов, каждый из которых содержит фрагмент информационного наполнения документа (текст, рисунок и т. д.) или выполняет роль контейнера для одного или нескольких объектов.

Для объектов-контейнеров доступны следующие метасвойства: *smet_c* — определяет объект в качестве контейнера; *mixed* — разре-

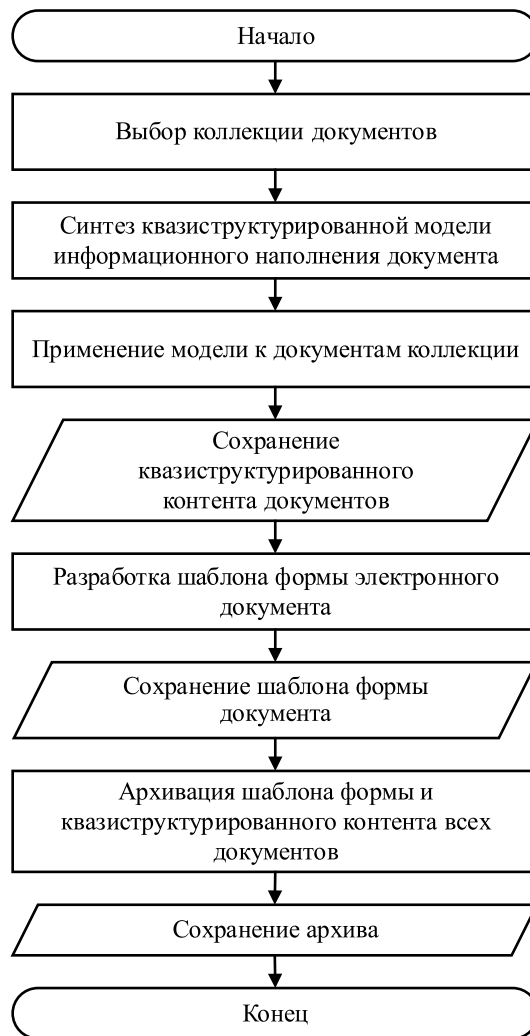


Рис. 1. Алгоритм сохранения электронных документов



Рис. 2. Алгоритм восстановления электронного документа

шает использование объектов-потомков в произвольном порядке. $LObj$ — отображение, определенное на множестве $sObj$, такое что $sObj \xrightarrow{LObj} \{obj_1, \dots, obj_n\}$, где $obj_i \in sObj$ — дочерний объект; n — число дочерних объектов; Obj_met — отображение, определенное на множестве $sObj$, такое что $sObj \xrightarrow{Obj_met} \{smet_c | smet_c, mixed | smet_1, \dots, smet_k\}$, где $smet_i \in sMet$ — метасвойство ограничения на содержимое объекта; $minOccurs$ — функция, определяющая минимально возможное число раз использования объекта в модели; $maxOccurs$ — функция, определяющая максимально возможное число раз использования объекта в модели.

Для разработки квазиструктурированной модели фактографического контента информационного наполнения документа "Информационная карта скважины" используем "Способ формирования квазиструктурированных моделей фактографического информационного наполнения документов" [5].

Структура модели документа "Информационная карта скважины" описывает основную информацию о газовых скважинах, такую как подключение, конструкция, информация об исследованиях, рабочие дебиты и т. д.

Модель "Информационная карта скважины" представлена базовым сегментом, рассмотрим его структуру подробнее (рис. 3).

Базовый сегмент состоит из двух сегментов: обязательного **BasicInfo** и необязательного **StatusInfo**. Сегмент **BasicInfo** хранит базовую информацию о пробуренной скважине. Он состоит из двух объектов **WellNum** — номер скважины и **DrillDate** — дата бурения, а также двух сегментов: **Connection** (рис. 4) и **ConsDescr** (рис. 5).

Обязательный сегмент **ConsDescr** используется для описания конструкции скважины и состоит из двух необязательных объектов: **PerfDiap** — информация о перфорации и **TrunkDepth** — информация об открытом стволе. Оба рассмотренных объекта могут быть использованы неограниченное число раз.

Сегмент **StatusInfo** хранит информацию о пусках и остановках скважины. Он состоит из двух сегментов: обязательного сегмента **Launch** (рис. 6), который может быть использован в контенте неограниченное число раз, и

необязательного сегмента **Break** (рис. 7), который также может быть использован неограниченное число раз.

Данный сегмент состоит из трех объектов: **FieldTitle** — название месторождения, **Train** — номер шлейфа, к которому подключена скважина и **UKPG** — номер установки комплексной подготовки газа (УКПГ), к которой поступает продукция от скважины. Рассмотренные объекты необязательны к использованию в документе.

Сегмент **Launch** (см. рис. 6) используется для хранения информации о работе скважины. Данный сегмент состоит из обязательного сегмента **Debit**, который может быть использо-

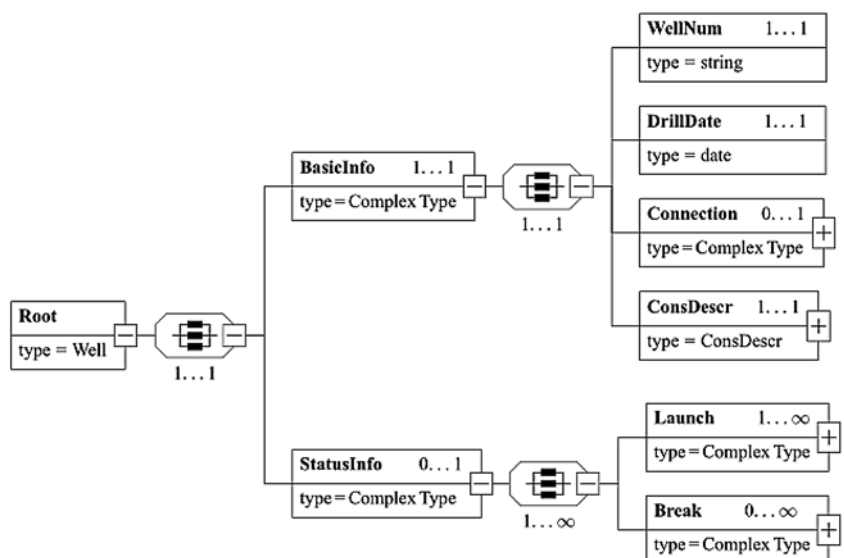


Рис. 3. Структура базового сегмента модели "Информационная карта скважины"

