

РОССИЙСКАЯ АКАДЕМИЯ НАУК
Отделение нанотехнологий и информационных технологий

САНКТ-ПЕТЕРБУРГСКИЙ
ИНСТИТУТ ИНФОРМАТИКИ И АВТОМАТИЗАЦИИ РАН

ТРУДЫ СПИИРАН

proceedings.spiiras.nw.ru



ВЫПУСК 5(42)



Санкт Петербург
2015

18+

Труды СПИИРАН

Выпуск № 5(42), 2015

Научный, научно-образовательный, междисциплинарный журнал с базовой специализацией в области информатики, автоматизации и прикладной математики

Журнал основан в 2002 году

Учредитель и издатель

Федеральное государственное бюджетное учреждение науки
Санкт-Петербургский институт информатики и автоматизации Российской академии наук
(СПИИРАН)

Главный редактор

Р.М. Юсупов, чл.-корр. РАН, д-р техн. наук, проф., С-Петербург, РФ

Редакционная коллегия

А.А. Ашимов, академик национальной академии наук Республики Казахстан д-р техн. наук, проф., Алматы, Казахстан

С.Н. Баранов, д-р физ.-мат. наук, проф., С.-Петербург, РФ

Н.П. Веселкин, академик РАН, д-р мед. наук, проф., С.-Петербург, РФ

В.И. Городецкий, д-р техн. наук, проф., С.-Петербург, РФ

О.Ю. Гусихин, Ph.D., Диаборн, США

В. Делич, д-р техн. наук, проф., Нови-Сад, Сербия

А.Б. Долгий, Dr. Habil., проф., Сент-Этьен, Франция

М. Железны, Ph.D., доцент, Пльзень, Чешская республика

Д.А. Иванов, д-р экон. наук, проф., Берлин, Германия

И.А. Каляев, д-р техн. наук, профессор, член-корреспондент РАН, Таганрог, РФ

Г.А. Леонов, член-корр. РАН, д-р физ.-мат. наук, проф., С.-Петербург, РФ

К.П. Марков, Ph.D., доцент, Аизу, Япония

Ю.А. Меркурьев, академик Латвийской академии наук, Dr. Habil., проф., Рига, Латвия

Р.В. Мещеряков, д-р техн. наук, профессор, Томск, РФ

Н.А. Молдовян, д-р техн. наук, проф., С.-Петербург, РФ

В.Е. Павловский, д-р физ.-мат. наук, профессор, Москва, РФ

А.А. Петровский, д-р техн. наук, проф., Минск, Беларусь

В.А. Путилов, д-р техн. наук, проф., Апатиты, РФ

В.Х. Пшихопов, д-р техн. наук, профессор, Таганрог, РФ

А.Л. Ронжин (зам. главного редактора), д-р техн. наук, проф., С.-Петербург, РФ

А.И. Рудской, член-корр. РАН, д-р техн. наук, проф., С.-Петербург, РФ

В. Сгурев, академик Болгарской академии наук, д-р техн. наук, проф., София, Болгария

В.А. Скормин, Ph.D., проф., Бингемптон, США

А.В. Смирнов, д-р техн. наук, проф., С.-Петербург, РФ

Б.Я. Советов, академик РАН, д-р техн. наук, проф., С.-Петербург, РФ

В.А. Сойфер, член-корр. РАН, д-р техн. наук, проф., Самара, РФ

Б.В. Соколов, д-р техн. наук, проф., С.-Петербург, РФ

Л.В. Уткин, д-р техн. наук, проф., С.-Петербург, РФ

А.Л. Фрадков, д-р техн. наук, проф., С.-Петербург, РФ

Н.В. Хованов, д-р физ.-мат. наук, проф., С.-Петербург, РФ

Л.Б. Шереметов, д-р техн. наук, Мехико, Мексика

А.В. Язенин, д-р техн. наук, профессор, Тверь, РФ

Адрес редакции

191778, Санкт-Петербург, 14-я линия, д. 39,

e-mail: publ@iias.spb.su, сайт: <http://www.proceedings.spiiras.nw.ru/>

Подписано к печати 15.10.2015. Формат 60×90 1/16. Усл. печ. л. 14,2. Заказ № 322. Тираж 200 экз., цена свободная
Отпечатано в Редакционно-издательском центре ГУАП, 190000, Санкт-Петербург, Б. Морская, д. 67

Журнал зарегистрирован Федеральной службой по надзору в сфере связи и массовых коммуникаций,
свидетельство ПИ № ФС77-41695 от 19 августа 2010 г.

Подписной индекс 29393 по каталогу «Почта России»

Журнал входит в «Перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные научные результаты диссертации на соискание ученой степени доктора и кандидата наук»

© Федеральное государственное бюджетное учреждение науки

Санкт-Петербургский институт информатики и автоматизации Российской академии наук, 2015

Разрешается воспроизведение в прессе, а также сообщение в эфир или по кабелю опубликованных в составе печатного периодического издания-журнала «Труды СПИИРАН» статей по текущим экономическим, политическим, социальным и религиозным вопросам с обязательным указанием имени автора статьи и печатного периодического издания-журнала «Труды СПИИРАН»

SPIIRAS Proceedings

Issue № 5(42), 2015

Scientific, educational, and interdisciplinary journal primarily specialized
in computer science, automation, and applied mathematics

Trudy SPIIRAN ♦ Founded in 2002 ♦ Труды СПИИРАН

Founder and Publisher

Federal State Budget Institution of Science

St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences
(SPIIRAS)

Editor-in-Chief

R.M. Yusupov, Prof., Dr. Sci., Corr. Member of RAS, St. Petersburg, Russia

Editorial Board Members

A.A. Ashimov, Prof., Dr. Sci., Academician
of the National Academy of Sciences of the
Republic of Kazakhstan, Almaty, Kazakhstan
S.N. Baranov, Prof., Dr. Sci., St. Petersburg, Russia
N.P. Veselkin, Prof., Dr. Sci., Academician of RAS,
St. Petersburg, Russia
V.I. Gorodetski, Prof., Dr. Sci., St. Petersburg, Russia
O.Yu. Gusikhin, Ph. D., Dearborn, USA
V. Delic, Prof., Dr. Sci., Novi Sad, Serbia
A. Dolgui, Prof., Dr. Habil., St. Etienne, France
M. Zelezny, Assoc. Prof., Ph.D., Plzen, Czech
Republic
I.A. Kalyaev, Prof., Dr. Sci., Corr. Member of RAS,
Taganrog, Russia
D.A. Ivanov, Prof., Dr. Habil., Berlin, Germany
G.A. Leonov, Prof., Dr. Sci., Corr. Member of RAS,
St. Petersburg, Russia
K.P. Markov, Assoc. Prof., Ph.D., Aizu, Japan
Yu.A. Merkurjev, Prof., Dr. Habil., Academician
of the Latvian Academy of Sciences, Riga, Latvia
R.V. Meshcheryakov, Prof., Dr. Sci., Tomsk, Russia
N.A. Moldovian, Prof., Dr. Sci., St. Petersburg, Russia
V.E. Pavlovskiy, Prof., Dr. Sci., Moscow, Russia
A.A. Petrovsky, Prof., Dr. Sci., Minsk, Belarus

V.A. Putilov, Prof., Dr. Sci., Apatity, Russia
V.K. Pshikhopov, Prof., Dr. Sci., Taganrog, Russia
A.L. Ronzhin (Deputy Editor-in-Chief),
Prof., Dr. Sci., St. Petersburg, Russia
A.I. Rudskoi, Prof., Dr. Sci., Corr. Member of RAS,
St. Petersburg, Russia
V. Sgurev, Prof., Dr. Sci., Academician
of the Bulgarian academy of sciences, Sofia,
Bulgaria
V. Skormin, Prof., Ph.D., Binghamton, USA
A.V. Smirnov, Prof., Dr. Sci., St. Petersburg, Russia
B.Ya. Sovetov, Prof., Dr. Sci., Academician of RAE,
St. Petersburg, Russia
V.A. Soyfer, Prof., Dr. Sci., Corr. Member of RAS,
Samara, Russia
B.V. Sokolov, Prof., Dr. Sci., St. Petersburg, Russia
L.V. Utkin, Prof., Dr. Sci., St. Petersburg, Russia
A.L. Fradkov, Prof., Dr. Sci., St. Petersburg, Russia
N.V. Hovanov, Prof., Dr. Sci., St. Petersburg,
Russia
L.B. Sheremetov, Assoc. Prof., Dr. Sci., Mexico,
Mexico
A.V. Yazenin, Prof., Dr. Sci. Tver, Russia

Editorial Board's address

14-th line VO, 39, SPIIRAS, St. Petersburg, 199178, Russia,

e-mail: publ@iias.spb.su, web: <http://www.proceedings.spiiras.nw.ru/>

Signed to print 15.10.2015

Printed in Publishing center GUAP, 67, B. Morskaya, St. Petersburg, 190000, Russia

The journal is registered in Russian Federal Agency for Communications and Mass-Media Supervision,
certificate ПИ № ФС77-41695 dated August 19, 2010 r.

Subscription Index 29393, Russian Post Catalog

© Federal State Budget Institution of Science

St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, 2015

СОДЕРЖАНИЕ

Гейда А.С., Лысенко И.В., Юсупов Р.М. ОСНОВНЫЕ КОНЦЕПТЫ И ПРИНЦИПЫ ИССЛЕДОВАНИЯ ОПЕРАЦИОННЫХ СВОЙСТВ ИСПОЛЬЗОВАНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	5
Егоров А.Н., Кузнецов В.А., Марлей В.Е., Назаргулов И.А. МОДЕЛЬ РАСПАРалЛЕЛИВАНИЯ ВЫЧИСЛЕНИЙ ДЛЯ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ВОССТАНОВЛЕНИЯ ДОСТУПА К ДАННЫМ В КОРПОРАТИВНЫХ СЕТЯХ	37
Диковицкий В.В. МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОЙ ОБРАБОТКИ И ПРЕДСТАВЛЕНИЯ ИНФОРМАЦИИ В МУЛЬТИПРЕДМЕТНЫХ ИНФОРМАЦИОННЫХ СИСТЕМАХ ПРОМЫШЛЕННЫХ ПРЕДПРИЯТИЙ	56
Тараканов О.В. СРАВНИТЕЛЬНОЕ ИССЛЕДОВАНИЕ СПОСОБОВ ХРАНЕНИЯ БОЛЬШИХ ДВОИЧНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ В БАЗАХ ДАННЫХ ORACLE	77
Кофнов О.В. МОДЕЛЬ И АЛГОРИТМЫ ИЗМЕРЕНИЯ ГЕОМЕТРИЧЕСКИХ ПАРАМЕТРОВ СТРУКТУР ТЕКСТИЛЬНЫХ МАТЕРИАЛОВ	90
Батенков К.А. МОДЕЛИРОВАНИЕ И СИНТЕЗ ЛИНЕЙНЫХ ДИСКРЕТНЫХ ОТОБРАЖЕНИЙ НЕПРЕРЫВНЫХ КАНАЛОВ СВЯЗИ	112
Михайлов С.Н., Малашенко О. И., Зайцева А.А. МЕТОДИКА ИНФОЛОГИЧЕСКОГО АНАЛИЗА СЕМАНТИЧЕСКОГО СОДЕРЖАНИЯ ОБРАЩЕНИЙ ПАЦИЕНТОВ ДЛЯ ОРГАНИЗАЦИИ ЭЛЕКТРОННОЙ ЗАПИСИ	140
Бирюков Д.Н., Ломако А.Г. ДЕНОТАЦИОННАЯ СЕМАНТИКА КОНТЕКСТОВ ЗНАНИЙ ПРИ ОНТОЛОГИЧЕСКОМ МОДЕЛИРОВАНИИ ПРЕДМЕТНЫХ ОБЛАСТЕЙ КОНФЛИКТА	155
Глыбовский П.А., Глухов А.П., Пономарев Ю.А., Шиленков М.В. ПОДХОД К ОЦЕНИВАНИЮ И ПРОГНОЗИРОВАНИЮ УРОВНЯ ЗАЩИЩЕННОСТИ ИНФОРМАЦИОННЫХ И ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМ	180
Бойко А.А. СПОСОБ АНАЛИТИЧЕСКОГО МОДЕЛИРОВАНИЯ ПРОЦЕССА РАСПРОСТРАНЕНИЯ ВИРУСОВ В КОМПЬЮТЕРНЫХ СЕТЯХ РАЗЛИЧНОЙ СТРУКТУРЫ	196
Шудрак М.О., Золотарев В.В. МОДЕЛЬ, АЛГОРИТМЫ И ПРОГРАММНЫЙ КОМПЛЕКС АВТОМАТИЗИРОВАННОГО ПОИСКА Уязвимостей в ИСПОЛНЯЕМОМ КОДЕ	212
Коломеец М.В., Чечулин А.А., Котенко И.В. ОБЗОР МЕТОДОЛОГИЧЕСКИХ ПРИМИТИВОВ ДЛЯ ПОЭТАПНОГО ПОСТРОЕНИЯ МОДЕЛИ ВИЗУАЛИЗАЦИИ ДАННЫХ	232

CONTENTS

Geida A.S., Lysenko I.V., Yusupov R.M. MAIN CONCEPTS AND PRINCIPLES FOR INFORMATION TECHNOLOGIES OPERATIONAL PROPERTIES RESEARCH	5
Yegorov A.N., Kuznetsov V.A., Marley V.Ye., Nazargulov I.A. THE MODEL OF PARALLEL COMPUTING TO EFFECTIVENESS IMPROVEMENT OF THE RESTORATION OF ACCESS TO DATA IN CORPORATE NETWORKS	37
Dikovitsky V.V. METHODS OF INTELLECTUAL DATA PROCESSING AND PRESENTATION IN MULTI- SUBJECT INFORMATION SYSTEMS OF INDUSTRIAL ENTERPRISES	56
Tarakanov O.V. A COMPARATIVE RESEARCH INTO WAYS TO STORE BINARY LARGE OBJECTS IN THE ORACLE DATABASES	77
Kofnov O.V. A MODEL AND ALGORITHMS FOR MEASUREMENT OF GEOMETRIC PARAMETERS IN TEXTILE STRUCTURES	90
Batenkov K.A. MODELING AND SYNTHESIS OF DISCRETE MAPPINGS OF LINEAR CONTINUOUS CHANNEL	112
Mikhailov S.N., Malashenko O.I., Zaytseva A.A. THE METHOD FOR THE INFOLOGY ANALYSIS OF PATIENTS COMPLAINTS SEMANTIC CONTENT IN ORDER TO ORGANIZE THE ELECTRONIC APPOINTMENTS	140
Biryukov D.N., Lomako A.G. DENOTATIONAL SEMANTICS OF KNOWLEDGE CONTEXTS IN ONTOLOGICAL MODELING OF SUBJECT DOMAINS OF THE CONFLICT	155
Glybovsky P.A., Gluhov A.P., Ponomarev Yu.A., Shilenkov M.V. APPROACH TO THE EVALUATION AND PREDICTION OF THE LEVEL OF SECURITY OF INFORMATION AND TELECOMMUNICATIONS SYSTEMS	180
Boyko A.A. METHOD OF ANALYTICAL MODELING OF VIRUSES PROPAGATION PROCESS IN COMPUTER NETWORKS WITH DIFFERENT TOPOLOGY	196
Shudrak M.O., Zolotarev V.V. A MODEL, ALGORITHMS AND SOFTWARE TOOL FOR VULNERABILITIES DETECTION IN MACHINE CODE	212
Kolomeec M.K., Chechulin A.A., Kotenko I.V. REVIEW OF METHODOLOGICAL PRIMITIVES FOR THE PHASED CONSTRUCTION OF DATA VISUALIZATION MODEL	232

А.С. ГЕЙДА, И.В. ЛЫСЕНКО, Р.М. ЮСУПОВ
**ОСНОВНЫЕ КОНЦЕПТЫ И ПРИНЦИПЫ ИССЛЕДОВАНИЯ
ОПЕРАЦИОННЫХ СВОЙСТВ ИСПОЛЬЗОВАНИЯ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

Гейда А.С., Лысенко И.В., Юсупов Р.М. Основные концепты и принципы исследования операционных свойств использования информационных технологий.

Аннотация. В статье рассмотрены основные концепты и принципы исследования операционных свойств использования информационных технологий. Исследование предполагается осуществить путем решения математических задач оценивания, анализа и синтеза по показателям операционных свойств, определяемых на основе прогнозных математических моделей эффектов использования информационных технологий на практике. Для определения таких показателей выполнен обзор подходов к описанию операционных свойств использования информационных технологий в литературе, анализ существующих практик исследования таких свойств, вскрыты основные закономерности проявления исследуемых свойств, обоснованы основные концепты и принципы исследования операционных свойств использования информационных технологий, разработана схема формирования эффектов использования информационных технологий, обоснована концептуальная схема оценивания исследуемых свойств. Полученные результаты позволяют перейти к определению необходимых показателей, рассчитываемых на математических моделях, к разработке концептуальной и математической постановок задач исследования и к их решению, как математических задач оценивания, анализа и синтеза.

Ключевые слова: операционные свойства, информационные технологии, эффективность, производительность, потенциал системы, автоматизация, потенциальные состояния, оценивание, цели, показатели, требования.

Geida A.S., Lysenko I.V., Yusupov R.M. Main Concepts and Principles for Information Technologies Operational Properties Research.

Abstract. Main concepts and principles for research problems of operational properties of information technologies statement and the solution as mathematical tasks are considered in the article. Such problems solutions could be based on properties indicators under study defined with use of prognostic mathematical models. For such indicators definition following results were obtained: the review of approaches to the description of operational properties of information technologies in literature; the practical analysis of such properties; properties main features; the main concepts and the operational properties of information technologies principles research; the scheme of effects of information technologies formation; the conceptual scheme of under study properties estimation. The results obtained allow passing to definition of indicators, calculated with use of mathematical models, passing to development of conceptual and mathematical statements of research problems under study, and to the solution of these tasks, as mathematical problems of estimation, analysis and synthesis of operational properties of information technologies.

Keywords: operational properties; efficiency; effectiveness; efficacy; productivity; performance; potential; potentiality; capabilities; capability; system potential; estimation; goals; indicators; requirements.

1. Введение. Попытки оценивания операционных свойств использования информационных технологий (ИТ), например, свойств, называемых исследователями «эффективность ИТ», «производитель-

ность ИТ», «результативность ИТ» вызывают много вопросов и дискуссий о том, что, как и почему следует оценивать при применении ИТ, каковы результаты ИТ. К сожалению, на многие из них пока нет таких ответов, которые бы позволили научно обоснованно, прагматично и конструктивно перейти к решению задач исследования указанных свойств на прогнозных математических моделях, а затем – к решению задач оценивания, анализа и синтеза по показателям этих свойств, как математических задач (например, оптимизационных). В статье сделана попытка вскрыть источники сложностей при поиске ответов на такие вопросы, а затем предложены концепты, принципы и схема оценивания показателей операционных свойств использования ИТ.

Основные понятия об операционных свойствах (ОС) систем и их функционирования, их особенности и отношения между ними были описаны в [1–5]. Эти определения используются, как базовые для исследования ОС использования ИТ [6,7]. Так, свойство, характеризующее приспособленность деятельности давать (в процессе использования для этой деятельности различных систем и реализации различных процессов их функционирования) требуемые эффекты называют *ОС деятельности*. *Примеры эффектов* – прибыль, затраты времени, затраты ресурсов. Давать *требуемые эффекты* означает, что характеристики эффектов должны находиться в требуемых отношениях между собой и (или) с характеристиками требований к этим эффектам. ИТ представляют собой, в своей сущности, способы реализации особого вида деятельности, состоящего в оперировании информацией о других (в частности, о предметно-преобразующих) видах деятельности, причем целью такого оперирования является совершенствование отражаемых с использованием этой информации объектов деятельности – так, чтобы достигались новые и(или) совершенствовалось достижение существующих целей субъектов деятельности. При этом достижение таких, новых целей в процессе постоянного совершенствования деятельности и улучшение достижения существующих (возможно, меняющихся) целей и есть положительный результат ИТ. ИТ могут быть реализованы либо только человеком, либо автоматизированно, то есть могут использовать для своей реализации еще и технические устройства, называемые средствами ИТ, или могут быть реализованы только техническими устройством, без участия человека, автоматически.

Для корректного введения понятий об ОС использования ИТ необходимо определить концепты, принципы такого исследования, описать механизм получения результатов ИТ на практике, определить концептуальную схему исследования ОС использования ИТ, основные требования к решению задач исследования ОС использования ИТ, а

затем выполнить постановку соответствующих задач и перейти к их решению.

2. Основные концепты использования информационных технологий. Ряд концептов использования информационных технологий уже имеет определения в нормативных документах. Рассмотрим их применимость к задачам исследования ОС использования ИТ.

Технология – способ преобразования вещества, энергии, информации в процессе изготовления продукции, обработки и переработки материалов, сборки готовых изделий, контроля качества, управления [8].

Способ – действие или система действий, применяемых при исполнении работы, при осуществлении чего-либо [9].

Информационная технология (ИТ) [10] – совокупность способов реализации информационных процессов в различных областях человеческой деятельности при производстве информационного продукта, или, в соответствии с ГОСТ серии 34 – приемы, способы и методы применения средств вычислительной техники при выполнении функций сбора, хранения, обработки, передачи и использования данных [из п. 4 Прил. 1 ГОСТ 34.003-90].

Представляется, что указанные определения обладают рядом недостатков, не позволяющих использовать их для исследования ОС использования ИТ. Так, не определено что такое информационные процессы в разных областях деятельности, зачем нужны информационные процессы и в каких именно видах деятельности, что у них общего, к чему ведут результаты ИТ при их использовании в деятельности, как информация связана с деятельностью. Чем способ отличается от деятельности. Для корректного вскрытия такого рода фундаментальных вопросов предлагается определить информацию и информационные технологии пользуясь базовыми философскими (а именно, гносеологическими) понятиями о мыслящем субъекте, объекте, окружающей природе, мыслях и затем, для исследования ОС использования ИТ, связать их с представлениями о деятельности и затем – ее эффективности [3,4].

Деятельность – совокупность действий, вид отношений субъекта с объектом, при реализации которых субъект достигает какие-либо цели, имеющиеся в его сознании. Субъект, объект, сознание, идеи – философские категории (фундаментальные понятия бытия).

Под *информацией* будем понимать форму существования идей, или мыслей (понятий, суждений, умозаключений) в материальном мире (то есть вне сознания).

Под *информационной технологией* будем понимать технологию действий (деятельности) с информацией. ИТ может реализоваться че-

ловеком или, с использованием других предметов, технических устройств.

Средства (реализации) технологии – предметы действительности (технические устройства, природные ресурсы), используемые для реализации технологии.

Способом действия (деятельности) назовем информацию о виде реализации какого-либо действия (деятельности). Тем самым, следует отличать способ, как информацию о действии и действие, реализуемое в соответствии с этим способом.

Технология – совокупность способов деятельности. Поэтому информационная технология – способы действий с информацией.

Результат использования ИТ – информация, требуемая людьми для какой-либо деятельности. Результат использования ИТ не обязательно представляется, как результат вычислений, более того, он может быть получен человеком без использования техники. Так, результатом использования ИТ является глиняная дощечка с описанием периодичности разливов рек Тигр и Евфрат, созданная жрецами Месопотамии.

Такое понимание информации и ИТ назовем прагматическим, т.е. относящимся к действию, служащим практике [11]. Оно является основой прагматического подхода к оцениванию ОС использования ИТ. *Прагматический подход* ограничивает представления о информации лишь той ее частью, что связана с деятельностью человека. Он позволяет конструктивно определить отношения ИТ с информацией, деятельностью, субъектом и объектом и затем, с ОС использования ИТ. Для получения результатов ИТ на практике информация должна быть использована при функционировании каких-либо систем, и *только по отношению к использованию на практике можно говорить об «ОС ИТ»*.

Оценивание ОС ИТ при использовании прагматического подхода основывается на конструктивном, основанном на описании природных закономерностей и соответствующих математических зависимостей, оценивании результатов использования информации в предметно-преобразующей деятельности. Деятельность реализуется субъектом в природе, на основе имеющейся у него свободы [12]. То, какой способ действия (в рамках свободы) выбирает человек, зависит от имеющейся у него информации о способах действий, об их возможных результатах и о своих целях. Триединство информации о этих аспектах действия – о способах действий, их результатах и целях назовем *информацией о действии*. Такая информация о действии, при этом, носит характер образов, символов, описаний, моделей – идеальных, а не природных объектов. При этом реализация ИТ позволяет получать дополнительную информацию о действиях, в частности – дополнительные способы

действий. Чем больше способов действий известно, тем больше возможностей выбора, больше возможностей получить требуемые для достижения целей (в том числе новых целей) эффекты, лучше операционные свойства. ИТ *закономерно ведут к расширению возможностей деятельности и именно этот результат – целевой эффект (ради которого на ИТ тратятся ресурсы).*

Достижимое на практике расширение возможностей деятельности в результате реализации ИТ проявляется в условиях ограничений, накладываемых действиями среды (природы, государства, коллектива, семьи) и в результате образуются возможности деятельности, которые, в соответствии с «capabilities approach» нобелевского лауреата 1998 г. А. Сена [13] и являются тем, *по чему следует судить о прогрессе.*

ОС описывают соответствие результатов действия в природе требованиям субъектов. Тем самым, для описания ОС ИТ необходимо, кроме природной, учесть идеальную составляющую действия и установить связи между информацией о действии, ее обработкой с использованием ИТ, получаемыми результатами такой обработки и тем, как затем человек использует информацию, что создает в природе, что он меняет в ней для действия, какие результаты дает действие тем или иным способом в природе, реализуемое с использованием полученной в результате ИТ информации. Практика свидетельствует, что при установлении таких связей возникает ряд существенных трудностей, ведущих к проявлению *проблемы оценивания ОС ИТ.*

3. Обзор исследований по проблеме оценивания операционных свойств информационных технологий. Проблема оценивания ОС ИТ – сложная научно-техническая проблема, научно обоснованное решение которой с опорой на математические прогнозные модели пока не получено. Об этом свидетельствуют, в частности, многочисленные и противоречивые публикации по проблеме. Так, например, [14, 15] отмечается, что *ИТ не ведут к улучшению производительности.* Многочисленные статистические исследования по проблеме не привели к однозначной трактовке влияния ИТ на результаты производства. В [14] автор сравнивает роль ИТ с ролью таких новых технологий, как передача и использование электроэнергии и железнодорожный транспорт. Однако, и для указанных технологий понятия о их ОС или о связях затрат на них с получаемыми предприятиями результатами не приводятся. В ряде работ [16–19] раскрыт «парадокс производительности ИТ»: при значительном и все возрастающем объеме расходов частных и государственных предприятий на ИТ нет возможности однозначно утвердительно ответить ни на вопрос о пользе, приносимой информационными технологиями, ни на вопрос о конкретных значениях отдачи

от вложений в информационные технологии. В работе [15] на основании исследований литературы по проблеме производительности ИТ (IT performance) сделан похожий вывод. Рядом авторов [18] отмечается, что выгоды от ИТ технологий могут не носить финансовый характер, результаты от внедрения могут проявляться в существенно различных областях деятельности, за счет внедрения ИТ часто создается пул неиспользуемых возможностей (“garbage can”), которые, с одной стороны, могут не использоваться, но с другой стороны, в ряде случаев при их отсутствии предприятия могут полностью терять конкурентоспособность и разоряться. При этом, использование различных возможностей характеризуется случайностями разной природы [18]. Отмечается, что «парадокс производительности» связан не с тем, что ИТ не дают результатов, а с тем, что эти результаты некорректно измеряются, что проявление этих результатов носит случайный характер и может протекать со значительным сдвигом во времени. Тем самым, существуют *концептуальные сложности* при описании того, что понимать под ОС технологии вообще и ИТ в частности, и за счет чего эти свойства проявляются.

При этом, имеются [20] указания на то, что ИТ не просто успешно используются, а что *именно ИТ и создают «новую экономику»* и именно ИТ привели к ряду явлений в экономике, которые не могли бы проявиться без существенного влияния эффектов ИТ. При этом, раз ИТ используются, то они должны приносить эффект и улучшать ОС. Однако, это ОС – не производительность, а другое по смыслу свойство, характеризующее возможности прогресса и развития за счет результатов ИТ. Так [15,21] отмечается, что результаты использования ИТ проявляются в экономике, но не в том виде, который следовало бы ожидать от повышения производительности труда, причем следующего непосредственно за инвестированием в ИТ. Так, улучшается удовлетворенность потребителей, расширяются возможности, достигаются новые бизнес цели, появляются новые рынки, наметился рост занятости и зарплат. В [22] проанализированы ожидания бизнеса от ИТ и указано, что к таким результатам относятся не столько улучшение бизнес-эффектов, сколько «улучшение бизнес-импульса» и обеспечение «информационного лидерства». Тем самым, результаты ИТ необходимо рассматривать *в связи с возможностями нововведений, развития, прогресса, расширения целей деятельности*. Они ведут к положительным результатам на практике, но мы не умеем оценивать эти результаты, хотя и *ощущаем в повседневной деятельности как возможности прогрессивных изменений, развитие, как возможности достижения новых целей*.

4. Исследования операционных свойств информационных технологий на основе «обобщений лучших практик». Имеется значительное число методов, методик, методологий совершенствования ОС и близких к ним свойств, основанных на обобщении практического опыта совершенствования функционирования систем (предприятий, организаций). Далее они называются «практиками» совершенствования ОС в связи с тем, что они, как правило, в большей мере опираются на опыт, апостериорную информацию из практики и их обобщение и в меньшей – на математические модели и методы исследований, позволяющие получить априорное описание эффектов. Тем самым, под «практиками» исследования ОС понимаются совокупность приемов, обобщающих имеющийся опыт и позволяющих решать некоторые задачи исследования ОС – как правило, на качественном уровне, а также за счет экспертного оценивания. Так, к «практикам» совершенствования ОС можно отнести управление эффективностью бизнеса (Business Performance Management – BPM), методы оценивания комплексной модели производительности и зрелости (Capability Maturity Model Integration – CMMI), методы построения и использования системы сбалансированных показателей (Balanced Score Card – BSC), методы оценки совокупной стоимости владения (Total Cost of Ownership – TCO) и другие практики совершенствования ОС, в том числе и ОС ИТ [23]. Практический и теоретический опыт в той или иной области, связанной, в том числе, и с исследованием ОС принято описывать с использованием сводов знаний (Body of Knowledge, BoK). Например, имеется такой свод знаний в области системной инженерии (System Engineering Body of Knowledge, SEBoK), в области управления проектами (Project Management Body of Knowledge, PMBoK). Оценивание эффективности (performance measurement) предполагается и при решении ряда задач исследования функционирования предприятий [24]. Исследуемое при этом операционное свойство отражает успешность действий, предпринимаемых на пути достижения целей организации. Однако, следует отметить, что оценивание эффективности в соответствии с указанными и многими другими практиками (используют еще понятие технологий), как правило, носит характер обобщения опыта, сбора и обработки апостериорных данных, а не вскрытия законов природы и проявлений эффектов, которое бы позволило строить *прогностические модели*.

При этом необходимость моделирования и оценивания показателей ОС с использованием математических моделей осознается большинством исследователей. Так, например, необходимость вскрытия зависимостей прогнозируемых значений показателей ОС от переменных и формализации решаемых задач совершенствования ОС, как

оптимизационных задач отмечается в [25] одним из известнейших практиков исследования ОС Г. Кокинзом. В [26] он отмечает, что математическое моделирование критично для улучшения решений, совершенствования, оптимизации функционирования. К сожалению, такого рода моделей, позволяющих описать комплексы преобразований эффектов – от эффектов-результатов ИТ до природных эффектов, описываемых прогнозными моделями, еще не создано. Сложность построения таких прогнозных математических моделей и последующего решения задач исследования ОС на основе этих моделей – источник трудностей при переходе от «практик» к исследованию ОС на прогнозных моделях.

5. Основные закономерности проявления операционных свойств использования информационных технологий. В соответствии с принятым прагматическим подходом нами обоснованы закономерности проявления ОС использования ИТ. ИТ могут давать эффекты в природе лишь опосредованно, в результате использования результатов применения ИТ в том или ином процессе деятельности по использованию того или иного объекта природы (например, некоторой технической системы, процессов ее создания, изменения, обслуживания, процессах целевого функционирования для достижения различных актуализируемых целей). Это следует из сути ИТ, как технологий обработки информации (пусть и описываемых с использованием знаков в природе), а не технологий преобразования вещества и энергии, описываемыми, отражаемыми этой информацией – для последующего использования на практике. Сама технология описывается с использованием способов действий – информации о деятельности. Тем самым ИТ – информация о деятельности особого вида (об информационной деятельности), которая производит информацию о других преобразованиях в процессе отражаемой практической деятельности (например, предметно-преобразующей). ИТ предоставляют информацию об отражаемой деятельности. Использование ее на практике (от греч. *Praktikos* — деятельный, активный) может (при реализации комплексе условий и в результате реализации комплексов действий) наступить, а может и не наступить, что и представляется *первой закономерностью* проявления ОС ИТ. Проявившись в области идеального, новое реализуется на практике не всегда и для его реализации (практического воплощения) требуется ряд условий и комплексов действий. Так, для того, чтобы результаты ИТ вызвали результаты в природе, может быть необходим комплекс действий, вызывающий комплекс обмена результатами (эффектами) разного вида и кроме того, необходимо, чтобы в природе в результате этого комплекса обменов создавалось нечто но-

вое (новый процесс, новое действие, новая система) или совершенствовалось нечто существующее на практике, соответствующее результатам ИТ. Такой комплекс обменов далее назовем *цепочками эффектов*, поскольку нас интересуют все последовательности эффектов, ведущие к требуемым эффектам. В связи с получением таких цепочек эффектов говорят об *использовании ИТ* – процессах получения эффектов, соответствующих полученной информации. Использование ИТ следует отличать от внедрения ИТ, как процесса изменения способов обработки информации, используемых затем для получения эффектов на практике. Использование ИТ во многом аналогично использованию инноваций, модернизации, реализуемых в результате использования новых способов действий на практике. Такое использование новых способов действий на практике изучается, например, в рамках исследования диффузии инноваций (innovations diffusion) и диффузии ИТ (IT diffusion) [27,29].

Второй закономерностью проявления ОС ИТ представляется то, что даже если в действительности (вне сознания) реализовалось внедрение новых результатов ИТ, началась реализация полученного при использовании ИТ результата, это еще не значит, что результат будет пригоден к масштабному использованию. Преобразование результата ИТ в вид, позволяющий использовать его при функционировании ряда систем требует *конверсии* результатов ИТ и систем, их использующих [18] и лишь затем результат ИТ может быть использован в последующих процессах функционирования тех или иных систем (с которыми выполнена конверсия). *Конверсией (результатов ИТ и систем, их использующих)* будем называть подготовку к использованию результатов ИТ на практике, такую, что она направлена на начало достижения какой-либо актуализированной цели. Так, для использования результатов ИТ следует изменить элементы систем, их отношения, действия при функционировании и лишь тогда можно рассчитывать на получение на практике требуемых эффектов. Но даже и после конверсии получение требуемых эффектов может и не произойти, если, например, план функционирования для достижения требуемых эффектов будет сорван.

Заметим, что на практике часто используют термин «внедрение ИТ», под которым понимают процесс изменения способов оперирования информацией. Внедрение ИТ должно вести к цепочке эффектов: изменение ИТ (способов оперирования информацией) – использование ИТ (получение новых способов действий в природе) – конверсия – функционирование – новые эффекты. Тем самым, при исследовании ОС внедрения ИТ следует рассмотреть цепочку эффектов начиная от

процессов изменения способов обработки информации и затратами на такое изменение и заканчивая эффектами действий, реализуемых в результате использования новой ИТ.

Эффекты проявляются в процессе функционирования для достижения заданной цели. Приспособленность достигать такую заданную цель функционирования традиционно [1–3] исследуют с использованием свойства *эффективности функционирования*.

Приспособленность системы к достижению целей (многих, изменяющихся) своего функционирования традиционно [5] характеризуют с использованием *потенциала системы*. Потенциал системы определяется как каждой из эффективностей функционирования системы для достижения какой-либо цели, так и приспособленностью системы к изменениям, нововведениям для перехода к достижению новых целей.

Третья закономерность проявления ОС ИТ состоит в том, что ИТ, а затем конверсия могут приводить к улучшению результатов, которые достигаются не при «целевом» функционировании. Такие результаты часто ведут к прогрессивным изменениям в процессах совершенствования систем и их функционирования. Так, при внедрении часто оказывается, что действия, связанные с обработкой информации не относятся к действиям, лежащим на цепочках преобразования ресурсов в целевой результат. А именно, ИТ могут служить, например, лишь средством улучшить снабжение, управление, приспособленность к изменениям. При этом, для определения ОС ИТ следует сравнивать весь комплекс эффектов и их изменений, реализуемых в разное время и при разных изменениях систем и процессов их функционирования – с использованием ИТ и без, например, сравнивая потенциалы систем до и после использования ИТ.

Четвертая закономерность проявления ОС ИТ. Часто оказывается, что получаемый в результате использования ИТ на практике результат получен при достижении новой цели. Так, может оказаться, что этот результат принципиально не мог быть достигнут без ИТ. В этом случае новый процесс может иметь другую цель и сравнивать его со старым процессом некорректно. В результате ИТ позволяют удовлетворить другие (новые) потребности, достигать новые цели, реализован прогресс, развитие. А развитие следует измерять с учетом изменений достигаемых целей.

Пятая закономерность проявления ОС ИТ. Получаемые за счет ИТ новые способы при своем использовании в процессах деятельности могут приводить как к *уменьшению рисков* (к уменьшению числа возможностей неблагоприятных событий), так и к *увеличению рисков* (к появлению новых возможностей неблагоприятных событий). Так, на-

пример, уменьшение рисков может достигаться за счет лучшей информированности о действиях, а увеличение – за счет противодействия использованию новых возможностей действий [28], за счет появления новых неблагоприятных результатов действий, за счет проявления новых возможностей целенаправленного нанесения ущерба.

Приведенные основные закономерности использования ИТ позволяют вскрыть механизм формирования эффектов ИТ, как результатов оперирования информацией, сначала создающих новые способы действий, а затем использующих эти способы для получения новых или лучших эффектов уже в природе.

6. Механизм формирования эффектов использования ИТ и операционных свойств использования ИТ. *Операционным свойством использования ИТ* будем называть свойство использования заданной ИТ, характеризующее приспособленность использования ИТ давать требуемые эффекты. Эффекты использования ИТ получают при функционировании различных систем, в различных процессах функционирования которых используется заданная ИТ. В зависимости от того, соответствие каких эффектов каких функционирований систем с использованием ИТ и каким требованиям оценивается, различают разные ОС.

Следует отметить, что понятие ОС использования ИТ отличается от понятий эффективность целенаправленного процесса функционирования (ЦПФС) и потенциал системы [3,5,29], но использует их:

1. ИТ, несмотря на использование материальных носителей при обработке информации, реализуются со знаковыми системами, с образами, а не с прообразами (денотатами) знаков и не могут давать требуемые эффекты в природе непосредственно, без перехода к природным денотатам, в результате реализации функционирования которых и получают требуемые (в соответствии с целями) эффекты.

2. Технологии вообще и ИТ в частности, будучи созданы, могут быть внедрены, а затем использоваться (и не использоваться) в ряде систем и при реализации ряда процессов функционирования для достижения разных целей.