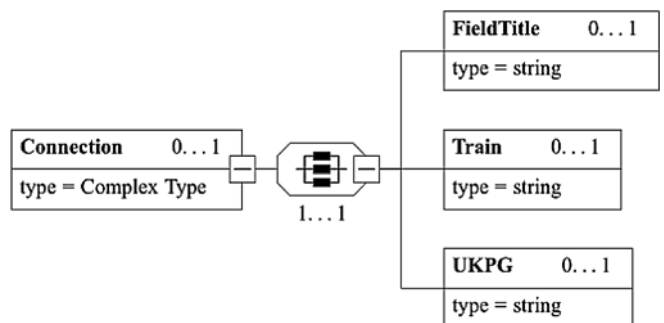


Рис. 4. Структура сегмента Connection

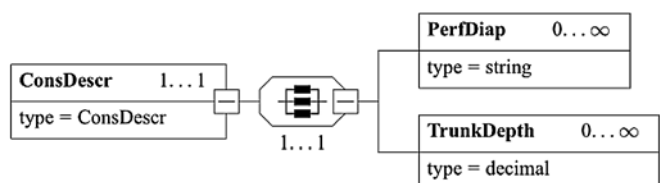


Рис. 5. Структура сегмента ConsDescr

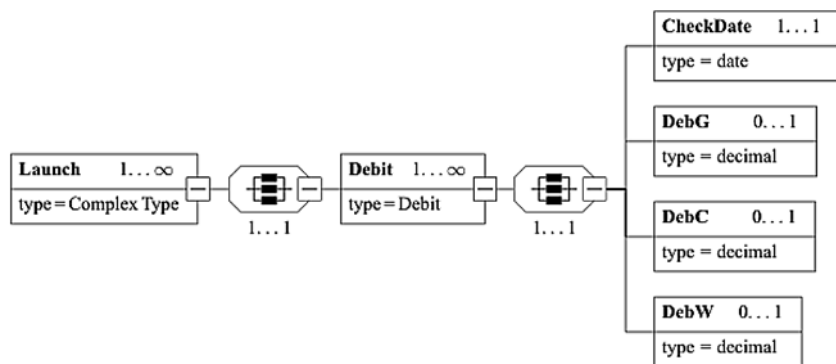


Рис. 6. Структура сегмента Launch

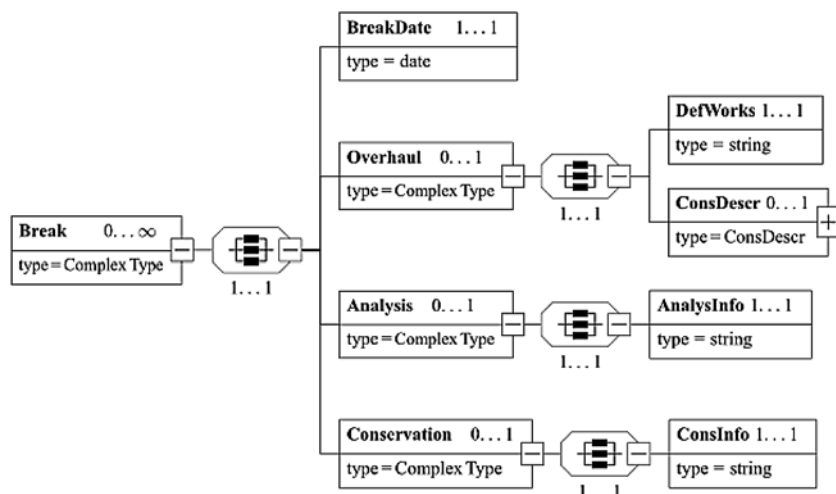


Рис. 7. Структура сегмента Break

ван неограниченное число раз. Сегмент **Debit**, в свою очередь, состоит из обязательного объекта **CheckDate** — дата отчетного месяца, и необязательных объектов **DebG** — месячный дебит газа, **DebC** — месячный дебит конденсата, **DebW** — месячный дебит воды.

Необязательный сегмент **Break** (рис. 7) предназначен для хранения информации об остановках и причинах этих остановок в работе скважины. Он состоит из обязательного объекта **BreakDate** — дата остановки, а также трех необязательных сегментов: **Overhaul** — используется для хранения информации о капитальном ремонте скважины, **Analysis** — используется для хранения информации о результатах гидродинамических исследованиях скважины и **Conservation** — используется для хранения информации о консервации скважины.

В состав сегмента **Overhaul** входят обязательный объект **DefWorks** — комплекс выполненных ремонтных работ и необязательный сегмент **ConsDescr**, хранящий информацию о текущем состоянии конструкции скважины.

Разметив содержимое электронных документов с помощью описанной модели, извлечем фактографические данные из документов. Полученные данные будут представлены в формате XML. Процесс разметки содержимого электронного документа с помощью модели может быть реализован как в ручном, так и в автоматизированном режиме [6].

Используя модель документа и формат Office Open XML, формируем шаблон формы электронного документа [7].

В результате проделанных операций для коллекции документов будет получено их фактографическое содержимое и шаблон формы электронного документа данного вида.

Таким образом, объем хранимой информации был существенно снижен без потери фактографических данных коллекции документов. Объем результирующей информации может быть дополнительно снижен за счет применения современных технологий архивирования данных.

Восстановление электронного документа до первоначального вида осуществляется с помощью алгоритма восстановления электронного документа, рассмотренного ранее, и может быть проиллюстрировано с помощью схемы, показанной на рис. 8.

Для оценки эффективности описанного способа хранения электронных документов с квазиструктурированным контентом были подготовлены наборы коллекций, состоящие из 1 000 и 2 000 документов "Информационная карта скважины". Результаты сравнения представлены в таблице.

Сравнение эффективности способа хранения электронных документов

Вариант хранения электронного документа	1000 документов	2000 документов
Исходный размер	29,2 Мбайт	58,5 Мбайт
Сжатие архивацией (7-Zip)	23,9 Мбайт	47,8 Мбайт
Сжатие алгоритмом	6,83 Мбайт	13,6 Мбайт
Сжатие алгоритмом + архивация (7-Zip)	1,06 Мбайт	2,11 Мбайт

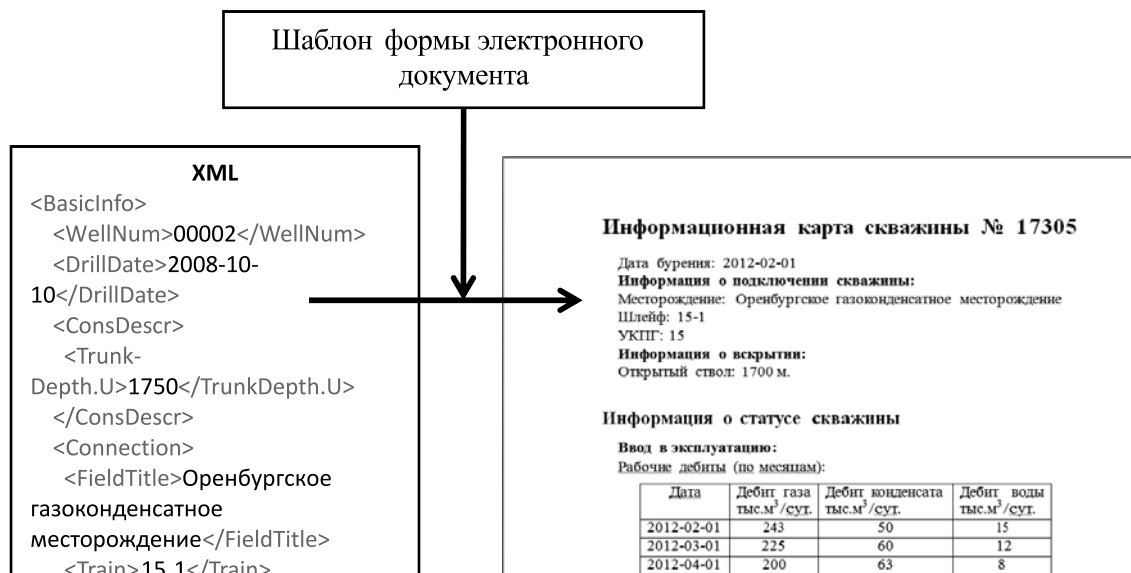


Рис. 8. Восстановление электронного документа

Заключение

Рассмотренный способ хранения электронных документов с квазиструктурированным контентом минимизирует их объем при хранении за счет выделения единого шаблона оформления документов, извлечения из документов фактографических данных с последующим сжатием шаблона и фактографических данных, что обеспечивает сжатие электронных документов до 22 раз эффективнее стандартных архиваторов.

На описанный в работе способ хранения электронных документов с квазиструктурированным контентом получен патент [8].

Список литературы

1. Палей Д. Моделирование квазиструктурированных данных // Открытые системы. 2002. № 9. С. 57–64.

2. Черников Б. В. Технология хранения слабоформализуемых документов на основе лексикологического синтеза // Информатика и ее применение. 2009. Т. 3, Вып. 4. С. 64–75.

3. Черников Б. В. Способ преобразования слабоформализуемых документов для минимизации их объема при хранении. Патент на изобретение, рег. № 2413985 от 10.03.2011. М.: Роспатент, 2011.

4. Полищук Ю. В., Черных Т. А. Синтез квазиструктурированных моделей информационного наполнения электронных // Вестник компьютерных и информационных технологий. 2012. № 6. С. 20–27.

5. Полищук Ю. В. Способ формирования квазиструктурированных моделей фактографического информационного наполнения документов. Патент на изобретение, рег. № 2517428 от 28.03.2014. М.: Роспатент, 2014.

6. Полищук Ю. В., Ларин А. В. Автоматизация процесса разметки управленческого контента электронных документов с квазиструктурированным информационным наполнением на основе паттернов // Вестник компьютерных и информационных технологий. 2013. № 3. С. 55–60.

7. Воутер В. В. Open XML — Кратко и доступно. Open XML Technical Evangelist. Microsoft, 2007. 101 с.

8. Полищук Ю. В., Полищук П. В. Способ преобразования документов для минимизации их объема при хранении электронных документов с квазиструктурированным информационным наполнением. Патент на изобретение, рег. № 2625611 от 17.07.2017. М.: Роспатент, 2017.

Yu. V. Polishuk, PhD, Associate Professor

of Computer Security mathematical software and information systems, Orenburg State University

The Method of Storing Electronic Documents with Semistructured Content

The accompanying operational documentation is formed in the work process of the enterprise, which, as a rule, is represented by a collection of documents, among which for documents of a single type it is possible to single out a general structure, but the design and procedure for placing information in them will be different. The latter is because the documentation is formed on the basis of internal standards of the enterprise or in accordance with the requirements of GOST in which the requirements for the content of documents are defined. The content of documents of this type is semistructured, i.e. is represented by semistructured

information. The semistructured information is understood as information in which a certain structure can be identified, but this structure is completely unknown in advance or may change with time. With a long period of the enterprise's work, a large number of electronic documentation with semistructured content is accumulated, as a rule, represented by MS Word documents and minimization of its storage volume is an actual task. The described method of storing electronic documents is to minimize the amount of storage of electronic documents with semistructured content by allocating a single template for processing documents, extracting factual data from documents, and then compressing the template and factographic data. The restoring the document is done by extracting from the archive the factual data of the form template and specified content of the document, and applying the form template to the newly received document content. The method provides compression of electronic documents with semistructured content up to 22 times more efficient than standard archivers.

Keywords: semistructured data; information compression; electronic documents processing

DOI: 10.17587/it.25.53-58

References

1. **Palej D.** *Modelirovanie kvazistrukturirovannykh dannykh* (Simulation of quasi-structured data), *Otkrytye Sistemy*, 2002, no. 9, pp. 57–64 (in Russian).
2. **Chernikov B. V.** *Tehnologiya hranenija slaboformalizuemykh dokumentov na osnove leksikologicheskogo sinteza* (Storage technology of poorly formalized documents based on lexicological synthesis), *Informatika i ee Primenenie*, 2009, vol. 3, iss. 4, pp. 64–75 (in Russian).
3. **Chernikov B. V.** *Sposob preobrazovaniya slaboformalizuemykh dokumentov dlja mini-mizacii ih ob#ema pri hranenii* (The method of converting poorly formalized documents to minimize their volume during storage), Patent na izobretenie, reg. № 2413985 ot 10.03.2011, Moscow, Rospatent, 2011 (in Russian).
4. **Polishhuk Ju. V., Chernyh T. A.** *Sintez kvazistrukturirovannykh modelej informacionnogo napolnenija jelektronnykh dokumentov* (Synthesis of quasi-structured models of informational filling of electronic), *Vestnik Komp'ju-Ternyh I Informacionnykh Tehnologij*, 2012, no. 6, pp. 20–27 (in Russian).
5. **Polishhuk Ju. V.** *Sposob formirovaniya kvazistrukturirovannykh modelej faktogra-ficheskogo informacionnogo napolnenija dokumentov* (The method of forming quasi-structured models of factual content of documents), Patent na izobretenie, reg. № 2517428 ot 28.03.2014, Moscow, Rospatent, 2014 (in Russian).
6. **Polishhuk Ju. V., Larin A. V.** *Avtomatizacija processa razmetki upravlencheskogo kontenta jelek-tronnykh dokumentov s kvazistrukturirovannym informacionnym napolneniem na osnove patternov* (Automating the process of marking management content of electronic documents with quasi-structured content based on patterns), *Vestnik komp'juternyh i infor-macionnykh tehnologij*, 2013, no. 3, pp. 55–60 (in Russian).
7. **Vouter V. V.** *Open XML — Kratko i dostupno* (Open XML — Brief and Available), Open XML Technical Evangelist, Microsoft, 2007, 101 p. (in Russian).
8. **Polishhuk Ju. V., Polishhuk P. V.** *Sposob preobrazovaniya dokumentov dlja minimizacii ih ob#joma pri hranenii jelektronnykh dokumentov s kvazistrukturirovannym informacionnym napolneniem* (The method of converting documents to minimize their volume when storing electronic documents with quasi-structured information content), Patent na izobretenie, reg. № 2625611 ot 17.07.2017, Moscow, Rospatent, 2017 (in Russian).