3. Для получения эффектов, т.е. для использования внедренных ИТ в «природе», требуется процесс перехода от оперирования информацией к выполнению действий в «природе» при том или ином функционировании той или иной системы.

Комплексные ОС ИТ следует поместить «выше» по отношению к потенциалу системы [5], поскольку ОС ИТ должны учитывать эффекты, получаемые при функционировании разных систем, в результате использования при их функционировании ИТ. Использование ИТ

требует перехода к денотатам информации в природе, обмениваемым веществом и энергией для получения требуемых эффектов. Этот переход осуществляется в виде свободного выбора людьми [12] одного из способов действий и последующей реализации действия в соответствии с выбранным способом. В рамках исследований ОС ИТ исследуются способы действий, приводящие к результатам в природе, поскольку в рамках прагматического подхода нас интересуют эффекты деятельности, получаемые в природе.

При исследовании ИТ исследуются способы действия с информационными объектами (знаками, системами знаков), а при исследовании ОС ИТ – комплексы действий по обработке информации, полученных способов действий и (соответствующих этим способам) действий на практике, которые и дают требуемые эффекты достижения различных целей в разных условиях. Такие комплексы, состоящие из цепочек обработки информации, полученных способов, соответствующих им комплексов действий и соответствующих комплексам действий цепочек эффектов – *цепочки проявлений эффектов с использованием ИТ*.

Реализация ИТ в виде разных способов действий с информационными объектами носит характер знаков и их систем, описывающих действия с другими знаками (данными, информационными объектами). Цель ИТ – получить способ (информацию о будущем действии) – такой, чтобы при его воплощении в природе при действии выбранным способом получать требуемые для достижения целей эффекты. Переход между полученным за счет ИТ способом действий и действием этим способом и есть переход от информации к действию.

Способ действий может меняться при изменении характеристик элементов действия, их отношений разного вида, характеристик этих элементов и отношений – то есть, при изменении идеальных объектов, соответствующих объектам в природе, которые человек может менять за счет своей воли и свободного выбора способа действия. В результате, указанные характеристики, отношения фигурируют *в качестве переменных в решаемых задачах*. Использование ИТ ведет к изменению способов, доступных для выбора, к изменению выбранных способов, к изменению отношений между способами действий. За счет этих изменений, протекающих в области оперирования знаками (описывающими способы, их отношения, выбор способов и отношений, результаты такого выбора) человек меняет затем (после выбора реализуемых способов) и отношения в природе (действия), а затем, в результате реализации отношений обмена в природе проявляются эффекты, как правило – цепочки эффектов, наблюдается их соответствие целям и в результате проявляются и могут быть спрогнозированы показатели ОС ИТ.

Результаты ИТ проявляются на практике в виде соответствующих им цепочек проявлений эффектов. Сначала это с необходимостью – результаты замены старых ИТ на новые, смена способов оперирования информацией. Это требует затрат ресурсов (в том числе времени). Затем, получают новые результаты ИТ (способы, другую информацию, затраты на ее обработку), затем реализуется выбор способа действий, переход к действиям, конверсия результатов ИТ и систем, а затем реализуют процессы функционирования, использующие результаты ИТ. Может быть необходимым повторение таких цепочек и накопление результатов (способов, результатов конверсий, результатов других действий), прежде чем деятельность сменится новой. Тем самым, механизм проявления ОС ИТ следует исследовать, изучая *изменения в разнообразных цепочках проявления эффектов с использованием ИТ*. Вскрытые закономерности и механизм проявления ОС ИТ позволяют объяснить ряд особенностей ИТ, проявляющихся на практике.

7. Подтверждение вскрытых закономерностей и механизма проявления ОС ИТ практикой использования ИТ. Вскрытые закономерности проявления ОС ИТ подтверждаются практикой. Так, известно [23], что затраты на ИТ и отдача от них коррелируют с капитальными затратами и затратами на управление. Однако, *капитальные затраты* связаны с созданием новых возможностей и лежат в начале цепочек преобразования эффектов. *Управленческие затраты* требуются, в том числе, для внедрения нового, для изменений в системе и процессе ее функционирования. Отмечалась наблюдаемая на практике схожесть влияния затрат на ИТ с затратами на *инфраструктурные технологии* (железные дороги, электрические сети) [14]. Если учесть, что такие инфраструктурные проекты меняют возможные *способы обмена эффектами* и, с одной стороны, сами по себе еще не ведут к изменению функционирования предприятий, а с другой стороны, лежат в основе изменения цепочек эффектов на предприятиях, то указанная схожесть тоже становится ясной. Другое наблюдающееся явление – *связь с квалификацией персонала*. Этот аспект становится ясным если учесть, что способы действия создаются и обрабатываются именно персоналом при реализации какой-либо ИТ. При этом отмечается [15,16,18,23,24,31–34], что нет явной связи *затрат на ИТ с производительностью*, затраты на ИТ ведут к прогрессу, но он проявляется на большом интервале времени. Это, с использованием обоснованных выше закономерностей, объясняется тем, что затраты на ИТ создают (и то лишь потенциально) новые способы, потенциальные возможности для смены используемых (старых) способов функционирования. Внедрение ИТ предполагает, что имеющиеся (где-то) ИТ сначала вводят в

состав системы, затем – определяют новые способы функционирования на основе ИТ, затем, реализуют действия по конверсии системы и ее функционирования для реализации этих способов и затем лишь (возможно) получают требуемые эффекты (результаты конверсии ИТ) за счет реализации новых способов в виде действий. Очевидно, что процесс получения эффектов на основе использования ИТ может затянуться во времени, может быть реализован с постепенным накоплением «положительных изменений» и накоплением новых способов действий и их цепочек до того, как они будут реализованы на практике. При этом, если изменений в результате ИТ не реализовывать, то предприятие будет функционировать по-старому, возможностей получения нового на основе ИТ не будет и, если отсутствие таких новых возможностей будет нечем компенсировать, предприятие может потерять возможность развиваться.

Роль ИТ в «новой экономике» [20] тоже может быть вскрыта с использованием закономерностей проявления ОС ИТ. Значительное число создаваемых способов действий позволяет строить значительное число новых цепочек эффектов функционирования предприятий, цепочек формирования «добавленной стоимости», получать новые результаты, удовлетворять новые потребности, которые не удовлетворялись раньше, и создавать стоимость, которая не создавалась раньше. Современные рынки вычислительной техники, электронных услуг, современного цифрового контента – пример проявления таких цепочек. *Цепочки создания нового* и получения затем, на этой основе, *новых или лучших эффектов* достижения новых или уже достигавшихся целей – основа «новой экономики». Для исследования характеристик таких цепочек проявлений ИТ следует выполнить схематизацию формирования эффектов в таких цепочках, что затем позволит ввести показатели ОС ИТ.

8. Схематизация формирования эффектов в результате использования информационных технологий. Новый способ действий, полученный за счет использования ИТ, как свидетельствует практика [35], воплощается как часть нового комплекса действий и в результате – вызывает новые цепочки проявления эффектов («Value Creation Hierarchy»). Схема формирования таких цепочек может быть условно представлена в виде схемы проявления эффектов (рисунок 1). Эта схема обобщает предложенную в [35] концепцию цепочки эффектов «установка – услуга заказчику – продукт –услуга заказчику» расширив ее переходами «получение и обработка информации – информация – установка». Полученная схема формирования позволяет вскрыть *общую закономерность формирования цепочек проявления*

эффектов ИТ – через человека, его решения, новые способы его действий, а затем – через новые изменения в системах, которые должны дать возможность последующих изменений в процессах функционирования этих систем. На схеме передача природных эффектов всегда соответствует связи между системой и процессом функционирования какой-либо системы. Для корректного описания ОС при рассмотрении цепочек преобразования эффектов следует рассматривать цепочки вида система – функционирование – эффект-система – функционирование, что и отражено на схеме (рисунок 1).

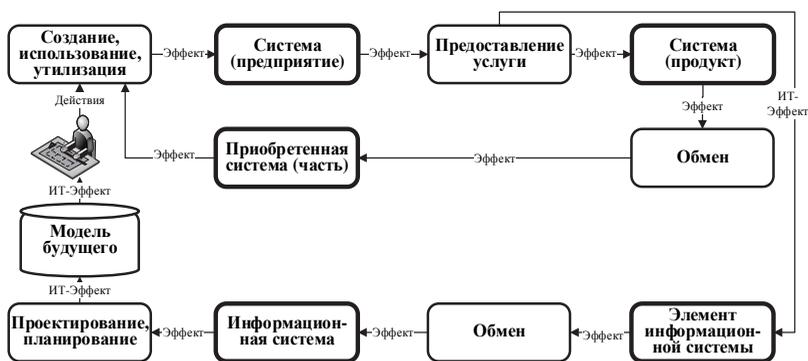


Рис. 1. Цепочки проявления эффектов

При этом, на схеме особо рассматривается случай, когда эффект – результат ИТ (*ИТ-эффект*), передается между процессом обработки информации (планирование, выбор способа действия) и человеком, который может осуществить действия созданными способами на практике и перенести способ действий в действительность [21] (“putting into actuality”). В рассмотренной схеме для упрощения описания цепочек преобразований информации и эффектов опущены отношения познания между человеком и действительностью. Эти отношения связывают все элементы схемы и человека и являются, в соответствии с определением информации, основой для получения результатов ИТ. Приведенная на рисунке 1 схема проявления эффектов характеризуется тем, что в ее составе имеются эффекты – результаты ИТ (в нижней части схемы), которые носят идеальный характер, получены на основе информации, полученной людьми и затем этот эффект используется людьми для получения (природных) эффектов создания, совершенствования систем, а затем процессов их функционирования.

В связи с указанными особенностями, человек – центральное

звено получения эффектов на основе ИТ и получения информации для реализации ИТ. Эта особенность может быть описана с использованием схемы: $A \leftrightarrow H \leftrightarrow I$, где H – человек (центр схемы), A – деятельность, затрагиваемая ИТ (верхняя часть схемы), I – ИТ и обрабатываемая информация (нижняя часть схемы), \leftrightarrow – реализация человеком действий различными способами (в природе) и получение информации (о природе), \Leftrightarrow – реализация человеком ИТ и получение результатов ИТ.

Части комплексов действий могут выполняться техническими устройствами вместо человека, тогда говорят об *автоматизации*.

При реализации процессов преобразования эффектов важнейший аспект исследования – множественность возможностей создания, изменения и использования систем, а затем – множественность проявлений эффектов (множественность возможностей деятельности). Такая множественность – результат наличия свободы выбора людей в человеческой деятельности [12] как в природе, так и при реализации ИТ. Эта множественность должна описываться множественностью информации о возможных действиях, и в частности – о способах действий при реализации ИТ, при создании систем, их элементов, их функционирований, действий с системами. На схеме (рисунок 2) множественность возможностей деятельности и соответствующих этим возможностям цепочек проявления эффектов проиллюстрирован с использованием тех же элементов диаграммы, что и на рассмотренной ранее схеме, но с учетом наличия возможностей выбора. Эти возможности выбора дают возможность улучшать ОС ИТ за счет увеличения числа способов и последующего *выбора лучших способов* и в результате, позволяют решать задачи совершенствования ОС ИТ, как математические задачи анализа и синтеза по показателям ОС ИТ.

Наличие возможностей выбора при использовании новых способов действий ведет, с одной стороны, к увеличению возможностей достижения различных целей, а с другой стороны – к необходимости рассмотрения всех возможных последовательностей способов, определения возможных результатов действий разными способами в разных условиях, а затем выбора лучших из них. Оценивание комплекса возможностей деятельности с учетом множественности достигаемых целей и с учетом результатов действий, предпринимаемых для перехода от достижения одних целей к достижению других традиционно [5,29] реализуется с использованием показателей потенциала систем. Поэтому *оценивание целевых результатов ИТ целесообразно выполнять с использованием показателей потенциала систем*. Затем следует со-

поставить получаемый целевой результат с результатами без использования ИТ и с затрачиваемыми на ИТ ресурсами.

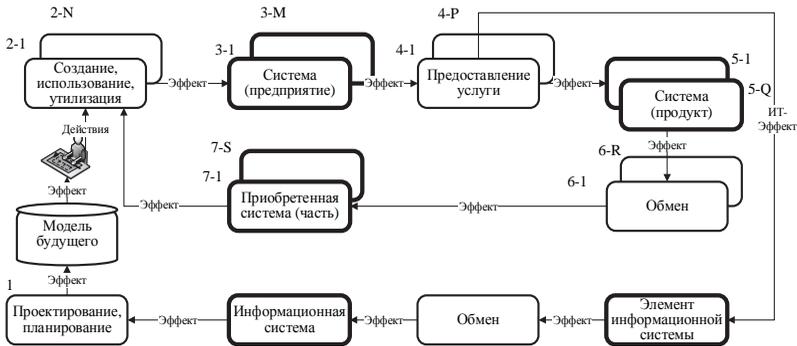


Рис. 2. Учет множественности возможностей деятельности и проявлений соответствующих эффектов

Использование результатов ИТ позволяет использовать новые способы обмена, осуществить переход к действиям этими способами, изменять элементы и их отношения так, что будут получены лучшие результаты действий. Для вскрытия того, как следует оценивать использование результатов ИТ предложены принципы исследования ОС ИТ.

9. Основные принципы исследования операционных свойств использования информационных технологий. Принципы исследования *сформулированы нами в соответствии с используемым прагматическим подходом* к исследованию ИТ. Цепочки проявления эффектов начинаются оперированием информацией, продолжают оперированием объектами «природы», а затем завершаются теми или иными действиями по достижению различных актуализируемых целей. Соответственно, при исследовании ОС ИТ необходимо вскрыть весь комплекс всех возможных действий целиком, с учетом всех возможностей, начиная от возможностей оперирования информацией и заканчивая возможностями действий по достижению целей. Такое исследование назовем комплексным исследованием цепочек проявлений эффектов. Оценивать ОС ИТ следует на основании оценивания того, какое улучшение ОС каких объектов, использующих ИТ дает использование той или иной ИТ. Необходимость такого комплексного исследования всяких объектов, использующих ИТ для оценивания ОС ИТ – один из основных принципов исследования ОС ИТ. Этот принцип назовем **принципом комплексного исследования всех возможных цепочек проявления эффектов ИТ** или – **принципом потенциал-**

ности. Он состоит в том, что для исследования ОС ИТ необходимо вскрыть и описать весь комплекс действий различного вида с разными объектами разной природы, таких, что эти комплексы действий, начинаясь оперированием информацией, заканчиваются, в конечном итоге, тем, что позволяют удовлетворять какие-либо цели при каком-либо функционировании каких-либо объектов, использующих ИТ. Этот принцип задает основу исследования ОС ИТ. Он развивает аналогичный принцип исследования потенциала систем [29]. В теории потенциала этот принцип, фактически, состоял в том, что исследовать ОС системы следует по всем эффектам комплексов действий, реализуемых с системой для конверсии системы и последующего достижения актуализируемых, в разных условиях, целей.

Этот принцип может трактоваться, как *переход* с уровня исследования отдельных процессов функционирования для достижения цели к рассмотрению комплексов всех затрагиваемых заданной ИТ процессов и переходов между ними в разных условиях, а также процессов перехода от одних процессов функционирования к другим. Это позволяет исследовать не только процессы преобразования вещества и энергии для получения требуемых целевых результатов, но и связанные с ними, предшествующие им процессы преобразования информации, реализуемые при создании новых и изменении существующих систем, их элементов и процессов их функционирования. Принцип потенциальности в теории потенциала позволил перейти к исследованию комплексного ОС системы – ее потенциала. Введенный принцип исследования ОС ИТ должен позволить исследовать потенциал объектов, использующих ИТ а затем – комплексные ОС ИТ, оцениваемые на его основе.

Для исследования ОС *заданной ИТ в соответствии с введённым принципом* необходимо:

1. Исследовать комплекс действий по реализации заданной ИТ, комплекс всех информационных результатов, получаемых в результате такой деятельности и комплекс ресурсов, необходимых для получения различных информационных результатов;

2. Исследовать комплекс действий (внедрения, конверсии), реализуемых со всеми объектами, которые используют результаты ИТ, реализуя переход от *результатов заданной ИТ (в виде способов действий)* к действиям этими способами с использованием объектов, использующих ИТ и затем, исследовать все эффекты таких действий.

3. Исследовать комплекс действий, реализуемых в природе для реализации ИТ.

4. Исследовать комплекс целей и комплекс событий, приводящих к их актуализации, таких что актуализируемые цели могут достигаться за счет исследованного комплекса действий, реализуемых с объектами, использующими ИТ.

5. Исследовать комплексы всех действий, которые могут быть реализованы для достижения какой-либо из актуализированных целей с тем или иным объектом, использующим ИТ и определить меры возможности достижения каждой из этих целей, как комплекс соответствий прогнозируемых результатов требованиям, выдвинутым при актуализации различных целей.

В результате выполнения 1-5 может быть получен комплекс результатов, позволяющих перейти к оцениванию потенциала объектов, использующих ИТ и затем, показателей ОС ИТ. Затем станет возможным выполнить анализ ИТ по показателю ОС ИТ и синтез характеристик ИТ и процессов использования ИТ по показателям ОС ИТ. В результате, становится возможным решать *ряд задач исследования ОС ИТ*, как математические задачи.

Принцип рекурсивного формирования эффектов ИТ при реализации цепочек действий. Этот принцип состоит в том, что эффекты от ИТ формируются рекурсивно, в результате реализации комплекса действий разного вида, причем таких, что началу цепочек проявления эффектов соответствуют действия по оперированию информацией. Определять меру соответствия этих эффектов требованиям возможно тоже рекурсивно, с использованием соответствующих математических моделей. Принцип дополняет принцип потенциальности указанием на то, как комплекс всех возможных действий на основе ИТ может быть вскрыт, поскольку в принципе потенциальности механизм образования цепочек нового не указан, описана лишь необходимость рассмотрения всех цепочек проявления эффектов. Для того, чтобы необходимые комплексы действий были выполнены так, чтобы были получены требуемые цепочки эффектов, необходимо эти действия планировать и реализовывать действия по плану. Для такого планирования необходимо решать задачи оценивания, анализа и синтеза характеристик исследуемых комплексов действий по показателям операционных свойств ИТ.

10. Концептуальная схема оценивания показателей операционных свойств использования информационных технологий. Как было показано ранее, ОС ИТ следует оценивать по тому, какие результаты дают ИТ за счет реализации новых способов в процессе осуществления действий этими способами. При этом, исследуются те виды деятельности, которые затронуты ИТ и которые реализуются с

исследуемыми системами – *направленные на затронутую деятельность и направленные на затронутые системы*. Эта деятельность меняется при применении ИТ (и без ИТ), а по тому, насколько она меняется и как меняются результаты деятельности при использовании ИТ и без них и каковы затраты ресурсов на эти ИТ и следует оценивать ОС ИТ. Исследоваться должны все процессы: целевого функционирования, переходные процессы, обеспечивающие процессы и процессы создания, изменения утилизации используемых элементов, создания изменения мероприятий, которые только могут быть реализованы в различных условиях реализации исследуемой деятельности – с использованием ИТ и без.

Такое оценивание реализуется на основе показателя комплексного ОС объекта с учетом результатов ИТ – потенциала объекта, на который направлены ИТ. Потенциал объекта – свойство объекта, характеризующее его приспособленность достигать с его использованием цели субъекта. Для его оценивания следует [5] определить все действия и их совокупности, выполняемые с использованием и без использования ИТ при реализации деятельности с исследуемым объектом, определить эффекты этих действий в разных условиях, а затем определить, насколько эти эффекты соответствуют требованиям к ним при актуализации различных целей. Затем следует оценить изменение потенциала объекта за счет использования ИТ. Полученное значение изменения потенциала объекта за счет использования ИТ и следует использовать, как показатель ОС ИТ для этого объекта. При наличии нескольких объектов, в которых используются ИТ следует рассмотреть комплекс объектов и определить ОС ИТ для комплекса в целом. Получаемые показатели должны позволять вскрыть зависимости между характеристиками ИТ, характеристиками способов действий (переменными), а затем – между действиями, их эффектами, а затем с требованиями к ним и наконец, со значениями определяемых показателей ОС ИТ. Для этого следует, с одной стороны (отражая происходящее на стороне ИТ) порождать последовательно модели символов и действий с символами, отражающие соответствующие им действия в природе и позволяющие при реализации ИТ создать новые способы действий, а затем, перейти к выбранным действиям в природе и теперь уже с другой стороны (отражая происходящее на стороне действительности, природы), порождать модели всевозможных видов действий разными способами, действий с разными элементами, затем с комплексами элементов, затем совокупностей действий, изменений действий и последовательностей разных видов деятельности с разными комплексами элементов, действий по изменению комплексов таких действий в при-

роде – так, чтобы в таком рекуррентном процессе породить модели эффектов и модели требований к ним, а затем – определить меру их соответствия при новой ИТ и старой, а затем определить ОС ИТ.

Такая, синтезированная в результате, связь между объектами ИТ и их оперированием и соответствующих этому оперированию изменению (прогрессу или регрессу) во всем спектре процессов деятельности, затрагиваемых ИТ и должна позволить решить задачу исследования ОС ИТ количественно, на основе прогнозных моделей и рассчитываемых с их помощью показателей. Полученные показатели *потенциала использования ИТ объектом, как целевого результата ИТ*, а затем показателей ОС ИТ, должны стать основой для определения других ОС ИТ, таких, как потенциал обмена ИТ, эффективность внедрения ИТ, а затем – решения комплекса задач исследования ОС ИТ.

Задачи исследования ОС ИТ состоят в *оценении* ОС ИТ, как свойств ИТ и деятельности, на которую ИТ направлены, *анализе* того, как при изменении характеристик ИТ, их объектов, деятельности и ее объектов меняются ОС ИТ, и наконец – *синтезе* таких характеристик ИТ и деятельности, их объектов, которые позволят достичь лучших результатов ИТ и деятельности, т.е. ИТ и деятельности с лучшими ОС ИТ. Концептуальная схема алгоритма решения задач исследования ОС ИТ показана ниже (рисунок 3).

Концептуальная схема алгоритма делится на три основных блока. В *первом блоке* определяются комплексы (рекуррентные цепочки) способов действий разного вида:

$$\sigma \in \Sigma, \Sigma \subseteq \Pi \times \Gamma \times \Lambda, \text{ где}$$

$\gamma \in \Gamma$ – возможные комплексы способов действий среды;

$\pi \in \Pi$ – возможные комплексы способов действий людей с исследуемым объектом, в зависимости от тех или иных действий среды;

$\lambda \in \Lambda$ – возможные комплексы способов действий по обработке информации в зависимости от действий среды и людей.

Указанные комплексы способов действий, как правило, определяются с использованием рекуррентно формируемых моделей разного вида, а для их определения используется разного рода информация – как описывающая наблюдаемые особенности объекта исследований, так и другие виды информации, прежде всего – априорная синтетическая [12] информация о характеристиках природных закономерностей обмена веществом и энергией, позволяющая прогнозировать результаты обмена.

Второй блок использует полученные в первом блоке результаты для определения отношений двух видов: вида $\tilde{R}(\delta)$ между характери-

стиками результатов \tilde{R} комплекса действий и характеристиками выбранных способов δ реализации действий на практике, и вида $\tilde{R}^o(\delta)$ между характеристиками требований к результатам \tilde{R}^o , предъявляемым при актуализации возможных целей и выбранными способами δ .

Отношения и первого и второго видов, определяются на моделях, построенных в первом блоке моделей. Они, как правило, описываются с использованием случайностей разного вида и могут описываться, например, случайными функциями, рекуррентными зависимостями, заданными с использованием случайных величин, ассоциированных с элементами моделей, построенных в первом блоке.

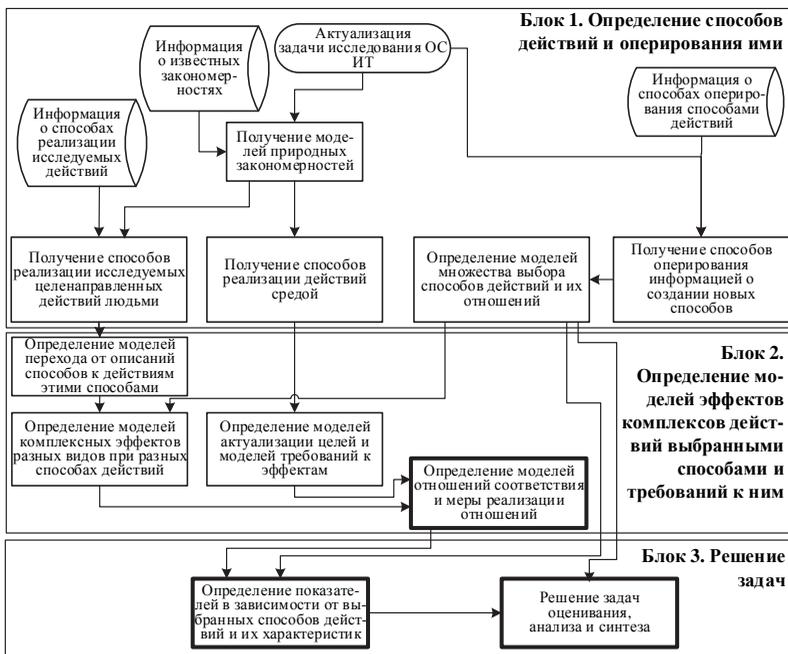


Рис. 3. Концептуальная схема алгоритма решения задач исследования ОС ИТ

Третий блок использует результаты, полученные во втором и первом блоке, для решения задач исследования, как задач оценивания, анализа и синтеза по показателям ОС ИТ. Для этого, в соответствии с актуализируемыми целями, определяются отношения \tilde{R}^c соответствия

$\tilde{R}(\delta)$ и $\tilde{R}^o(\delta)$ в разное время в разных условиях. Затем, определяют показатель $W^o(\delta)$ потенциала объекта, с учетом использования ИТ.

Он имеет вид *меры возможности* соответствия характеристик результатов требованиям к ним в разных условиях среды, для разных процессов функционирования объекта, *число от 0 до 1*:

$$W^o(\delta) \equiv Poss(\tilde{R}^c(\tilde{R}(\delta), \tilde{R}^o(\delta))). \quad (1)$$

Объект, который использует ИТ, может иметь вид, например, систем, процессов функционирования, комплекса процессов функционирования комплекса систем. Соответственно, будет меняться оцениваемое ОС и вид используемого показателя (1).

Определение способов действий, обеспечивающих наилучший потенциал объекта с учетом ИТ может быть представлено в виде задачи поиска экстремума вида:

$$\delta^* \in \underset{\delta \in \Sigma}{ArgMax}\{W^o(\delta)\}.$$

Здесь $W^o(\delta)$ рассчитывается по (1), как функция способов действия, в том числе – способов действий по обработке информации.

11. Пример использования концептуальной схемы оценивания показателей эффективности внедрения ИТ. При внедрении ИТ старые способы обработки информации, используемые ИТ объекта внедрения, заменяются новыми. На процесс такой замены тратятся ресурсы, в том числе время, как ресурс особого вида. Результат внедрения новой ИТ в объект внедрения – функционирование ИТ объекта внедрения новыми способами. Внедрение новой ИТ – создание новых способов обработки информации, которые, в свою очередь, могут создавать новые способы действий. Эти новые способы действий могут вызывать новые действия в природе – так, что функционирование этими способами для достижения той или иной (часто, новой) цели позволяет лучше достигать цели. Лучше – значит с ОС лучше, чем до внедрения ИТ.

Обозначим комплекс цепочек способов действий при использовании заданной ИТ I через $\delta^{IT} = \delta(I)$. Эти способы отличаются от способов действий $\delta^{I_0} = \delta(I_0)$, при использовании предшествующей ИТ I_0 . Может быть получена разность ОС ΔW^o после использования I и до ее использования (при использовании ИТ I_0):

$$\Delta W^{\circ}(I, I_0) \equiv W^{\circ}(\delta^{IT}, I) - W^{\circ}(\delta^{ITp}, I_0). \quad (2)$$

Эта разность $\Delta W^{\circ}(I, I_0)$ может использоваться, как показатель комплексного ОС ИТ для заданного объекта и внедряемой новой ИТ. При использовании новых ИТ число способов действий увеличивается, поэтому, как правило, значение выражения (2) положительно и лежит в замкнутом интервале $[0,1]$.

Значение (2), если оно положительно, имеет смысл меры возможности того, что за счет использования новой ИТ будут реализованы цепочки полезных нововведений и затем будут достигаться новые или лучшие достигаться старые цели, с учетом возможностей актуализации целей и в условиях рисков – число в замкнутом интервале $[0,1]$.

Если же значение (2) отрицательно, то это означает, что ИТ дало негативные результаты, цепочки нововведений оказались вредны. В этом случае модуль (2) имеет смысл меры возможности того, что в результате использования новой ИТ наступили негативные последствия для достижения целей. Такие негативные последствия, а также риски использования ИТ планируется рассмотреть в последующих публикациях.

Если значение (2) равно 1, то новая ИТ всегда и во всех условиях позволяет достигать все возможные актуализируемые цели, а старая никогда и ни при каких условиях не позволяла достигать никаких (не было возможности получать с использованием старой ИТ информацию о том, как можно достигать такие цели), то есть, старая ИТ была бесполезна.

Предполагается, что некоторая ИТ имеется всегда, но она может не позволять достигать некоторые цели в некоторые моменты в некоторых условиях. Образно говоря, если (2) принимает значение 1, то новая ИТ оказалась «панацеей», «философским камнем» для исследуемого объекта – она описывает, как сделать все то, что раньше сделать с объектом за счет старых ИТ было невозможно, и не делалось, но было нужно. Если значение (2) равно 0, то новая ИТ не дает преимуществ по сравнению с предшествующей ИТ.

Описанная разность должна быть сопоставлена с обеспечивающими ее получение результатами. Это целесообразно сделать с использованием показателя успешности внедрения новой ИТ, имеющего вид, аналогичный (1), но имеющий характер показателя мета-уровня для цепочки нововведений, результаты которой (результаты использования ИТ) характеризуются показателем (2). А именно, в этом показателе

теле мера возможности того, что характеристики целевого эффекта удовлетворяют требованиям – $\Delta W^o(I, I_o)$.

Рассмотрим остальные эффекты, соответствие которых требованиям отражается в этом показателе, характеризующем успешность внедрения новой ИТ. Для упрощения примера ограничимся двумя обеспечивающими эффектами – промежутком времени на создание новой ИТ на основе старой $T(I, I_o)$ и стоимостью $C(I, I_o)$ получения новой ИТ на основе старой. Требования к ним будем считать детерминированными константами T^o и C^o , а отношение соответствия примем имеющим вид «эффект не больше требования». Тогда, ОС процесса перехода от старой ИТ к заданной новой для заданного объекта может быть оценено с использованием следующего показателя:

$$W^C(I, I_o) \equiv \Delta W^o(I, I_o) \cdot Poss\left(\left(\tilde{C}(I, I_o) \leq C^o\right) \cap \left(\tilde{T}(I, I_o) \leq T^o\right)\right). \quad (3)$$

Выражение для ОС ИТ (3) имеет смысл *эффективности внедрения новой ИТ*. Для определения (3) необходимо определить целевой эффект – ОС ИТ по отношению к существовавшей ранее ИТ, в соответствии с показателем ОС ИТ (2), а затем определить соответствие затрат ресурсов на внедрение ИТ требованиям и оценить значение (3), имеющее вид *меры возможности достижения цели внедрения ИТ, число в замкнутом интервале [0,1]*. Затем, решают имеющиеся задачи, как математические задачи, имеющие вид поиска экстремума, например, задачи, имеющие вид выбора внедряемой ИТ и плана ее внедрения из множества возможных ИТ и планов внедрений мощностью J :

$$I^*(I_o) \in \underset{I_j \in \{I_j, j=\overline{1, J}\}}{\text{ArgMax}}\{W^C(I_j, I_o)\}.$$

12. Заключение. В статье сделана попытка вскрыть причины сложностей, возникающих при оценивании ОС использования ИТ, определить источники сложностей при решении задач оценивания, анализа и синтеза по показателям ОС использования ИТ, а затем предложены концепты, принципы, концептуальная схема оценивания их показателей. Полученные результаты, на наш взгляд – основа для дискуссии, к которой приглашаются все заинтересованные исследователи. По мнению авторов, полученные результаты позволяют перейти к определению прогнозных, рассчитываемых на математических моделях, показателей ОС использования ИТ, а затем, на этой основе – к разработке концепту-

альной и математической постановок задач исследования и к решению этих задач, как математических задач оценивания, анализа и синтеза по показателям ОС использования ИТ. Для решения такого рода задач разработанные концепты и принципы оценивания ОС использования ИТ должны быть отражены в концептуальных и математических моделях решения задач оценивания, анализа, синтеза по их показателям, а затем – в методах и методиках решения этих задач.

Литература

1. Колмогоров А.Н. Число попаданий при нескольких выстрелах и общие принципы оценки эффективности системы стрельбы // Сборник статей по теории стрельбы, Тр. Матем. ин-та им. В.А. Стеклова, № 12, Изд-во АН СССР, М.–Л., 1945, С. 7–25.
2. Иоффе А.Я., Морозов Л.М., Петухов Г.Б., Юсупов Р.М. Актуальные проблемы теории эффективности // Л.: МО СССР. 1977. 37с.
3. Петухов Г.Б. Основы теории эффективности целенаправленных процессов. Методология, методы, модели // М: МО СССР. 1989. 606 с.
4. Петухов Г.Б., Якунин В.И. Методологические основы внешнего проектирования целенаправленных процессов и целеустремленных систем // М. АСТ. 2006. 504 с.
5. Гейда А.С., Исмаилова З.Ф., Клитный И.В., Лысенко И.В. Задачи исследования операционных и обменных свойств систем // Труды СПИИРАН.2014. Вып. 35. С. 136–160.
6. Юсупов Р. М., Заболотский В. П. Показатели оценивания состояния и результатов развития информационного общества // Труды СПИИРАН. Вып. 15. 2010. С. 75–94.
7. Юсупов Р.М., Заболотский В.П. Концептуальные и научно-методологические основы информатизации // СПб. Наука, 2009. 542 с.
8. Борисов А. Большой экономический словарь // М. «Книжный мир». 2010. 864 с.
9. Ожегов С.И., Шведова Н.Ю. Толковый словарь русского языка // «Азъ». 1992.
10. ГОСТ 27000-2012. Информационная технология. Методы и средства обеспечения безопасности. Системы менеджмента информационной безопасности. Обзор и терминология // М. «Стандартинформ». 2014. 17 с.
11. Философский энциклопедический словарь // М.: Советская энциклопедия. 1983. 840 с.
12. Кант И. Критика чистого разума. Перевод Н. Лосского // М.: Мысль, 1994. 591 с.
13. Kuklys, W. Amartya Sen's capability approach theoretical insights and empirical applications // Studies in Choice and Welfare. XVII. Berlin New York: Springer. 2005. 117 p.
14. Карр Н. Блеск и нищета информационных технологий. Почему ИТ не являются конкурентным преимуществом // М. Издательский дом «Секрет Фирмы». 2005. 176 с.
15. Brynjolfsson E., Yang S. Information Technology and Productivity: A Review of the Literature // In.: Advances in Computers. Vol. 43. Academic Press, NY. 1996. 323 p.
16. Каминский Д.С. Дисс. Кандидата экономических наук. Оценка влияния информационных технологий на деятельность предпринимательских структур // Специальность 08.00.05. М. 2005. 166 с.
17. Мошелла Д. Бизнес-перспективы информационных технологий: как заказчик определяет контуры технологического роста // М. Альпина Бизнес Букс. 2004. 113 с.
18. Solow R.M. We'd Better Watch Out // New York Times Book Review. 1987. 36 p.
19. Lucas H.C. Information Technology and the Productivity Paradox: Assessing the Value of Investing in IT // Oxford University Press. 1999. 240 p.
- 30 SPIIRAS Proceedings. 2015. Issue 5(42). ISSN 2078-9181 (print), ISSN 2078-9599 (online) www.proceedings.spiiras.nw.ru

20. *Tanaka G.* Digital Deflation: The Productivity Revolution and How It Will Ignite the Economy // McGraw-Hill. 2003. 418 p.
21. *Patel N.V.* Organization and Systems Design. Theory of Deferred Action // Palgrave MacMillan. NY. USA. 2006. 288 p.
22. *Yatim T.* Чтого хочет бизнес от ИТ. Стратегия эффективного сотрудничества руководителей бизнеса и ИТ-директоров // Минск, «Гревцов паблишер», 2007. 256 с.
23. *Лугачев М., Скрипкин К., Ананьин В., Зимин К.* Семь столпов эффективности инвестиций в ИТ. Альманах лучших работ // Information management. Научно-методический журнал для профессионалов ИТ. № 08-10. 2012.
24. *Neely A.D., Gregory M.J., Platts K.W.* Performance measurement system design: a literature review and research agenda // International Journal of Operations & Production Management. 2005. vol. 15 no. 4. pp. 80-116.
25. *Cokins G.* Why is Modeling Foundational to Performance Management? // Dashboard inside newsletter. 2009. 13 p.
26. *Cokins G.* Performance Management: Myth or Reality? // Performance Management: Integrating Strategy Execution, Methodologies, Risk, and Analytics. Wiley, 2009. 274 p.
27. *Beyond Productivity: Information technology, innovation and creativity.* William J. Mitchell, Alan S. Inouye, Marjory S. Blumenthal (Eds) // National Research Council. 2003. 268 p.
28. *Дьякова Е.Г., Трахтенберг А.Д.* Как работать с кнопочками? Информатизация здравоохранения и проблема информационного сопротивления // Информационное общество. 2015. № 1. С. 30–36.
29. *Гейда А. С., Лысенко И. В.* Задачи исследования потенциала социально-экономических систем // Труды СПИИРАН. 2009. № 10. С. 63–84.
30. *Woodside A. G., Pattinson H.* Innovation and diffusion of software technology: Mapping Strategies // Emerald Group Publishing Limited. UK. 2007. 478 p.
31. *Новак Е.В.* информационно-коммуникационные технологии: оценка эффективности // Информационные технологии. 2014. №8. С. 74–80.
32. *Гейда А. С., Лысенко И. В.* Автоматизация решения задач исследования потенциала систем и эффективности их функционирования // Труды СПИИРАН. 2012. № 22. С. 260–281.
33. *Schlenker L., Matcham A.* The effective organization. The Nuts and Bolts of Business Value // John Wiley & Sons. 2005. 202 p.
34. *Taticchi P.* Business Performance Measurement and Management: New Contexts, Themes and Challenges // Springer Science & Business Media. 2010. 376 p.
35. *Handbook on Business Process Management: Introduction, Methods, and Information Systems* // Jan vom Brocke, Michael Rosemann (Eds.). Springer. 2010. 612 p.
36. *Aslaksen E. W.* The System Concept and Its Application to Engineering // Springer. 2013. 266 p.

References

1. Kolmogorov A.N. [Several shots number of hits and the general principles of an assessment of shooting system effectiveness]. *Sbornik statej po teorii strel'by, Trudy Matematicheskogo instituta imeni steklova* [The collection of articles about theory of shooting. Proceedings of Steklov Mathematical institute]. Izdatelstvo AN SSSR, M. 1945, vol. 12, pp. 7–25. (In Russ.).
2. Ioffe A.Ya., Morozov L.M., Petuhov G.B., Yusupov R.M. *Aktual'nye problemy teorii jeffektivnosti* [Urgent problems of the efficiency theory]. L.: MO SSSR. 1977. 37 p. (In Russ.).
3. Petuhov G.B. *Osnovy teorii jeffektivnosti celenapravlennyh processov. Metodologija, metody, modeli* [The theory of efficiency of purposeful processes foundations. Methodology, methods, models.]. M: MO SSSR, 1989. 606 p. (In Russ.).