УДК 004.421

DOI: 10.17587/it.25.58-63

Т. О. Перемитина, канд. техн. наук, доц., e-mail: peremitinat@mail.ru,

И. Г. Ященко, канд. геол.-минер. наук, e-mail: sric@ipc.tsc.ru,

Федеральное государственное бюджетное учреждение науки Институт химии нефти СО РАН, Томск

АЛГОРИТМ КОМПЛЕКСНОГО АНАЛИЗА МНОГОМЕРНЫХ ДАННЫХ ОБ ЭКОЛОГИЧЕСКОМ СОСТОЯНИИ ОКРУЖАЮЩЕЙ СРЕДЫ

Рассматриваются методические вопросы реализации и практического применения комплексного подхода к анализу многомерных данных об экологическом состоянии пространственно-распределенных объектов исследования. Подход основан на сочетании метода главных компонент и метода пространственного анализа с применением геоинформационных систем. Метод главных компонент применяется для статистической обработки и анализа данных, метод пространственного анализа используется для учета пространственных свойств исследуемых объектов. В статье приведены результаты анализа многомерных данных о состоянии природной среды и здоровья населения территорий Сибирского и Дальневосточного федеральных округов.

Ключевые слова: многомерные данные, метод главных компонент, геоинформационные системы, окружающая среда, экология

Введение

Мониторинг состояния окружающей среды сопряжен с обработкой постоянно увеличивающегося массива разрозненной совокупной информации об объектах исследования. Аналитические процессы предполагают обработку

вающегося массива разрозненной совокупной информации об объектах исследования. Аналитические процессы предполагают обработку

огромного количества материалов и включают такие трудоемкие процессы, как систематизация, выявление каких-либо закономерностей и преобразование данных.

Широко известен и используется для анализа многомерных данных метод главных компонент (МГК) [1–3]. Однако для анализа объектов исследования, имеющих выраженные пространственные свойства, применение МГК оказывается недостаточным, так как он не позволяет учитывать их пространственные характеристики. В связи с этим возникла необходимость разработки нового комплексного подхода к анализу многомерных данных, направленного на учет пространственных свойств объектов исследования.

Анализ экологического состояния окружающей природной среды, выявление причин загрязнения и прогнозирование дальнейшего состояния являются актуальными и важными задачами, поскольку именно качество природных объектов во многом определяет состояние здоровья людей, которые являются лишь частью природных экосистем [4].

В данной работе обсуждается комплексный анализ многомерных данных о состоянии природной среды и здоровья населения территории Сибирского и Дальневосточного Федеральных округов (ФО) и его практическое применение.

Методические вопросы реализации комплексного анализа

Основная идея комплексного анализа заключается в сочетании МГК и метода пространственного анализа с применением геоинформационных систем (ГИС). МГК применяется для статистической обработки различных количественных характеристик, а методом пространственного анализа исследуются пространственные свойства объектов.

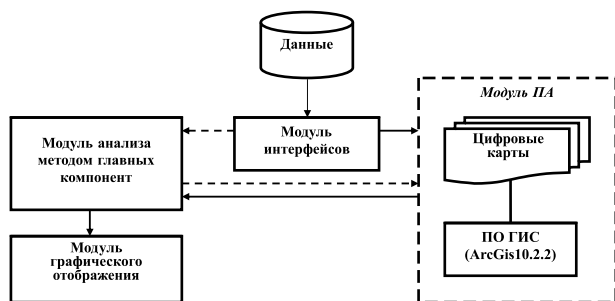


Рис. 1. Схема комплексного анализа многомерных данных на основе сочетания методов главных компонент и пространственного анализа

На рис. 1 приведена обобщенная схема проведения комплексного анализа многомерных данных о пространственно-распределенных объектах.

Модуль анализа МГК включает (рис. 1):

- процедуры нормирования и стандартизации исходного массива данных;
- вычисление ковариационной матрицы;
- нахождение собственных значений и собственных векторов ковариационной матрицы методом вращения Якоби;
- вычисление главных компонент;
- вычисление вклада каждой главной компоненты в общую дисперсию;
- анализ вкладов главных компонент в общую дисперсию и отбор главных компонент для проведения дальнейшего анализа;
- вычисление проекций исходных наблюдений (нагрузок) на отобранные главные компоненты;
- табличное представление нагрузок на отобранные главные компоненты.

Модуль пространственного анализа (ПА) данных включает:

- подготовку атрибутивной таблицы — создание массива исследуемых данных средствами ArcGis 10.2.2;
- отображение объектов на цифровых картах — создание цифрового слоя исследуемых объектов;
- выявление пространственных группирований объектов — в атрибутивной таблице каждому объекту приписывается номер класса, к которому он отнесен.

Модуль графического отображения включает:

- графическое отображение объектов исследования в пространстве двух главных компонент;
- графическое представление нагрузок на две главные компоненты;
- векторное представление нагрузок в пространстве двух главных компонент;
- графическое представление результатов анализа с отображением средних значений и доверительных областей.

Модуль интерфейсов предназначен для осуществления взаимодействия между такими разнородными частями системы, как модуль пространственного анализа (ПО ГИС) и модуль МГК-анализа (среда разработки Delphi).

Результаты комплексного анализа

В данном исследовании был проведен комплексный анализ состояния природной среды

Перечень субъектов РФ, включенных в анализ

Федеральный округ	№	Классификатор объектов административно-территориального деления
Сибирский федеральный округ	1	Алтайский край
	2	Забайкальский край
	3	Иркутская область
	4	Кемеровская область
	5	Красноярский край
	6	Новосибирская область
	7	Омская область
	8	Республика Алтай
	9	Республика Бурятия
	10	Республика Тыва
	11	Республика Хакасия
	12	Томская область
Дальневосточный федеральный округ	13	Амурская область
	14	Еврейская автономная область
	15	Камчатский край
	16	Магаданская область
	17	Приморский край
	18	Республика Саха (Якутия)
	19	Сахалинская область
	20	Хабаровский край
	21	Чукотский автономный округ

и здоровья населения территорий Сибири и Дальнего Востока. В таблице приведен перечень субъектов Российской Федерации, включенных в анализ. Открытые литературные источники и официальный сайт Федеральной службы государственной статистики [5] позволили собрать следующие количественные показатели о состоянии природной среды и здоровья населения исследуемых субъектов РФ в 2015 г. (в скобках указаны единицы измерения):

- 1) нарушено земель в связи с несельскохозяйственной деятельностью (тыс. га);
- 2) несельскохозяйственные земли, пройденные пожарами (га);
- 3) лесные земли, пройденные пожарами (га);
- 4) территория с погибшим молодняком (га);
- 5) территория со сгоревшим лесом на корню (м^3);
- 6) объем сброса загрязненных сточных вод по бассейнам рек и озер (млрд м^3);
- 7) объем загрязненных вод из сброшенных сточных вод (млн м^3);
- 8) количество образованных отходов производства и потребления за отчетный год (тыс. т);
- 9) количество выброшенных в атмосферу загрязняющих веществ, отходящих от стационарных источников (тыс. т);

10) количество выброшенных без очистки загрязняющих веществ за отчетный год организованными источниками загрязнения (тыс. т);

11) количество выброшенных загрязняющих веществ в атмосферу (тыс. т);

12) количество выброшенных без очистки загрязняющих веществ (тыс. т);

13) заболеваемость населения злокачественными новообразованиями на 100 тысяч населения (человек);

14) заболеваемость населения на 100 тысяч населения, с диагнозом, установленным впервые (человек);

15) заболеваемость населения, взятого под наблюдение в отчетном году с впервые установленным диагнозом на 100 тысяч населения (человек).