4. Petuhov G.B., Yakunin V.I. *Metodologicheskie osnovy vneshnego proektirovaniya celenapravlennykh processov i celeustremlyennykh sistem* [Methodological foundations of purposeful processes and purpose-oriented systems external design]. M.: 2006. 504 p. (In Russ.).
5. Geyda A.S., Ismailova Z.F., Clitnyu I.V., Lysenko I.V. [Research problems of operational and exchange properties of systems]. *Trudy SPIIRAN - SPIIRAS Proceedings*. 2014. vol. 35. pp.136-160. (In Russ.).
6. Yusupov R.M., Zabolotskiy V.P. [Indicators of estimation of a state and results of development of information society.]. *Trudy SPIIRAN - SPIIRAS Proceedings*. 2010. vol. 15. pp. 75–94. (In Russ.).
7. Yusupov R.M., Zabolotskiy V.P. *Konceptual'nye i nauchno-metodologicheskie osnovy informatizatsii* [Conceptual and methodological foundations of informatization.]. SPb. Nauka, 2009. 542 p. (In Russ.).
8. Borisov A. *Bol'shoy jekonomicheskij slovar'* [Large economic dictionary]. M.: Knizhnyu Mir, 2010. 864 p. (In Russ.).
9. Ozhegov S.I., Shvedova N.Yu. *Tolkovyy slovar' russkogo jazyka* [Explanatory dictionary of Russian language]. Az. 1992. 923 p. (In Russ.).
10. GOST 27000-2012. [Information technology. Methods and means of ensuring of safety. Systems of management of information security. General review and terminology]. M. StandartInform. 2014. 17 p. (In Russ.).
11. *Filosofskij jenciklopedicheskij slovar'* [Encyclopedic dictionary of philosophy]. M.: Sovietskaya Enciclopedia. 1983. 840 p. (In Russ.).
12. Kant I. *Kritika chistogo rasuma*. [Critique of the pure reason]. M.: Myusl. 1994. 591 p. (In Russ.).
13. Kuklys W. Amartya Sen's capability approach theoretical insights and empirical applications. *Studies in Choice and Welfare, XVII*. Berlin New York: Springer. 2005. 117 p.
14. Karr N. *Blesk i nishheta informacionnykh tehnologij. Pochemu IT ne javljajutsja konkurentnym preimushhestvom* [Gloss and poverty of information technologies. Why IT aren't competitive advantage.]. M. Izdatelskiy dom Sekret Firmyu. 2005. 176 p. (In Russ.).
15. Brynjolfsson E., Yang S. Information Technology and Productivity: A Review of the Literature. *Advances in Computers*. Academic Press, NY. 1996. vol. 43. 323 p.
16. Kaminsky D.S. Dissertation. *Ocenka vlijaniya informacionnykh tehnologij na dejatel'nost' predprinimatel'skih struktur* [Information technologies influence assessment on the activity of enterprise structures.]. Speciality 08.00.05. M. 2005. 166 p. (In Russ.).
17. Moshella D. *Biznes-perspektivy informacionnykh tehnologij: kak zakazchik opredeljaet kontury tehnologicheskogo rosta* [Business perspectives of information technologies: How the customer defines contours of technological growth.]. M. Alpina Business Books. 2004. 113 p. (In Russ.).
18. Solow R. M. We'd Better Watch Out. *New York Times Book Review*. 1987. 36 p.
19. Lucas H. C. Information Technology and the Productivity Paradox: Assessing the Value of Investing in IT. Oxford University Press. 1999. 240 p.
20. Tanaka G. *Digital Deflation: The Productivity Revolution and How It Will Ignite the Economy*. McGraw-Hill. 2003. 418 p.
21. Patel N. V. *Organization and Systems Design. Theory of Deferred Action*. Palgrave McMillan, NY, USA. 2006. 288 p.
22. White T. *Chego hochet biznes ot IT. Strategija jeffektivnogo sotrudnichestva rukovoditelej biznesa i IT-direktorov* [What is wanted by business from IT. Strategy of effective cooperation of business chiefs and IT directors.]. Minsk, Grevzov Publisher. 2007. 256 p. (In Russ.).

23. Lugachev M., Skripkin K., Ananyun V., Zimin K. [Seven pillars of efficiency of investments into IT. Almanac of the best works]. *Information management*. 2012. vol. 08-10. (In Russ.).
24. Neely A.D., Gregory M.J., Platts K.W. Performance measurement system design: a literature review and research agenda. *International Journal of Operations & Production Management*, 2005. vol. 15 no. 4. pp. 80-116.
25. Cokins G. Why is Modeling Foundational to Performance Management? Dashboard inside newsletter. 2009. 13 p.
26. Cokins G. Performance Management: Myth or Reality?. Performance Management: Integrating Strategy Execution, Methodologies, Risk, and Analytics. Wiley, 2009. 274 p.
27. Beyond Productivity: Information technology, innovation and creativity. William J. Mitchell, Alan S. Inouye, Marjory S. Blumenthal (Eds). National Research Council. 2003, USA. 268 p.
28. Dyakova E.G., Trahtenberg A.D. How to work with buttons? Informatization of health care and problem of information resistance. *Informacionnoe obshchestvo - Information Society*. 2015. vol. 1. pp. 30-36. (In Russ.).
29. Geyda A.S., Lysenko I.V. [Social and economic systems capabilities research problems]. *Trudy SPIIRAN - SPIIRAS Proceedings*. 2009. vol. 10, pp. 63–84. (In Russ.).
30. Woodside A. G., Pattinson H. Innovation and diffusion of software technology: Mapping Strategies. Emerald Group Publishing Limited, UK. 2007. 478 p.
31. Novak E.V. [Information and communication technologies: efficiency assessment]. *Informacionnyeologii - Information technologies*. 2014. vol. 8. pp. 74–80. (In Russ.).
32. Geyda A.S., Lysenko I.V. Automation of the systems capability research problems solution and efficiency of their functioning. *Trudy SPIIRAN - SPIIRAS Proceedings*. 2012. vol. 22. pp. 260–281. (In Russ.).
33. Schlenker L., Matcham A. The effective organization. The Nuts and Bolts of Business Value. John Wiley & Sons. 2005. 202 p.
34. Taticchi P. Business Performance Measurement and Management: New Contexts, Themes and Challenges. Springer Science & Business Media, 2010. 376 p.
35. Handbook on Business Process Management: Introduction, Methods, and Information Systems. Jan vom Brocke, Michael Rosemann (Eds.). Springer. 2010. 612 p.
36. Aslaksen E.W. The System Concept and Its Application to Engineering. Springer. 2013. 266 p.

Гейда Александр Сергеевич — к-т техн. наук, доцент, старший научный сотрудник лаборатории информационно-аналитических технологий в экономике, Федеральное государственное бюджетное учреждение науки Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН). Область научных интересов: анализ и синтез организационно-технических, социально-экономических систем, оценивание эффективности их функционирования, потенциала организационно-технических и социально-экономических систем в условиях риска. Число научных публикаций — 121. geida@iias.spb.su, http://www.mathnet.ru/php/person.phtml?personid=58804&option_lang=rus; Россия, 199178, г. Санкт-Петербург, 14 линия, 39; п.т.: +7812 3283257.

Geida Alexander Sergeevich — Ph.D., assistant professor, senior researcher, Laboratory for Information-Analytic Technologies for Economics, St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS). Research interests: analysis and synthesis of techno-organizational, socio-economical systems, their functioning efficiency estimation, estimation of techno-organizational, socio-economical systems capabilities under risk conditions. The number of publications — 121. geida@iias.spb.su,

http://www.mathnet.ru/php/person.phtml?personid=58804&option_lang=rus; 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone: +7812 3283257.

Лысенко Игорь Васильевич — д-р техн. наук, профессор, заведующий лабораторией информационно-аналитических технологий в экономике, Федеральное государственное бюджетное учреждение науки Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН). Область научных интересов: моделирование, информационно-аналитические технологии, экономический анализ функционирования организационно-технических систем, программно-целевое планирование и управление, разработка теории нечетких чисел и функций с приложениями. Число научных публикаций — 243. ilya@iias.spb.su; 14-я линия В.О., д. 39, г. Санкт-Петербург, 199178; р.т.: +7(812)328-3257, Факс: +7(812)328-4450.

Lysenko Igor Vasilievich — Ph.D., Dr. Sci., professor, head of laboratory for information-analytic technologies for economics, St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS). Research interests: modeling, information-analytic technologies, economic analysis of techno-organizational systems functioning, fuzzy numbers theory and applications. The number of publications — 243. ilya@iias.spb.su; 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone: +7(812)328-3257, Fax: +7(812)328-4450.

Юсупов Рафаэль Мидхатович — д-р техн. наук, профессор, член-корреспондент РАН, заслуженный деятель науки и техники РФ, директор, Федеральное государственное бюджетное учреждение науки Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН), президент, НП Национальное общество имитационного моделирования («НОИМ»). Область научных интересов: теория управления, информатика, теоретические основы информатизации и информационного общества, информационная безопасность. Число научных публикаций — 390. spiiran@iias.spb.su; 199178, Санкт-Петербург, 14 линия, д. 39; р.т.: +7-812-328-3311.

Yusupov Rafael Midhatovich — Dr. Sci., professor, Corr. Member of RAS, director, St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS), president of NP National Simulation Society («NSS»). Research interests: control theory, informatics, theoretic basics of informatization and information society, information security. The number of publications — 390. spiiran@iias.spb.su; 39, 14-th Line, St. Petersburg, 199178, Russia; office phone: +7-812-328-3311.

РЕФЕРАТ

Гейда А.С., Лысенко И.В., Юсупов Р.М. **Основные концепты и принципы исследования операционных свойств использования информационных технологий.**

В статье вскрыты основные связи понятий о информации, действиях, способах действия, о связях информационных технологий и деятельности. Показано, что информационные технологии закономерно ведут к расширению возможностей деятельности и именно этот результат – целевой, ради которого на них тратятся ресурсы. Выполнен обзор исследований по проблеме оценивания операционных свойств информационных технологий. Показано, что результаты информационных технологий необходимо рассматривать в связи с возможностями нововведений, развития, прогресса, расширения целей деятельности. Они ведут к положительным результатам на практике, но мы не всегда умеем оценивать эти результаты, хотя и ощущаем в повседневной деятельности достижения новых целей.

Вскрыты основные закономерности проявления операционных свойств информационных технологий, механизмы формирования эффектов информационных технологий и их операционных свойств. Сделан вывод о том, что создание новых способов и затем, накопление возможностей их использования на практике в виде цепочек новых действий определяет механизм проявления операционных свойств информационных технологий. Вскрытые закономерности и механизм проявления операционных свойств информационных технологий позволили объяснить ряд особенностей информационных технологий, проявляющихся на практике. Для описания цепочек использования информационных технологий выполнена схематизация проявления соответствующих эффектов, что затем позволило ввести необходимые показатели операционных свойств.

Показано, что результаты информационных технологий позволяют создавать и использовать новые способы обмена веществом и энергией, осуществить переход к действиям этими способами, изменять элементы и их отношения так, что будут получены лучшие результаты действий, позволяют достигать новые цели. Для вскрытия того, как следует оценивать использование результатов информационных технологий предложены принципы исследования. С их использованием разработана концептуальная схема оценивания показателей операционных свойств информационных технологий. Предложен пример использования полученной концептуальной схемы для оценивания эффективности внедрения информационных технологий. Полученные результаты позволяют перейти к определению прогнозных, рассчитываемых на математических моделях, показателей исследуемых свойств, а затем, на этой основе – к разработке концептуальной и математической постановок задач исследования и к решению этих задач, как математических задач оценивания, анализа и синтеза по показателям операционных свойств информационных технологий.

SUMMARY

Geida A.S., Lysenko I.V., Yusupov R.M. **Main Concepts and Principles for Information Technologies Operational Properties Research.**

The article discusses main connections between such concepts as the concept of information, ways of actions, capabilities concepts, connections between information technologies and human action. It is shown that information technologies naturally lead to expansion of opportunities of human action and this is the target result of information technologies, exchanged on resources spent. The review of the research on the problem of information technologies operational properties estimation has been made. It is revealed that results of information technologies shall be considered in connection with the possibilities of innovations, development, progress, expansions of human action goals. They lead to positive results in practice, but we are not able yet to estimate these results properly, though we can see that results in daily activity as possibilities of progressive changes, development, as possibilities of achievement of the new objectives we were not able to achieve before.

Regularities of operational properties of information technologies, mechanisms of information technologies effects formation and their operational properties formation have been discussed. It was concluded that creation of new capabilities, and then accumulation of those capabilities in the form of new actions chains define the mechanism of operational properties of information technologies formation. The regularities and the mechanism of information technologies operational properties development discussed allowed explaining a number of features of the information technologies, manifested in practice. For the description of information technologies usage chains, the systematization of the corresponding effects development has been made. It allows introducing necessary indicators of operational properties.

It is indicated that results of information technologies allow creating and using new capabilities of an exchange of substance and energy, allow carrying out transition to human action according to these capabilities, to change elements of action and their relations so that to receive the best results of actions. Thus, they will allow achieving the new objectives of actions. For analysis of information technologies effects development, the principles of research have been suggested. With their use we developed the conceptual scheme of information technologies operational properties indicators estimation. An example of the conceptual scheme usage for estimation of efficiency of information technologies application has been offered. The results received allow passing to definition (with use of predictive mathematical models) of the studied properties indicators, and then, on this basis – to the development of conceptual and mathematical statements of research problems and to the solution of these problems as mathematical problems of estimation, analysis and synthesis of information technologies operational properties indicators.

А.Н. Егоров, В.А. Кузнецов, В.Е. Марлей, И.А. Назаргулов
**МОДЕЛЬ РАСПАРАЛЛЕЛИВАНИЯ ВЫЧИСЛЕНИЙ ДЛЯ
ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ВОССТАНОВЛЕНИЯ
ДОСТУПА К ДАННЫМ В КОРПОРАТИВНЫХ СЕТЯХ**

Егоров А.Н., Кузнецов В.А., Марлей В.Е., Назаргулов И.А. Модель распараллеливания вычислений для повышения эффективности восстановления доступа к данным в корпоративных сетях.

Аннотация. В статье рассмотрены модель и особенности реализации программной оболочки распараллеливания вычислений. Приведены результаты сравнительной оценки производительности решения задачи восстановления доступа к данным на различных аппаратных средствах, используя для этих целей, как последовательный алгоритм вычислений, так и реализацию на основе программной оболочки распараллеливания.

Ключевые слова: параллельные алгоритмы, технология CUDA, восстановление доступа к данным, восстановление ключа.

Yegorov A.N., Kuznetsov V.A., Marley V.Y., Nazargulov I.A. The Model of Parallel Computing to Effectiveness Improvement of the Restoration of Access to Data in Corporate Networks.

Abstract. In the article, the model and implementation feature of the program shell of parallelizing are considered. Results of the comparative performance evaluation of solving the task of access restoring to data on various hardware, using for these purposes, both a sequential algorithm of computing, and implementation on the basis of the program shell of parallelizing are given.

Keywords: parallel algorithms, CUDA technology, restoration of access to data, key recovery.

1. Введение. В условиях функционирования корпоративных сетей, в которых для передачи данных, как правило, используются защищенные каналы, предполагающие их передачу в зашифрованном виде, возникают сбои, связанные с потерей или искажением секретных ключей. В такой ситуации актуальной становится задача автоматизации восстановления этих ключей с целью обеспечения доступа к зашифрованным данным, что позволит в значительной степени снизить уровень влияния человека на восстановление работоспособности системы.

Для решения этой проблемы, в основном, применяют подходы [1, 2], предполагающие резервное копирование ключей при их генерации. Копии помещаются в хранилища, из которых в случае необходимости ключ можно восстановить. Политикой безопасности определяется формат таких копий: ключ хранится целиком в исходном или зашифрованном виде, либо он разделяется на части, которые помещаются в отдельные хранилища [3], и, наконец, может сохраняться ряд параметров, связанных с ключом [2, 4].

Хранилища реализуются как программными, так и аппаратными средствами [4], которые, помимо безопасного хранения ключа, могут поддерживать функции шифрования, дешифрования, генерации ключевой пары, а также хеш-функции, что в свою очередь значительно снижает нагрузку на программную часть криптографической системы.

Достоинством такого подхода является скорость восстановления ключа. Однако, хранение копий ключей является дополнительной уязвимостью системы и требует обеспечения надежной защиты хранилищ.

Для ее реализации могут применяться гибридные и (или) альтернативные методы. Например, в [2] резервирование ключа осуществляется в два этапа. Сначала вычисляются параметры ключа, а затем генерируется множество подобных данных того же формата, содержащих произвольные значения параметров, из которых невозможно восстановить ключ. В результате создается так называемый шум, маскирующий исходные параметры. Такой подход ведет к существенному увеличению времени восстановления ключа. Для решения этой проблемы применяют распределенную вычислительную систему с большим (порядка нескольких тысяч) количеством узлов [2].

Восстановление ключей может осуществляться автоматически либо с помощью одного или нескольких специальных агентов [5]. Одной из последних разработок в этой сфере является система HADM-KRS [6], которая реализует децентрализованный подход к организации взаимодействия клиента и агентов. Это позволяет отказаться от использования единого центра восстановления ключа, что снижает затраты на обслуживание и повышает доступность системы. Однако, использование дополнительного звена приводит к увеличению временных затрат на реализацию процесса восстановления.

В России был предложен один из эффективных методов надежного хранения и восстановления секретного ключа, основанный на применении нейросетевых преобразователей «Биометрия – код доступа» [7]. Подобные системы относятся к классу биометрических криптографических систем с генерацией ключа, особенность которых заключается в том, что ключ извлекается из биометрических данных пользователя и не хранится в базе данных [8]. Это исключает необходимость использования резервного копирования. Реализация подобных систем подразумевает использование специального и, зачастую, дорогостоящего оборудования, что снижает их привлекательность. Дополнительная проблема рассматриваемого метода возникает при долгосрочном хранении данных. Поскольку расшифровать их может только обладатель биометрических

параметров, то в случае, например, его увольнения, требуется предварительная перешифровка необходимых данных другим ключом.

В статье рассматривается реализация, при которой для восстановления ключа используется исходный и зашифрованный текст. Несомненно, такой подход требует существенных временных затрат на вычисления. Это связано с высокой степенью защиты современных криптографических систем.

Для сокращения времени обработки больших объемов информации применяют параллельные алгоритмы, повышение эффективности которых достигается увеличением числа одновременно работающих устройств (цифровой сигнальный процессор, центральный процессор, графический процессор, программируемая логическая интегральная схема и т.д.), используемых при распараллеливании.

Цель работы заключается в построении математической модели распараллеливания вычислений и ее интеграции в универсальную программную оболочку распараллеливания (ПОР) [9].

Рассмотрим механизмы, реализованные в ПОР, и эффективность ее применения для решения задачи восстановления доступа к данным, используя полный перебор возможных ключей.

2. Модель ПОР. Основное назначение программной реализации ПОР – организация эффективного исполнения прикладного решения пользователя. Для этого ПОР использует вычислительные мощности устройств компьютера.

ПОР представляет собой совокупность взаимодействующих модулей [10]. Связующим звеном этих модулей является ядро программной оболочки (ЯПО). ЯПО включает в себя базовую функциональность ПОР и имеет самую высокую независимость от других модулей. Вся основная логика работы ПОР заложена во взаимодействии ЯПО с модулями устройств. Модули устройств (МУ) выполняют основную работу вычислений ПОР – решение пользовательских задач (ПЗ). Предполагается также, что МУ имеют одно из двух состояний: busy (устройство занято) и free (устройство свободно).

При формировании требований к параллельным алгоритмам наиболее важным является определение последовательности выполнения определенных событий для каждого вычисления программы [11]. Это позволяет для описания модели ПОР использовать язык темпоральной логики линейного времени (LTL), в котором явно учтен феномен времени. Каждое событие будет характеризоваться булевой переменной, которая принимает значение «Истина» тогда и только тогда, когда наступает событие.

В LTL помимо булевых логических связей для описания причинно-следственной зависимости событий во времени используются темпоральные операторы:

– $X\varphi$ «в следующий момент времени», указывает на то, что φ выполняется на следующем состоянии пути;

– $F\varphi$ «когда-то в будущем», указывает на то, что φ будет соблюдаться в каком-то последующем состоянии пути;

– $G\varphi$ «всегда», указывает на то, что φ выполняется в каждом состоянии пути;

– $\varphi U \psi$ «до тех пор пока», указывает на то, что φ выполняется до тех пор, пока не станет верно ψ ;

– $\varphi R \psi$ «высвободить, открепить», указывает на то, что ψ может перестать быть верным только после того, как станет верно φ .

Для описания модели введем следующие атомарные высказывания, соответствующие основным событиям вычислений программы:

- try_i – i -й пользователь собирается отправить задачу;
- add_i – i -я ПЗ добавляется в очередь задач;
- del_i – i -я ПЗ удаляется из очереди задач;
- $send_i$ – i -я ПЗ отправляется на решение МУ;
- $create_i$ – i -я подзадача формируется из ПЗ;
- $solve_i$ – i -й МУ решил подзадачу;
- $recv_i$ – i -й результат подзадачи МУ отправляется ЯПО;
- res_i – формируется i -й результат ПЗ;
- $free$ – МУ свободен;
- $busy$ – МУ занят;
- end – получен конечный результат ПЗ;
- $empty$ – очередь задач пуста.

Рассмотрим математическую модель программной оболочки, построенную на основе языка LTL:

1. всякий раз, когда пользователь собирается запустить ПЗ, ЯПО добавляет себе ПЗ в очередь задач:

$$G (try_i \rightarrow add_i); \quad (1)$$

2. пока очередь задач не окажется пустой ($empty$), ЯПО будет отправлять задачи на решение свободным МУ ($free$):

$$G ((free \rightarrow send_j) \cup empty); \quad (2)$$

3. всякий раз, когда ЯПО отправляет задачу МУ, формируется подзадача для этого модуля:

$$G (send_i \wedge X create_i); \quad (3)$$

4. всякий раз, когда у МУ формируется подзадача, он становится занятым (busy):

$$G(\text{create}_j \rightarrow \text{busy}); \quad (4)$$

5. всякий раз, когда МУ становится занятым, должна когда-нибудь решиться подзадача:

$$G(\text{busy} \rightarrow F \text{ solve}_i); \quad (5)$$

6. после решения подзадачи МУ возвращает результат ЯПО, и становится свободным (free):

$$G(\text{solve}_i \rightarrow X(\text{recv}_j \wedge \text{free})); \quad (6)$$

7. всякий раз, когда ЯПО получает результат решения подзадачи от МУ, ЯПО формирует результат соответствующей ПЗ:

$$G(\text{recv}_i \rightarrow X \text{ res}_j); \quad (7)$$

8. всякий раз, когда ЯПО формирует результат решения ПЗ, оболочка проверяет, получен или нет конечный результат. Если этот результат получен, то ПЗ удаляется и очереди:

$$G((\text{res}_i \wedge \text{end}) \wedge \text{del}_j). \quad (8)$$

Полученные зависимости темпоральной логики позволяют представить псевдокод основных процессов модели ПОР (листинги 1, 2). Для каждой операции псевдокода ставится счетчик. Следует заметить, что псевдокод МУ является инвариантным по отношению ко всем вычислительным устройствам.

```
while (true)
{
1.1.   if (try)
1.2.     add();
1.3.   while (empty && status == free)
1.4.     send();
1.5.   if (recv())
      {
1.6.     res();
1.7.     if (end)
1.8.       del();
      }
}
```

Листинг 1. Псевдокод ЯПО модели ПОР

```

while (true) {
2.1.   if (create()) {
2.2.       status = busy;
2.3.       while (!solve) {}
2.4.       rcv();
2.5.       status = free;
        }
}

```

Листинг 2. Псевдокод МУ модели ПОР

В соответствии с построенным псевдокодом строится размеченная система переходов LTS (Labelled Transition System), где учитываются всевозможные состояния ПОР, основными из которых являются:

- счетчики выполнения команд;
- состояние решения задачи или подзадачи (решено/не решено);
- состояние модуля устройства (занят/свободен).

На рисунке 1 изображены схемы LTS каждого отдельного потока: единственного – для ЯПО и множества потоков МУ. В узлах через запятую показаны значения основных состояний ПОР.

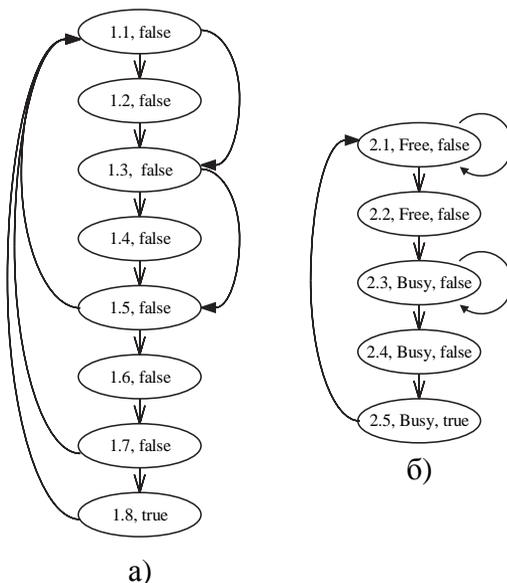


Рис. 1. LTS потоков ПОР: а) - ЯПО; б) - МУ

Поскольку программная реализация обеспечивает работу со многими процессами МУ, рассмотрим LTS–схему, демонстрирующую динамику взаимодействия двух процессов МУ и одного процесса ЯПО (рисунок 2).

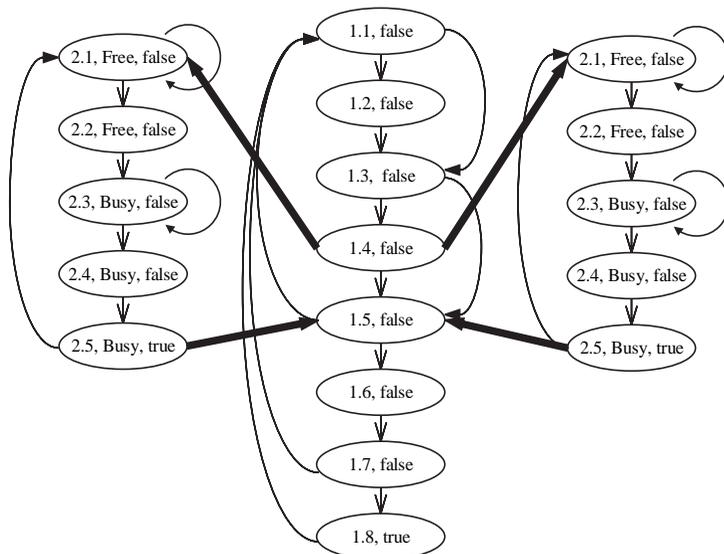


Рис. 2. LTS взаимодействия процесса ЯПО с двумя процессами МУ

ЯПО управляет работой процессов МУ, выполняемых параллельно. Взаимодействие МУ и ЯПО, осуществляемое во временные отметки отправки (блок состояния ЯПО – «1.4, false») и получения (блок состояния ЯПО – «1.5, false») результата решения подзадачи МУ, должно быть синхронизировано. На схеме синхронизация процессов показана жирными линиями.

3. Верификация модели ПОР. Обязательным этапом при разработке и реализации любых программных продуктов, включая в них параллельно исполняемые программы, является проверка корректности работы приложения. К параллельным программам могут успешно применяться следующие методы анализа корректности [12]:

- 1) тестирование;
- 2) формальная верификация: дедуктивная (доказательная); проверка на модели (model checking).

Э. Дейкстра доказал, что тестирование не позволяет разработчику доказательно удостовериться в корректности выполнения программы [13].

В общем случае формальная верификация представляет собой процесс доказательства с помощью формальных методов корректности или некорректности алгоритмов, программ и систем в соответствии с заданным описанием их свойств [12].

Задача дедуктивной верификации формулируется в виде доказательства теоремы (в системе математических доказательств), что предполагает огромную ручную работу, делая это метод малоприменимым на практике.

Верификацией на модели позволяет для заданной модели поведения системы с конечным, даже очень большим, числом состояний, проверить выполнимость некоторого логического требования (спецификации). Следует отметить, что спецификации обычно формулируют в терминах языка темпоральной логики. Это позволяет проверить не только условия на мгновенное состояние системы, но и историю его развития со временем.

Для верификации модели ПОР воспользуемся верификатором NuSMV, принимающим на вход модель, описанную на языке SMV. Указанный язык поддерживает задание спецификаций на языках CTL и LTL. Модель на языке SMV разделяется на модули, каждый из которых может содержать в себе набор переменных, правила переходов и требования [12].

В общем виде структура модели ПОР на языке SMV может быть представлена следующим образом (рисунок 3):

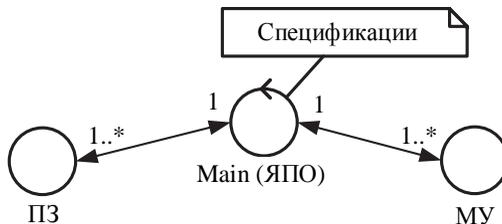


Рис. 3. Структура модели ПОР на языке SMV

Модуль main является обязательным для модели на языке SMV. При верификации модели ПОР данный модуль описывает алгоритм работы ЯПО согласно LTS потоков ПОР, представленных на рисунке 1.а). Main содержит спецификации, выдвигаемые к модели. Модуль МУ описывает логику работы программы в соответствии с LTS-схемой (рисунок 1.б). Для удобства описания модели ПОР на языке SMV состояния ПЗ и правила их переходов выделены в отдельном модуле.

В общем виде алгоритм проверки спецификаций, выдвигаемых к модели ПОР на языке SMV, описан схемой (рисунок 4). Схема логически разделена на 3 части, каждая из которых показывает действия, происходящие в соответствующем модуле.

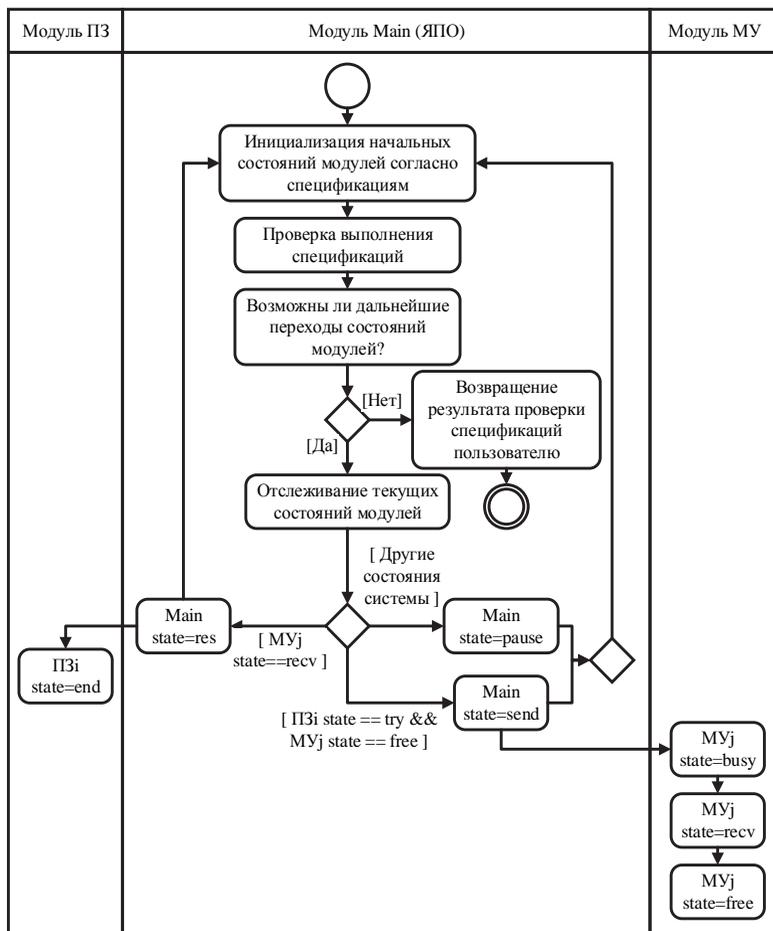


Рис 4. Обобщенная схема алгоритма проверки спецификаций выдвигаемых к модели ПОР на языке SMV

Функционал программной оболочки направлен на вычисления ПЗ, т.е., если пользователь отправляет задачу ПОР, то программная оболочка когда-нибудь должна вернуть результат решения этой

задачи. Данный факт необходимо проверить на модели. В терминах LTL требование выражается так:

$$G(\text{try}_i \rightarrow F \text{end}) \quad (9)$$

По результатам, приведенным на рисунке 5, видно, что модель ПОР соответствует требованию (9).

```
*** This version of NuSMV is linked to the CUDD library version 2.4.1
*** Copyright (c) 1995-2004, Regents of the University of Colorado

*** This version of NuSMV is linked to the MiniSat SAT solver.
*** See http://www.cs.chalmers.se/Cs/Research/FormalMethods/MiniSat
*** Copyright (c) 2003-2005, Niklas Een, Niklas Sorensson

NuSMV > process_model -i E:\por.txt
WARNING *** Processes are still supported, but deprecated. ***
WARNING *** In the future processes may be no longer supported. ***

WARNING *** The model contains PROCESSES or ISAs. ***
WARNING *** The HRC hierarchy will not be usable. ***
The computation of reachable states has been completed.
The diameter of the FSM is 1.
-- specification G (pz1.state = try -> F pz1.state = end) is true
-- specification G (pz2.state = try -> F pz2.state = end) is true
-- specification G (pz3.state = try -> F pz3.state = end) is true
NuSMV >
```

Рис. 5. Результат верификации ПОР верификатором NuSMV

4. Программная реализация. Программная реализация оболочки предназначена для распараллеливания процесса вычислений в прикладных решениях, используя для этих целей различные аппаратные средства. Каждому типу устройств соответствует свой обработчик. Поддержка прикладных решений реализуется в виде dll - библиотек, разработанных для каждого типа устройства, с которыми работает ПОР. Каждая библиотека должна содержать значения параметров, требуемых для настройки среды программной оболочки, таких как: размер массива бинарных данных задачи и результата ее решения; идентификатор задачи, и др. Так же необходимо реализовать ряд экспортных функций, обеспечивающих выполнение следующих операций [14]:

1. Score – оценка производительности устройства;
2. Parser – проверка, обработка и формирование данных ПЗ;
3. Cut – формирование данных подзадачи;
4. SimSim – решение подзадачи;
5. Result и ResultBuild – формирования общего решения ПЗ.

ПОР состоит из модулей, работу которых организует ЯПО [10]. Схема взаимодействия модулей программной реализации изображена на рисунке 6, на котором экспортные функции dll - библиотек представлены в виде отдельных модулей и выделены серым цветом. Сплошными линиями показаны связи между модулями, образующими

ЯПО, пунктирными динамика процесса решения ПЗ. Модули, взаимодействия которых обозначены жирными линиями, выполняются параллельно.

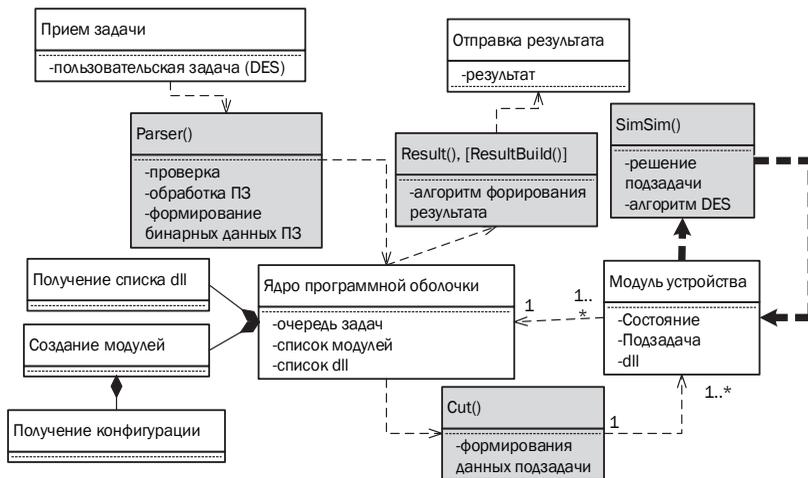


Рис. 6. Схема взаимодействия модулей ПОР

Работу ПОР опишем в виде пошагового алгоритма.

Этап 1. Создание модулей устройств (МУ), отвечающих за взаимодействие с конкретным аппаратным средством, на основе информации, сформированной в результате предварительного анализа вычислительных устройств компьютера, с которыми поддерживает работу ПОР.

Этап 2. Инициализация задачи посредством считывания и обработки данных конфигурационного файла, созданного пользователем, который содержит информацию о типе задачи и бинарные данные задачи, соответствующие выбранному типу.

Этап 3. Решение полученной задачи:

- шаг 1. С помощью функции «Score», вызванной из соответствующей dll - библиотеки, оценивается скорость решения текущей ПЗ на определенном устройстве;

- шаг 2. ЯПО отправляет задачу на решение множеству параллельно работающих МУ, при этом, в зависимости от оценки производительности устройства, модуль «Cute» формирует данные подзадачи для каждого МУ;

– шаг 3. В процессе решения подзадачи модуль «SimSim» осуществляет дополнительное распараллеливание вычислительного процесса, используя особенности аппаратного средства;

– шаг 4. После решения подзадачи МУ отправляет результат ЯПО.

Этап 4. Формирование общего результата решения задачи, на основе информации полученной от МУ, и отправка его пользователю.

При создании комплекта dll - библиотек для решения определенной задачи реализация функций «Score» и «SimSim», является уникальной по отношению к различным аппаратным средствам. В свою очередь, функции Parser, Cute, Result и ResultBuild, зачастую, являются инвариантными, однако в ряде случаев реализации отличаются. Например, в процессе распараллеливания вычислительного процесса функцией «SimSim» на конкретном устройстве могут понадобиться дополнительные параметры, в связи с чем, формирование данных подзадачи функцией «Cute» будет другим.

Архитектура ПОР позволяет практически не ограниченно расширять список поддерживаемых устройств и задач. На текущий момент программная оболочка может использовать центральные процессоры и графические видеокарты, которые поддерживают технологию CUDA компании NVIDIA. CUDA – это платформа параллельных вычислений и модель программирования, позволяющая существенно увеличить вычислительную производительность за счет максимально эффективного использования ресурсов видеокарты (графических процессоров и памяти) компании NVIDIA.

В рамках задачи восстановления доступа к данным были созданы библиотеки для восстановления секретного ключа алгоритма шифрования DES. При реализации библиотек для отладки параллельных алгоритмов использовался NVIDIA Nsight, позволяющий оптимизировать производительность вычислений как для центрального, так и для графических процессоров. В общем виде реализованный по технологии CUDA алгоритм поиска ключа DES изображен на рисунке 7. Узлы, выполняемые в параллельных потоках, на схеме выделены серым цветом.

Вся совокупность потоков делится на блоки, каждый из которых работает со своим объемом встроенной памяти. Рациональная настройка, связанная с выбором количества таких блоков, в конечном итоге определяет количество потоков для распараллеливания.

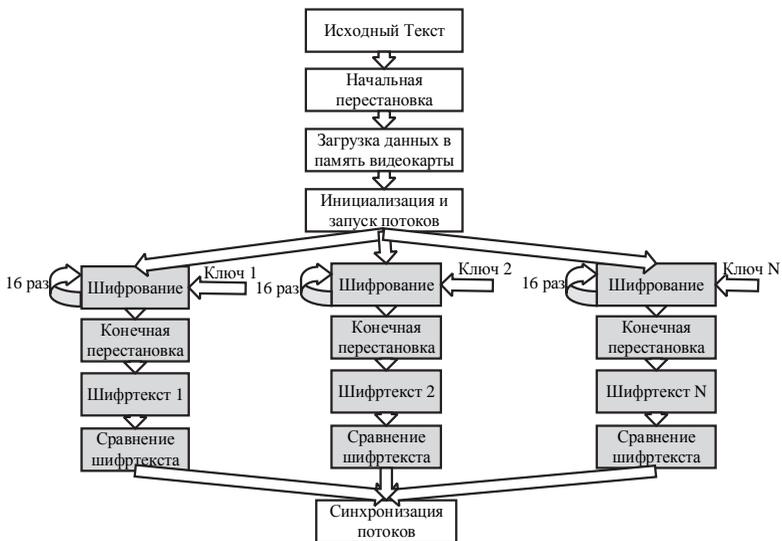


Рис. 7. Алгоритм поиска ключа DES на основе технологии CUDA

Количественная оценка производительности решения задачи восстановления ключа алгоритма шифрования DES в разработанной среде ПОР с применением технологий CUDA представлена в таблице 1. Значения таблицы, показывают эффективность применения ПОР для восстановления доступа к данным.

Таблица 1. Результаты оценки производительности решения задачи по восстановлению ключа алгоритма шифрования DES на конкретных вычислительных устройствах

Устройство	Режим решения		
	Последовательный (ключей/сек)	Параллельный на основе ПОР (ключей/сек)	Параллельный на основе ПОР с применением технологии CUDA (ключей/сек)
Процессор Intelcorei5	~160 000	~15 000 000	
Процессор Intelcorei7	~200 000	~20 000 000	
Видеокарта NVIDIA GTX 256			~156 000 000
Видеокарта NVIDIA GTX 590			~1 000 000 000

5. Заключение. Изложенный в статье подход к распараллеливанию вычислений при восстановлении секретного ключа позволяет поднять производительность процессоров до 10 раз. При использовании алгоритма на основе технологии CUDA выигрыш может достигать значения в 5000 раз по сравнению с последовательной реализацией. Учитывая тот факт, что в настоящее время персональный компьютер может иметь в комплекте до 7 видеокарт, эффективность их применения для параллельных вычислений не вызывает сомнений.

Дальнейшее развитие ПОР предполагает реализацию объединения компьютеров в единую вычислительную сеть и использование более эффективных алгоритмов для восстановления секретных ключей, базирующихся на основе применения радужных таблиц для вычислений [15]. Это позволит еще существенно поднять производительность.

Литература

1. *Полянская О.Ю., Горбатов В.С.* Инфраструктуры открытых ключей. Учебное пособие // М.: Интернет-Университет Информационных Технологий: БИНОМ. Лаборатория знаний. 2007. 367 с.
2. *Jajodia S., Litwin W., Schwarz Th.* Recoverable Encryption through a Noised Secret over a Large Cloud // *Transactions on Large-Scale Data- and Knowledge-Centered Systems*. 2013. vol. 9. pp. 42–64.
3. *Jajodia S., Litwin W., Schwarz Th.* LH*RE: A Scalable Distributed Data Structure with Recoverable Encryption // *Proceedings third IEEE International Conference on Cloud Computing (CLOUD 2010)*. Miami. Fl. July 2010. pp. 354–361.
4. Способ и устройство для хранения и восстановления криптографического секретного ключа: пат. № 2279766. РФ. 2006. Бюл. № 19. 24 с.
5. *Barker E., Branstad D., Chokhani S., and Smid M.* A Framework for Designing Cryptographic Key Management Systems, Draft Special Publication 800–130. National Institute of Standards and Technology. 2010. 89 p.
6. *Kanyamee K., Sathitwiriawong C.* High-availability decentralized cryptographic multi-agent key recovery // *The International Arab Journal of Information Technology*. 2014. vol. 11. no. 1. pp. 52–58.
7. *Гончаров С.М., Боршевников А.Е.* Построение нейросетевого преобразователя «Биометрия – код доступа» на основе параметров визуального вызванного потенциала электроэнцефалограммы // *Доклады ТУСУР*. 2014. Вып. 2(32). С. 51–55.
8. *Бардаев С.Э., Финько О.А.* Многофакторная биометрическая пороговая криптосистема // *Известия ЮФУ. Технические науки*. 2010. Вып 4. С. 148–155.
9. *Егоров А.Н., Кузнецов В.А., Назаргулов И.А.* Программная оболочка распараллеливания процесса вычисления прикладных решений. № 2014619266. РФ. 2014.
10. *Егоров А.Н., Кузнецов В.А., Назаргулов И.А.* Программная оболочка распараллеливания для восстановления доступа к данным // *Региональная*

информатика (РИ-2014) (г. Санкт-Петербург, 29-31 октября 2014 г.). Материалы конференции. СПб.: СПОИСУ. 2014. С. 71–72.

11. *Стальмаков В. А.* Параллельный генетический алгоритм для решения задачи составления расписания прохождения судов через шлюзованные системы и его верификация // Вестник ГУМРФ имени адмирала С. О. Макарова. СПб.: ГУМРФ имени адмирала С. О. Макарова. 2014. Вып. 1. С. 93–102.
12. *Чебатуркин А.А., Мазин М.А.* Методы верификации конечных автоматов, взаимодействующих по акторной модели // СПб.: ИТМО. 2010. 47 с.
13. *Dahl O.-J., Dijkstra E.W., Hoare C.A.R.* Structured Programming // Academic Press. 1972. 220 p.
14. *Егоров А.Н., Кузнецов В.А., Назаргулов И.А.* Параллельные алгоритмы для восстановления доступа к данным // Современные технологии и управление. Сборник научных трудов III Международной научно-практической конференции (р. п. Светлый Яр, 20–21 ноября 2014 г.). Светлый Яр: филиал ФГБОУ ВО МГУТУ имени К. Г. Разумовского (ПКУ) в р. п. Светлый Яр Волгоградской области. 2014. С. 39–42.
15. *Oechslin P.* Making a Faster Cryptanalytic Time-Memory Trade-Off // Proceedings of the 23rd Annual International Advances in Cryptology. Santa Barbara. USA. 2003. vol. 2729. pp. 617–630.

References

1. Polyanskaya O.Yu., Gorbatov V.S. *Infrastruktury otkrytyh kljuchej* [Public key infrastructure]. M.: Internet-Universitet Informacionnyh Tehnologij: BINOM. Laboratorija znaniy. 2007. 367 p. (In Russ.).
2. Jajodia S., Litwin W., Schwarz Th. Recoverable Encryption through a Noised Secret over a Large Cloud. Transactions on Large-Scale Data- and Knowledge-Centered Systems. 2013. vol 9. pp 42–64.
3. Jajodia S., Litwin W., Schwarz Th. LH*RE: A Scalable Distributed Data Structure with Recoverable Encryption. Proceedings third IEEE International Conference on Cloud Computing (CLOUD 2010). Miami. Fl. 2010. pp 354–361.
4. *Sposob i ustrojstvo dlja hranenija i vosstanovlenija kriptograficheskogo sekretnogo kljucha* [Method and device for the storage and recovery of a cryptographic secret key]: patent No. 2279766. Patent RF. 2006. bulletin 19. 24 p. (In Russ.).
5. Barker E., Branstad D., Chokhani S., and Smid M. A Framework for Designing Cryptographic Key Management Systems, Draft Special Publication 800–130. National Institute of Standards and Technology. 2010. 89 p.
6. Kanyamee K., Sathitwiryawong C. High-availability decentralized cryptographic multi-agent key recovery. *The International Arab Journal of Information Technology*. 2014. vol. 11. no. 1. pp. 52–58.
7. Goncharov S.M., Borshevnikov A.E. [Construction of neural network transformer «Biometrics – access code» based on the parameters of the visual evoked potential electroencephalogram]. *Doklady TUSUR – Proceedings of TUSUR*. 2014. vol. 2. no. 32. pp. 51–55. (In Russ.).
8. Bardaev S.E., Finko O.A. [Multifactor biometric threshold cryptosystem]. *Izvestija JuFU. Tehnicheskie nauki – Izvestiya SFedU. Engineering sciences*. 2010. vol. 4. pp. 148–155. (In Russ.).
9. Yegorov A.N., Kuznetsov V.A., Nazargulov I.A. *Programmijnaja obolochka rasparallelivanija processa vychislenija prikladnyh reshenij* [Program shell of parallelizing the computation of application solutions]. No. 2014619266. Patent RF. 2014. (In Russ.).

10. Yegorov A.N., Kuznetsov V.A., Nazargulov I.A. [Program shell of parallelizing for restoration of access to data]. *Regional'naja informatika (RI-2014). Materialy konferencii*. [Regional Informatics (RI-2014) Collected papers.]. St.Petersburg. 2014. pp. 71–72. (In Russ.).
11. Stal'makov. V.A. [Parallel genetic algorithm for solving scheduling passing through the gateways system and its verification]. *Vestnik GUMRF imeni admirala S.O. Makarova – Herald of Admiral Makarov SUMIS*. 2014. vol. 2. pp. 93–102. (In Russ.).
12. Chebaturkin A.A., Mazin M.A. *Metody verifikacii konechnyh avtomatov, vzaimodejstvujushhih po aktornoj modeli*. [Verification methods of finite automats, interacting on actor model]. St.Petersburg: ITMO University Publ. 2014. 47 p. (In Russ.).
13. Dahl O.-J., Dijkstra E.W., Hoare C.A.R. *Structured Programming*. Academic Press. 1972. 220 p.
14. Yegorov A.N., Kuznetsov V.A., Nazargulov I.A. [Parallel algorithms for restoration of access to data]. *Sovremennye tehnologii i upravlenie. Sbornik trudov*. [Modern technology and management. Collected papers.]. Svetly Yar: Offices of MSUTM named after K.G. Razumovsky in Svetly Yar Publ. 2014. pp. 39–42. (In Russ.).
15. Oechslin P. Making a Faster Cryptoanalytic Time-Memory Trade-Off. *Proceedings of the 23rd Annual International Advances in Cryptology*. Santa Barbara. USA. 2003. vol. 2729. pp. 617–630.

Егоров Александр Николаевич — к-т техн. наук, доцент, профессор кафедры вычислительных систем и информатики, ФГБОУ ВО Государственный университет морского и речного флота имени адмирала С.О. Макарова. Область научных интересов: информационные системы и сети, технологии корпоративных сетей. Число научных публикаций — 52. eanspb@rambler.ru; ул. Двинская, д. 5/7, Санкт-Петербург, 198035; р.т.: +7(812)334-3874, Факс: +7(812)334-3874.

Yegorov Alexander Nikolayevich — Ph.D., associate professor, professor of computer system and informatics department, Admiral Makarov State University, of Maritime and Inland Shipping. Research interests: information systems and networks, technologies of enterprise networks. The number of publications — 52. eanspb@rambler.ru; 5/7, Dvinskaya str., St.Petersburg, 198035; office phone: +7(812)334-3874, Fax: +7(812)334-3874.

Кузнецов Виталий Александрович — аспирант, ФГБОУ ВО Государственный университет морского и речного флота имени адмирала С.О. Макарова. Область научных интересов: распределенные и параллельные системы, технология программирования, криптографические системы. Число научных публикаций — 2. kuznecov.v.spb@mail.ru; ул. Двинская, д. 5/7, Санкт-Петербург, 198035; р.т.: +7(904)616-59-03.

Kuznetsov Vitaliy Aleksandrovich — Ph.D. student, Admiral Makarov State University, of Maritime and Inland Shipping. Research interests: distributed and parallel systems, programming technology, cryptographic systems. The number of publications — 2. kuznecov.v.spb@mail.ru; 5/7, Dvinskaya str., St.Petersburg, 198035; office phone: +7(904)616-59-03.

Марлей Владимир Евгеньевич — д-р техн. наук, профессор, заведующий кафедрой вычислительных систем и информатики, ФГБОУ ВО Государственный университет морского и речного флота имени адмирала С.О. Макарова. Область научных интересов: автоматизация моделирования и программирования, моделирование, моделирование исторического процесса, мониторинг состояния объектов, теоретическое

программирование, распределенные и параллельные системы. Число научных публикаций — 150. vmarley@rambler.ru, <http://marley.spb.ru>; ул. Двинская, д. 5/7, Санкт-Петербург, 198035; р.т.: +7(921)900-0006.

Marley Vladimir Yevgenyevich — Dr. Sci., professor, head of computer system and informatics department, Admiral Makarov State University, of Maritime and Inland Shipping. Research interests: automation of modeling and programming, simulation of the historical process, condition monitoring of objects, theoretical programming, distributed and parallel systems. The number of publications — 150. vmarley@rambler.ru, <http://marley.spb.ru>; 5/7, Dvinskaya str., St.Petersburg, 198035; office phone: +7(921)900-0006.

Назаргулов Ильшат Азатович — инженер-программист, компания «Като». Область научных интересов: информационные системы и сети, криптографические системы. Число научных публикаций — 2. nazargulov.i@gmail.com, <https://kato.im/>; ул. Нагорная, д. 22, кв. 33, дер. Первомайская, Мелеузовский район, республика Башкортостан, 453874; р.т.: 8(911)945-44-03.

Nazargulov Ilshat Azatovich — programming engineer, «Kato». Research interests: information systems and networks, cryptographic systems. The number of publications — 2. nazargulov.i@gmail.com, <https://kato.im/>; 22, Nagornaja str., apt. 33, Pervomajskija, Republic of Bashkortostan, 453874, Russia; office phone: 8(911)945-44-03.

РЕФЕРАТ

Егоров А.Н., Кузнецов В.А., Марлей В.Е., Назаргулов И.А. **Модель распараллеливания вычислений для повышения эффективности восстановления доступа к данным в корпоративных сетях.**

Рассмотренная в статье программная оболочка распараллеливания (ПОР) реализует универсальный подход к организации эффективного исполнения прикладных решений пользователя. В соответствии с этим утверждением практически любую задачу, требующую для повышения производительности использования распараллеливания вычислений, можно решить в этой программной среде, реализовав необходимые библиотеки. Каждая из этих библиотек должна содержать значения параметров, требуемых для настройки среды программной оболочки и функции, реализующие вычисление оценок производительности, необходимых для эффективного распределения нагрузки между вычислительными устройствами, инициализацию задачи и ее декомпозицию на подзадачи, решение подзадач и обработку результатов их решения.

Архитектура ПОР позволяет практически не ограниченно расширять список поддерживаемых устройств и задач. На текущий момент программная оболочка может использовать центральные процессоры и графические видеокарты, которые поддерживают технологию CUDA компании NVIDIA.

Изложенный в статье подход к распараллеливанию вычислений при восстановлении секретного ключа позволяет поднять производительность для процессоров до 10 раз. При использовании алгоритма на основе технологии CUDA выигрыш может достигать значения в 5000 раз по сравнению с последовательной реализацией. Учитывая тот факт, что в настоящее время персональный компьютер может иметь в комплекте до 7 видеокарт, эффективность их применения для параллельных вычислений не вызывает сомнений.

Дальнейшее развитие ПОР предполагает реализацию объединения компьютеров в единую вычислительную сеть и использование более эффективных алгоритмов для восстановления секретных ключей, базирующихся на применении радужных таблиц для вычислений. Это позволит еще существеннее поднять производительность.

SUMMARY

Yegorov A.N., Kuznetsov V.A., Marley V.Y., Nazargulov I.A. **The Model of Parallel Computing to Effectiveness Improvement of the Restoration of Access to Data in Corporate Networks.**

The considered program shell of parallelizing implements a universal approach to organizing the effective execution of application solutions of a user. As consistent with this statement, almost any task a user is requiring to improve the performance of with the use of parallel computing, can be solved in this software environment by implementing the necessary libraries. Each of these libraries should contain the parameter values required to configure an environment of program shell and functions implementing power evaluation of hardware needed for effective load distribution between computing devices, tasks initialization and decomposition of the task into sub-task, sub-task solution and processing of the sub-task solutions.

Architecture program shell of parallelizing allows almost unlimited expanding the list of supported devices and tasks. At the moment the shell program can use central processors and graphic video cards which support the CUDA technology of the NVIDIA company.

The outlined approach to parallelization computing with restoring the secret key allows improving performance for processors by 10 times. When using algorithm on the basis of the CUDA technology the performance can improve by 5000 times in comparison with sequential implementation. Considering that now a personal computer can have in a set up to 7 video cards, efficiency of their use for parallel computing doesn't raise doubts.

Further development of program shell of parallelizing involves implementation of combining computers in a single computer network and use of more effective algorithms for restoration of the secret keys which are based on the use of rainbow tables for computation.

В.В. Диковицкий
**МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОЙ ОБРАБОТКИ И
ПРЕДСТАВЛЕНИЯ ИНФОРМАЦИИ В МУЛЬТИПРЕДМЕТНЫХ
ИНФОРМАЦИОННЫХ СИСТЕМАХ ПРОМЫШЛЕННЫХ
ПРЕДПРИЯТИЙ**

Диковицкий В.В. Методы интеллектуальной обработки и представления информации в мультипредметных информационных системах промышленных предприятий.

Аннотация. В работе рассматривается задача формирования единого информационного пространства промышленного предприятия в виде мультипредметной информационной системы промышленного предприятия (МИСПП). Предложена архитектура мультипредметной информационной системы промышленного предприятия, метод автоматизированного формирования семантической модели предметной области информационной системы на основе принципа «пользователь как эксперт», метод формирования когнитивных пользовательских интерфейсов, адаптированных для различных категорий пользователей, и метод поиска, обеспечивающий автоматизированное расширение запроса и оценку релевантности результатов поиска на основе совместного анализа формализованной ментальной модели и семантической модели предметной области с учетом субтрактивных отношений.

Ключевые слова: мультипредметная ИС, навигационная структура, когнитивный пользовательский интерфейс.

Dikovitsky V.V. Methods of Intellectual Data Processing and Presentation in Multi-Subject Information Systems of Industrial Enterprises.

Abstract. In this paper, the problem of improving the technology of formation and functioning of multi-subject intellectualized information systems of industry is considered. As a solution to this problem, we propose an architecture of multi-subject IS information system of an industrial enterprise; an automated method of formation of a semantic domain model based on the principle of "user as an expert."; a method of formation of cognitive user interfaces adapted for different categories of users; and search method, providing automated query expansion and evaluation of the relevance of search results based on a joint analysis of the formal mental models and semantic models of a domain with the subtractive relationship.

Keywords: multisubject IS, navigation structure, cognitive user interface.

1. Введение. Деятельность современных промышленных предприятий требует аккумуляции и обновления знаний в различных предметных областях. Диверсификация современных производств влечет рост количества процессов, отражающихся в нормативно-справочной информации (НСИ) промышленного предприятия, являющейся объектом рассмотрения в данной работе. НСИ современного производственного предприятия претерпевает быстрые изменения, одни и те же процессы и объекты описываются в различных документах и могут рассматриваться специалистами с различных точек зрения [1], что усугубляет проблему разнородности и рассогласованности информации.

Существующие корпоративные информационные системы обладают рядом недостатков, делающих их неприменимыми в существ-

вующих условиях производства. К ним относятся привлечение экспертов по знаниям для поддержания актуальности знаний, реорганизация управления, ориентированность на четко структурированную иерархическую систему процессов, комплексный подход к интеграции затрудняет применение уже существующих на производстве информационных сервисов[2].

Это обуславливает необходимость формирования единого информационного пространства промышленного предприятия в виде мультипредметной информационной системы [3] промышленного предприятия (МИСПП). Предлагаемые в данной работе методы основаны на использовании формализованных знаний [4-7] в интеллектуализированной замкнутой информационной системе в условиях динамических изменений предметной области и отсутствии специалиста по знаниям. Интеллектуализированная замкнутая информационная система – это информационная система, основанная на знаниях, обладающая обратной связью, позволяющей пользователю влиять на процесс обработки информации с целью осуществления эффективного поиска. Повышение эффективности достигается за счет динамического обновления и адаптации к различным ментальным стереотипам пользователей семантической модели предметной области (СМПО) на основе пользовательского опыта (концепция “user as an expert” [8]).

Далее представлены архитектура МИСПП, принципы построения и формальное описание ментальных моделей категорий пользователей, метод автоматизированного формирования СМПО, формальная постановка задачи и метод формирования адаптивных пользовательских интерфейсов, метод поиска информации с учетом субтрактивных отношений СМПО - отношений между концептами СМПО, имеющих отрицательный весовой коэффициент. Использование субтрактивных отношений СМПО позволяет автоматизировать процесс добавления ограничений в расширенный запрос. Данные методы направлены на решение задачи организации эффективного доступа средствами поиска и навигации к ресурсам информационной системы промышленного предприятия.

2. Модель мультипредметной информационной системы промышленного предприятия. Предметная область современных производственных процессов по составу и структуре динамична. Для поддержки актуальности знаний требуется верифицируемая семантическая модель предметной области. Реализации обратной связи с целью адаптации представления информации и верификации СМПО требует хранения и учета пользовательского опыта в виде формализованных ментальных моделей (ФММ). Наличие ФММ и СПМО в

структуре ИС вносит изменения в процесс обработки данных. Логическая структура МИСПП представлена на рисунке 1.

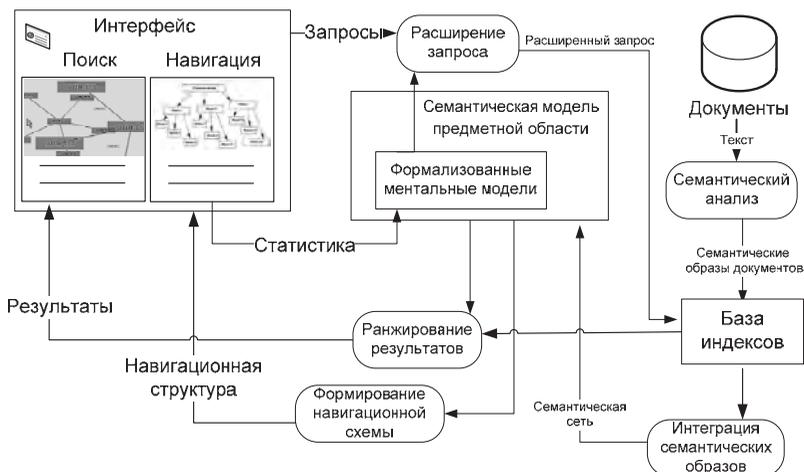


Рис. 1. Логическая структура мультипредметной информационной системы промышленного предприятия

Основу МИСПП составляют семантическая модель предметной области (СМПО), формализованная ментальная модель пользователей (ФММ), база индексов документов НСИ. Входными данными, обеспечивающими функционирование МИСПП, являются документы предприятия и статистика работы пользователя. Выходными данными являются результаты обработки запросов и навигационная структура интерфейса. СМПО формируется в результате интеграции семантических образов, вновь появляющихся в информационных хранилищах документов, и последующей верификации структуры модели на основе формализованных ментальных моделей пользователей. Семантический образ документа – семантическая сеть, множество вершин которой составляют понятия СМПО, присутствующие в документе, множество ребер – множество двухместных отношений над понятиями. Запрос – множество понятий предметной области, представленных множеством ключевых слов или множеством элементов навигационного интерфейса. Информационный элемент – документ информационной системы предприятия. Формализованная ментальная модель - семантическая сеть, множество вершин которой составляют понятия СМПО, которыми оперирует пользователь, множество ребер – множество взвешенных двухместных отношений над понятиями. ФММ служит «эталоном» в процессе формирования пользовательского интерфейса и оцен-

ки его когнитивности как степени соответствия структуры интерфейса формализованной ментальной модели пользователя. ФММ формируется на основе обработки запросов пользователя и статистики его работы с информационными элементами. Схема взаимодействия пользователя и мультимедийной информационной системой может быть представлена следующим алгоритмом (рисунок 2):

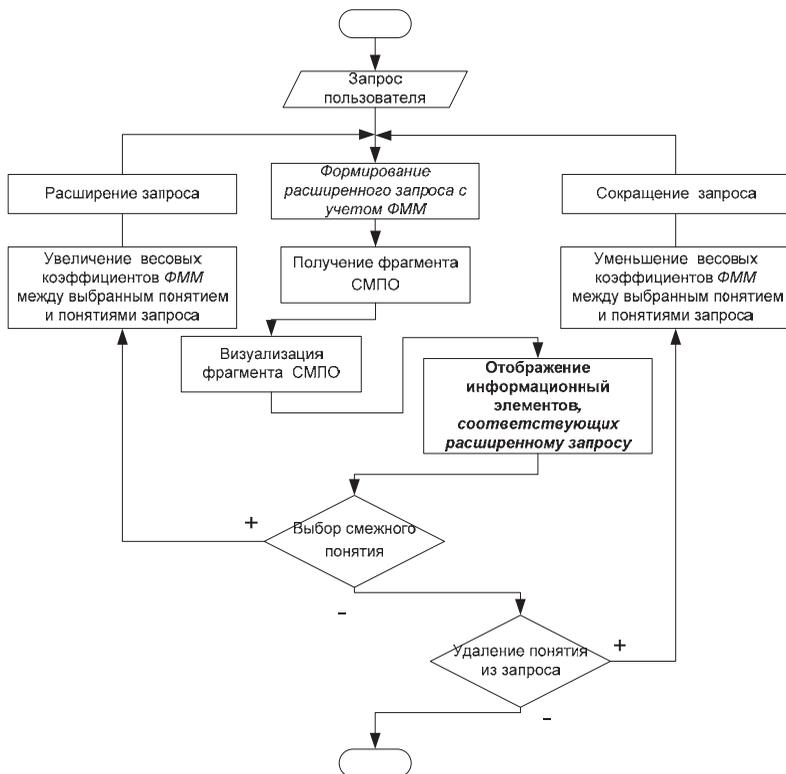


Рис.2. Алгоритм взаимодействия пользователя и ИС

Взаимодействие заключается в итеративном расширении запросов пользователя на основе ФММ, обеспечении возможности коррекции запроса средствами адаптивного интерфейса, а также учета пользовательских предпочтений путем коррекции весовых коэффициентов между понятиями ФММ. При совместном употреблении в запросе пар понятий, входящих в состав ФММ, увеличиваются весовые коэффициенты отношений между данными понятиями. Системные информаци-

онные базы используются в дальнейшем для формирования навигационно-поискового интерфейса, обеспечивающего эффективный, в смысле скорости и релевантности, доступ пользователя к требуемым данным. Последовательность действий, описывающих взаимодействие компонентов МИСПП, представлена следующими диаграммами:

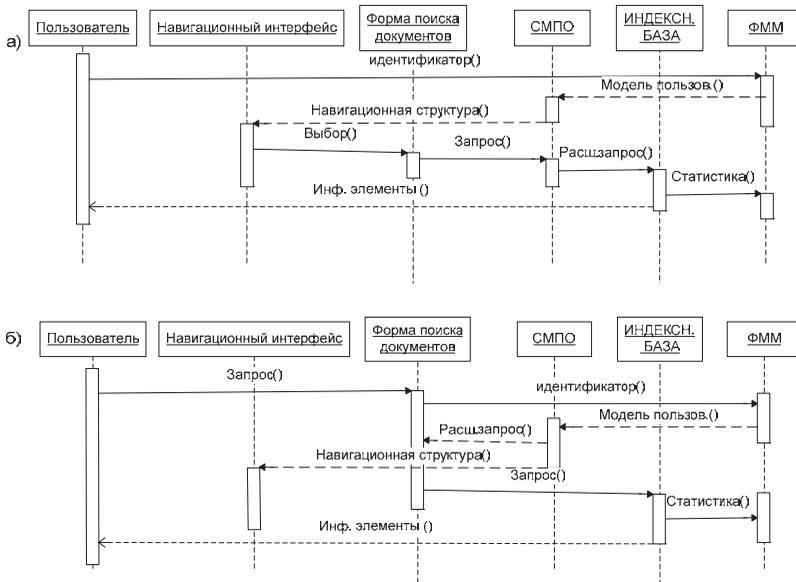


Рис.3. Диаграмма последовательности действий, а) навигация б) поиск

С целью снижения объемов хранимых системных данных модель строится не для каждого пользователя в отдельности, а для категорий пользователей. Формально категории пользователей определяются следующим образом: C - некоторое множество понятий (концептов), U - множество пользователей. Каждый концепт c имеет множество атрибутов:

$$A(c) = \{a(c)_i\}, a(c)_i \in C, i = \overline{1, N_c},$$

Упорядочив множество атрибутов по убыванию степени их значимости для пользователя u , получим последовательность, характеризующую его представление о данном концепте:

$$A^u(c) = \{a^u(c)_i\}, i = \overline{1, N_c} : a^u(c)_i \varphi^u a^u(c)_j, \forall i \leq j,$$

где φ^u - отношение, задающее значимость атрибутов для пользователя u ; $a \varphi^u b$ означает, что «для пользователя u a не менее значим, чем b ».

Определим группу пользователей, имеющих схожие представления о понятиях из некоторого множества C . Назовем подобную группу пользовательской *категорией* k -го порядка на множестве концептов C , и определим ее следующим образом:

$$U_C^k = \{u \mid \{a^u(c)_i\} = \{a^{u'}(c)_i\}, i = \overline{1, k}, \forall c \in C, \forall u' \in U_C^k\},$$

Ментальная модель некоторой k -й категории пользователей представлена взвешенным мультиграфом:

$$UM_k = \{C, L_k\}, L_k = \{l_k^{jm}\},$$

где C – множество вершин графа, представляющих понятия СМПО, L_k – множество взвешенных дуг, вес которых характеризует значимость семантического отношения m -го типа между i -м и j -м понятиями для k -й категории пользователей.

В совокупности, ФММ всех пользователей образуют фрагмент базы знаний МИСПП, представляющий собой мультиграф с векторными весами дуг. Матрица инцидентности мультиграфа имеет размерность 3:

$$M_l : C \times L \times U_k \rightarrow w_k^{ij},$$

где C – множество вершин графа, представляющих понятия СМПО, L – множество дуг, задающих отношения над C , U_k – множество категорий пользователей. Элементами матрицы являются весовые коэффициенты w_k^{ij} , задающие вес связи между концептами c_i и c_j для k -й категории пользователей. На рисунке 4 представлены фрагменты ментальных моделей пользователей различных категорий.



Рис.4. Атрибуты понятия «Карьерный экскаватор» для различных категорий пользователей

3. Метод автоматизированного формирования семантической модели предметной области промышленного предприятия информационной системы на основе принципа «пользователь как эксперт».

Формально СМПО представлена n-арной семантической сетью:

$$KB = \{C, L, Tr\}, Tr = \{synonymOf, HyponymOf, associateWith, subStruct\}$$

где C - множество концептов, L - множество отношений над концептами. Tr – множество типов отношений. Основой СМПО выступает тезаурус русского языка, расширяемый семантическими образами документов информационной системы. Процесс формирования семантической модели предметной области предприятия на основе коллекции документов информационной системы и расширяемого тезауруса состоит из следующих этапов:

1. Формирование семантического образа документа D статистическими и лексико-грамматическими методами обработки текста:

$$D = \{C_D, L_D\}, C_D \subset C, L_D = C_D \times C_D, L_D \subset L,$$

где D - семантическая сеть, C_D - множество концептов, выделенных в

документе, L_D - множество отношений, выделенных в документе, C - множество всех концептов СМПО, L - множество всех отношений СМПО.

2. Интеграция семантических образов D в СМПО на основе модифицированной составной семантической метрики [9], заключающейся в оценке близости двух концептов путем вычисления следующих оценок:

I. Вычисление оценки сходства контекста концептов документа C_D с контекстом СМПО:

$$\forall c_i^D \in C : \text{Syn}(c_i^D, c_j^{KB}) = |C_s^D \cap C_s^{KB}|, i = \overline{1..n}, j = \overline{1..m},$$

где C_s^D, C_s^{KB} - множество синонимов концепта c_i^D и c_j^{KB} соответственно.

II. Вычисление оценки сходства структурного положения концептов документа C_D и структурного положения концептов СМПО как количества общих гипонимов:

$$\forall c_i^D \in C : \text{Poseq}(c_i^D, c_j^{KB}) = |C_H^D \cap C_H^{KB}|, i = \overline{1..n}, j = \overline{1..m}$$

где C_H^D, C_H^{KB} - множество гипонимов концепта c_i^D и c_j^{KB} соответственно.

III. Вычисление оценки сходства имен понятий c документа D с именами понятий СМПО:

$$\forall c_i^D \in C : \text{Eq}(c_i^D, c_j^{KB}) = x/l, i = \overline{1..n}, j = \overline{1..m}$$

где x – длина эквивалентной цепочки символов в именах понятий c_i^D и c_j^{KB} , l – длина имени понятий.

IV. Добавление отсутствующих в СМПО концептов и отношений на основании результатов вычисления пороговой функции от среднего трех оценок:

$$f(c_i^D) = \frac{a \cdot \text{Eq}(c_i^D, c_j^{KB}) + b \cdot \text{Poseq}(c_i^D, c_j^{KB}) + \text{Syn}(c_i^D, c_j^{KB})}{3} > z,$$

$$i = \overline{1..n}, j = \overline{1..m},$$

где a, b, c – некоторые коэффициенты, z – значение пороговой функции, которые определяются экспертно, исходя из объема и разнородности

предметной области.

3. Верификация СМПО осуществляется изменением весовых коэффициентов существующих отношений между понятиями. Данный процесс инициируется при совместном использовании двух понятий в одном пользовательском запросе. Величина изменения весового коэффициента определяется следующим образом:

$$dWU'(t, n_i) = \frac{\sum_{\langle t, n_j \rangle \in (ACW)} (WU_k(t, n_j))}{|ACW|} * (WU(t, n_i) + dWU(t, n))$$

– для весового коэффициента отношения между совместно использованными понятиями, и

$$dWU'(t, n_i) = - \frac{\sum_{\langle t, n_j \rangle \in (ACW)} (WU_k(t, n_j))}{|CW / ACW|} * (WU(t, n_i) + dWU(t, n))$$

– для весового коэффициента отношения между остальными (неиспользованными) понятиями, где $dWU(t, n)$ - предыдущее значение изменения коэффициента, ACW и CW – множества отношений с совместно использованными и неиспользованными понятиями.

Предложенный метод предлагается использовать для автоматизированного формирования и обеспечения актуального состояния семантической модели предметной области производственного предприятия в условиях динамических изменений предметной области без привлечения эксперта по знаниям. На основе полученной модели решаются задачи обеспечения эффективного доступа пользователей к НСИ предприятия, а именно реализуются методы поиска и формирования адаптивных пользовательских интерфейсов.

5. Метод формирования адаптивных пользовательских интерфейсов обеспечивает повышение когнитивности МИС за счет автоматизированного формирования интерфейсов, адекватных моделям пользовательских интересов.

Пользовательский интерфейс представляет собой пару:

$$UI = \langle I, s \rangle,$$

где I – множество информационных элементов; s – навигационная структура. Навигационная структура определяет иерархию групп информационных элементов (ИЭ). На каждом уровне иерархии исходное множество информационных элементов делится на

подмножества в соответствии с одним или несколькими классификационными признаками. В качестве классификационных признаков используются атрибуты понятий предметной области. Очевидно, что при использовании на одном уровне навигационной структуры нескольких атрибутов, полученные множества ИЭ могут пересекаться. Введем следующие обозначения:

$\Gamma^l = \{G_i^l\}$ – множество разделов l -го уровня навигационной структуры;

G_i^l – i -я группа информационных элементов l -го уровня навигационной структуры;

$P^l = \{p_i^l\}$ – множество классификационных атрибутов, используемых для формирования групп ИЭ на l -м уровне навигационной структуры.

Взаимодействия пользователя и МИСПП заключается в поиске некоторых информационных элементов по имеющемуся у пользователя образу. При этом образ, чаще всего, неточный: в нем специфицируются лишь некоторая часть идентифицирующих атрибутов. Вследствие этого, пользователь с разной степенью уверенности может предполагать в какой из групп ИЭ на некотором уровне навигационной структуры находится искомый элемент. Эта уверенность тем выше, чем более точно представляет пользователь информационные элементы, составляющие потенциальное содержимое группы. Введем следующую функцию, задающую числовую оценку степени уверенности пользователя u (чем выше значение, тем выше степень уверенности):

$$p^u : \Gamma^l \rightarrow (0,1] .$$

Оценка времени, требуемого для доступа к искомому информационному элементу в рамках навигационной структуры на l -м уровне, будет равна:

$$O\left(\frac{\max_i |G_i^l|}{p^u(G_i^l)}\right).$$

Таким образом, при прочих равных условиях, степень уверенности пользователя в принадлежности информационного элемента к той или иной группе определяет качество интерфейса в смысле скорости доступа к требуемой информации.

Пусть навигационная структура интерфейса имеет глубину \hat{l} уровней. Тогда в качестве количественной оценки когнитивности

интерфейса для пользователя u может использоваться сумма:

$$\sum_{l=1}^{\hat{l}} p^u(l),$$

где \hat{l} – количество уровней навигационной структуры интерфейса, $p^u(l)$ – функция, задающая числовую оценку степени уверенности пользователя u в какой из групп ИЭ на уровне l навигационной структуры находится искомый элемент.

Данная мера может использоваться для оценки уже существующих интерфейсов, когда известно значение \hat{l} . Для решения же прямой задачи, то есть структуризации исходного множества информационных элементов в рамках навигационной структуры, требуется учитывать дополнительные ограничения. Эти ограничения обусловлены психологией восприятия человека, ограничивающей максимальное количество одновременно эффективно воспринимаемых объектов. Вследствие этого необходимо ограничивать размер группы информационных элементов, а также глубину навигационной структуры.

С учетом сказанного, оптимальная для пользователя u структура интерфейса есть решение следующей задачи с ограничениями [10]:

$$\begin{aligned} \max_s \sum_{l=1}^{\hat{l}(s)} p^u(l) \\ g(s) \leq K, \\ \hat{l}(s) \leq K'. \end{aligned}$$

Здесь $\hat{l}(s)$ – количество уровней в навигационной структуре s ; $g(s)$ – максимальный размер группы информационных элементов $\hat{l}(s)$ – го в навигационной структуре s ;

K – когнитивная константа, определяющая максимальное число одновременно предъявляемых пользователю информационных элементов для их эффективного восприятия;

K' – когнитивная константа, определяющая максимальное число уровней навигационной структуры, в рамках которых поиск информации для пользователя остается комфортным. Константы K и K' определяются экспертно исходя из соображений психологии восприятия [11–14].

Для формирования навигационного интерфейса, удовлетворяющего приведенной формулировке задачи синтеза, предлагается процедура, основанная на формализованной ментальной модели. Процедура содержит несколько этапов:

1. Определение информационной потребности специалиста на основе модели пользовательских интересов и текущего запроса:

$$UQ = f_{tr}(Q, UM), UM : \exists l \exists L : t_i \in Q, t_j \in UM,$$

где UM - модель пользовательских интересов; Q - запрос специалиста, f_{tr} - функция трансляции запроса в модель пользовательских интересов;

2. Определение множества информационных элементов навигационной структуры, соответствующих информационной потребности специалиста.

$$G = \{g \mid \exists a^u = g\}, a^u \in UQ,$$

где G - множество информационных элементов навигационной структуры; a^u - концепты текущей информационной потребности UQ .