Возможности сайта Федеральной службы государственной статистики не позволили наглядно отобразить значения количественных показателей для всех исследуемых субъектов. В связи с отсутствием возможности отобразить в виде графика, гистограммы или картографического отображения по 15 количественным показателям о состоянии природной среды и здоровья населения 21 субъекта РФ была использована одна из функций сайта — выгрузка (экспорт) многомерных данных (рис. 2). Выбранные данные были выгружены (экспортированы) в виде таблицы "объект—свойство", где представлены объекты исследования — субъекты РФ (21 строка) и их свойства — количественные характеристики природных ресурсов и здоровья населения (15 столбцов).

На первом шаге проведения комплексного анализа многомерных данных проводился МГК-анализ статистических данных. Основ-

Территория	Выброшено в атмосферу загрязняющих веществ, отходящих от стационарных источников (тыс. тонн)	Выброшено без очистки наиболее распространенных загрязняющих веществ за отчетный год (тыс. тонн)
Алтайский край	255.159	25.829
Амурская область	91.752	16.083
Иркутская область	504.93	34.422
Камчатский край	46.821	9.072
Кемеровская область	1225.182	54.017
Красноярский край	2599.226	29.354
Магаданская область	36.817	5.976
Приморский край	351.936	16.399
Новосибирская область	198.049	18.572
Омская область	238.349	18.877
Сахалинская область	100.003	12.014
Томская область	254.598	24.747

Рис. 2. Возможности сайта Федеральной службы государственной статистики

ная идея МГК состоит в переходе к новой системе координат и оценке важности, информативности каждой из новых переменных — главных компонент. При проведении МГК-анализа вычисляются ортогональные векторы (главные компоненты), учитывающие максимальную долю дисперсии исходного распределения. Поскольку исходными данными для нахождения главных компонент являются выборочные коэффициенты корреляции, то очевидно, что необходимо проверить значимость, достоверность полученных результатов. Разработанный программный комплекс анализа многомерных данных (см. рис. 1) включает процедуру оценки информативности главных компонент, на основании чего можно говорить о степени целесообразности перехода к новым системам координат (в пространство главных компонент) и проведения дальнейшего МГК-анализа.

Для сформированной выборки результаты МГК-анализа показали, что первая (F1) и вторая (F2) главные компоненты описывают более 90 % общей дисперсии, что позволяет перейти к двумерному пространству главных компонент. Для изучения особенностей исследуемых объектов необходимо проанализировать значения нагрузок на главные компоненты.

Матрица нагрузок — это матрица перехода из исходного пространства переменных (15-мерного) в пространство главных компонент (двумерное). График нагрузок применяется для исследования роли переменных, их информативности и значимости [6]. Графическое представление нагрузок на главные компоненты позволило оценить влияние каждого количественного показателя анализируемого субъекта РФ на их распределение в пространстве главных компонент и определить набор наиболее информативных параметров. Анализ нагрузок на первую главную компоненту показал, что для данного набора показателей наиболее информативными (вариабельными) являются всего пять из пятнадцати количественных показателей состояния природной среды и здоровья населения субъектов РФ. Таким образом, в дальнейший анализ необходимо включать только эти пять характеристик, остальные можно не принимать в рассмотрение в силу их малой вариабельности от объекта к объекту. Алгоритм комплексного анализа позволил исключить из анализа незначимые характеристики, и для анализа состояния природной среды и здоровья населения Сибири, Дальнего Востока были отобраны пять наиболее значимых количественных показателей:

- 1) количество выброшенных в атмосферу загрязняющих веществ, отходящих от стационарных источников (тыс. т);
- 2) количество выброшенных без очистки загрязняющих веществ (тыс. т);
- 3) количество выброшенных без очистки загрязняющих веществ за отчетный год организованными источниками загрязнения (тыс. т);
- 4) объем загрязненных вод из сброшенных сточных вод (млн м³);
- 5) заболеваемость населения злокачественными новообразованиями на 100 тысяч населения (человек).

На рис. 3 показано взаимное распределение всех исследуемых субъектов РФ в двумерном пространстве главных компонент F1 и F2.

Из рис. 3 видно, что максимальными значениями первой главной компоненты обладают Красноярский край (5) и Кемеровская область (4), что говорит о высоких значениях всех количественных показателей состояния природной среды и состояния здоровья населения для данных субъектов.

На рис. 4 представлены значения нагрузок пяти количественных показателей для 21 субъекта РФ, где (по горизонтальной оси) 1 — количество выброшенных в атмосферу загрязняющих веществ, отходящих от стационарных источников (тыс. т); 2 — количество