3. Разбиение множества информационных элементов навигационной структуры на подмножества и их ранжирование в соответствии с весовыми коэффициентами модели пользовательских интересов.

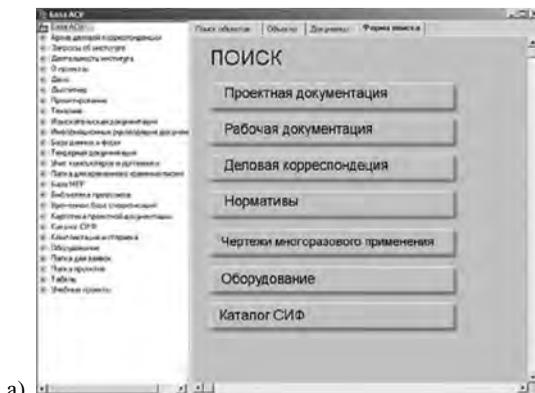
$$G_i^d = \{g_k \mid (\forall g_k, g_m \exists l \in L : w(g_k, g_m) > x) \wedge (\exists g_z : \exists l_{sup}(g_k, g_z), l_{sup}(g_m, g_z) \in L)\}$$

$$k, m = \overline{1..n}, z = \overline{1..d}, G_i^d \subset G,$$

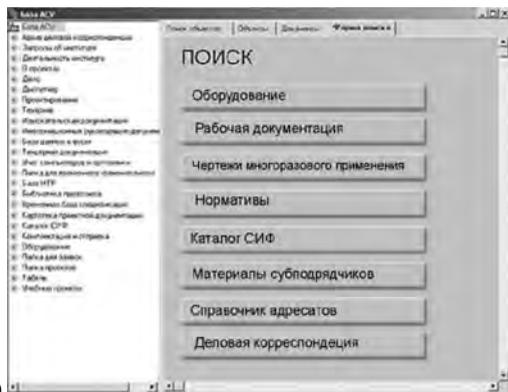
где G_i^d - i -я группа информационных элементов d -го уровня навигационной структуры; $l_{sup}(g_k, g_z)$ - отношение классификации в модели интересов пользователя;

$w(g_k, g_m)$ - весовой коэффициент отношения l над информационными элементами g_k, g_m ; d - количество уровней навигационной структуры; x - порог вхождения информационного элемента в навигационную структуру, задается на основе ограничений на максимальное число одновременно предъявляемых пользователю информационных элементов K .

На основе групп множеств G формируется интерфейс мультипредметной информационной системы в виде множества меток («tags»), отражающих информационное содержание ресурсов в терминах СМПО с учетом модели категории пользователей. На рисунке 5 представлен адаптивный пользовательский интерфейс, динамически изменяющий навигационную структуру в зависимости от ментальной модели пользователя.



а)



б)

Рис.5. Вид пользовательского интерфейса для категорий пользователей:
а) отделов «Управление производством»; б) «Техническое планирование»

6. Метод поиска информации с учетом субтрактивных отношений позволяет увеличить эффективность поиска в смысле релевантности результатов за счет автоматического включения в запрос ограничений. Ограничения формируются на основе субтрактивных отношений искомого концепта, заданных в СМПО. Использование субтрактивных отношений СМПО позволяет автоматизировать процесс добавления ограничений в расширенный запрос.

Метод поиска включает 3 составляющих: Способ представления документа, способ представления запроса и функцию соответствия запроса результатам. Документ представлен фрагментом СМПО и множеством ключевых слов (меток) в базе индексов D , выделенных семантическим анализатором на этапе индексации. Запрос представлен

в виде множества ключевых слов Q . Процесс поиска документов, соответствующих запросу, состоит из следующих этапов:

1. Формирование запроса в терминах СМПО:

1.1. Формирование расширенного запроса EQ , содержащего отношения и соответствующие запросу $Q = \{c_i\}$ концепты СМПО:

$$EQ = f_{tr}(KB, Q) = \{T^q, L^q \mid Eq(c_i^Q, t_j^{KB}) > 1 - \varepsilon\}, L = \{l\} = \{< t_i, t_j, tp, \bar{w}_k >\},$$

$$i = \overline{1..n}, j = \overline{1..m}, tp \in Tp, k = \overline{1..r},$$

где KB – семантическая модель предметной области, T^q – множество концептов СМПО, L^q – множество отношений над концептами, $f_{tr}()$ – функция трансляции запроса в модель пользовательских интересов, устанавливающая соответствие между словами запроса и концептами базы знаний, $Eq()$ – оценка сходства концептов, t_i, t_j – концепты, tp – тип отношения, \bar{W} – вектор весовых коэффициентов отношений, компоненты которого соответствуют уровню значимости отношений для различных групп пользователей, r – количество групп пользователей, ε – погрешность оценки сходства концептов.

1.2. Расширение запроса с учетом весовых коэффициентов отношений и субтрактивных отношений, ограничивающих контекст запроса:

$$EQ = \{T^q, L^q\} \cup \{T', L' \mid \forall l \in L' : t_i \in T^q, t_j \in T', f_u(\bar{w}_k, n) > x\},$$

$$\{T', L'\} \in KB, \bar{w}_k = \langle w_1, \dots, w_k, \dots, w_r \rangle,$$

где T' – множество концептов СМПО, связанных с концептами множества T^q отношениями из множества L' вида, $f_u(\bar{w}_k, r)$ – функция получения n -ой компоненты вектора весовых коэффициентов отношения l , x – порог включения отношения в расширенный запрос, определяемый экспертно исходя из точности и полноты поиска по коллекции документов.

2. Получение множества документов D , соответствующих расширенному запросу EQ :

$$D = \{d_i \mid T^{d_i} \cap T^q \neq \emptyset\},$$

где T^{d_i} – множество терминов СМПО, присутствующих в документе d_i , T^q – множество терминов СМПО, присутствующих в запросе EQ .

3. Ранжирование множества документов с учетом весовых коэффициентов отношений:

$$R(d_k) = \sum_{L_{d_k}} (f_u(\bar{w}_k, r)) - \sum_{L'_{d_k}} (f_u(\bar{w}_k, r)),$$

$$L_{d_k} = \{l^d \mid (t_i, t_j \in d_k) \wedge (tp \in \{\text{synonymOf}, \text{HyponymOf}, \text{associateWith}\})\},$$

$$L'_{d_k} = \{l^d \mid (t_i, t_j \in d_k) \wedge (tp \in \{\text{subStract}\})\}, \quad i, j = \overline{1..n}, k = \overline{1..m}, tp \in Tp,$$

где $f_u(\bar{w}_k, r)$ – функция получения компоненты вектора весовых коэффициентов отношений из множества L_{d_k} между концептами t_i и t_j , присутствующими в документе d_k , для категории пользователей r . Tp – множество типов отношений. Результатом ранжирования является упорядоченное по убыванию оценки $R(d_k)$ множество документов $D = \{d_k\}$, представляющих результаты поиска. Таким образом, на ранжирование результатов оказывают влияние наличие понятий формализованной ментальной модели в документе, весовые коэффициенты между понятиями формализованной ментальной модели, коррелирующие с частотой использования понятий пользователем, а также наличие субтрактивных отношений.

7. Апробация. Предложенные методы были апробированы в информационной системе крупного промышленного предприятия региона – ОАО «Оленегорский горно-обогатительный комбинат». Семантическую модель предметной области мультипредметной информационной системы составляет русскоязычный тезаурус WordNet 3.0, расширенный результатами работы семантического анализатора над коллекцией нормативно-справочных документов предприятия. Эффективность предложенных методов была проверена путем натурального эксперимента с привлечением тестовой выборки пользователей, которым предлагалось решить схожие информационно-поисковые задачи с помощью существующих систем поиска с использованием интерфейса строки поиска и с помощью адаптивного интерфейса мультипредметной информационной системы.

Для оценки характеристик адаптивного пользовательского интерфейса был использован метод GOMS [15], заключающиеся в сравнении временных характеристик выполнения пользователями производственных задач в рамках существующего и разрабатываемого интерфейса. Целью пользователей являлось получение доступа к требуе-

тому документу. Среднее время выполнения операции представлены в таблице 1.

Таблица 1. Оценка показателей использования методов поиска

Оцениваемые характеристики	Навигация с использованием существующего интерфейса	Навигация с использованием адаптивного интерфейса
Среднее время, затраченное на навигационный поиск, с.	31,8	20,85

Для оценки метода поиска осуществлялся поиск по заранее проиндексированной коллекции объемом 14 тыс. документов. Экспертами оценивались процесс поиска результаты выполнения 10 запросов. В качестве критериев оценки выступали скорость поиска – время, затраченное на удовлетворение информационной потребности, выраженной одним запросом; точность – соответствие результатов запросу; и полнота результатов – полнота охвата документов с упоминанием об объекте поиска:

$$Precision = \frac{|D_{rel} \cap D_{retr}|}{|D_{retr}|}, \quad Recall = \frac{|D_{rel} \cap D_{retr}|}{|D_{rel}|},$$

где D_{rel} – множество релевантных документов в базе, D_{retr} – множество документов, найденных системой.

Для оценки альтернатив экспертам была предложена лингвистическая шкала измерений.

Оценка i -й альтернативы производилась j -м экспертом по формуле:

$$v_{ij} = 1 - \frac{(l-1)}{k},$$

где l – индекс значения лингвистической шкалы; k – количество значений этой шкалы.

Для оценки i -й альтернативы n экспертами используется формула:

$$s_i = \sum_{j=1}^n v_{ij}.$$

Результаты оценки характеристик приведены в таблице 2.

Таблица 2. Оценка показателей использования методов поиска

Оцениваемые характеристики	Используемый метод поиска документов	
	Поиск по ключевым словам с использованием интерфейса строки ввода	Поиск с использованием метода поиска информации с учетом субтрактивных отношений
Оценка скорости выполнения поиска	0,45±0,09	0,82±0,15
Точность результатов	0,96±0,04	0,91±0,08
Полнота результатов	0,73±0,09	0,94±0,06
Среднее значение оценок	0,71	0,89

Высокая точность поиска с использованием интерфейса строки ввода обусловлена знакомством пользователей с предметной областью и, как следствие, малой неопределенностью при формировании запроса, а также относительно малым объемом коллекции документов. Тем не менее, результаты экспериментов позволяют сделать вывод о корректности и обоснованности использования динамически формируемой и верифицируемой СМПО для реализации адаптивного интерфейса и метода поиска информации с учетом субтрактивных отношений в мультипредметных информационных системах промышленных предприятий.

8. Заключение. Рост объема, разнородность и динамика изменения информации современных промышленных предприятий обуславливают необходимость развития методов оперативного доступа к ней. Одним из путей решения данной проблемы является построение мультипредметных информационных систем, способных предоставить пользователю необходимые функциональные возможности для оперирования большими массивами данных. В данной работе представлены архитектура МИСПП, принципы построения и формальное описание ментальных моделей категорий пользователей, метод автоматизированного формирования СМПО, формальная постановка задачи и метод формирования адаптивных пользовательских интерфейсов, метод поиска информации с учетом субтрактивных отношений СМПО. В рамках документооборота промышленного предприятия проведена программная реализация и оценка эффективности разработанных методов. Результаты проведенного анализа эффективности разработанных методов мультипредметных информационных систем с использованием разработанного программного обеспечения показали, что при осуществлении навигации обеспечивается сокращение времени доступа к необходимым документам в 1,5 раза, при осуществлении информационного поиска обеспечивается сокращение времени поиска в 1,9 раза, и увеличение полноты результатов в 1,2 раза.

Литература

1. *Acker L., Porter B.* Extracting viewpoints from knowledge bases // Proceedings of The 12th National Conference on Artificial Intelligence. 1994. pp. 547–552.
2. *Трубейко К.Н., Кулешов А.А., Клебанов А.Ф., Владимиров Д.Я.* Современные системы управления горнотранспортными комплексами // СПб.: Наука. 2007. 344 с.
3. *Диковицкий В.В., Ломов П.А., Сепеда-Эррера Р.Р., Шишаев М.Г.* Современные методы создания мультипредметных веб-ресурсов на базе визуализации и обработки формализованной семантики // Вестник Кольского научного центра РАН. 2011. №3. С.63–73.
4. *Смирнов А.В., Пацкин М.П., Шилов Н.Г., Левашова Т.В.* Онтологии в системах искусственного интеллекта: способы построения и организации // Новости искусственного интеллекта. 2002. №1. С.3–13.
5. *Гаврилова Т.А., Гулякина Н.В., Голенков В.В.* Технология проектирования интеллектуальных систем // Информационные системы и технологии (IST'2009): Мат. V Междунар. конф.-форума. Минск: А.Н.Вараксин. 2009. С.93–96.
6. *Коголовский М.Р.* Перспективные технологии информационных систем // М.: Компания АйТи, 2003. 288 с.
7. *Кучуганов В.Н.* Элементы теории ассоциативной семантики // Управление большими системами: Сб. научн. тр. М.: ИПУ РАН. 2012. Вып. 40. С. 30–48.
8. *Шишаев М.Г., Ломов П.А., Диковицкий В.В.* Использование концепции «user as an expert» в разработке мультипредметных веб-ресурсов, основанных на онтологиях // Труды Института системного анализа РАН: Прикладные проблемы управления макросистемами. 2012. Т. 62. С.40–47.
9. *Шишаев М.Г., Ломов П. А.* Интеграция семантически связанных информационных ресурсов на основе онтологий // Информационные технологии в региональном развитии: Сборник научных трудов ИИММ КНЦ РАН. Апатиты: Изд-во КНЦ РАН. 2007. Вып. VII. С. 61–68.
10. *Шишаев М.Г., Ломов П.А., Диковицкий В.В.* Формализация задачи построения когнитивных пользовательских интерфейсов мультипредметных информационных ресурсов // Труды Кольского научного центра РАН. Информационные технологии. 2013. С.90–97.
11. *Koffka K.* Principles of Gestalt psychology // N.Y.: Routledge. 1935. 720 p.
12. *Miller G.* The Magical Number Seven, Plus or Minus Two // The Psychological Review. 1956. pp. 81–97.
13. *Brooke J., Jordan P.W., Thomas B., Weerdmeester B.A., McClelland I.L.* SUS: A 'quick and dirty' usability scale // Usability evaluation in industry. London: Taylor & Francis. 1996. pp.189–194.
14. *Салвенди Г.* Человеческий фактор // М: Мир. 1991.
15. *John, B. E., Kieras, D. E.* The GOMS family of user interface analysis techniques: Comparison and contrast // ACM Transactions on Computer-Human Interaction. 1996. pp. 320–351.

References

1. *Acker L., Porter B.* Extracting viewpoints from knowledge bases. Proceedings of The 12th National Conference on Artificial Intelligence. 1994. pp. 547–552.
2. *Trubeckoj K.N., Kuleshov A.A., Klebanov A.F., Vladimirov D.Ja.* *Sovremennyye sistemy upravleniya gorno-transportnymi kompleksami* [Modern control systems of mining and transport]. St. Petersburg. 2007. 344 p. (In Russ.).
3. *Dikovickij V.V., Lomov P.A., Sepeda-Jerrera R.R., Shishaev M.G.* [Modern methods of creating multi-subject web resources on the basis of visualization and processing of

- formal semantics]. *Vestnik Kol'skogo nauchnogo centra RAN – Herald of Kola Science Centre RAS*. 2011. vol. 3. pp. 63–73. (In Russ.).
4. Smirnov A.V., Pashkin M. P., Shilov N. G., Le-vashova T. V. [Ontologies in artificial intelligence systems: methods of construction and organization] *Novosti iskusstvennogo intellekta – AI NEWS Newsletter*. Moscow. 2002. vol. 1. pp.3–13. (In Russ.).
 5. Gavrilova T.A., Guljakina N.V., Golenkov V.V. [Technology of designing intelligent systems]. *Informacionnye sistemy i tehnologii (IST'2009): materialy V Mezhdunar. konf.-foruma [Information Systems and Technology (IST'2009): Collected papers]*. Minsk: A.N.Varaksin. 2009. pp. 93–96. (In Russ.).
 6. Kogalovskij M.R. *Perspektivnye tehnologii informacionnyh sistem [Promising Information Systems Technology]*. Moscow. 2003. 288 p. (In Russ.).
 7. Kuchuganov V.N. [Elements of the theory of associative semantics]. *Upravlenie bol'shimi sistemami: Sbornik trudov [Managing large systems: Collected papers]*. Moscow: IPU RAN. 2012. pp. 30–48. (In Russ.).
 8. Shishaev M.G., Lomov P.A., Dikovickij V.V. [Using the concept of «user as an expert» in the development of multi-subject web resources, based on ontologies]. *Trudy Instituta sistemnogo analiza RAN: Prikladnye problemy upravlenija makrosistemami – Proceedings of the Institute for Systems Analysis: Applied management problems of macrosystems*. Moscow. 2012. pp.40–47. (In Russ.).
 9. Shishaev M.G., Lomov P.A. [The integration of semantically related information resources on the basis of ontologies]. *Informacionnye tehnologii v regional'nom razviii: Sbornik nauchnyh trudov IIMM KNC RAN – Information technologies in regional development: Proceedings of the IIMM KSC*. Apatity: KNC RAN. 2007. vol. VII. pp. 61–68. (In Russ.).
 10. Shishaev M.G., Lomov P.A., Dikovickij V.V [The formalization of the task of building cognitive user interfaces for multi-subject Information Resources]. *Trudy Kol'skogo nauchnogo centra RAN. Informacionnye tehnologii – Proceedings of the Kola Scientific Center, Russian Academy of Sciences. Information technology*. 2013. pp. 90–97. (In Russ.).
 11. Koffka K. Principles of Gestalt psychology. N.Y.: Routledge. 1935. 720 p.
 12. Miller G. The Magical Number Seven, Plus or Minus Two. *The Psychological Review*. 1956. pp. 81–97.
 13. Brooke J., Jordan P.W., Thomas B., Weerdmeester B.A., McClelland I.L. SUS: A 'quick and dirty' usability scale. *Usability evaluation in industry*. London: Taylor & Francis. 1996. pp. 189–194.
 14. Salvendi G. *Chelovecheskij faktor [Human factor]*. Moscow: Mir, 1991. (In Russ.).
 15. John B.E., Kieras D.E. The GOMS family of user interface analysis techniques: Comparison and contrast. *ACM Transactions on Computer-Human Interaction*. 1996. pp. 320–351.

Диковицкий Владимир Витальевич — младший научный сотрудник, Федеральное государственное бюджетное учреждение науки Институт информатики и математического моделирования технологических процессов Кольского научного центра Российской академии наук. Область научных интересов: интеллектуализированные системы, адаптивный интерфейс, структурирование знаний, семантика. Число научных публикаций — 22. dikovitsky@gmail.com; ул. Ферсмана, д.30, кв.57, Апатиты, Мурманская обл., 184209; р.т.: +79021364888.

Dikovitsky Vladimir Vital'evich — junior researcher, Establishment of Russian Academy of Sciences Institute for Informatics and Mathematical Modelling of Technological Processes of the Kola Science Center RAS. Research interests: intellectualized systems, adaptive interface, structured knowledge, semantics. The number of publications — 22. dikovitsky@gmail.com; 30-57, Fersman St., Apatity, Murmansk Region, 184209; office phone: +79021364888.

РЕФЕРАТ

Диковицкий В.В. **Методы интеллектуальной обработки и представления информации в мультипредметных информационных системах промышленных предприятий.**

В работе рассмотрена задача формирования единого информационного пространства промышленного предприятия в виде мультипредметной информационной системы промышленного предприятия (МИСПП).

Во введении перечислены проблемы предметной области, в качестве решения предлагается формирование единого информационного пространства промышленного предприятия в виде мультипредметной информационной системы на основе использования формализованных знаний.

Рассмотрена логическая структура МИСПП, перечислены основные компоненты, обеспечивающие функционирование ИС, а так же принципы формирования семантической модели предметной области (СМПО), формализованной ментальной модели пользователей (ФММ), базы индексов документов.

Представлен метод автоматизированного формирования семантической модели предметной области промышленного предприятия на основе принципа «пользователь как эксперт». Формально описана СМПО и этапы процесса ее формирования.

Представлен метод формирования адаптивных пользовательских интерфейсов, обеспечивающий повышение когнитивности мультипредметной информационной системы за счет автоматизированного формирования интерфейсов, адекватных моделям пользовательских интересов. На основе ФММ и ограничений задачи синтеза предложена процедура формирования структуры навигационного интерфейса.

Представлен метод поиска, обеспечивающий автоматизированное расширение запроса и оценку релевантности результатов поиска на основе совместного анализа формализованной ментальной модели и семантической модели предметной области с учетом субтрактивных отношений. Рассмотрены способ представления документа, способ представления запроса и функция соответствия между ними, составляющие соответствующие этапы процедуры поиска, отличающейся автоматической выработкой ограничений на основе СМПО и адаптивным ранжированием результатов поиска на основе ФММ.

В заключении приводятся результаты апробации предложенных методов.

SUMMARY

Dikovitsky V.V. **Methods of Intellectual Data Processing and Presentation in Multi-Subject Information Systems of Industrial Enterprises.**

The paper considers the problem of forming a unified information space of an industrial enterprise in the form of a multi-subject information system of industrial enterprise (MISIE).

In introduction, problems of subject domain are listed. As a solution to these problems, formation of a unified information space of the industrial enterprise in the form of multi-subject information system based on the use of formalized knowledge is proposed.

The logical structure of MISIE is considered. The main components providing functioning of the IS as well as principles of the formation of a semantic domain model (SDM), a formalized mental model of users (FMM), and index databases are listed.

The method of automated formation of a semantic domain model of an industrial enterprise based on the principle "user as an expert" is presented. SDM and the steps in the process of its formation have been formally described.

The method of the formation of an adaptive user interface, providing increased cognition in multi-subject information system by automated formation of adequate for models of user interests interfaces. On the basis of the FMM and the constraints of the problem of synthesis a procedure for forming the structure of navigation interface is proposed.

The search method provides an automated query expansion and evaluation of the relevance of search results through a sharing of analysis of the formal mental models and semantic domain model with the subtractive relations. The method of presentation of the document, the method of the query presentation and the function of correspondence between them, constituting the corresponding steps in the procedure of search, which provides an automatic generation of restrictions based on SDM and adaptive ranking of search results based on the FMM are examined.

The results of the proposed technologies testing are presented in conclusion.

О.В. ТАРАКАНОВ
**СРАВНИТЕЛЬНОЕ ИССЛЕДОВАНИЕ СПОСОБОВ ХРАНЕНИЯ
БОЛЬШИХ ДВОИЧНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ В БАЗАХ
ДАННЫХ ORACLE**

Тараканов О.В. Сравнительное исследование способов хранения больших двоичных последовательностей в базах данных ORACLE.

Аннотация. Работа выполнена с целью построения математического описания условий пригодности основных способов хранения больших двоичных последовательностей (BLOB и BFILE) в базах данных на платформе ORACLE. Исследование выполнено путем восстановления уравнений множественной линейной регрессии, учитывающих значимые на наш взгляд условия применения заявленных типов данных. Сформулированы некоторые практические рекомендации для обоснования выбора типа данных для хранения больших двоичных последовательностей.

Ключевые слова: база данных, большая двоичная последовательность, большой двоичный объект, BLOB, BFILE, уравнение множественной линейной регрессии.

Tarakanov O.V. A Comparative Research into Ways to Store Binary Large Objects in the ORACLE Databases.

Abstract. The research purpose was to describe the conditions of formation of mathematical suitability of main ways to store binary large sequences (BLOB or BFILE) in ORACLE databases. The solution is obtained by restoring multiple linear regression equations taking into account significant conditions of use of these datatypes as BLOB and BFILE. Some practical recommendations are formulated to justify the selection of the datatype to store binary large sequences.

Keywords: database; binary large sequence; binary large object; BLOB; BFILE; multiple linear regression equation.

1. Введение. Ведение двоичных данных в фактографических базах данных сопряжено с рядом трудностей и противоречий: необходимо обеспечить их связь с соответствующими идентификаторами, добиться выровненности строк, соблюдения эргономичности или хотя бы одинаковость обращения с ними пользователя. При этом не снизить эффективность системы баз данных в целом. Противоречивость требований к ведению двоичных последовательностей в базах данных обуславливает актуальность сравнительного исследования основных способов хранения двоичных последовательностей.

Современный уровень развития технологий баз данных предлагает два основных способа хранения больших двоичных последовательностей – внутренний и внешний. В первом случае последовательность помещается в ячейку реляционной таблицы с соответствующим типом данных. Второй способ предусматривает хранение в таблице базы данных ссылку (обычно объемом в 25 кБайт) на внешний файл, содержащий двоичную последовательность. Внутреннее хранение

двоичной последовательности обеспечивает защищенность объекта на уровне безопасности собственно базы данных, сильную связанность с соответствующими идентификаторами, следовательно, достаточную целостность, согласованность и непротиворечивость (обеспечивается базовыми механизмами реляционной модели данных) [1, 7, 8, 10].

Внешнее хранение обеспечивает сильную связь с сервером приложений, породившим последовательность, не нарушает выровненность строк таблицы, но не реализует связь с индексами, не гарантирует уровень защищенности не ниже, чем в базе данных [1, 6-8, 10]. Наивысшая опасность для надежности хранения обусловлена допустимостью размещения двоичной последовательности вне системной директории (каталога) системы баз данных.

Вариативность способов хранения двоичных последовательностей не обеспечивает однозначность в выборе наилучшего варианта. Свойства каждого из них являются достоинствами или недостатками только в определенных условиях применения. На основании данных положений можно считать задачу исследования взаимосвязи свойств способов хранения двоичных последовательностей в системах баз данных достаточно актуальной. Исследование проведено для систем баз данных на платформе ORACLE.

2. Постановка задачи. Имеется тестовый набор двоичных последовательностей объемом 14.2, 923 и 0.957 мБайт, предназначенных для хранения в базе данных ORACLE. Последовательности представляют собой аудио, видео записи и оцифрованную фотографию (статистические межсимвольные зависимости и преобладания символов одного вида не устранялись). Для их хранения имеется база данных, содержащая два односхемных фрагмента, отличающихся только типом данных атрибута, предназначенного для хранения больших двоичных последовательностей. В первом случае применен тип BLOB, а во втором – BFILE. Инфологическая модель базы данных предусматривает некоторую семантическую окраску, предназначенную для получения связанных таблиц (выполнение запросов с вложениями – рекурсивного типа).

Предполагается, что основными показателями эффективности использования больших двоичных последовательностей являются время их загрузки, выгрузки и удаления (t_3 , t_B , t_U соответственно), а также затраты дискового пространства для размещения на сервере – V . Это обусловлено значимостью соответствующих свойств – быстродействия и ресурсоемкости, компонента информационной системы (результативность не рассматривается) [2, 9]. Ожидается, что влияние на значения показателей будут оказывать: характеристики аппаратно-программной платформы сервера баз данных – x_1 ; размер большой

двоичной последовательности – x_2 ; тип запроса в базу данных по манипулированию последовательностью – x_3 . Другие факторы, влияющие на изменение значимых свойств исследуемой системы, отнесены в ограничения вследствие роста сложности исследовательской задачи.

Для обеспечения применимости линейного регрессионного анализа необходимо доказать, что значения показателей t_3 , t_B , t_Y и V распределены нормально [3]. Доказательство может быть построено в соответствии с логикой центральной предельной теоремы, когда изыскивается совокупность случайных составляющих исследуемого параметра и оценивается их вклад в текущее значение. Для данного случая имеется достаточная статистика, позволяющая получить доверительные оценки параметров распределения случайных величин t_3 , t_B , t_Y и V , на основании которых можно утверждать, что они распределены нормально при неизменных условиях реализации (постоянных значениях x_1, x_2, x_3).

Таким образом, задача состоит в определении вида зависимости значений временных и ресурсных показателей вариантов хранения двоичных последовательностей в базах данных ORACLE от характеристик (типа) аппаратно-программной платформы сервера, собственно размера двоичной последовательности и типа запроса в базу данных на манипулирование соответствующим объектом хранения двоичной последовательности. Формально необходимо восстановить уравнение множественной линейной регрессии вида:

$$y = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + b_3 \cdot x_3 + \varepsilon. \quad (1)$$

Здесь y обозначает одну из заявленных характеристик основных способов хранения двоичных последовательностей (среднее время выполнения запроса на выгрузку, загрузку или удаление двоичной последовательности); b_1 – коэффициенты линейной регрессии; b_0 – постоянная составляющая (необъяснимая) регрессии; ε – случайная погрешность (возмущение).

Влияние возмущения обычно оценить не возможно, поэтому перейдем к оценкам истинных значений параметров y_j , и в уравнении регрессии (1) не будем его учитывать. Тогда задачей исследования будет установление вида функции регрессии на двух выборках случайных величин показателей для двух вариантов хранения двоичных последовательностей, а именно:

$$\widehat{y}_1 = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + b_3 \cdot x_3, \quad (2)$$

$$\widehat{y}_2 = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + b_3 \cdot x_3. \quad (3)$$

Ограничения, при которых решается задача, сформулированы выше, а именно, случайное возмущение считается не соизмеримо малым – $\varepsilon = 0$; иные кроме x_1, x_2, x_3 факторы считаются незначимыми, а их влияние учтено постоянной составляющей b_0 .

3. Проведение эксперимента и восстановление уравнения регрессии. Исследование вида зависимости эффективности хранения больших двоичных последовательностей от совокупности разнородных факторов (в том числе – способа хранения последовательности) предполагает получение состоятельной статистики на правдоподобных операциях. Для получения необходимых исходных данных выполнен эксперимент с использованием системы баз данных ORACLE версии 11g R2. Целевая установка эксперимента состоит в определении степени зависимости времени выполнения запроса (вставка – загрузка, выборка – выгрузка, удаление) и затрачиваемого дискового ресурса от варианта организации хранения двоичной последовательности, производительности аппаратно-программной платформы сервера баз данных.

Время модификации последовательности не исследовалось ввиду не востребованности данной информационной операции. Действительно, при хранении последовательности во внешнем файле модификация данных осуществляется приложением-сервером, с помощью которого она была создана. В случае внутреннего хранения (домен BLOB) модификация возможна только после выгрузки последовательности во внешнюю прикладную программу с последующей загрузкой результата в файл данных. Данные положения позволяют обосновать отсутствие актуальности исследования времени модификации данных непосредственно в базе данных (типовая операция UPDATE).

Исходные данные получены в результате дробнофакторного эксперимента со значимостью $\alpha = 0.05$ и $\beta = 0.02$. Вероятность ошибки второго рода вычислена по семейству оперативных кривых [4], описывающих изменение мощности эксперимента при фиксированном числе степеней свободы случайного отклика и переменном числе реплик. Принадлежность экспериментальных данных генеральной совокупности определена на основании критерия Диксона [5] для текущих условий экспериментирования.

Экспериментальный стенд представляет собой совокупность трех независимых друг от друга серверов баз данных с различной аппаратно-программной платформой. Конкретные показатели платформы инвариантны результатам эксперимента [9, 11]. Основное условие – существенное различие вариантов платформ по производительности на операциях ввода/вывода и специфических вычислениях при выполнении запросов в базу данных на языке SQL (интерпретация текста

запроса, планирование процедуры выполнения, обмен с памятью). В эксперименте использованы имеющиеся у исследователей платформы. Вариант А: двухпроцессорная архитектура AMD Phenom™ II X4 925 2,81 ГГц /2048 мБайт/160 гБайт; вариант Б: 2x Intel(R) Core(TM) i5 CPU 750 2,67 ГГц /8192 мБайт/1 тБайт; вариант В: 2xIntel Xeon E5-2680V2/3x8192 мБайт/2x1 тБайт SATA RAID0.

Во всех вариантах применена локальная вычислительная сеть 100 мб/с Ethernet без внешней нагрузки (задержки в сети минимальны и одинаковы для всех вариантов аппаратно-программной платформы). Клиентское программное обеспечение ORACLE развернуто на персональном компьютере на базе архитектуры Intel Core i5 2390T/16384 мБайт/750 мБайт, одним и тем же для всех трех серверных платформ (реплики выполнялись последовательно).

Логическая структура базы данных для способов хранения отличалась только типом данных атрибута, предназначенного для ведения двоичной последовательности. В первом случае применен тип BLOB, во втором – BFILE. Мощность контрольного примера одинакова – 9 кортежей. Логическая структура предусматривает четыре связанных между собой отношения, все связи идентифицирующие. Степень нормализации до нормальной формы Бойса-Кодда. Для измерения времени отклика базы данных использован встроенный хронометр утилиты ORACLE SQLDeveloper.

В обеспечение применимости методов регрессионного анализа выполнено исследование закона распределения случайной величины отклика. Сформулировано предположение о нормальном распределении и выполнена проверка принадлежности случайной величины (времени выполнения запроса) нормально распределенной генеральной совокупности. Для всех уровней фактора (вид запроса в комбинации с вариантом аппаратно-программной платформы и размером двоичной последовательности) получены вероятности того, что время выполнения запроса отклоняется от своего математического ожидания на величину, меньшую, чем утроенное среднеквадратичное отклонение. Они принадлежат диапазону от 0.70628 до 0.98826. В соответствии с данными результатами сформулирован вывод о состоятельности гипотезы о смещенном нормальном распределении времени отклика базы данных на запрос по всем уровням фактора. Для построения уравнения регрессии выполнена спецификация – обоснование набора независимых переменных (факторов влияния), существенно влияющих на время выполнения запроса. По сути, доказана применимость сформулированного при постановке задачи набора факторов, влияющих на

время выполнения запроса по загрузке, выгрузке или удалению большого двоичного объекта.

Факторы влияния в исследуемой области качественные. Для построения математической модели применен способ оцифровки качественных параметров, варьирующихся в трех уровнях. Использовано серийно-порядковое кодирование. Данный метод предполагает формирование всех комбинаций уровней фактора, например, три варианта применяемой аппаратно-программной платформы сервера баз данных, с последующей нумерацией по порядку внутри серий – вариативных условий экспериментирования. Сформированный код играет роль индекса при соответствующей независимой переменной x_{ij} . Результаты кодировки факторов представлены в таблице. Здесь значение "1" указывает на факт использования данного параметра в опыте.

Таблица. Кодировка переменных уравнения регрессии

Наименование уровня фактора	Возможные значения	Значение фиктивной переменной		Математическое обозначение в уравнении
Тип аппаратно-программной платформы	Вариант А	0	0	x_{11}
	Вариант Б	1	0	x_{12}
	Вариант В	0	1	x_{13}
Размер хранимого объекта	14,2 мБайт	0	0	x_{22}
	923 мБайт	1	0	x_{21}
	0,957 мБайт	0	1	x_{23}
Тип запроса	Удаление	0	0	x_{31}
	Выгрузка	1	0	x_{32}
	Загрузка	0	1	x_{33}
Время выполнения запроса при способе хранения	BLOB			y_1
	BFILE			y_2

Для определения коэффициента множественной корреляции измеренные в ходе опытов величины времени выполнения запросов (получены с помощью утилиты SQLDeveloper) введены в лист данных пакета прикладных программ "Statistika 6.0". С помощью данного программного средства выполнена статистическая обработка. Результаты для способа хранения BLOB получены на 270 наблюдениях. Точечная диаграмма распределения времени манипулирования тремя большими двоичными объектами представлена на рисунке 1. Здесь же приведена линейная аппроксимация роста задержки выполнения запроса в зависимости от объема обрабатываемого большого двоичного объекта.

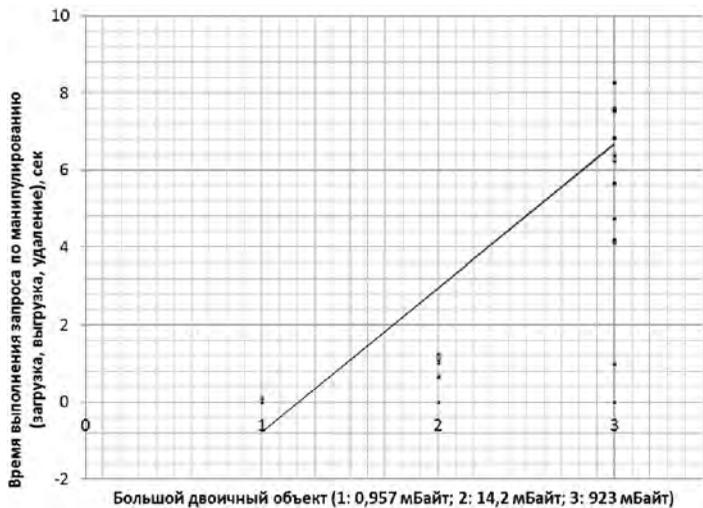


Рис. 1. Время выполнения операции манипулирования для трех разных по объему больших двоичных объектов

На графике рисунке 2 представлены результаты измерения времени выполнения основных операций по загрузке, выгрузке и удалению по основным способам хранения большой двоичной последовательности.

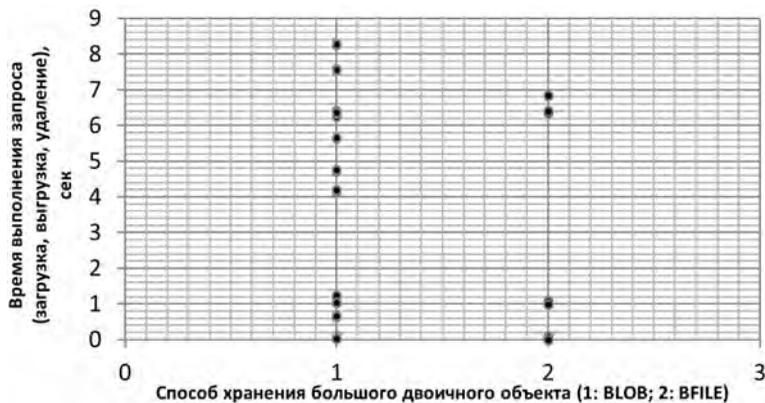


Рис. 2. Время манипулирования данными большого двоичного объекта для двух вариантов организации хранения

Отклики времени манипулирования большими двоичными объектами на различных аппаратно-программных платформах сервера баз данных приведены на рисунке 3.