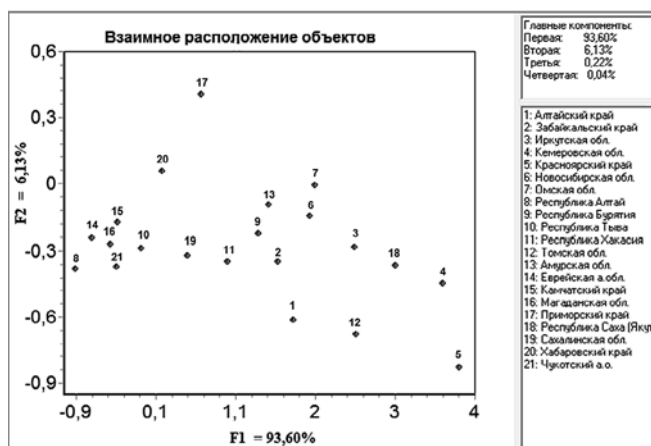


Рис. 3. Результаты МГК-анализа

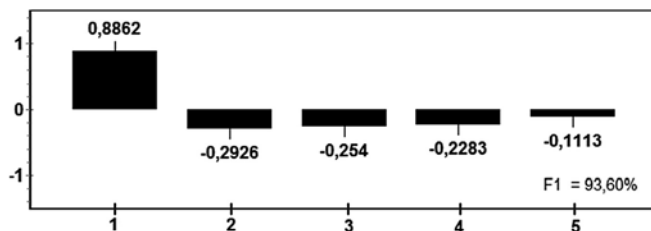


Рис. 4. Нагрузки на первую главную компоненту

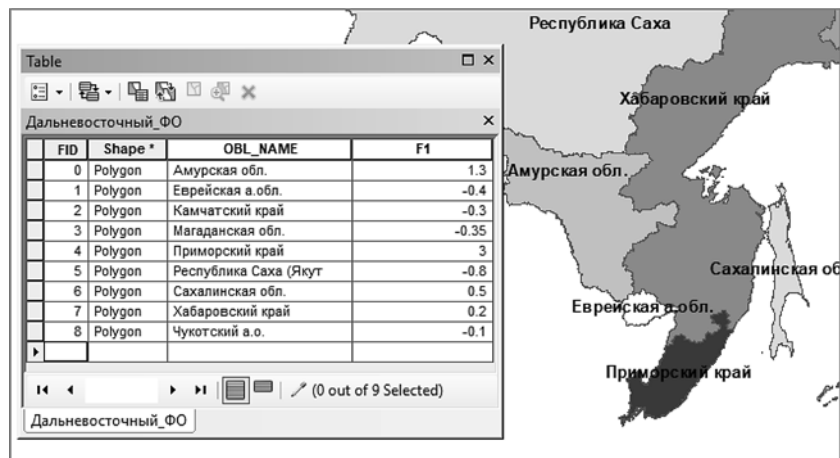


Рис. 5. Картирование результатов МГК-анализа данных о природной среде и состоянии здоровья населения субъектов РФ

выброшенных без очистки загрязняющих веществ (тыс. т); 3 — количество выброшенных без очистки загрязняющих веществ за отчетный год организованными источниками загрязнения (тыс. т); 4 — из сброшенных сточных вод — объем загрязненных (млн м³); 5 — заболеваемость населения злокачественными новообразованиями на 100 тысяч населения (человек).

Из рис. 4 видно, что наиболее значимым параметром для 21 субъекта РФ является характеристика — "количество выброшенных в атмосферу загрязняющих веществ, отходя-

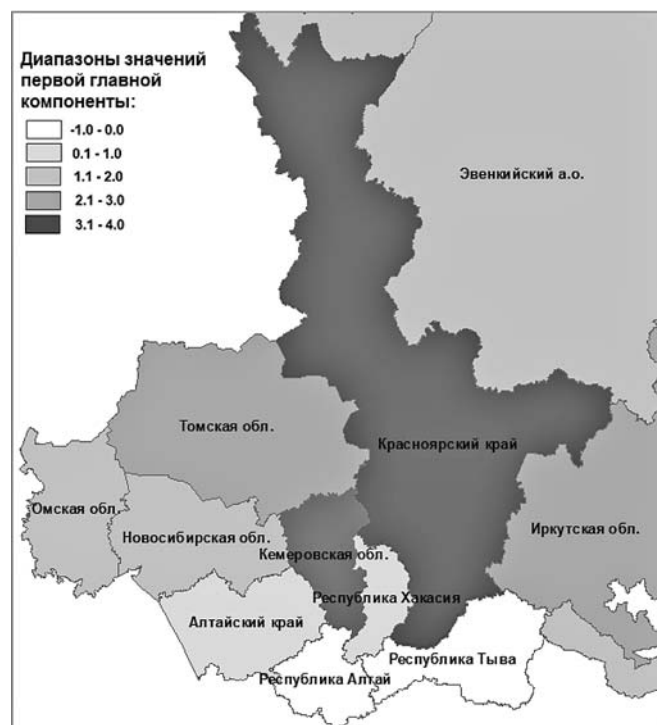


Рис. 6. Картирование результатов МГК-анализа

щих от стационарных источников (тыс. т)" с положительным значением 0,8862. Нагрузки для 2-, 3- и 4-го показателей практически одинаковы и имеют отрицательные значения от $-0,2926$ до $-0,2283$. Самая малая нагрузка с отрицательным значением $-0,1113$ имеет характеристика "заболеваемость населения злокачественными новообразованиями на 100 тысяч населения (человек)".

Средствами геоинформационной системы ArcGis 10.2.2 результаты МГК-анализа были добавлены (экспортированы) в атрибутивную таблицу [7] исследуемых субъектов РФ (рис. 5).

На рис. 6 показано картирование результатов МГК-анализа и ранжирование субъектов РФ Сибирского ФО по значению первой главной компоненты, которая характеризует пять количественных параметров состояния природной среды и здоровья населения. На рис. 6 наглядно отображено, что максимальные значения первой главной компоненты имеют субъекты Красноярский край и Кемеровская область, минимальные значения — Республика Алтай и Республика Тыва.

Пространственный анализ показал, что на каждой исследуемой территории (Сибирский ФО и Дальневосточный ФО) есть субъекты, отличающиеся максимальными значениями первой главной компоненты. Наиболее загрязненными является территории Красноярского края, Кемеровской области и Приморского края.

Заключение

При описании природных объектов в задачах мониторинга состояния окружающей среды и природопользования, как правило, используется большой объем пространственной информации. Традиционный метод главных компонент не позволяет учитывать пространственные свойства объектов, что требует разработки методических, алгоритмических и программных средств его использования в сочетании с методами пространственного анализа, реализуемыми с помощью ГИС-технологий.

Применение разработанного подхода к обработке многомерных данных о состоянии природной среды и здоровья населения Сибирского и Дальневосточного ФО позволило проанализи-

зировать особенности исследуемых территорий. Применение МГК позволило выявить наиболее значимые характеристики для территорий, а пространственный анализ дал возможность наглядно отобразить результаты МГК-анализа и выявить экологически благополучные и неблагополучные регионы.

Список литературы

1. **Abson D. J., Dougill A. J., Stringer L. C.** Using Principal Component Analysis for information-rich socio-ecological vulnerability mapping in Southern Africa // *Applied Geography*. 2012. Vol. 35, Iss. 1–2. P. 515–524. DOI: <https://doi.org/10.1016/j.envpol.2005.10.028>
2. **Андрукович П. Ф.** Применение метода главных компонент в практических исследованиях. М.: Изд-во МГУ, 1973. Вып. 36. 124 с.
3. **Zhang C.** Using multivariate analyses and GIS to identify pollutants and their spatial patterns in urban soils in Gal-

way, Ireland // *Environmental Pollution*. 2006. Vol. 142, Iss. 3. P. 501–511. DOI: <https://doi.org/10.1016/j.apgeog.2012.08.004>

4. **Суслов М. И., Булкина К. А., Бубнов А. Г., Буймова С. А., Царев Ю. В.** Статистический анализ данных экологического мониторинга родниковых вод урбанизированных территорий // *Южно-Сибирский научный вестник*. 2016. № 3(15). С. 15–19.

5. **Федеральная служба государственной статистики:** Окружающая среда. URL: http://www.gks.ru/wps/wcm/connect/rosstat_main/rosstat/ru/statistics/environment/ (дата обращения: 22.06.2018).