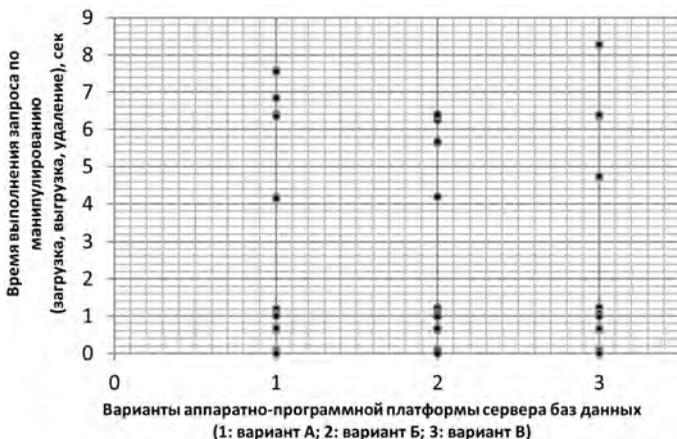


Рис. 3. Время выполнения единичных операций манипулирования большими двоичными последовательностями на различных аппаратно-программных платформах

Коэффициент множественной корреляции для способа хранения BLOB составляет 0.972, что позволяет считать зависимость между факторами и откликом как сильную. Для способа BFILE статистика рассчитана также на основании 270 измерений и показывает сильную множественную корреляцию между фактором и откликом: $R = 0.714$. Методом наименьших квадратов восстановлены уравнения множественной линейной регрессии для основных способов хранения:

$$y_1 = 55.0203 + 0.2382 \cdot x_{12} + 0.4793 \cdot x_{13} - 61.8185 \cdot x_{22} - 63.0109 \cdot x_{23} + 8.566 \cdot x_{32} + 12.1851 \cdot x_{33}, \quad (4)$$

$$y_2 = 15.0655 + 0.0009 \cdot x_{12} - 0.0013 \cdot x_{13} - 21.659 \cdot x_{22} - 22.7687 \cdot x_{23} + 23.4377 \cdot x_{32} - 0.0254 \cdot x_{33}. \quad (5)$$

Детальное рассмотрение уравнений (4), (5) и результатов статистической обработки данных эксперимента позволяет сформулировать ряд утверждений. Очевидно, что производительность аппаратно-программной платформы оказывает прямое влияние на время выполнения запроса. При этом существенно более производительная платформа по варианту В не дает подавляющего преимущества в сравнении с показателями платформы по варианту А. Преимущество внешнего способа хранения (BFILE) по времени загрузки, выгрузки и удаления объекта определяется отсутствием дополнительной ступени преобразования двоичной последовательности при взаимодействии с файлом данных и словарем данных. Не зависимо от размера двоичной последовательности этот выигрыш приблизительно втроекратный в среднем. Иллюстрация приведена на рисунке 4.

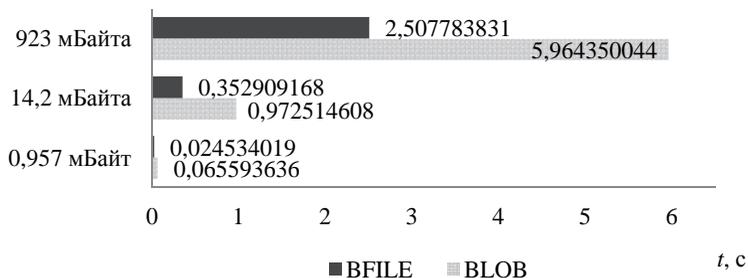


Рис. 4. Сравнительная диаграмма отличия временных задержек при манипулировании двоичными последовательностями разных объемов

Еще одним очевидным результатом является тот факт, что среднее время манипулирования двоичной последовательностью прямо зависит от ее размера. При этом для последовательностей размером до 1 мБайта преимущества способа хранения не наблюдается – среднее время манипулирования отличается в рамках статистической погрешности измерений. С увеличением размера двоичной последовательности наблюдается расхождение производительности системы хранения по манипулированию данными: как для размера порядка 15 мБайт, так и для последовательностей размера около 1 гБайта – задержки для способа внешнего хранения меньше в пять – восемь раз, чем для способа внутреннего хранения. Сравнительная диаграмма представлена на рисунке 5 (здесь показано изменение времени загрузки последовательности для разных вариантов аппаратно-программной платформы). Задача восстановления вида зависимости времени манипулирования двоичной последовательностью от ее размера при разных способах хранения может быть исследована отдельно.

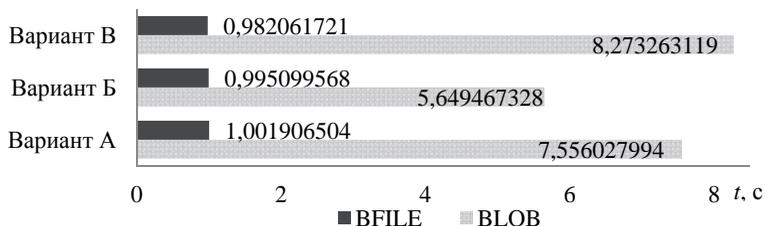


Рис. 5. Сравнительная диаграмма зависимости времени манипулирования данными при разных объемах и способах хранения большой двоичной последовательности

Эффект от применения различных способов хранения больших двоичных последовательностей иллюстрируется диаграммой на рисунке 6. Здесь представлены результаты статистического исследования времени, затрачиваемого на загрузку, выгрузку и удаление одинаковых по объему двоичных последовательностей. Получены ожидаемые результаты при учете алгоритмов обработки последовательностей в базе данных. Практическое значение сведений на рисунке 6 состоит в том, что становится очевидной инвариантность выгрузки двоичной последовательности и сильная зависимость времени загрузки от способа хранения. Данный результат является основанием для выбора способа хранения для систем с преобладающей операцией загрузки, например, видео контента.

4. Заключение. Экспериментальные исследования в целом подтверждают противоречивую природу основных способов хранения больших двоичных последовательностей в базах данных на примере систем на основе решений ORACLE. Нельзя утверждать, что какой-либо основной способ хранения подобных объектов лучше. При выборе способа хранения больших двоичных последовательностей основным фактором является предполагаемый режим манипулирования ими (загрузка, выгрузка или удаление). Для систем, накапливающих данные (автоматизированные системы контроля, автоматизированные системы научных исследований, системы автоматизированного проектирования и им подобные), предпочтительнее использовать внешний способ хранения больших двоичных последовательностей (тип данных BFILE). В системах управления и регулирования, где высока доля операций по выгрузке больших двоичных последовательностей, предпочтительнее применение внутреннего способа хранения (тип данных BLOB).

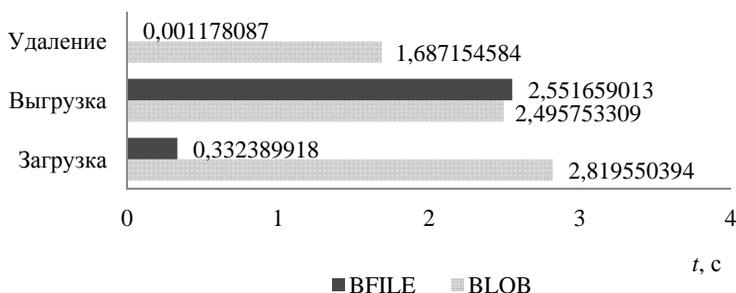


Рис. 6. Сравнительная оценка времени манипулирования двоичной последовательностью при использовании разных способов хранения

Данные утверждения обоснованы для случаев хранения объектов размером в десятки мБайт и более. Двоичные объекты не значительных размеров (фотографии, аудио записи в форматах сжатия с потерями и им подобные) инвариантны к применяемым способам хранения с точки зрения временных задержек при манипулировании. В данном случае обоснование выбора способа хранения целесообразно основывать на функциональных показателях и свойствах, обеспечивающих защищенность.

Исследование подтвердило физическую природу механизмов и алгоритмов обработки и хранения больших двоичных последовательностей в базах данных, основанных на решениях ORACLE. Статистическая независимость исследования позволяет воспроизвести его для других популярных платформ. При этом следует ожидать результатов, объясняемых заложенными в программное обеспечение серверов баз данных программно-техническими решениями. Воспроизведение представленного исследования может быть полезно при оценивании возможностей по хранению больших двоичных последовательностей в так называемых «проприетарных» системах с неизвестными или неясными механизмами манипулирования данными.

Литература

1. *Гарсиа-Молина Г., Ульман Дж., Уидом Дж.* Системы баз данных. Полный курс: пер. с англ. // М.: Издательский дом "Вильямс". 2003. 1088 с.
2. *Анфилатов В.С., и др.* Системный анализ в управлении: Учеб. пособие. / Под ред. А.А. Емельянова. // М.: Финансы и статистика. 2002. 368 с.
3. *Бронштейн И.Н., Семендяев К.А.* Справочник по математике для инженеров и учащихся втузов. – 13-е изд., исправленное. // М.: Наука, Гл. ред. физ.-мат. лит. 1986. 544 с.
4. *Монтгомери Д.К.* Планирование эксперимента и анализ данных: пер. с англ. // Л.: Судостроение. 1980. 384 с.
5. *Блохин В.Г., Глудкин О.П., Гуров А.И., Ханин М.А.* Современный эксперимент: подготовка, проведение, анализ результатов / Под ред. О.П. Глудкина // М.: Радио и связь. 1997. 232 с.
6. *Алапани С.Р.* Oracle Database 11g: руководство администратора баз данных: пер. с англ. // М.: ООО "И.Д. Вильямс". 2010. 1440 с.
7. *Kaim T.* Oracle для профессионалов: пер. с англ. // Спб.: ООО "ДиаСофтЮП". 2003. 672 с.
8. *Панченко Б.Е.* Каркасное проектирование доменно-ключевой схемы реляционной базы данных // Кибернетика и системный анализ. 2012. № 3. С. 174–187.
9. *Стукалов В.А., Савотченко С.Е.* Моделирование процесса загрузки данных в автоматизированные библиотечные информационные системы с открытым кодом // Фундаментальные исследования. 2013. № 8–1. С. 69–72.
10. *Григорьев Ю.А.* Алгоритм синтеза частично оптимальной схемы реляционной базы данных // Наука и образование: электронное научно-техническое издание. 2012. № 1. С. 31–35.

11. *Сорокин А.А., Королев С.П., Смагин С.И., Поляков А.Н.* Макет отказоустойчивой информационной системы для облачного хранения наборов научных данных // Вычислительные технологии. 2013. Том 18. № 1. С. 87–95.

References

1. Garsia-Molina G., Ul'man Dzh., Uidom Dzh. *Sistemy baz dannyh. Polnyj kurs*. [The database system. Full course]; Per. s angl. M.: Izdatel'skij dom "Vil'jams", 2003. 1088 p. (In Russ.).
2. Anfilatov V.S., et al. *Sistemnyj analiz v upravlenii* [System analysis in management]: Ucheb. posobie. Ed. A.A. Emel'janova. M.: Finansy i statistika [Finance and Statistics], 2002. 368 p. (In Russ.).
3. Bronshtejn I.N., Semendjaev K.A. *Spravochnik po matematike dlja inzhenerov i uchashhihsja vtuzov* [Handbook of mathematics for engineers and students of technical colleges]. M.: Nauka, Gl. red. fiz.-mat. lit., 1986. 544 p. (In Russ.).
4. Montgomeri D.K. *Planirovanie jeksperimenta i analiz dannyh*. [Planning the experiment and analysis of data]; Per. s angl. L.: Sudostroenie, 1980. 384 p. (In Russ.).
5. Blohin V.G., Gludkin O.P., Gurov A.I., Hanin M.A. *Sovremennyj jeksperiment: podgotovka, provedenie, analiz rezul'tatov* [Modern experiment preparation, conduct, analysis of the results]. Ed. O.P. Gludkina. M.: Radio i svjaz', 1997. 232 p. (In Russ.).
6. Alapati S.R. *Oracle Database 11g: rukovodstvo administratora baz dannyh* [Oracle Database 11g: Administrator's Guide database]; Per. s angl. M.: OOO "I.D. Vil'jams", 2010. 1440 p. (In Russ.).
7. Kajt T. *Oracle dlja professionalov* [Oracle professionals.]. Spb.: OOO "DiaSoftJuP", 2003. 672 p. (In Russ.).
8. Panchenko B.E. [Framework design of a domain-key scheme of a relational database]. *Kibernetika i sistemnyj analiz – Cybernetics and system analysis*. 2012. no. 3. pp. 174–187. (In Russ.).
9. Stukalov V.A., Savotchenko S.E. [Simulation of loading data into the automated library information systems with open source]. *Fundamental'nye issledovanija – Fundamental research*. 2013. no. 8-1. pp. 69–72. (In Russ.).
10. Grigor'ev Ju.A. [The algorithm of synthesis partially optimal relational database schema]. *Nauka i obrazovanie: jelektronnoe nauchno-tehnicheskoe izdanie – Sciences and Educational: electronic science-technical issue*. 2012. no. 1. pp. 31–35. (In Russ.).
11. Sorokin A.A., Korolev S.P., Smagin S.I., Poljakov A.N. [Layout fault-tolerant information system for the cloud storage of scientific data sets]. *Vychislitel'nye tehnologii – Computational technologies*. 2013. vol. 18. no. 1. pp. 87–95. (In Russ.).

Тараканов Олег Викторович — к-т техн. наук, доцент, сотрудник, Академия ФСО России. Область научных интересов: Базы данных, структуры данных, информационные технологии, обработка данных. Число научных публикаций — 100. ole_g66@list.ru; 302040, г. Орел, ул. 8 Марта, д. 66, кв. 222; р.т.: +79208019202.

Tarakanov Oleg — Ph.D., assistant professor, collaborator, Academy of the Federal Security Service of the Russian Federation. Research interests: Database, data model, information technology, data processing. The number of publications — 100. ole_g66@list.ru; 8 Marta, 66, fl. 222, Oryol, 302040; office phone: +79208019202.

РЕФЕРАТ

Тараканов О.В. Сравнительное исследование способов хранения больших двоичных последовательностей в базах данных ORACLE.

В статье рассматриваются вопросы сравнения основных способов хранения больших двоичных последовательностей в базах данных ORACLE по их отдельным прагматическим свойствам. Исследованы типы данных BLOB и BFILE. Статья содержит отчет о результатах эксперимента, проведенного над некоторой базой данных, представленной двумя версиями. В одной применен тип данных BLOB, а в другой – BFILE. В результате статистической обработки полученных экспериментальных данных восстановлены уравнения линейной множественной регрессии. Они являются математическими описаниями корреляции между временем выполнения запроса и факторов, обусловленных типом аппаратно-программной платформы сервера баз данных, размером двоичной последовательности и собственно типом выполняемого запроса (выгрузка, загрузка, удаление). Восстановленные уравнения линейной множественной регрессии интерпретированы для применения по обоснованию выбора способа хранения больших двоичных последовательностей в базах данных ORACLE.

SUMMARY

Tarakanov O.V. A Comparative Research into Ways to Store Binary Large Objects in the ORACLE Databases.

The article deals with the comparison of the individual pragmatic properties of the main ways to store binary large sequences in databases ORACLE. The research into data types BLOB and BFILE is conducted. The article contains a report on the results of an experiment conducted on some database, presented in two versions. One type of data applies BLOB, and the other - BFILE. As a result of statistical processing of the experimental data, linear multiple regression equations are restored. They are mathematical descriptions of the correlation between the query execution time and factors arising from the type of hardware and software platform of the database server, the size of the binary sequence and the actual type of the query executed (uploading, modification, deletion). Recovered linear multiple regression equations are interpreted for use in justifying the choice of method of storing large binary sequences in the ORACLE databases.

О.В. КОФНОВ

МОДЕЛЬ И АЛГОРИТМЫ ИЗМЕРЕНИЯ ГЕОМЕТРИЧЕСКИХ ПАРАМЕТРОВ СТРУКТУР ТЕКСТИЛЬНЫХ МАТЕРИАЛОВ

Кофнов О.В. Модель и алгоритмы измерения геометрических параметров структур текстильных материалов.

Аннотация. Рассмотрены методы бесконтактного определения величин угловых и линейных геометрических параметров в структурах текстильных материалов. Разработаны алгоритм моделирования изображений дифракционных картин с использованием быстрого преобразования Фурье, алгоритм измерения угла кручения нити по цифровой фотографии её структуры, алгоритм измерения перекоса нитей в ткани и алгоритм измерения расстояния между соседними элементами структуры с использованием метода двойного преобразования Фурье.

Ключевые слова: алгоритм, текстиль, структура материала, модель дифракционной картины, преобразование Фурье, крутка нити, угол кручения, перекося, ткань.

Kofnov O.V. A Model and Algorithms for Measurement of Geometric Parameters in Textile Structures.

Abstract. Non-contact methods measuring angular and linear geometric parameters in textile structures are considered. The diffraction pattern modeling algorithm based on the fast Fourier transform, the algorithm counting the yarn twist angle, the algorithm counting the skewness of the weft thread in fabric and the Double Fourier transform algorithm of determining the period of a material structure are developed.

Keywords: algorithm, textile, structure, diffraction pattern modeling, Fourier transform, yarn twist, twist angle, skewness, fabric.

1. Введение. Одной из важных особенностей текстильных материалов является то, что они представляют собой упорядоченное переплетение объектов малого размера (волокон и нитей) [1]. В частности, в крученой нити витки волокон закручиваются в определённом направлении, а ткань представляет собой переплетение нитей в строгом порядке, называемом раппортом. Таким образом, можно сказать, что структура большинства текстильных материалов обладает периодичностью микроскопических элементов, из которых она состоит [2]. К геометрическим параметрам такой микроструктуры можно отнести: диаметр текстильной нити; величину крутки нити (угол кручения); плотность ткани по основе и утку; петельный шаг и высоту петельного ряда трикотажного изделия; поверхностную плотность ткани/трикотажа; размер сквозных пор. От этих параметров в значительной степени зависят физические и технологические характеристики текстильных изделий: прочность, поверхностная плотность, воздухо- и водопроницаемость, износостойкость, а также внешний вид и потребительские свойства. Поэтому актуальной является задача измерения величин этих параметров как на стадии

контроля качества готового изделия, так и на более ранних стадиях производства.

ГОСТ 8846-87, 3812-72, 6611.3-2003 и ISO 2061:2010-12 определяют набор методов, применяемых в настоящее время для контроля геометрических параметров структуры текстильных материалов. Однако все эти методы либо требуют разрушения исследуемого образца, либо применения ручного труда (непосредственный подсчет элементов структуры лаборантом с использованием лупы или микроскопа). Таким образом, актуальной является задача разработки бесконтактных способов измерения и контроля количества и размеров повторяющихся элементов структур текстильных материалов.

К настоящему времени были разработаны методы и устройства оптического контроля качества производимых текстильных изделий, например, [3]. Следует отметить, что подобные приборы имеют высокую стоимость и сложны в эксплуатации. В [2] обосновываются преимущества одного из подклассов оптических методов контроля: дифракционного метода. Он заключается в получении дифракционной картины при освещении исследуемого образца текстильного материала лучом монохромного света. В частности, на практике подобный метод был реализован в [4]. Установка состоит из источника когерентного света, освещающего исследуемую нить, системы линз и зеркал, а также оптических датчиков, регистрирующих интенсивность света в результирующей дифракционной картине, имеющей вытянутую форму с ярко выраженными большой и малой осями. Угол кручения нити определяется как угол между большой осью, перпендикулярной продольной направляющей витков, и прямой, перпендикулярной оси нити.

Несомненным преимуществом дифракционного метода является то, что световые волны, отраженные от периодических элементов структуры, размеры которых сопоставимы с длиной волны используемого когерентного света, взаимодействуют между собой, что даёт в результирующей дифракционной картине упорядоченные ряды максимумов интенсивности. По взаимному расположению этих дифракционных максимумов можно определить истинные размеры и форму элементов исследуемой микроструктуры. К недостаткам следует отнести необходимость освещения образца источником когерентного света (лазер) и необходимость проецировать дифракционную картину на удалённый экран, что повышает громоздкость установки. Собственно, подобные недостатки присущи

всем аппаратным методам, физически моделирующим требуемый процесс.

Современное развитие вычислительной техники позволяет во многих случаях перейти к математическому моделированию физических процессов. Образец материала в таком случае можно заменить его компьютерным изображением (цифровой фотографией, сделанной с требуемым увеличением). Разработаны методы анализа подобных изображений тканей с помощью фильтров, основанных на быстром преобразовании Фурье [5], на преобразовании Хафа [6] и на сопоставлении изображений, сделанных с разных ракурсов [7]. В данной работе предлагается непосредственное математическое моделирование самого процесса дифракции. Это позволит применять уже апробированные аппаратные дифракционные методы [2] без необходимости физического воспроизведения дифракционной картины с помощью лазера и использовать предлагаемый способ анализа изображений не только для тканей, но и для всех видов текстильных материалов с периодической микроструктурой.

2. Постановка задачи. Задача бесконтактного оптического контроля геометрических параметров материалов с периодической структурой формулируется следующим образом.

Дано: материал со структурой, состоящей из повторяющихся элементов (витки, петли, волокна, нити в переплетении), размеры и расстояния между которыми сопоставимы с длинами волн света в оптическом диапазоне. Уже существуют апробированные методы получения дифракционных картин при освещении таких материалов лучом когерентного света и последующего определения по распределению максимумов интенсивности в таких картинах геометрических параметров поверхностных структур таких материалов. Однако эти методы требуют использования соответствующих оптических установок, включающих источник когерентного света (лазер), удаленный экран для проецирования дифракционной картины, светочувствительные датчики, фиксирующие расположение дифракционных максимумов.

Требуется: разработать алгоритмы измерения величин геометрических параметров структур текстильных материалов, использующие математическую модель процесса дифракции, построенную с помощью компьютерной программы. Эта же программа осуществляет анализ смоделированной дифракционной картины с целью определения зависимостей в расположении дифракционных максимумов и расчет на их основе истинных величин геометрических параметров исходной микроструктуры. Входными

данными являются цифровая фотография структуры материала, сделанная с необходимым увеличением, и массив допустимых диапазонов контролируемых величин, определённых на основе требований к качеству и технологическому процессу производства материала. Выходные данные: выводимое пользователю на экран компьютера изображение дифракционной картины, значения автоматически рассчитанных величин геометрических параметров и отчет о соответствии этих величин заданным критериям качества, определённым введёнными контрольными диапазонами. Таким образом, конечный пользователь системы получает наглядную информацию о процессе контроля, структуре и качестве исследуемого материала.

Задача сводится к математическому моделированию физического процесса, положенного в основу уже известных способов контроля параметров текстильных материалов. Такое моделирование стало возможным благодаря появлению технологии цифровой фотографии, позволяющей получать изображение объекта в виде массива точек (пикселей), характеризующихся яркостью по каждому из цветовых каналов. Цифровое изображение образца представляет собой таблично заданную двумерную функцию распределения яркости, которая прямо пропорциональна интенсивности излучения единицы поверхности объекта. От момента наблюдения (съёмки объекта) и до получения результата точность контроля геометрических параметров структуры текстильных материалов зависит только от точности вычислений компьютера при математическом моделировании процесса контроля.

Рассматриваемая система, описывающая процесс контроля величин геометрических параметров структуры текстильных материалов, представляет собой целостный объект, состоящий из частей (являющихся в свою очередь подсистемами) и связей между ними [8]. На этапе проектирования технологический процесс контроля представлен схемой (рисунок 1).

Изображение, выступающее в качестве входных данных структуры контролируемого материала, предварительно преобразуется в массив данных, на основе которых осуществляется компьютерное моделирование изображения дифракционной картины. По взаимному расположению дифракционных максимумов рассчитываются искомые величины линейных и угловых геометрических параметров структуры материала, которые сравниваются с заданными в качестве входных данных контрольными значениями.

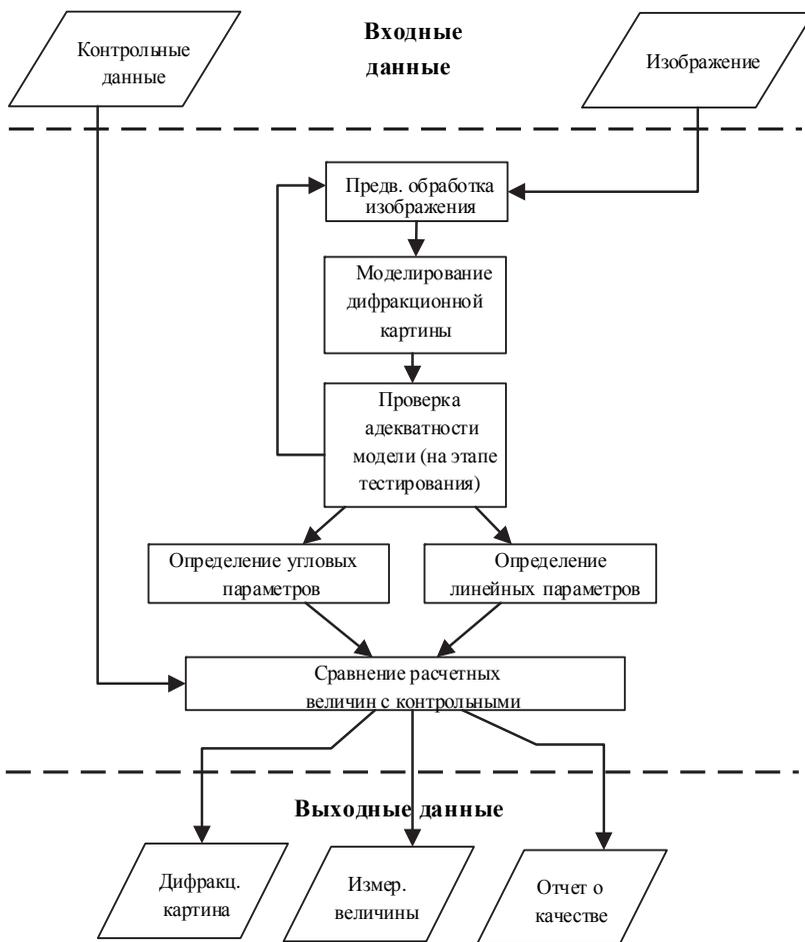


Рис. 1. Схема технологического процесса контроля геометрических параметров структуры материала

На этапе настройки системы (а такая настройка обязательно требуется, ибо результат всегда зависит от условий съемки, качественных характеристик материала и т.д.) предусмотрена отрицательная обратная связь между блоком проверки адекватности модели (который задействуется только на этапе тестирования) и блоком предварительной обработки изображения, необходимая для подбора таких параметров преобразования исходного изображения, при которых измерение искомым геометрических параметров структуры материала осуществляется с максимальной точностью. Использование отрицательной обратной связи

обеспечивает устойчивость процесса измерения, заключающуюся в минимальной погрешности при обработке изображений различных образцов материалов.

3. Математическое моделирование дифракции.

Компьютерное изображение исследуемого образца материала, полученное в результате цифровой фотосъемки, представляет собой функцию двух переменных:

$$I_0 = f(\xi, \eta), \quad (1)$$

где ξ и η – координаты точки в двумерной системе координат исходного изображения, I_0 – интенсивность светового сигнала, которая пропорциональна яркости точки. Здесь и в дальнейшем для модели процесса существенно только взаимное расположение точек на компьютерном изображении с максимальной по отношению к соседним точкам величиной интенсивности, поэтому абсолютная величина интенсивности светового сигнала в модели не учитывается.

Дифракцией света называют совокупность явлений, возникающих при огибании световой волной малых препятствий, сравнимых по размеру с длиной волны. Существуют аналитические решения задачи расчета дифракционной картины для образцов простых форм (одиночное отверстие, дифракционная решетка) [9] и численные решения для объектов сложных форм.

Известно, что дифракция Фраунгофера на микроструктурах позволяет реализовать подобную модель контроля. На удалённом экране после освещения периодического образца лучом монохроматического света отображаются ряды дифракционных максимумов. Таким образом, осуществляется такое преобразование функции интенсивности, которое эквивалентно построению модели дифракционной картины по информации об исходном изображении исследуемой структуры:

$$I = F(f(\xi, \eta), x, y), \quad (2)$$

где x и y – координаты точки модели в новой системе координат, связанной с дифракционной картиной; I – интенсивность сигнала в соответствующей точке модели дифракционной картины. Следует отметить, что масштаб систем координат (x, y) и (ξ, η) – разный.

В общем случае для моделирования дифракционной картины может быть использован интеграл Френеля-Кирхгофа [10]:

$$U(x, y) = C \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} U_0(\xi, \eta) \exp[-i \frac{2\pi}{\lambda L} (x\xi + y\eta)] d\xi d\eta, \quad (3)$$

где U – амплитуда светового сигнала в точке с координатами (x,y) дифракционной картины; U_0 – амплитуда светового сигнала в точке изображения с координатами (ξ,η) ; λ – длина световой волны, м; L – расстояние от объекта до экрана, м ($L \gg x,y$); i – мнимая единица; C – некоторая константа, зависящая от положения источника света и точки наблюдения. Координаты (x,y) и (ξ,η) , как и прочие величины расстояний в формуле, измеряются в метрах.

Вместо комплексной величины амплитуды сигнала при моделировании изображения дифракционной картины используется вещественная величина интенсивности сигнала I в каждой точке изображения:

$$I = UU^* = \text{Re}(U)^2 + \text{Im}(U)^2, \quad (4)$$

где $U = \text{Re}(U) + i\text{Im}(U)$ и $U^* = \text{Re}(U) - i\text{Im}(U)$ – комплексно сопряженные числа; $\text{Re}(U)$ – вещественная часть амплитуды U , $\text{Im}(U)$ – мнимая часть амплитуды U .

В качестве исходных данных для моделирования используется цифровое изображение структуры материала, представляющее собой массив точек с некоторой яркостью (заданной величиной от 0 до 255), пропорциональной амплитуде светового сигнала. Результатом моделирования является массив интенсивностей светового сигнала в каждой точке дифракционной картины.

Для двумерного массива амплитуд светового сигнала в каждой точке дискретного компьютерного изображения интеграл (3) можно заменить конечной суммой:

$$U_{k,l} = C\Delta^2 \sum_{n=-N/2}^{N/2-1} \exp\left[\frac{-2i\pi kn\Delta^2}{\lambda L}\right] \sum_{m=-M/2}^{M/2-1} u_{n,m} \exp\left[\frac{-2i\pi lm\Delta^2}{\lambda L}\right], \quad (5)$$

где M и N – ширина и высота компьютерного изображения соответственно; k и l , n и m – индексы пикселей соответствующего и исходного изображений по ширине и высоте; Δ – расстояние между пикселями, м.

Согласно [11] при компьютерном моделировании величины λ , L , N и Δ можно задать таким образом, что:

$$\frac{N\Delta^2}{\lambda L} = I, \quad (6)$$

при условии, что $N = M$, то есть изображение имеет форму квадрата.

Тогда выражение (5) можно переписать в виде:

$$U_{k,l} = C\Delta^2 \sum_{n=-N/2}^{N/2-1} \exp\left[-\frac{2i\pi kn}{N}\right] \sum_{m=-M/2}^{M/2-1} u_{n,m} \exp\left[-\frac{2i\pi lm}{M}\right], \quad (7)$$

сходном с формулой двумерного дискретного преобразования Фурье, где $u_{n,m}$ – относительная величина амплитуды сигнала, равная яркости пикселя в цифровой фотографии структуры образца материала. Выражение (6) также позволяет отказаться от использования размерностей и перейти к безразмерным относительным величинам.

В [11] показано, что если $M=N=2^t$, где t – целое положительное число, то для вычисления (7) может быть использован алгоритм, аналогичный алгоритму быстрого дискретного преобразования Фурье с оценкой быстродействия $O(N=M) = N^2 \log_2 N$ (вместо $O(N=M) = N^3$ при прямом суммировании (7)).

Перед построением модели дифракционной картины данные исходного цифрового изображения структуры материала предварительно обрабатываются. Изображение приводится к квадратному с одинаковым количеством пикселей по ширине и высоте, равным целой степени двойки (условие использования алгоритма быстрого преобразования Фурье) и преобразуется в монохромное, в котором яркость каждого пикселя определяется одной величиной.

Освещение структуры материала на цифровой фотографии лучом лазера имитируется следующим образом. Особенность луча лазера заключается в том, что интенсивность света в лазерном пятне круглого сечения убывает экспоненциально от центра пятна до его края и на границе пятна плавно обращается в ноль [2]. В [11] предложена следующая математическая модель освещения образца материала лазерным пятном (рисунок 2).

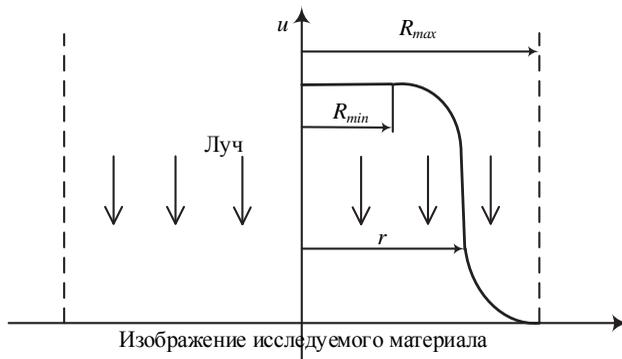


Рис. 2. Схема имитации светового пятна

Имитируемое пятно отличается от реального лазерного пятна тем, что характеризуется не одним, а двумя радиусами R_{\max} (радиус всего пятна) и R_{\min} (радиус внутренней части пятна, в пределах которого величина светового сигнала исходного изображения не меняется). Внутри кольца между R_{\min} и R_{\max} световой сигнал гасится до нуля по экспоненциальному закону.

Если обозначить относительную величину светового сигнала, пропорциональную яркости пикселя исходного изображения, как u_0 , то после наложения светового пятна она примет значение u в диапазоне $R_{\min}^2 \leq \xi^2 + \eta^2 \leq R_{\max}^2$:

$$u(\xi, \eta) = u_0(\xi, \eta) \exp\left[-\frac{\sigma(R_{\min} - \sqrt{\xi^2 + \eta^2})^2}{R_{\max}^2}\right], \quad (8)$$

где σ – коэффициент, рассчитанный из условия убывания u до нуля на участке $[R_{\min}; R_{\max}]$. Координаты (ξ, η) относительно центра изображения могут быть вычислены через индексы пикселя n и m .

Соответственно, для $\xi^2 + \eta^2 > R_{\max}^2$:

$$u(\xi, \eta) = 0, \quad (9)$$

и для $\xi^2 + \eta^2 < R_{\min}^2$:

$$u(\xi, \eta) = u_0(\xi, \eta). \quad (10)$$

Преобразование изображения структуры с помощью (8)-(10) преследует две цели. Во-первых, на изображении стирается резкая граница, образуемая краем изображения, что может приводить к искажению моделируемой дифракционной картины за счёт появления так называемых N -1 максимумов [12]. Во-вторых, при интегрировании по всей бесконечной поверхности световой волны амплитуда светового сигнала за пределами пятна всегда тождественно равна нулю.

Для измерения величин геометрических параметров структур текстильных материалов разработаны пять алгоритмов.

Алгоритм 1 предварительной обработки исходного цифрового изображения состоит из следующих шагов.

Шаг 1.1. Обрезать края изображения, превратив его в квадрат со стороной $N' = 2^t$, где t – максимальное целое число, удовлетворяющее условию $2^t \leq \min(M, N)$.

Шаг 1.2. Преобразовать изображение в монохромное.

Шаг 1.3. Смоделировать освещение центра изображения лучом лазера, рассчитав величину светового сигнала в каждой точке подготовленного изображения с использованием (8)-(10).

После предварительной обработки компьютерное изображение периодической структуры, представляющее собой двумерный массив комплексных чисел, каждое из которых соответствует амплитуде сигнала в соответствующей точке исходного изображения, преобразуется с помощью суммы (7). Для ускорения вычисления суммы используется алгоритм быстрого дискретного преобразования Фурье, модифицированный для рядов с отрицательными значениями индексов [11]. Результатом является численная модель дифракционной картины в виде массива комплексных чисел того же размера, что и массив исходного изображения, значение каждого элемента которого равно комплексной амплитуде светового сигнала в соответствующей точке дифракционной картины. Массив комплексных амплитуд может быть приведён к массиву интенсивностей сигналов с помощью выражения (4). Для последующего анализа расположения дифракционных максимумов может быть применён как непосредственно этот массив распределения интенсивности световых сигналов в построенной модели, так и его визуализация в виде цифрового изображения дифракционной картины, в которой яркость каждого пикселя прямо пропорциональна числовому значению элемента полученного массива интенсивностей сигналов.

Алгоритм 2 расчета дифракционной картины по цифровой фотографии структуры материала с использованием математической модели [11] может быть записан в виде следующих шагов обработки изображения.

Шаг 2.1. Подготовить компьютерное изображение структуры материала с помощью алгоритма 1.

Шаг 2.2. Скопировать величины яркостей световых сигналов в каждом пикселе подготовленного на шаге 2.1 изображения в двумерный массив комплексных чисел.

Шаг 2.3. Для двумерного массива, полученного на шаге 2.2, вычислить (7) с помощью быстрого преобразования Фурье [11].

Шаг 2.4. Для каждой комплексной величины из массива комплексных чисел, полученного на шаге 2.3, вычислить интенсивность I , используя (4), и результат записать в новый массив вещественных чисел. Полученный массив содержит интенсивности светового сигнала в каждой точке смоделированной дифракционной картины.

Шаг 2.5. Для визуализации результата моделирования построить новое цифровое изображение, где яркость каждого пикселя выражена целым числом от 0 до 255, пропорциональным значению соответствующего элемента массива интенсивностей, полученного на шаге 2.4.

4. Алгоритмы анализа дифракционных картин. С использованием предложенной математической модели [11] по алгоритму 2 были построены и проанализированы изображения дифракционных картин различных микроструктур. Анализ проводился с целью выявления закономерностей в расположении дифракционных максимумов для структур различной формы и последующего синтеза способов автоматического определения величин угловых и линейных геометрических параметров структур различных текстильных материалов по рассчитанному массиву интенсивностей световых сигналов.

Первоначально алгоритм моделирования дифракционных картин был апробирован на структурах простых форм, для которых существуют аналитические решения распределения интенсивностей в дифракционной картине, например прямоугольное отверстие (рисунок 3).

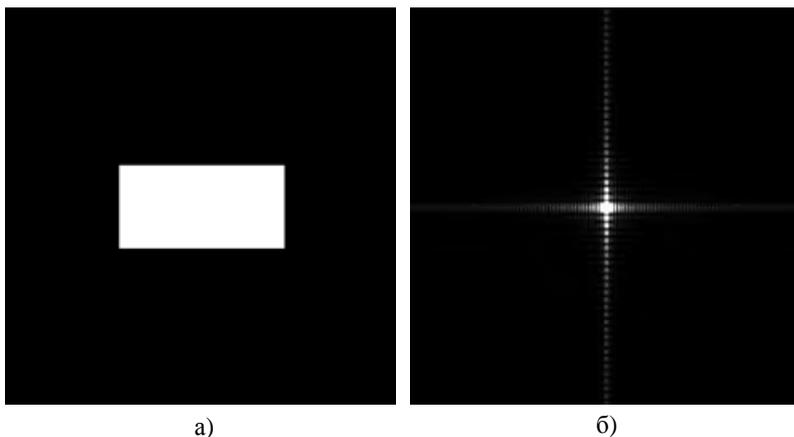


Рис. 3. Дифракция на прямоугольном отверстии: а) исходное изображение; б) смоделированная дифракционная картина

Существует аналитическое выражение для расчета интенсивности I для дифракционной картины от прямоугольного отверстия [9]:

$$I(x, y) = \frac{l_X^2 l_Y^2}{\lambda^2 L^2} \text{SINC}^2\left(\frac{l_X x}{\lambda L}\right) \text{SINC}^2\left(\frac{l_Y y}{\lambda L}\right), \quad (11)$$

где x и y – координаты точки на экране, отображающем дифракционную картину, м (точка начала системы отсчета располагается в центре изображения); l_X и l_Y – соответственно ширина и высота прямоугольного отверстия, м; λ – длина волны монохроматического света, освещающего отверстие, м; L – расстояние от отверстия до экрана, м; функция $\text{SINC}(\gamma) = (\sin\gamma)/\gamma$. Величина интенсивности I является безразмерной (приведённой к величине самой яркой точки дифракционной картины – центрального максимума). Численное моделирование с помощью (7) и (11) показало, что рассчитанные обоими способами величины интенсивности совпадают. Модель дифракции с использованием модели (7) также была апробирована при проведении численного расчета модели дифракции на синусоидальных решетках и сравнении результата с аналитическим решением, предложенным в [9]. Исследование дифракционных картин крученых нитей показало, что дифракционные максимумы располагаются в направлении, перпендикулярном продольной направляющей витков нити (рисунок 4).