6. **Лучкова С. В., Перемитина Т. О., Ященко И. Г.** Использование нечеткого моделирования для повышения репрезентативности информации на примере анализа характеристик нефти // *Информационные технологии*. 2015. Т. 21, № 5. С. 336–340.

7. **Brewer C. A.** Basic Mapping Principles for Visualizing Cancer Data Using Geographic Information Systems (GIS) // *American Journal of Preventive Medicine*. 2006. Vol. 30, Iss. 2. P. S25–S36. DOI: <https://doi.org/10.1016/j.amepre.2005.09.007>

T. O. Peremitina, Ph.D. in Technical Science, e-mal: peremitinat@mail.ru,

I. G. Yashchenko, Ph.D. in Geology and Mineralogy, e-mal: sric@ipc.tsc.ru,

Institute of Petroleum Chemistry, Siberian Branch of the Russian Academy of Sciences

Algorithm for Complex Analysis of Environment Ecological State

There is considered methodical questions of realization and practical application of the complex approach to the analysis of multidimensional data on the ecological state of spatially distributed objects of research. The approach is based on a combination of the method of principal components and the method of spatial analysis using geoinformation systems. The principal components analysis (PCA) is used for statistical processing and data analysis, the method of spatial analysis is used to take into account the spatial properties of the objects under study. The article presents the results of the analysis of multidimensional data on the state of the natural environment and the health of the population of the territories of the Siberian and Far Eastern Federal Districts.

Keywords: multidimensional data, principal component analysis, geographic information systems, environment, ecology

DOI: 10.17587/it.25.58-63

References

1. **Abson D. J., Dougill A. J., Stringer L. C.** Using Principal Component Analysis for information-rich socio-ecological vulnerability mapping in Southern Africa, *Applied Geography*, 2012, vol. 35, iss. 1–2, pp. 515–524, DOI: <https://doi.org/10.1016/j.envpol.2005.10.028>
2. **Андрукович П. Ф.** *Применение метода главных компонент в практических исследованиях* (Application of principal components analysis (PCA) in practical research), Moscow, Publishing house of MGU, 1973, iss. 36, 124 p. (in Russian).
3. **Zhang C.** Using multivariate analyses and GIS to identify pollutants and their spatial patterns in urban soils in Galway, Ireland, *Environmental Pollution*, 2006, vol. 142, iss. 3, pp. 501–511, DOI: <https://doi.org/10.1016/j.apgeog.2012.08.004>
4. **Suslov M. I., Bulkina K. A., Bubnov A. G., Bujmova S. A., Caryov Yu. V.** *Statisticheskij anali z dannyh ehkologicheskogo monitoringa rodnikovyh vod urbanizirovannyh territorij* (Statistical

analysis of data of environmental monitoring of spring water in urban areas), *YUzhno-Sibirskij nauchnyj vestnik*, 2016, no. 3(15), pp. 15–19 (in Russian).

5. **Federal'naya sluzhba gosudarstvennoj statistiki: Okruzhayushchaya sreda** (Federal state statistics service), available at: http://www.gks.ru/wps/wcm/connect/rosstat_main/rosstat/ru/statistics/environment/ (data obrashcheniya: 22.06.2018) (in Russian).

6. **Luchkova S. V., Peremitina T. O., Yashchenko I. G.** *Ispol'zovanie nechetkogo modelirovaniya dlya povysheniya reprezentativnosti informacii na primere analiza harakteristik nefi* (Usage fuzzy modelling for raising a sample representative on the example the analysis of characteristics oils), *Informacionnye Tekhnologii*, 2015, vol. 21, no. 5, pp. 336–340 (in Russian).

7. **Brewer C. A.** Basic Mapping Principles for Visualizing Cancer Data Using Geographic Information Systems (GIS), *American Journal of Preventive Medicine*, 2006, vol. 30, iss. 2, pp. S25–S36. DOI: <https://doi.org/10.1016/j.amepre.2005.09.007>

3—7 июня 2019 года,
Санкт-Петербургский политехнический университет Петра Великого (СПбПУ),
Санкт-Петербург, Россия

21—25 октября 2019 года,
Белорусский национальный технический университет (БНТУ),
Минск, Беларусь

30 октября — 1 ноября 2019 года,
Саратовский государственный технический университет имени Гагарина Ю. А.,
Саратов, Россия

**Распределенная во времени и пространстве конференция
XXXII Международная научная конференция**

"МАТЕМАТИЧЕСКИЕ МЕТОДЫ В ТЕХНИКЕ И ТЕХНОЛОГИЯХ — ММТТ-32"

1. Качественные и численные методы исследования дифференциальных и интегральных уравнений.
 2. Оптимизация, автоматизация и оптимальное управление технологическими процессами.
 3. Математическое моделирование технологических и социальных процессов.
 4. Математическое моделирование и оптимизация в задачах САПР, аддитивных технологий.
 5. Математические методы в задачах радиотехники, радиоэлектроники и телекоммуникаций, геоинформатики, авионики и космонавтики.
 6. Математические методы и интеллектуальные системы в робототехнике и мехатронике.
 7. Математические методы в медицине, биотехнологии и экологии.
 8. Математические методы в экономике и гуманитарных науках.
 9. Информационные и интеллектуальные технологии в технике и образовании.
 10. Математические и инструментальные методы технологий Индустрии 4.0.
 11. Обсуждение квалификационных работ.
- Школа молодых ученых — ШМУ. Конкурс УМНИК.

**Подробная информация о конференции и условиях участия в ней
размещена на сайте <http://mmtt.sstu.ru/>**

Адрес редакции:

107076, Москва, Стромынский пер., 4

Телефон редакции журнала **(499) 269-5510**

E-mail: it@novtex.ru

Технический редактор *Е. В. Конова*.

Корректор *Н. В. Яшина*.

Сдано в набор 08.11.2018. Подписано в печать 25.12.2018. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ ИТ119. 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.