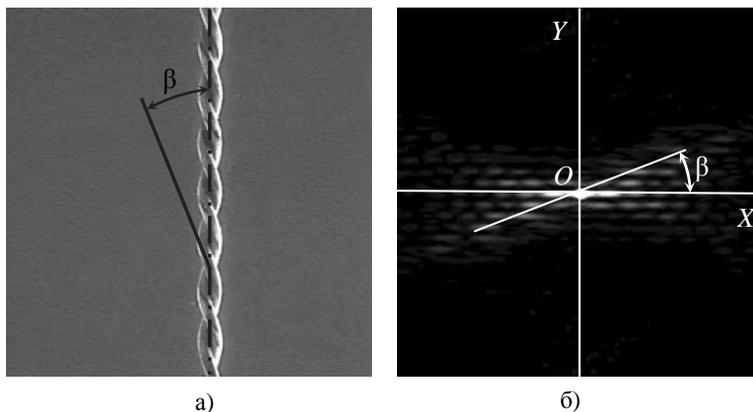


Рис. 4. Дифракция света на крученой нити: а) исходное изображение; б) дифракционная картина

Если ось нити на исходной фотографии расположена вертикально (рисунок 4а), то величина угла кручения β равна углу между горизонтальной осью координат и линией, проведённой из центра изображения через середины дифракционных

максимумов (рисунок 4б). В [13] для определения величины угла β предлагается использовать диаграмму распределения интенсивности дифракционных максимумов в полярных координатах (рисунок 5).

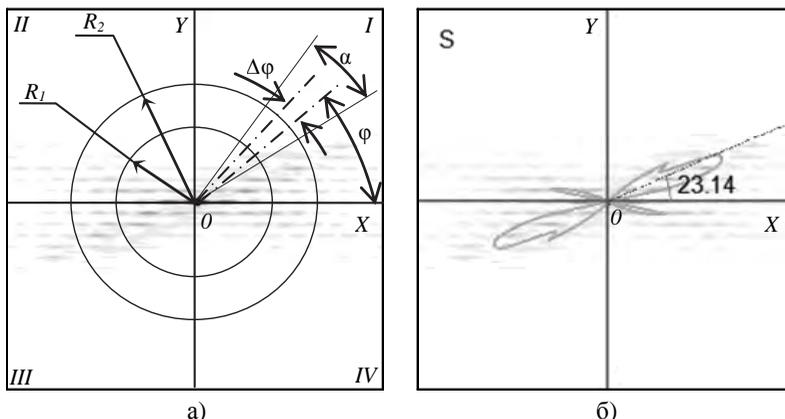


Рис. 5. Построение диаграммы распределения интенсивностей для дифракционной картины крученой нити

Алгоритм 3 построения диаграммы распределения интенсивности и последующего определения величины угла кручения нити с использованием способа, предложенного в [13], состоит из следующих шагов.

Шаг 3.1. Цифровая фотография нити ориентируется так, чтобы ось нити располагалась строго вертикально (рис. 4а).

Шаг 3.2. Моделируется дифракционная картина по алгоритму 2.

Шаг 3.3. На изображение рассчитанной дифракционной картины накладывается кольцо с внутренним радиусом R_1 и внешним R_2 . Внутри кольца на угловом расстоянии φ от оси OX строится сектор, ограниченный внутренней и наружной границами кольца и с угловым размером α . Определяется суммарная интенсивность дифракционных максимумов внутри сектора. Это значение откладывается на диаграмме интенсивности как величина радиус-вектора для угла φ .

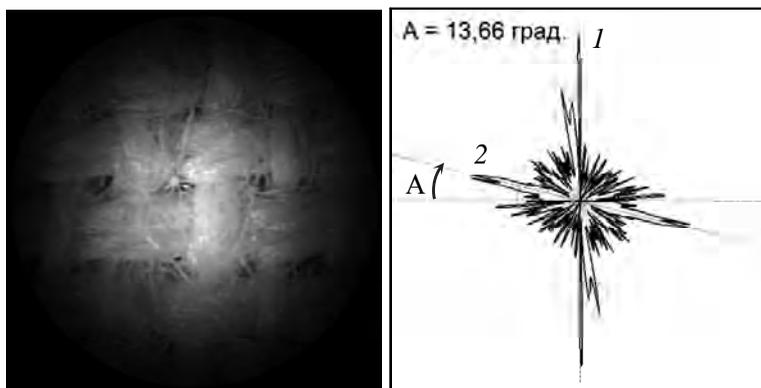
Шаг 3.4. Сектор, построенный на шаге 3.3, сдвигается на угол $\Delta\varphi$. Как и на шаге 3.3, рассчитывается суммарная интенсивность дифракционных максимумов внутри сектора после сдвига. Это значение откладывается на диаграмме как величина радиус-вектора для угла $(\varphi + \Delta\varphi)$.

Шаг 3.5. Повторяется шаг 3.4 для $0 \leq \varphi \leq 2\pi$. При этом конец радиус-вектора опишет кривую диаграммы.

Шаг 3.6. Построенная на шаге 3.5 диаграмма имеет две точки максимума (симметричные относительно начала координат). Через точку максимума, находящуюся в первых двух четвертях изображения, и центр изображения проводится прямая. Угол между этой прямой и осью OX равен искомому углу кручения.

С использованием алгоритма 3 выполнены измерения угла кручения нитей из полиэстера и смеси хлопка и вискозы, выпускаемых отечественной промышленностью.

Аналогичная диаграмма может быть использована для определения величины перекоса уточной нити в ткани (рисунок 6). В этом случае на построенной поверх смоделированной дифракционной картины диаграмме распределения интенсивностей в полярных координатах присутствуют два ярко выраженных максимума 1 и 2 (рисунок 6б) [14]. Эти пики диаграммы соответствуют расположению наборов дифракционных максимумов от уточных нитей и нитей основы. При строго перпендикулярном расположении нитей в переплетении угол между направлениями из центра изображения на эти два пика диаграммы равен 90° . В противном случае отклонение величины этого угла от 90° равно величине перекоса.



а) б)

Рис. 6. Определение перекоса в тканях

Алгоритм 4 определения величины перекоса между нитями утка и основы ткани по цифровому изображению согласно способу, описанному в [14], состоит из следующих шагов.

Шаг 4.1. Осуществляется расчет (моделирование) дифракционной картины по цифровой фотографии образца ткани, сделанной с двадцатикратным увеличением, по алгоритму 2.

Шаг 4.2. Для рассчитанной дифракционной картины строится диаграмма распределения интенсивности в полярных координатах (шаги 3.3–3.5 алгоритма 3).

Шаг 4.3. Осуществляется поиск максимума диаграммы (наиболее удалённой от начала отсчета точки I диаграммы на рисунок 6б).

Шаг 4.4. В диапазоне углов диаграммы $[\psi; \pi + \psi]$, где ψ – угловая координата максимума диаграммы I , осуществляется поиск второго по величине максимума 2.

Шаг 4.5. Угол между направлениями из начала отсчета на максимумы I и 2 определяется как разность между угловыми координатами этих максимумов на диаграмме. Разница между 90° и этим углом равна величине перекоса нитей в переплетении.

С использованием алгоритма 4 осуществлялся контроль перекоса уточных нитей в процессе сушки на ширильной машине плащевой ткани и полотна.

При освещении лучом монохроматического света объекта с периодической структурой расстояния между соседними дифракционными максимумами обратно пропорциональны расстояниям между элементами этой структуры. Например, для синусоидальной дифракционной решетки [9]:

$$D_X = f\lambda L, \quad (12)$$

где D_X – расстояние между соседними дифракционными максимумами, м; f – частота решетки, обратно пропорциональная расстоянию между соседними штрихами ($f=1/T$), m^{-1} ; λ – длина волны, м; L – расстояние от решетки до экрана, на котором наблюдается дифракционная картина, м.

В [11] расстояния между периодическими элементами структуры текстильного материала и расстояния между дифракционными максимумами в смоделированной дифракционной картине определяются выражениями:

$$T_X = \frac{K}{D_X}; T_Y = \frac{K}{D_Y}, \quad (13)$$

где T_X – расстояния между соседними элементами структуры вдоль горизонтальной оси OX (например, между соседними нитями утка в ткани), м; T_Y – расстояния между соседними элементами структуры вдоль вертикальной оси OY (например, между соседними нитями

основы в ткани), м; D_x и D_y – соответственно расстояния между соседними дифракционными максимумами в горизонтальном и вертикальном направлениях, м; K – коэффициент, учитывающий разрешение цифрового фотоаппарата, усиление микроскопа и расстояние от объектива до исследуемого материала [11], м².

В [15] предложено применить повторно алгоритм расчета дифракционной картины к дифракционной картине, смоделированной по исходному цифровому изображению структуры материала. Так как в обоих преобразованиях для вычисления суммы (7) применялся алгоритм быстрого дискретного двумерного преобразования Фурье, этот способ был назван авторами способом «двойного Фурье-преобразования».

Пусть T_x и T_y – расстояния между соседними периодическими элементами структуры на исходном цифровом изображении текстильного материала. Тогда с учетом (13) расстояния между дифракционными максимумами на смоделированной дифракционной картине:

$$T_{X1} = \frac{K}{T_x}; T_{Y1} = \frac{K}{T_y}, \quad (14)$$

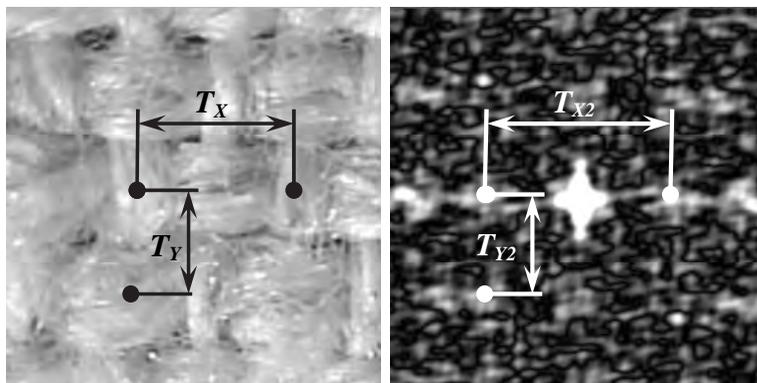
где T_{X1} – расстояния между соседними дифракционными максимумами вдоль горизонтальной оси Ox , м; T_{Y1} – расстояния между соседними дифракционными максимумами вдоль вертикальной оси Oy (например, между соседними нитями основы в ткани), м; K – некоторый коэффициент, м².

Результатом моделирования дифракционной картины является двумерный массив интенсивностей точек изображения. Повторное применение к нему алгоритма моделирования дифракционной картины в предположении, что коэффициент K не изменился, так как не менялись условия съемки, позволяет определить расстояния между дифракционными максимумами на второй дифракционной картине:

$$T_{X2} = \frac{K}{T_{X1}} = \frac{K}{K/T_x} = T_x; T_{Y2} = \frac{K}{T_{Y1}} = \frac{K}{K/T_y} = T_y, \quad (15)$$

где T_{X2} и T_{Y2} – расстояния между соседними дифракционными максимумами во второй модели соответственно вдоль горизонтальной и вертикальной осей координат, м.

Таким образом, после двух последовательных построений дифракционных картин с использованием алгоритма преобразования Фурье расстояния между соседними дифракционными максимумами в точности равны расстояниям между периодическими элементами на исходном изображении структуры материала (рисунки 7).



а б
Рис. 7. Метод «двойного» преобразования Фурье

Алгоритм 5 измерения расстояний между периодическими элементами структуры текстильного материала, разработанный на основании способа [15], описывается следующими шагами.

Шаг 5.1. Подготовить цифровую фотографию структуры материала по алгоритму 1.

Шаг 5.2. Выполнить расчет сумм (7) для каждого пикселя подготовленного изображения с использованием алгоритма быстрого дискретного преобразования Фурье (шаги 2.2-2.4 алгоритма 2 расчета дифракционной картины по цифровой фотографии структуры материала). В результате получено цифровое изображение дифракционной картины и двумерный массив интенсивностей, рассчитанных с помощью (4).

Шаг 5.3. Повторить шаги 2.3-2.4 алгоритма 2 для полученного на шаге 5.2 цифрового изображения дифракционной картины. В результате построено новое изображение дифракционной картины и двумерный массив интенсивностей точек в ней.

Шаг 5.4. Проанализировать расположение максимумов интенсивностей в полученной на шаге 5.3 модели дифракционной картины с помощью любого алгоритма поиска экстремумов функции, заданной таблично. По найденным координатам дифракционных максимумов определить средние расстояния между ними вдоль горизонтальной и вертикальной осей: T_{x2} и T_{y2} .

Шаг 5.5. Согласно (15) определённые на шаге 5.4 средние расстояния между дифракционными максимумами равны средним расстояниям между элементами периодической структуры материала на цифровой фотографии в пикселях. Далее полученные величины

умножаются на коэффициент разрешения цифрового фотоаппарата, использованного для съемки исследуемой структуры.

С помощью алгоритма 5 выполнено определение расстояния между петельными рядами и столбиками для чулочных изделий, а также определение расстояния между соседними нитями в ткани с переплетением «тартан».

5. Заключение. Дифракционная картина, получаемая на удалённом экране при освещении образца текстильного материала с периодической структурой лучом монохроматического когерентного света, может быть рассчитана с использованием интеграла Френеля-Кирхгофа. Линейные размеры периодических элементов структуры материала определяются по расстояниям между главными дифракционными максимумами с помощью любого алгоритма поиска экстремумов. Угловые размеры (углы кручения нити и перекоса в ткани) определяются с помощью диаграммы распределения интенсивностей дифракционных максимумов, построенной в полярных координатах.

Разработаны алгоритмы дифракционных способов контроля геометрических параметров структуры текстильных материалов, которые могут быть реализованы с помощью компьютерной программы, обрабатывающей исходное цифровое изображение материала, сделанное с достаточным увеличением. Предложенные алгоритмы отличаются простотой реализации и не требуют значительных вычислительных ресурсов. Для определения точности измерений с помощью указанных алгоритмов использовались нити, выпускаемые ОАО «Прядильно-ниточный комбинат им. С.М. Кирова» (г. Санкт-Петербург). Точность измерения угла составила $\pm 1^\circ$ при вычисленном уровне значимости статистики Стьюдента (t -критерия) $\alpha_{\text{выч.}} < 0,05$. Благодаря использованию высокопроизводительных алгоритмов предлагаемый комплекс может применяться для оперативного измерения геометрических параметров текстильных материалов в процессе их производства. Ввод в качестве исходных данных границ допустимых диапазонов измеряемых параметров и контроль вхождения измеряемых величин в эти диапазоны позволяет использовать данную систему для автоматического контроля качества производимого материала по геометрическим параметрам его структуры.

Литература

1. *Шустов Ю.С.* Основы текстильного материаловедения: учебное пособие // М.: МГТУ им. А.Н. Косыгина: Совьяж Бево. 2007. 302 с.

2. *Шляхтенко П.Г.* Оптические методы контроля параметров волоконсодержащих материалов. Контроль структуры текстильных материалов // Saarbrücken: LAP LAMBERT Academic Publishing GmbH & Co. KG. 2012. 347 с.
3. *Kretzschmar S.D., Furter R.* Uster© Tester 5-S800. Application Report // Uster: Uster Technologies AG. 2009. 40 p.
4. *Durand B.* Process And Device For Measuring The Twist Of A Textile Yarn // Patent US. no. 5210594. 1993.
5. *Calvimontes A., Badrul Hasan M.M., Dutschk V.* Effects of Topographic Structure on Wettability of Woven Fabrics // *Woven Fabric Engineering*. Rijeka: Sciyo. 2010. pp. 71–92.
6. *Pan R.* Automatic Inspection of Woven Fabric Density of Solid Colour Fabric Density by the Hough Transform // *Fibres & Textiles in Eastern Europe*. 2010. vol.18. no. 4(81). pp. 46–51.
7. *Лутсгартен Н.В., Сокова Г.Г., Сергеев А.С.* Бесконтактный способ анализа структуры ткани // Патент РФ. № 2131605. 1999. Бюл. № 16.
8. *Переудов Ф.И., Тарасенко Ф.П.* Введение в системный анализ // М.: Высшая школа. 1989. 360 с.
9. *Гудмен Дж.У.* Введение в Фурье-оптику // М.: Мир. 1970. 364 с.
10. *Борн М.* Основы оптики // М.: Наука. 1973. 720 с.
11. *Шляхтенко П.Г., Пименов В.И., Кофнов О.В.* Использование двумерного дискретного преобразования Фурье для компьютерного анализа материала с повторяющейся структурой // *Автоматизация и современные технологии*. 2013. №7. С. 20–27.
12. *Shlyakhtenko P.G., Shkuropat S.N., Nefedov V.P.* Computer analysis of models of optical images of plain weaves // *Journal of Optical Technology*. 2007. vol. 74. pp. 476–478.
13. *Шляхтенко П.Г., Кофнов О.В.* Способ определения угла крутки нити // Патент РФ. № 2534720. 2014. Бюл. № 30.
14. *Shlyakhtenko P.G., Kofnov O.V., Sukharev P.A.* Method of determining the skewness of the weft thread in fabric // *Journal of Optical Technology*. 2014. vol. 81. Issue 2. pp. 111–113.
15. *Шляхтенко П.Г., Кофнов О.В., Рудин А.Е.* Способ измерения геометрических параметров структуры текстильных материалов // Патент РФ. № 2508537. 2014. Бюл. № 6.

References

1. Shustov Y.S. *Osnovy tekstil'nogo materialovedeniya: uchebnoe posobie* [Basics of Textile Materials: Textbook]. Moscow: MGTU A.N. Kosygina: Sov'yazh Bevo. 2007. 302 p. (In Russ.).
2. Shlyakhtenko P.G. *Opticheskie metody kontrolya parametrov voloknosoderzhashchih materialov. Kontrol' struktury tekstil'nyh materialov* [Optical methods of controlling the parameters of fiber materials. Controlling of the structure of textile materials]. Saarbrücken: LAP LAMBERT Academic Publishing GmbH & Co. KG. 2012. 347 p. (In Russ.).
3. Kretzschmar S.D., Furter R. Uster© Tester 5-S800. Application Report. Uster: Uster Technologies AG. 2009. 40 p.
4. Durand B. Process And Device For Measuring The Twist Of A Textile Yarn. Patent US. no. 5210594. 1993.
5. Calvimontes A., Badrul Hasan M.M., Dutschk V. Effects of Topographic Structure on Wettability of Woven Fabrics. *Woven Fabric Engineering*. Rijeka: Sciyo. 2010. pp. 71–92.

6. Pan R. Automatic Inspection of Woven Fabric Density of Solid Colour Fabric Density by the Hough Transform. *Fibres & Textiles in Eastern Europe*. 2010. vol. 18. no. 4(81). pp. 46–51.
7. Lustgarten N.V., Sokova G.G., Sergeev A.S. *Beskontaktnyj sposob analiza struktury tkani* [The contactless method of analyzing the structure of the fabric]. Patent RF. no. 2131605. Bul. no. 16. (In Russ.).
8. Peregudov F.I., Tarasenko F.P. *Vvedenie v sistemnyj analiz* [Introduction to systems analysis]. M.: Vysshaya shkola. 1989. 360 p. (In Russ.).
9. Gudmen J.W. *Vvedenie v Fur'e-optiku* [Introduction to Fourier optics]. M.: Mir. 1970. 364 p. (In Russ.).
10. Born M. *Osnovy optiki* [Principles of optics]. M.: Nauka. 1973. 720 p. (In Russ.).
11. Shlyahthenko P. G. Pimenov V.I., Kofnov O.V. *Ispol'zovanie dvumernogo diskretnogo preobrazovaniya Fur'e dlya komp'yuternogo analiza materiala s povtoryayushchejsya strukturoj* [Use of two-dimensional discrete fourier transform for the computer analysis of a material with repeating structure]. *Avtomatizatsiya i sovremennye tekhnologii – Automation and modern technology*. 2013. vol. 7. pp. 20–27. (In Russ.).
12. Shlyakhtenko P.G., Shkuropat S.N., Nefedov V.P. Computer analysis of models of optical images of plain weaves. *Journal of Optical Technology*. 2007. vol. 74. pp. 476–478.
13. Shlyahthenko P.G., Kofnov O.V. *Sposob opredeleniya ugla krutki niti* [A method of determining the yarn twist angle] Patent RF. no. 2534720. 2014. Bul. no. 30. (In Russ.).
14. Shlyakhtenko P.G., Kofnov O.V., Sukharev P.A. Method of determining the skewness of the weft thread in fabric. *Journal of Optical Technology*. 2014. vol. 81. Issue 2. pp. 111–113.
15. Shlyahthenko P.G., Kofnov O.V., Rudin A.E. *Sposob izmereniya geometricheskikh parametrov struktury tekstil'nyh materialov* [A method of measuring geometric parameters of the structure of textile materials. Patent RF. no. 2508537. 2014. Bul. no. 6. (In Russ.).

Кофнов Олег Владимирович — соискатель ученой степени кандидата технических наук кафедры механической технологии текстильных материалов, Санкт-Петербургский государственный университет технологии и дизайна. Область научных интересов: технология производства текстильных материалов, машинное зрение, обработка астрономических каталогов, программирование на Java и C#. Число научных публикаций — 14. kofnov@mail.ru; Большая Морская ул., 18, Санкт-Петербург, 191186; р.т.: +7(812)9413288.

Kofnov Oleg Vladimirovich — Ph.D. student of fibrous materials mechanical technology department, St. Petersburg State University of Technology and Design. Research interests: textile technology, machine vision, astronomical catalogs processing, Java and C# programming. The number of publications — 14. kofnov@mail.ru; 18, Bolshaya Morskaya St., St. Petersburg 191186, Russia; office phone: +7(812)9413288.

РЕФЕРАТ

Кочнов О.В. **Модель и алгоритмы измерения геометрических параметров структур текстильных материалов.**

В настоящей статье рассматриваются моделирование дифракционного процесса контроля геометрических параметров структуры текстильных материалов и алгоритмы измерения величин этих параметров. Известны методы контроля, в которых дифракционная картина получается при освещении образца материала лучом лазера. Геометрические параметры структуры материала определяются по расположению дифракционных максимумов.

Предлагается вместо этого использовать математическое моделирование явления дифракции. В качестве исходных данных выступает цифровое изображение структуры текстильного материала. Расчет дифракционной картины для этого изображения осуществляется с использованием интеграла Френеля-Кирхгофа. Предложен алгоритм такого расчета, основанный на двумерном дискретном преобразовании Фурье. Результатом расчета является массив интенсивностей света в каждой точке дифракционной картины.

По расположению максимумов интенсивности в дифракционной картине, построенной для крученой нити, можно определить угол кручения нити. От величины угла кручения зависит прочность нити. Для определения угла кручения используется диаграмма распределения интенсивности в дифракционной картине, построенная в полярных координатах. Предложен алгоритм построения диаграммы и измерения угла кручения нити.

Аналогичная диаграмма может быть использована для определения угла перекося уточной нити в ткани. Такой перекося является дефектом ткани и связан с неисправностью оборудования. Предложен алгоритм определения величины перекося с использованием диаграммы распределения интенсивности в полярных координатах.

В дифракционной картине, смоделированной для изображения материала, расстояния между дифракционными максимумами обратно пропорциональны расстояниям между периодическими элементами структуры материала. Если выполнить то же преобразование для полученной дифракционной картины, то во второй дифракционной картине расстояния между дифракционными максимумами равны периодическим расстояниям на цифровом изображении структуры материала. Это упрощает контроль за этими величинами. Предложен алгоритм двух последовательных расчетов дифракционных картин с помощью двумерного дискретного преобразования Фурье и определения периодических расстояний между элементами структуры.

SUMMARY

Kofnov O.V. Model and Algorithms for Measurement of Geometric Parameters in Textile Structures.

The model of diffraction and the algorithms for measuring of geometric parameters in textile are considered. There are a few methods based on diffraction after illuminating the textile by a laser. Geometric parameters of that structure can be determined after measuring distances between maximums in the diffraction pattern.

Instead of using a real diffraction process the mathematical modeling of diffraction patterns are proposed. The source data about lights in the structure can be extracted from the digital image of textile. The diffraction pattern model can be constructed using Fresnel–Kirchhoff diffraction formula. The algorithm of diffraction pattern digital image calculation is proposed. This algorithm is based on the two-dimensional discrete Fourier transform. In result the two-dimension intensity array in every pixel of the diffraction image is produced.

The placement of diffraction maximums provides the measurement of the twist angle of yarn. It is extremely important because the yarn strength depends on the twist angle. This angle can be measured in the intensity diagram of a yarn diffraction pattern which is drawn in polar coordinates. The algorithm of this diagram construction and angle measurement is developed.

The same diagram can be used to determine the skewness of the weft thread in fabric. It is a defect of fabric. The algorithm of the skewness determination is proposed.

After the first transform the digital image of textile to the diffraction pattern image the distances between diffraction maximums are reciprocally proportional to distances in the periodical structure of the source object so additional calculations are required. After the second transform the distances between them are equal to distances between repeating elements in the source material. The offered algorithm combines two two-dimensional discrete fast Fourier transforms to calculate a diffraction pattern from a diffraction pattern image of material under consideration.

К.А. БАТЕНКОВ
**МОДЕЛИРОВАНИЕ И СИНТЕЗ ЛИНЕЙНЫХ ДИСКРЕТНЫХ
ОТОБРАЖЕНИЙ НЕПРЕРЫВНЫХ КАНАЛОВ СВЯЗИ**

Батенков К.А. Моделирование и синтез линейных дискретных отображений непрерывных каналов связи.

Аннотация. Формализованы дискретные отображения непрерывных каналов связи в виде операторов преобразования метрических конечномерных пространств. Показана их связь с представлением в виде интегральных преобразований на основе предельного перехода от представления в виде рядов. Показано, что операторы аналого-дискретных преобразований дискретных отображений непрерывных каналов связи на основе канонического разложения В. С. Пугачева, вычисляются путем итерационной процедуры, последовательно определяющей базисные функции на выходе канала в виде рекуррентного операторного преобразования входных базисных функций.

Ключевые слова: непрерывный канал связи, дискретное отображение непрерывных каналов связи, оператор, базис пространства, каноническое разложение В. С. Пугачева, системная характеристика, аддитивная помеха.

Batenkov K.A. **Modeling and Synthesis of Discrete Mappings of Linear Continuous Channel.**

Abstract. Linear continuous channel discrete mappings in the form of metric finite-dimensional space operators are formalized. Its coupling with representation in the form of integral transforms on the basis of limiting process from series view is shown. It is shown that semi-digital operators of linear continuous channel discrete mappings on the basis of Pugachev canonical presentation are calculated by force of iterative procedure consistently conditioning basis functions on input and output channel in terms of recurrence input transformation.

Keywords: continuous channel, discrete mappings of linear continuous channel, operator, space basis, Pugachev canonical expansion, system characteristic, additive noise.

1. Введение. При синтезе систем связи одним из этапов является формирование канала дискретного времени, предполагающее исследование конечномерных пространств и преобразований их в бесконечномерные и обратно, что является следствием не только необходимости практической реализации систем передачи, но и требований к моделированию с позиций теории вероятности на основе вероятностных мер [1]. Подобные преобразования в научной литературе определяются термином «дискретные отображения непрерывных каналов связи», акцентирующем внимание на том, что осуществляется процедура перехода от канала с непрерывными по времени и состоянию входом и выходом к каналу с непрерывными по состоянию и дискретным по времени входами и выходами [2].

Следует подчеркнуть, что данная задача тесно взаимосвязана с наиболее острой проблемой, существующей с момента возникновения связи как средства коммуникации и по настоящее время, – это постоян-

ный рост требований к скорости и качеству передачи информации со стороны пользователей. При этом актуальность данной проблемы ощущается как в области беспроводных, так и проводных систем связи [3, 4].

Естественно, что наилучшим решением подобной задачи является совместная оптимизация дискретно-аналогового и аналого-дискретного преобразователей в целом [5]. Однако в данном случае существуют серьезные затруднения, связанные, прежде всего, со сложностью описываемых математическими моделями каналов связи, и как следствие невозможностью получения аналитических решений в явном виде [6].

В то же время для ряда простейших моделей, например канала связи с аддитивным белым гауссовским шумом, аналитический вид оптимальных дискретных отображений формализован [7]. Однако используемый критерий качества (отсутствие шумов ортогональности) приводит к тому, что система связи оказывается настроенной не на максимизацию достоверности либо скорости передачи информации, а, по сути, на поиск решения, которое позволяет получить в явном виде функцию правдоподобия результирующего дискретного канала связи и, как следствие, обеспечить простоту дальнейших математических выкладок при проектировании других функциональных устройств системы (кодера и декодера). Использование других критериев (минимального среднего риска в общем случае) наталкивается на серьезные трудности вследствие сложности аналитического описания как функции правдоподобия самого непрерывного канала связи, так и трудностью математических преобразований общей формы среднего риска [8].

Одним из способов решения возникающих затруднений оказывается использование линейных как дискретно-аналогового, так и аналого-дискретного преобразователей. В результате появляется возможность описания дискретных отображений в аналитическом виде для целого набора критериев качества. Причем для ряда модулей непрерывных каналов связи, например с аддитивным гауссовским шумом, оказываются справедливы теоремы об отсутствии информационных потерь при подобной линейной трансформации [9]. Однако более сложные модели каналов не обладают схожими свойствами с точки зрения простоты математических выкладок, и как следствие не позволяют даже для линейных дискретно-аналогового и аналого-дискретного преобразователей получать приемлемые отображения.

В результате для более точного согласования линейных дискретно-аналогового и аналого-дискретного преобразователей со свойствами канала связи вводят дополнительные устройства, например

банки фильтров, позволяющие несколько видоизменить передаточную функцию как модулятора, так и демодулятора [10, 11]. Однако даже в подобной ситуации зачастую оказывается не под силу воспроизвести максимально достижимые показатели качества дискретного канала связи.

Во многих задачах теории связи так же пользуются приемом, когда один из блоков (дискретно-аналоговый или аналого-дискретный преобразователь) считают заданным и оптимизация производится либо только для другого блока, либо же осуществляют последовательную оптимизацию: сначала при фиксированном аналого-дискретном преобразователе оптимизируют дискретно-аналоговый преобразователь, затем при полученном дискретно-аналоговом – аналого-дискретный преобразователь, а далее процедуру циклически повторяют. В результате получают дискретные отображения, являющиеся, по сути, компромиссом между достигаемым техническим эффектом и требуемой вычислительной сложностью [12, 13]. Однако данный подход оказывается допустимым так же лишь для ряда достаточно простых моделей каналов связи, а для более точных моделей трудоемкость аналитических выкладок сопоставима с общим подходом совместной оптимизации дискретно-аналогового и аналого-дискретного преобразователей.

Наиболее широко используемым методом синтеза дискретных отображений непрерывных каналов связи в настоящее время является метод, который предполагает оптимизацию лишь аналого-дискретного преобразователя при фиксированности дискретно-аналогового [14, 15]. При этом дискретно-аналоговый преобразователь пытаются как можно лучше согласовать со свойствами непрерывного канала либо на основе интуитивного подхода, используя, например, гармонические сигналы, либо проводят предварительную оптимизацию для более простых, но близких по природе моделей каналов связи. В этой связи возникает две основные проблемы. Первая – это аналитическая сложность в общем виде получения структуры аналого-дискретного преобразователя, а вторая – необходимость дополнительной подстройки дискретно-аналогового преобразователя под свойства используемой модели канала и меняющего аналого-дискретного преобразователя.

Первую трудность преодолевают различными путями – применяют различного рода квазиоптимальные алгоритмы аналого-дискретного преобразования (итерационные, линейные и другие), зачастую существенно проигрывающие оптимальным схемам обработки, но значительно менее требовательные с точки зрения вычислений [16, 17].

Для разрешения второй проблемы разработаны достаточно мощные методы, по сути, представляющие собой отдельное направление в теории кодирования, нашедшие наиболее широкое применение как в проводной, так и беспроводной связи при использовании систем с несколькими входами и выходами (с векторными каналами). Однако в настоящее время найдено лишь ограниченное число кодов, позволяющих существенно повысить технический эффект системы связи, причем модели каналов, для которых они разработаны так же весьма просты и зачастую не совсем полно отражают физику реального канала связи.

Таким образом, современные дискретно-аналоговые и аналого-дискретные преобразования предполагают их разбиение на последовательность преобразований, каждое из которых переводит лишь элементы определенного выделенного подпространства сигналов (векторов или многомерных функций), а не всего пространства. При этом зачастую в качестве правил выделения подпространств выступают заданные ограничения параметров сигналов, например пространственные, частотные или временные аргументы, что приводит к сужению классов применяемых преобразований. Кроме того, данные правила в большинстве случаев не учитывают целостность пространств сигналов и, как следствие, взаимосвязанность подпространств, а следовательно вносят дополнительные информационные потери. В результате, дискретное отображение непрерывного канала связи формируется на основе последовательности преобразований заданного класса, осуществляющих трансформацию только выделенной области пространства сигналов. Следовательно и само дискретное отображение в данном случае сводится к последовательности фиксированных классов отображений (не обязательно дискретных) операторов каналов связи с областями определения и значения, являющимися подпространствами исходных векторных и функциональных пространств сигналов на входах и выходах дискретно-аналогового и аналого-дискретного преобразователей.

В настоящей работе предлагается подход к синтезу линейных дискретных отображений непрерывных каналов связи в обобщенной форме, позволяющий рассматривать линейные дискретные отображения не только как совокупность последовательности преобразований, но и как единое целостное преобразование непрерывного канала связи с произвольными свойствами, описываемыми на уровне первых двух статистических моментов.

2. Дискретные отображения непрерывных каналов связи как операторы преобразования метрических пространств общего

вида. Дискретным аналогом интегральных преобразований, позволяющим анализировать динамические системы на ограниченном интервале анализа, а также исследовать ошибки аппроксимации, является бесконечный обобщенный ряд, по сути, сопоставляющий произвольный сигнал некоторому числовому вектору бесконечной размерности [18]:

$$x(t, \mathbf{r}) = \sum_{i=1}^N x_i \phi_i(t, \mathbf{r}), \quad (1)$$

$$z(t', \mathbf{r}') = \sum_{i=1}^{N'} z_i \phi_i'(t', \mathbf{r}'), \quad (2)$$

где x_i, z_i – входные и выходные коэффициенты разложения соответственно; $\phi_i(t, \mathbf{r}), \phi_i'(t', \mathbf{r}')$ – входные и выходные базисные функции разложения соответственно; $t, t', \mathbf{r}, \mathbf{r}'$ – временные и пространственные координаты на входе и выходе канала связи соответственно.

Как и в случае интегральных преобразований, выбор входных и выходных базисов достаточно произволен и может как быть продиктован условиями решаемой задачи, так и являться ее решением.

В целом представление в виде интегральных преобразований является следствием того или иного предельного перехода от представления в виде рядов [19]. Так, устремление смещения во времени и пространстве дискретного базиса многомерных дельта-функций, представляющих сигналы в виде последовательности отсчетов, к нулю приводит к непрерывному базису многомерных дельта-функций, интерпретирующих сигналы в непрерывном виде, то есть:

$$\int_{\Omega, \Omega'} x(t', \mathbf{r}') \delta(t - t', \mathbf{r} - \mathbf{r}') dt' d\mathbf{r}' = \lim_{\Delta t, \Delta \mathbf{r} \rightarrow 0} \Delta t \Delta \mathbf{r} \sum_{i, j=-\infty}^{\infty} x(i\Delta t, j\Delta \mathbf{r}) \delta(t - i\Delta t, \mathbf{r} - j\Delta \mathbf{r}).$$

Подобные переходы возможно также осуществить и для ряда других базисов, например базиса гармонических функций, экспоненциальных функций с гармоническим заполнением, функций Френеля и т. п.

По аналогии с интегральными преобразованиями [20, 21] и входным базисным функциям можно взаимнооднозначно сопоставить выходные, в виде реакций канала связи на первые из них:

$$\phi_i'(t', \mathbf{r}') = H\{\phi_i(t, \mathbf{r})\}. \quad (3)$$

Тогда выходной сигнал представим в виде ряда, в котором базисом выступает реакция (3), а коэффициентами разложения – входные коэффициенты разложения сигнала, то есть $z_i = x_i$:

$$z(t', \mathbf{r}') = \sum_{i=1}^N x_i H' \{ \varphi_i(t, \mathbf{r}) \}. \quad (4)$$

Выбор базисов, как и в случае интегральных разложений, произволен, поэтому не обязательно входные и выходные базисные функции должны быть идентичны (ситуация смешанных базисов). Следовательно в общем случае разложения в ряд, в отличие от интегрального представления, оператор преобразования в канале связи имеет матричный вид [19, 20]:

$$\mathbf{z} = \mathbf{K} \cdot \mathbf{x}, \quad (5)$$

где $\mathbf{z} = (z_0, z_1, \dots, z_\infty)^T$, $\mathbf{x} = (x_0, x_1, \dots, x_\infty)^T$ – бесконечномерные вектора коэффициентов разложения на выходе и входе соответственно;

а элементы матрицы преобразования в канале связи $\mathbf{K} = \{ K_{i,j} \}$, $i, j = \overline{0, \infty}$, по сути являющейся системной характеристикой канала, задаются выражениями:

$$K_{i,j} = \int_{\Omega} \int_{\Omega'} \varphi_i^{-1}(t', \mathbf{r}') H' \{ \varphi_j(t, \mathbf{r}) \} dt' d\mathbf{r}' \quad (6)$$

и определяются на основе сопряженных выходных базисных функций $\varphi_i^{-1}(t, \mathbf{r})$, удовлетворяющих условию ортогональности:

$$\int_{\Omega} \int_{\Omega'} \varphi_i^{-1}(t, \mathbf{r}) \varphi_j^{-1}(t, \mathbf{r}) dt d\mathbf{r} = \delta_{i,j},$$

где $\delta_{i,j}$ – символ Кронекера.

Таким образом, как и при интегральных разложениях, представление как сигналов, так и преобразований в канале связи может быть различно и задается в первую очередь базисами метрических пространств, в рамках которых рассматриваются исследуемые каналы связи. Однако сравнение выражений (4) и (5) позволяет сделать вывод о возможности представления сигнала на выходе линейного канала связи как в виде линейной комбинации системной характеристики канала и коэффициентов разложения входного сигнала, так и в виде дискретной свертки матричной системной характеристики и коэффициентов

разложения входного сигнала, вне зависимости от выбора формы базисов. В результате выходной сигнал можно рассматривать в виде дискретного разложения в базисе дискретной системной характеристики [21].

Таким образом, при использовании обобщенного разложения в ряд любой нестационарный канал связи, в том числе и стохастический, описывается в виде матричной системной характеристики (в общем случае бесконечномерной), в литературе [22, 23, 24] называемой спектральной, или проекционной. Данное обстоятельство устанавливает наличие алгебраической связи между коэффициентами разложения входных и выходных сигналов.