Рисунки к статье М. В. Сарамуда, И. В. Ковалева, В. В. Лосева,
М. О. Петросян, А. О. Калинина

«СРАВНЕНИЕ ОТКАЗОУСТОЙЧИВЫХ МОДЕЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ИМИТАЦИОННОЙ СРЕДЕ ИСПОЛНЕНИЯ»

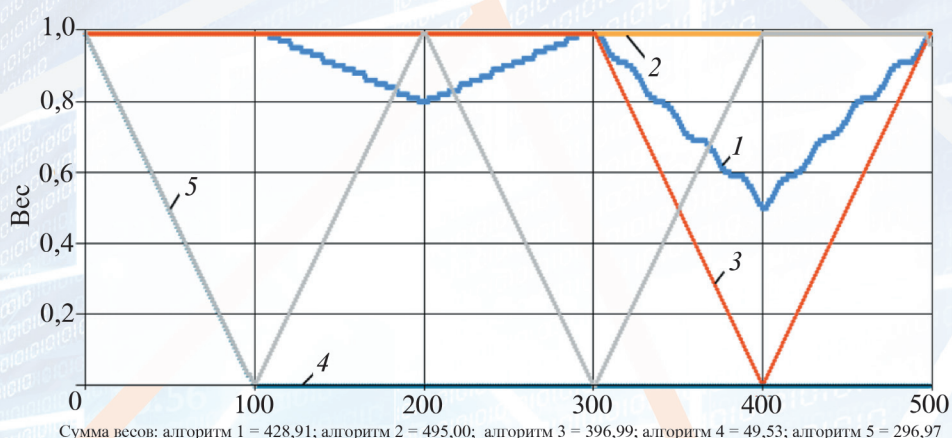


Рис. 3. Результат работы среды при $E = 0,045$:

1 – алгоритм 1; 2 – алгоритм 2; 3 – алгоритм 3; 4 – алгоритм 4; 5 – алгоритм 5

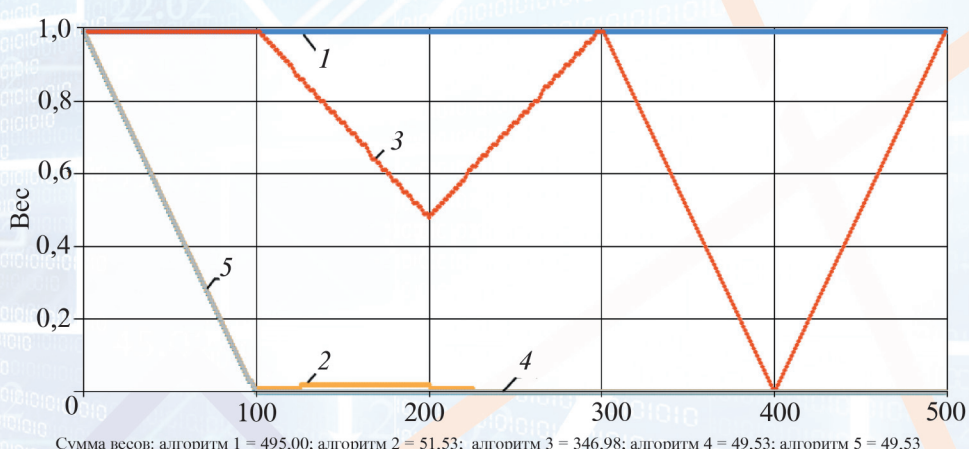


Рис. 4. Результат работы среды при $E = 0,001$:

1 – алгоритм 1; 2 – алгоритм 2; 3 – алгоритм 3; 4 – алгоритм 4; 5 – алгоритм 5

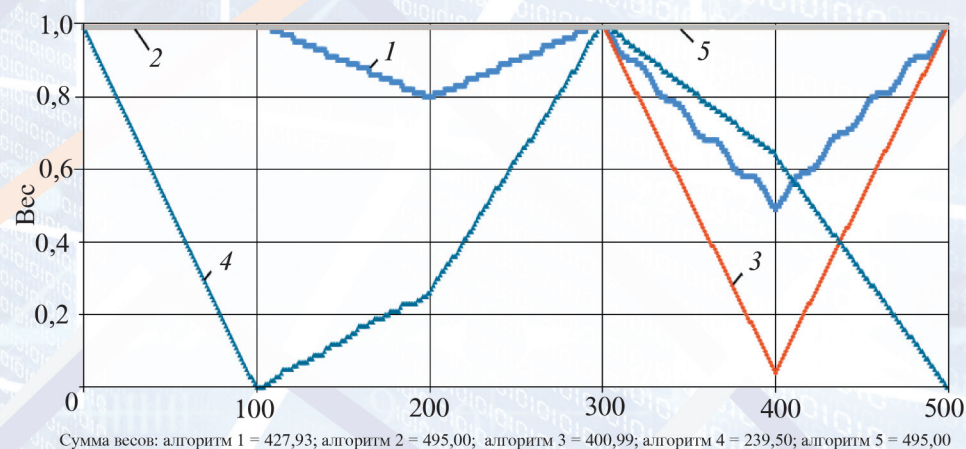


Рис. 5. Результат работы среды при $E = 3$:

1 – алгоритм 1; 2 – алгоритм 2; 3 – алгоритм 3; 4 – алгоритм 4; 5 – алгоритм 5

Издательство «НОВЫЕ ТЕХНОЛОГИИ» выпускает научно-технические журналы



Ежемесячный теоретический и прикладной научно-технический журнал **ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ**

В журнале освещаются современное состояние, тенденции и перспективы развития основных направлений в области разработки, производства и применения информационных технологий.

Подписной индекс по Объединенному каталогу
«Пресса России» – 72656



Научно-практический
и учебно-методический журнал

БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ

В журнале освещаются достижения и перспективы в области исследований, обеспечения и совершенствования защиты человека от всех видов опасностей производственной и природной среды, их контроля, мониторинга, предотвращения, ликвидации последствий аварий и катастроф, образования в сфере безопасности жизнедеятельности.

Подписной индекс по
Объединенному каталогу
«Пресса России» – 79963

Ежемесячный
междисциплинарный
теоретический и прикладной
научно-технический журнал

НАНО- и МИКРОСИСТЕМНАЯ ТЕХНИКА

В журнале освещаются современное состояние, тенденции и перспективы развития нано- и микросистемной техники, рассматриваются вопросы разработки и внедрения нано микросистем в различные области науки, технологии и производства.



Подписной индекс по
Объединенному каталогу
«Пресса России» – 79493



Ежемесячный теоретический
и прикладной
научно-технический журнал

МЕХАТРОНИКА, АВТОМАТИЗАЦИЯ, УПРАВЛЕНИЕ

В журнале освещаются достижения в области мехатроники, интегрирующей механику, электронику, автоматику и информатику в целях совершенствования технологий производства и создания техники новых поколений. Рассматриваются актуальные проблемы теории и практики автоматического и автоматизированного управления техническими объектами и технологическими процессами в промышленности, энергетике и на транспорте.

Подписной индекс по
Объединенному каталогу
«Пресса России» – 79492

Теоретический
и прикладной
научно-технический журнал

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

В журнале освещаются состояние и тенденции развития основных направлений индустрии программного обеспечения, связанных с проектированием, конструированием, архитектурой, обеспечением качества и сопровождением жизненного цикла программного обеспечения, а также рассматриваются достижения в области создания и эксплуатации прикладных программно-информационных систем во всех областях человеческой деятельности.



Подписной индекс по
Объединенному каталогу
«Пресса России» – 22765

Адрес редакции журналов для авторов и подписчиков:

107076, Москва, Стромынский пер., 4. Издательство "НОВЫЕ ТЕХНОЛОГИИ".
Тел.: (499) 269-55-10, 269-53-97. Факс: (499) 269-55-10. E-mail: antonov@novtex.ru