3. Дискретные отображения непрерывных каналов связи как операторы преобразования метрических пространств с заданными базисами. Как и в случае интегральных преобразований, в качестве базисных функций может использоваться произвольный ряд линейно независимых функций, причем не обязательно ортогональных [19]. Поэтому спектральные характеристики как системные, так и сигнальные трактуются как некоторые коэффициенты разложения их исходных непрерывных аналогов относительно выбранной системы линейно независимых функций, или базиса [23]. Таковым базисом в зависимости от исходных данных решаемой задачи может быть и система тригонометрических функций (Фурье), и система экспоненциальных функций с тригонометрическим заполнением (Лапласа), и система функций Уолша, а также ряд других функций, по сути являющихся дискретным аналогом ядер интегральных преобразований. Следует отметить, что дискретные во времени и пространстве сигналы в данном контексте могут рассматриваться в виде спектральных разложений в базисе дискретных дельта-функций.

Преимущественное использование при синтезе систем связи базисов Фурье и Лапласа:

$$\varphi_i(t, \mathbf{r}) = \left\{ e^{i\{i_0\omega_t + \omega_r^T \text{diag}(i_r)\mathbf{r}\}} \mid i = i_0 + \sum_{j=1}^n \max(i_{r_{j-1}}) \cdot i_{r_j} \right\}, \quad (7)$$

$$\varphi_i(t, \mathbf{r}) = \left\{ e^{i_{r_{n+1}}\sigma_t + \sigma_r^T \text{diag}(i_r)\mathbf{r} + i\{i_0\omega_t + \omega_r^T \text{diag}(i_r)\mathbf{r}\}} \mid i = i_0 + \sum_{j=1}^{2n+1} \max(i_{r_{j-1}}) \cdot i_{r_j} \right\}, \quad (8)$$

где $i_r = (i_{r_1}, \dots, i_{r_n})$, $i_{\mathbf{r}} = (i_{r_{n+2}}, \dots, i_{r_{2n+1}})$ – n -мерные вектора дискретных пространственных координат; ω_t , ω_r – временная частота и вектор про-

пространственных частот дискретизации соответственно; σ_t , σ_r – коэффициенты затухания по временной и пространственным координатам соответственно;

связано, как и в случае интегральных преобразований, прежде всего с тем, что они являются собственными функциями линейных стационарных систем, а значит сохраняют свою форму при прохождении через подобные системы [25]. В результате матричная системная характеристика канала связи \mathbf{K} приобретает диагональный вид, а следовательно алгебраическая форма преобразований канала (5) еще больше упрощается и каждый коэффициент разложения пространственно-временных сигналов может рассматриваться независимо от других, то есть:

$$z_i = K_{i,i} x_i, \quad i = \overline{0, N}. \quad (9)$$

Различия в условиях применения преобразований Лапласа и Фурье подобны рассмотренным в рамках интегральных преобразований. При этом при несоблюдении условий стационарности простейшее выражение вида (9) оказывается недопустимым поскольку нарушается ортогональность откликов линейной системы на гармоническое воздействие вследствие изменения их формы при ограниченных интервалах анализа как во временной, так и пространственной области. В итоге для обеспечения независимости спектральных составляющих пространственно-временных сигналов необходимо решать достаточно сложную задачу по поиску собственных функций и чисел линейной нестационарной системы, хорошо известную в математике и физике [26].

Развитие электронно-вычислительных устройств привело к широкому использованию кусочно-постоянных функций, основными среди которых являются функции Уолша и Хаара [25]. Основным преимуществом их использования по сравнению с функциями Фурье и Лапласа является постоянство первых из них на некотором, пусть и не слишком большом, интервале. В результате число проводимых операций умножения значительно сокращается, поскольку отпадает необходимость нахождения каждого произведения сигнала на базисные функции в каждой точке области анализа. Система функций Уолша может определяться различными способами – существует упорядочение по Уолшу, Пэли и Адамару [25, 27]. Наиболее простым с точки зрения определения и генерации является упорядочение по Пэли, согласно которому функции Уолша ранжируются по возрастанию их номеров в двоичном представлении.

Несмотря на значительное снижение требуемых операций умножения системы функций Уолша обладают существенным недостатком. Ее корреляционные свойства оказываются неудовлетворительными для ряда задач, связанных с разделением сигналов [27]. Это является следствием значительных отклонений функции распределения корреляционной функции данной системы на краях диапазона от нормальной плотности распределения, то есть вероятность появления сильно коррелированных функций Уолша существенно выше, чем у полного кода, обладающего предельными характеристиками независимости. При этом коэффициент эксцесса распределения корреляционной функции систем Уолша значительно превосходит единицу, что свидетельствует о возможности усовершенствования систем базисных функций, в частности путем разработки так называемых производных систем, составленных из функций, представляющих собой произведение более элементарных.

Функции Хаара имеют вид:

$$\varphi_i(t, \mathbf{r}) = \left\{ \prod_{j=0}^n h(r_j) \mid h(x) = \begin{cases} 2^{m_j/2}, & (i_{r_j} - 1)m_j^{-1} \leq x < (i_{r_j} - 1/2)m_j^{-1}, \\ -2^{m_j/2}, & (i_{r_j} - 1/2)m_j^{-1} \leq x < i_{r_j}m_j^{-1}, \\ 0, & x \notin [(i_{r_j} - 1)m_j^{-1}, i_{r_j}m_j^{-1}), \end{cases} \right. \quad (10)$$

$$i = i_{r_0} + \sum_{j=1}^n \max(i_{r_{j-1}}, i_{r_j}), \quad m_j = \log_2 \{ \max(i_{r_j}) \}, \quad r_0 = t \}.$$

В отличие от функций Уолша, принимающих в области анализа значения равные ± 1 , функции Хаара при условии соблюдения условия ортонормированности принимают одно из трех возможных значений, включающих нуль. Это обстоятельство позволяет несколько по иному учитывать поведение исходных пространственно-временных сигналов при вычислении коэффициентов разложения (спектрального состава), чем при расчете в базисе Уолша. Основное различие заключается в наборе точек, в которых рассматривается поведение сигналов. Так, разложение в ряд Уолша требует использования всех значений сигнала на интервале анализа, в то время как базис Хаара предполагает учет значений лишь в точках, достаточно близкорасположенных друг относительно друга. В результате число операций умножения может быть еще снижено по сравнению с применением разложения в ряд Уолш [25].

4. Дискретные отображения непрерывных каналов связи как операторы преобразования метрических пространств с базисом в виде координатных функций интегрального канонического

представления В. С. Пугачева. По аналогии с интегральным каноническим разложением в [28] введено понятие канонического разложения случайной функции (процесса), называемого также разложением В. С. Пугачева [29]. При этом основным отличием от ранее описанных базисных функций является их хотя и детерминированный, но не структурированный характер, поскольку форма функций вычисляется исходя из требуемых свойств коэффициентов разложения (спектрального состава). Таким образом, отправной точкой в данном разложении служит не форма базиса, а свойства результирующего (дискретного) случайного процесса [30].

Стохастический пространственно-временной сигнал представляется в виде линейной комбинации некоррелированных случайных величин, имеющих равные нулю математические ожидания, то есть коэффициентами данного разложения служит дискретный некоррелированный шум (в общем случае белый неоднородный как по пространству, так и по времени [31, 32]):

$$M\{\mathbf{x}\} = 0, \quad M\{\mathbf{x} \cdot \mathbf{x}^T\} = \mathbf{E} \cdot \mathbf{d}_x, \quad (11)$$

где \mathbf{E} – единичная (бесконечномерная) матрица; $\mathbf{d}_x = (d_{x_0}, d_{x_1}, \dots, d_{x_n})^T$ – вектор дисперсий дискретного нестационарного белого шума (коэффициентов разложения \mathbf{x}).

Вид разложения соответствует формулам (1) и (2), в которых выбор входных и выходных базисных функций разложения (координатных функций канонического разложения) осуществляется в соответствии с требованиями (11) и подразумевается центрированность исходного пространственно-временного сигнала. В случае невыполнения этого условия всегда можно свести нецентрированный процесс к центрированному путем вычитания из первого его математического ожидания. В результате каноническое разложение будет включать еще одно детерминированное слагаемое – функцию математического ожидания.

Как и в случае интегрального представления, нахождение в явном виде базисных функций разложения сопряжено с рядом часто непреодолимых трудностей [33]. Однако вследствие представления пространственно-временного сигнала в каждой точке области анализа в виде суммы случайных величин значительно упрощаются процедуры нахождения его характеристик [32]. Зависимость же от временных и пространственных координат целиком заключена именно в базисных функциях разложения, являющихся детерминированными функциями времени. В результате осуществление преобразований случайных про-

цессов и полей, в том числе и нелинейных, значительно упрощаются по сравнению с исходным описанием.

Необходимыми условиями осуществимости канонического представления являются коррелированность пространственно-временного сигнала и коэффициентов разложения, а также конечность их дисперсий, что выражается в виде функциональной зависимости для функций разложения [28]:

$$\varphi_i(t, \mathbf{r}) = \frac{M\{x_i \cdot x(t, \mathbf{r})\}}{d_{x_i}}. \quad (12)$$

Данное соотношение совместно с (1) или (2) позволяет формально определить стохастический пространственно-временной сигнал в виде суммы элементарных случайных составляющих, представляющих собой произведение некоторой детерминированной функции и случайного коэффициента. Однако практические способы выбора коэффициентов разложения и вычисления соответствующих базисных (координатных) функций оказываются достаточно произвольными. Кроме того, сходимость рядов (1) или (2) также задается в некотором смысле, а поэтому также должна быть жестко определена.

Так, в случае использования простейшей зависимости между коэффициентами разложения и исходным сигналом в виде скалярного произведения его самого на соответствующую базисную функцию:

$$x_i = \int_{\Omega} \int_{\Omega} x(t, \mathbf{r}) \varphi_i(t, \mathbf{r}) dt d\mathbf{r}. \quad (13)$$

Условие некоррелированности приводит к уравнению, которому должны удовлетворять базисные функции:

$$\int_{\Omega} \int_{\Omega} \varphi_i(t, \mathbf{r}) \int_{\Omega} \int_{\Omega} K_x(t, \mathbf{r}, t', \mathbf{r}') \varphi_j(t', \mathbf{r}') dt' d\mathbf{r}' dt d\mathbf{r} = d_{x_i} \delta_{i,j}, \quad (14)$$

где $K_x(t, \mathbf{r}, t', \mathbf{r}') = M\{x(t, \mathbf{r})x(t', \mathbf{r}')\}$ – корреляционная функция пространственно-временного сигнала.

Наложение условия ортонормированности базисных функций:

$$\int_{\Omega} \int_{\Omega} \varphi_i(t, \mathbf{r}) \varphi_j(t, \mathbf{r}) dt d\mathbf{r} = \delta_{i,j} \quad (15)$$

сводит задачу к поиску решений однородного интегрального уравнения Фредгольма второго рода [34]:

$$d_{x_i} \varphi_i(t, \mathbf{r}) = \int_{\Omega_x} \int_{\Omega_x} K_x(t, \mathbf{r}, t', \mathbf{r}') \varphi_i(t', \mathbf{r}') dt' d\mathbf{r}',$$

которое приводит к разложению Карунена-Лоэва [33, 35]. Таким образом, данное разложение делает возможным описание стохастического пространственно-временного сигнала на уровне его вторых моментов и его представление в виде некоррелированных случайных величин в бесконечномерном пространстве детерминированных базисных функций, определенных на некотором ограниченном интервале анализа. Данное обстоятельство одновременно является и достоинством, и недостатком подобного представления. С одной стороны оно позволяет достаточно просто проводить преобразования в линейных и нелинейных системах посредством матричных операций со случайными векторами коэффициентов разложения. С другой же стороны, за исключением гауссовских процессов, в общем случае не обеспечивает независимость коэффициентов разложения, поскольку в данном случае коррелированность не влечет за собой независимости [36]. К тому же следует еще раз подчеркнуть, что разложение Карунена-Лоэва является частным случаем канонического разложения и отличается от последнего наложением условия ортонормированности функций разложения (15) и заданием коэффициентов разложения в форме скалярного произведения исходного сигнала и базисных функций (13). Для самого же канонического разложения помимо условий (11) никаких дополнительных ограничений не накладывается.

В [37] показывается, что подобное представление является оптимальным как с точки зрения минимизации среднеквадратической ошибки аппроксимации пространственно-временного сигнала конечным числом членов ряда (1) или (2), так и с позиции минимизации энтропии дискретного набора дисперсий коэффициентов разложения при условии соблюдения нормировки сигнала для класса ортонормированных функций разложения [38].

Для более обширных классов функций, описываемых каноническим разложением, в [28] доказывается лишь оптимальность с позиции минимума среднеквадратической ошибки аппроксимации конечным числом членов ряда. Вопрос о минимизации энтропии остается открытым, однако разумно предположить, что подобное представление все же существует, поскольку каноническое разложение не требует задания коэффициентов разложения в виде скалярного произведения (13).

В общем случае для представления стохастического пространственно-временного сигнала в виде ряда элементарных случайных

функций при выполнении условия некоррелированности коэффициентов разложения (11) необходимо обобщить частный случай линейного преобразования в виде скалярного произведения (13) путем рассмотрения произвольных линейных функционалов:

$$x_i = \phi_i \{x(t, \mathbf{r})\}.$$

Другими частными случаями данных функционалов являются линейные комбинации значений стохастического сигнала в дискретном наборе точек:

$$\phi_i \{x(t, \mathbf{r})\} = \sum_{j, j'} a_{i, j, j'} x(t_j, \mathbf{r}_{j'}),$$

являющиеся частным случаем линейных комбинаций значений сигнала и его производных до заданного порядка в дискретном наборе точек:

$$\phi_i \{x(t, \mathbf{r})\} = \sum_{k_0, \dots, k_n, j, j'} a_{i, k_0, \dots, k_n, j, j'} \frac{\partial^{k_0} x(t, \mathbf{r})}{\partial t^{k_0} \prod_{k=1}^n \partial r_{k'}^{k_k}} \Bigg|_{t=t_j, \mathbf{r}=\mathbf{r}_{j'}},$$

являющиеся в свою очередь частным случаем более общего скалярного произведения:

$$\phi_i \{x(t, \mathbf{r})\} = \int_{\Omega_t} \int_{\Omega_{\mathbf{r}}} x(t, \mathbf{r}) \varphi_i(t, \mathbf{r}) dt d\mathbf{r}, \quad (16)$$

в котором произвольные функции $\phi_i(t, \mathbf{r})$ содержат линейные комбинации дельта-функций до соответствующего порядка. Кроме того, возможны еще более общие случаи линейных функционалов ϕ_i .

В результате условие некоррелированности коэффициентов разложения (11) приводит к уравнениям, которым должны удовлетворять линейные функционалы:

$$\varphi_{i, t, \mathbf{r}} \left[\varphi_{j, t', \mathbf{r}'} \{K_x(t, \mathbf{r}, t', \mathbf{r}')\} \right] = d_{x_i} \delta_{i, j}, \quad (17)$$

где нижние индексы t, \mathbf{r} в функционале ϕ означают применение этого функционала к функции по данным аргументам t, \mathbf{r} (остальные аргументы рассматриваются при фиксированных значениях).

Данное уравнение по своей форме сходно с условиями, которым должны соответствовать базисные функции в разложении Карунена-

Лоэва (14), однако в отличие от него уравнение (17) не содержит в явном виде выражения для базисных функций, что делает их форму более произвольной, поскольку не налагается дополнительное ограничение на их ортогональность. Вид же базисных функций определяется оператором Фредгольма путем постановки (16) в (12):

$$\phi_i(t, \mathbf{r}) = \frac{1}{d_{x_i}} \varphi_{i', \mathbf{r}'} \{K_x(t, \mathbf{r}, t', \mathbf{r}')\}. \quad (18)$$

Используя данную формулу на основании требований ортогональности линейных функционалов (17), в [28] выводится необходимое и достаточное условие некоррелированности коэффициентов разложения в виде условий биортогональности линейных функционалов базисных функций:

$$\varphi_i \{ \phi_j(t, \mathbf{r}) \} = \delta_{i,j}. \quad (19)$$

В результате возможна реализация последовательной процедуры нахождения требуемых линейных функционалов ϕ_i на основе произвольно выбранной последовательности линейных функционалов γ_i . Данная процедура такова [28]. Первоначально определяют линейный функционал ϕ_0 произвольным образом, то есть:

$$\phi_0 = \gamma_0. \quad (20)$$

После этого согласно (17) вычисляют дисперсии коэффициентов разложения (при условии $i = j$), а по формуле (18) определяют базисную функцию разложения $\phi_0(t, \mathbf{r})$. На ее основе вычисляют линейный функционал ϕ_1 в соответствии с рекуррентной формулой:

$$\phi_i = \gamma_i - \sum_{j=0}^{i-1} \gamma_i \{ \varphi_j(t, \mathbf{r}) \} \phi_j, \quad i = \overline{1, \infty}. \quad (21)$$

Далее производится расчет базисной функции по выражению (18). Процедура продолжается до вычисления необходимого числа членов ряда канонического разложения. Очевидно, вследствие произвольности выбора функционалов начального приближения γ_i существует бесчисленное множество канонических разложений стохастического пространственно-временного сигнала в отличие от единственного разложения Карунена-Лоэва. Получить каноническое раз-

ложение из разложения Карунена-Лоэва и наоборот возможно путем поворота осей координат [29], однако каноническое разложение не связывает линейные функционалы ϕ_i с характеристиками случайного процесса, в то время как ортогональное предполагает строгий вид этих функционалов, подынтегральной функцией в которых является базисная функция, то есть $\phi_i(t, \mathbf{r}) = \varphi_i(t, \mathbf{r})$. В результате возможно варьирование вида линейных функционалов ϕ_i для достижения определенной цели канонического разложения – минимума некоторой ошибки аппроксимации, энтропии и т. п. В том числе возможно наложение дополнительных условий в виде независимости коэффициентов разложения, или по крайней мере равенства нулю некоторых их центральных моментов, что позволяет применять данное разложение в случае негауссовских процессов.

Данные разложения позволяют описывать преобразования в канале путем матричных операций вида (5), причем естественно, что на выходе детерминированной линейной системы свойства канонических разложений сохраняются [32], то есть коэффициенты разложения остаются некоррелированными случайными величинами, имеющими характер белого шума, и определяются более простыми выражениями (9). Однако в случае прохождения сигналов через стохастические системы, в том числе определяемые детерминированной системной характеристикой и шумом, подобное свойство может не сохраняться. В результате требуется отыскание статистических характеристик на выходе канала связи, причем канонические разложения в этом случае приходится выполнять и над случайным процессом на выходе.

Таким образом, можно заключить, что, как и в случае интегральных разложений, выбор базиса в данном контексте счетного, пусть и бесконечного, числа функций, целиком определяется условиями решаемой задачи исходя из удобства представления и целей синтеза. При этом выбор базиса оказывается также достаточно произвольным.

Следует отметить еще одно немаловажное обстоятельство. В практических целях стохастический сигнал как на входе канала, так и на выходе описывается в виде конечномерного случайного вектора, обычно получающегося путем учета в разложениях (1) и (2) конечного числа членов ряда. Причем выбор этого числа определяется в основном эвристически, например первых столько-то, либо на основании максимальных значений некоторого параметра соответствующих им базисных функций. В результате с точки зрения описания как самих сигналов, так и канала связи неизбежна ошибка аппроксимации, вели-

чина которой зависит не только от типа канала связи и используемых сигналов, но и от формы базисных функций. Таким образом, чрезмерная вольность выбора формы базиса или свойств результирующих коэффициентов разложения приводит к недостаточно полному учету требований построения системы связи с точки зрения близости к практически нереализуемым бесконечномерным системам.

Именно поэтому при разложении следует учитывать не только свойства системной характеристики канала связи, но и характер неизбежно присутствующей в любом канале связи аддитивной помехи, которая, как указывалось ранее, может оказывать существенное влияние на статистические взаимосвязи коэффициентов разложения, в том числе и канонического.

В общем случае в канале связи присутствует пространственно-распределенная помеха $n(t', \mathbf{r}')$, которая аддитивно взаимодействует с полезной составляющей принимаемого сигнала. Причины ее возникновения могут быть самыми разнообразными, причем их проявление принимает различные формы, определяемые конкретной реализацией случайного процесса [5]. В результате подобные преобразования, включающие элемент стохастичности, оказываются принципиально необратимыми в отличие от линейных в отсутствие помех.

Таким образом, при описании аддитивных помех следует учитывать два немаловажных обстоятельства. Во-первых, модели аддитивных помех достаточно разнообразны и в общем случае должны характеризоваться довольно сложными многомерными законами распределения своих параметров, в том числе и мгновенных значений. Во-вторых, складываясь с полезным сигналом, подобные помехи изменяют статистическую структуру передаваемых данных. В результате принципиально важным фактом оказывается внесение неопределенности в передаваемые сообщения (даже при детерминированном характере канала связи), что приводит к невозможности безошибочной передачи исходных сообщений и необратимости преобразований в результирующем канале.

Следствием данных обстоятельств оказывается дополнительное накопление ошибки аппроксимации при синтезе систем на основе конечного набора базисных функций рядов (1) и (2). Так, в отсутствие аддитивной помехи и детерминированности системной характеристики канала связи N пространственно-временной сигнал на выходе канала связи однозначно определяется выражением (4) с той лишь разницей, что число членов ряда является конечным. При данных условиях корреляционная функция выходного сигнала получается путем дву-

кратного преобразования корреляционной функции исходного входного сигнала оператором системной характеристики H' [28]. Поскольку он является детерминированным, то в выражении для нахождения базисных функций (18) его можно поменять местами с оператором Фредгольма ϕ , что приводит к подобной достаточно простой операторной взаимосвязи базисных функций на входе и на выходе в виде (3). Однако даже в этом случае наилучшее приближение трактуется в смысле минимально возможной величины среднеквадратического отклонения усеченного ряда относительно исходного сигнала [28, 37]. При синтезе же систем связи данный критерий оказывается не всегда допустимым, в результате применение аппроксимации в виде даже наиболее общего случая канонического разложения является довольно грубым и требующим дополнительных исследований влияния конечности числа членов ряда разложения на получаемые оптимальные решения.

В общем же случае стохастичности оператора преобразования канала связи H' , обусловленной наличием аддитивной помехи и/или случайностью его системной характеристики H , каноническое разложение определяется на основе математического ожидания и корреляционной функции выходного пространственно-временного сигнала согласно выражениям (18), (20) и (21), а выходные базисные функции не имеют операторной зависимости (3) от входных [39]. В общем виде второй начальный момент выходного сигнала определяется как усреднение соответствующих случайных функций [32]:

$$M(x'x') = M([H'\{x\} + n][H'\{x\} + n]). \quad (22)$$

Для устранения громоздкости описания в данной и последующих формулах, если не оговорено дополнительно, аргументы при случайных функциях опущены, а при записи произведения двух функций подразумевается, что они зависят от различных аргументов, например $x'x' = x'(t, \mathbf{r})x'(t', \mathbf{r}')$. Корреляционная функция же вычисляется как разность между вторым начальным моментом и произведением математических ожиданий:

$$K_x = M(x'x') - M(x')M(x'). \quad (23)$$

Применение теорем сложения и умножения математических ожиданий [40], а также использование леммы о существовании оператора математического ожидания $M_{H'}$ случайного оператора системной

характеристики H' [41] и его линейности и независимости от передаваемого сигнала, приводит к выражению:

$$M(x') = M_{H'}\{M(x)\} + M(n). \quad (24)$$

Следует отметить, что в отличие от оператора системной характеристики H' оператор математического ожидания $M_{H'}$ имеет строго детерминированный характер и трактуется как усреднение ядра интегрального оператора либо коэффициентов дифференциального уравнения. Так, в случае интегральных преобразований и независимости импульсной характеристики от передаваемого пространственно-временного сигнала оператор математического ожидания имеет вид:

$$M_{H'}\{M(x)\} = \int_{\Omega} \int_{\Omega} M\{x(t, \mathbf{r})\} M\{h(t', \mathbf{r}', t, \mathbf{r})\} dt d\mathbf{r}.$$

Все последующие выражения для нахождения высших моментов случайных операторов имеют подобную же трактовку с той лишь разницей, что используется понятие оператора математического ожидания произведения случайных операторов [41].

Раскрытие скобок в (22) и применение теоремы сложения математических ожиданий, а также учет линейности оператора системной характеристики преобразует данное выражение к виду:

$$M(x'x') = M(H'[H'\{x\}]) + M(H'\{x\}n) + M(nH'\{x\}) + M(nn). \quad (25)$$

Использование свойства независимости случайного оператора системной характеристики, аддитивной помехи и передаваемого сигнала в совокупности с теоремой умножения математических ожиданий и леммы о существовании оператора математического ожидания произведения случайных операторов $M_{H'H}$, а также применение равенства (23) ко второму начальному моменту аддитивной помехи видоизменяют уравнение (25) следующим образом:

$$M(x'x') = M_{H'H}\{M(x)\} + M_{H'}\{M(x)\}M(n) + M(n)M_{H'}\{M(x)\} + K_n + M(n)M(n), \quad (26)$$

где K_n – корреляционная функция аддитивной помехи.

Очевидно, что сумма второго, третьего и пятого слагаемого в левой части данной формулы представляет собой разность между произведением математических ожиданий выходного сигнала (24) и произведением операторов математических ожиданий системной характе-

ристики $M_H\{M(x)\}$. Следовательно вычитание из (26) произведения (24) на самого себя, но с зависимостями от других переменных, приводит к выражению для корреляционной функции выходного сигнала:

$$K_x = M_{HH}\{M(x)\} + K_n - M_H\{M(x)\}M_H\{M(x)\}.$$

Применение (23) к операторам математического ожидания произведения случайных операторов системной характеристики и второго начального момента входного сигнала, а также приведение подобных слагаемых преобразует данное выражение к виду:

$$K_x = K_H\{K_x + M(x)M(x)\} + M_H\{M_H\{K_x\}\} + K_n. \quad (27)$$

где K_H – корреляционная функция случайного оператора системной характеристики H .

Частный случай данной формулы для преобразований линейных интегральных случайных операторов без учета воздействующей в канале аддитивной помехи описан В. С. Пугачевым в [28], а для случая равенства нулю математических ожиданий – в работах М. Лоэва. Кроме того, в случае отсутствия преобразований в канале связи следствием данной формулы оказывается хорошо известное выражение для корреляционной функции суммы двух независимых случайных процессов [32].

Выражение (27) иллюстрирует, что вид корреляционных связей выходного пространственно-временного сигнала существенно отличается от входного. А в результате зависимость выходных от входных базисных функций отличается от выражения (3). Применение в качестве начального приближения γ'_i при каноническом разложении выходных базисных функций ϕ'_i последовательности функционалов ϕ_i , определяющих случайные коэффициенты разложения входного сигнала, то есть предположение, что $\gamma'_i = \phi_i$, а также выражений (27) (18), (20), (21), в которых используются линейные функционалы для канонического разложения выходных базисных функций (функционалы ϕ_i , γ_i и базисные функции ϕ_i на входе в данных формулах заменены на соответствующие функционалы ϕ'_i , γ'_i и базисные функции ϕ'_i на выходе) и свойства линейности операторов математического ожидания M_H , корреляционной функции K_H и функционалов канонического разложения ϕ'_i делает возможным представ-

ление выходных базисных функций в виде рекуррентного операторного преобразования входных:

$$\varphi'_i = \frac{1}{d_{x'_i}} \left\{ \phi_i [K_H \{K_x + M(x)M(x)\}] + \phi_i [M_H \{M_H(K_x)\}] + \right. \\ \left. + \phi_i(K_n) - \sum_{j=0}^{i-1} \phi_i(\varphi'_j) d_{x'_j} \varphi'_j \right\}. \quad (28)$$

Первое и второе слагаемые в данном выражении предполагают необходимость осуществления функционального преобразования над операторами корреляционной функции K_H и математического ожидания системной характеристики M_H . Следует отметить, что линейный функционал ϕ_i применяется непосредственно лишь к данным операторам, не затрагивая составляющие, находящиеся в фигурных скобках, то есть сумму корреляционной функции входного сигнала и произведения его математических ожиданий, а также оператора математического ожидания системной характеристики от корреляционной функции входного сигнала.

При этом первое слагаемое существует только в случае стохастичности системной характеристики, а второе – ее нецентрированности. Причем в отличие от первого детерминированность системной характеристики приводит лишь к некоторому упрощению формы оператора математического ожидания, преобразующегося в детерминированный оператор системной характеристики H .

Третье слагаемое задается корреляционными свойствами аддитивной помехи, а соответственно отлично от нуля при условии ее учета. Требование биортогональности выходных базисных функций (19) предопределяет существование четвертого слагаемого, независимо от свойств исследуемого канала связи.

Следует также отметить, что найденная для общего случая стохастического канала связи корреляционная функция пространственно-временного сигнала на его выходе (27) позволяет рекуррентно вычислять выходные базисные функции на основе (28). Следует отметить бесчисленное количество их возможных форм, поскольку даже в случае строго задания начального приближения γ'_i последовательностью функционалов ϕ_i , определяющих случайные коэффициенты разложения входного сигнала, начальные приближения для входного канонического разложения γ_i являются произвольными. С одной стороны это оказывается достоинством при синтезе систем связи, так как делает

возможным варьирование вида базисных функций для достижения предполагаемого оптимума согласно некоторому критерию. С другой же стороны такая неопределенность может приводить к значительным различиям в получающихся решениях при условии их реализации в разнесенных системах, например на приеме и передаче, поскольку выбор начальных приближений может также включать элемент стохастичности. В результате вероятно наличие систематических ошибок передаваемой информации, обусловленных именно различием в форме используемых базисных функций.

5. Заключение. Возможность представления входного и выходного пространственно-временных сигналов в виде бесконечного ряда некоррелированных случайных функций не позволяет точно описывать процессы в канале связи, поскольку практическая реализация осуществима лишь для конечного числа слагаемых. Однако в большинстве случаев полезный сигнал изменяется существенно медленнее, чем помеха, либо имеет какую-либо локализацию, выражающуюся в различиях их статистических свойств. В результате в каноническом разложении пространственно-временного сигнала на выходе канала связи можно ограничиться некоторым числом членов ряда, дающих с определенной степенью точности описание входного сигнала. При этом либо число, либо последовательность коэффициентов разложения входного сигнала и выходного, искаженного стохастическим каналом связи, оказываются несогласованными в смысле точности их аппроксимации, что приводит к необходимости одновременного учета ошибок приближения, как на входе, так и на выходе канала связи. Очевидно, что с точки зрения среднеквадратической ошибки аппроксимации пространственно-временных сигналов на входе и выходе канале связи конечным рядом ответом на данный вопрос служит какая-нибудь форма канонического представления [28]. Однако при синтезе систем связи критерии в основном направлены на точность воспроизведения передаваемого сигнала, то есть задачи синтеза трактуются как задачи принятия статистических решений, показателями, качества которых являются те или иные степени схожести переданного сообщения и вынесенного решения [33]. Следовательно, разложение в ряд должно рассматриваться с позиции точного воспроизведения передаваемых сигналов на приемной стороне, а не с точки зрения точности аппроксимации передаваемых и принимаемых сигналов. Произвольность выбора базисных функций в ограниченном каноническом разложении позволяет подбирать их таким образом, чтобы обеспечить в некотором смысле точность приема сигнала. Другие же способы конечномерного

представления входных и выходных сигналов (например разложение Карунена-Лозва, различного рода системы ортогональных функций) делают возможным лишь выбор наиболее лучших представителей из заданного класса функций.

В целом же достаточность корреляционных свойств выходного пространственно-временного сигнала для его представления в виде канонического разложения скрывает в себе и серьезный недостаток. В данном случае фиксируется класс функций, придающих коэффициентам соответствующего разложения свойство некоррелированности, что с одной стороны делает возможным более простое описание дальнейших преобразований этих коэффициентов, а с другой – не только не обеспечивает их независимости в наиболее общих случаях негауссовских процессов, но и потенциально снижает качество синтезируемой системы, так как сокращает вариативность получаемых решений. Следовательно в общей постановке (негауссовские процессы) задача синтеза должна формулироваться без введения ограничений на класс функций или же свойства коэффициентов разложения. Даже независимость коэффициентов разложения не может гарантировать оптимальность по определенному критерию синтеза. Таким образом, в общем случае дискретное отображение непрерывного канала связи должно основываться на разложениях типа (1) и (2) с конечным числом членов ряда, в которых базисные функции вычисляются в соответствии с критерием оптимальности воспроизведения передаваемой информации, что является направлением дальнейших исследований и в настоящей работе не рассматривается.

Литература

1. *Возенкрафт Дж. М., Джекобс И.М.* Теоретические основы техники связи : пер. с англ. / Под ред. Р.Л. Добрушина // М.: Мир. 1969. 640 с.
2. *Батенков К.А.* Дискретные отображения непрерывного канала связи на основе обобщенного ряда Фурье // Вестник Рязанского государственного радиотехнического университета. Рязань: 2013. № 1(43). С. 12–20.
3. *Kazovsky L.G., Cheng N., Shaw W., et al.* Broadband optical access networks // Hoboken. New Jersey : John Wiley & Sons, Inc. 2011. 283 p.
4. *Horak R.* Telecommunications and data communications handbook // Hoboken. New Jersey : John Wiley & Sons, Inc. 2007. 791 p.
5. *Зоко А.Г., Кловский Д.Д., Коржик В.И., Назаров М.В.* Теория электрической связи : учеб. для вузов // М.: Радио и связь, 1999. 432 с.
6. *Кловский Д.Д., Конторович В.Я., Широков С.М.* Модели непрерывных каналов связи на основе стохастических дифференциальных уравнений // М.: Радио и связь, 1984. 247 с.
7. *Cioffi J.M.* Advanced Digital Communication. Class reader EE379C // Stanford University. 2005. URL: <http://www.stanford.edu/class/ee379c/>. (дата обращения: 12.09.2012).
8. *Elahmadi S., Srinath M.D., Rajan D., Haberman R.* Capacity and Modeling of Nonlinear Fiber Optic Communications as a Frequency-Selective Fading Channel //

- International Conference on Computing, Networking and Communications, Signal Processing for Communications Symposium. 2012. pp. 922–928.
9. *Зяблов В.В., Коробков Д.Л., Портной С.Л.* Высокоскоростная передача сообщений в реальных каналах // М.: Радио и связь. 1991. 288 с.
 10. *Cherubini G., Eleftheriou E., Olcer S., Cioffi J.M.* Filter Bank Modulation Techniques for very High-speed Digital Subscriber Lines // IEEE communication magazine. 2000. pp. 98–104.
 11. *Regalia P.* Filter Banks for Next Generation Multicarrier Wireless Communications // EURASIP Journal on Advances in Signal Processing. 2010. 147 p.
 12. *Zhang J. Djordjevic I.B.* Optimum Signal Constellation Design for Rotationally Symmetric Optical Channel with Coherent Detection // OSA/OFC/NFOEC. 2011.
 13. *Kahn J.M., Ho K.* Spectral Efficiency Limits and Modulation/Detection Techniques for DWDM Systems // International Conference on Computing, Networking and Communications, Signal Processing for Communications Symposium. 2012. pp. 922–928.
 14. *Tavassoli V.* High Capacity Phase/Amplitude Modulated Optical Communication Systems and Nonlinear Inter-Channel Impairments : diss. ... PhD in the Dep. of El. and Comp. Eng. // University of Victoria. 2012. 104 p.
 15. *Ip E., Lau A. P. T, Barros D. J. F., Kahn J. M.* Coherent detection in optical fiber systems // Optics express. 2008. vol. 16. no. 2. pp. 753–791.
 16. *Louchet H.* Top-Down Analysis of High-Capacity Fiber-Optic Transmission : diss. ... Doktor der Ingenieurwissenschaften Dr.-Ing. genehmigte. // Berlin. Technischen Universität Berlin. 2006. 123 p.
 17. *Caio Ya., Musslimani Z.H., Titi E.S.* Spectral Efficiency Limits and Modulation/Detection Techniques for DWDM Systems // Numerical Functional Analysis and Optimization. 2009. vol. 30(1–2). pp. 46–69.
 18. *Баскаков С.И.* Радиотехнические цепи и сигналы: учебник // М.: Высш. Школа. 1983. 536 с.
 19. *Френкс Л.* Теория сигналов : пер. с англ. // М.: Советское радио. 1974. 344 с.
 20. *Батенков К.А.* Математическое моделирование непрерывных многопараметрических каналов связи в операторной форме // Телекоммуникации. 2013. № 10. С. 2–4.
 21. *Батенков К. А.* Моделирование непрерывных каналов связи в форме операторов преобразования некоторых пространств // Труды СПИИРАН. 2014. № 1(32). С. 171–198.
 22. Методы классической и современной теории автоматического управления: учебник. В 5 т. Математические модели, динамические характеристики и анализ систем автоматического управления / Под ред. К. А. Пупкова, Н. Д. Егупова. 2-е изд. перераб. и доп. // М.: Издательство МГТУ им. Н. Э. Баумана. 2004. Т.1: 656 с.
 23. *Солодовников В. В., Дмитриев А. Н., Егупов Н. Д.* Спектральные методы расчета и проектирования систем управления // М.: Машиностроение, 1986. 440 с.
 24. *Пупков К.А., Егупов Н.Д., Макаренко А.М., Трофимов А.И.* Теория и компьютерные методы исследования стохастических систем // М.: ФИЗМАТЛИТ. 2003. 400 с.
 25. *Залманзон Л. А.* Преобразование Фурье, Уолша, Хаара и их применение в управлении, связи и других областях // М.: Наука. Гл. ред. физ.-мат. лит. 1989. 496 с.
 26. *Коллатц Л.* Задачи на собственные значения : пер. с нем. // М.: Наука, 1968. 504 с.
 27. *Варакин Л. Е.* Системы связи с шумоподобными сигналами // М.: Радио и связь. 1985. 384 с.
 28. *Пугачев В.С.* Теория случайных функций и ее применение к задачам автоматического управления // М.: Физматгиз. 1962. 883 с.

29. Драган Я.П. Модели сигналов в линейных системах // АН УССР. Физ.–мех. ин-т. Киев: Наукова думка. 1972. 302 с.
30. Батенков К.А. Подходы к решению задачи оптимального дискретного отображения непрерывного канала связи на основе обобщенной штрафной функции // Информационные системы и технологии. 2014. № 2(82). С. 78–83.
31. Рытов С.М., Крайнов Ю.А., Татарский В.И. Введение в статистическую радиофизику. Ч. 2. // М.: Наука. 1978. 464 с.
32. Вентцель Е.С., Овчаров Л.А. Теория случайных процессов и ее инженерные приложения : учеб. пособие для вузов. 2-е изд. стер. // М.: Высш. школа. 2000. 383 с.
33. Тихонов В.И. Статистическая радиотехника // М.: Советское радио. 1966. 681 с.
34. Васильева А.Б., Тихонов Н.А. Интегральные уравнения. 2-е изд. стереот. // М.: Физматлит. 2002. 160 с.
35. Ван Трис Г. Теория обнаружения, оценок и модуляции: пер. с англ. В 3 т. Теория обнаружения, оценок и линейной модуляции // М.: Советское радио, 1977. Т. 1. 744 с.
36. Тихонов В.И., Харисов В.Н. Статистический анализ и синтез радиотехнических устойчивых систем : учеб. пособие для вузов // М.: Радио и связь. 1991. 608 с.
37. Фу К. Последовательные методы в распознавании образов и обучение машин // М.: Наука. 1971. 320 с.
38. Соифер В.А. Прикладная теория информации: учебное пособие // Куйбышев: КуАИ. 1985. 93 с.
39. Батенков К.А. Проблема синтеза функциональных узлов дискретного канала связи по информационным критериям // Труды СПИИРАН. 2014. № 2(33). С. 5–23.
40. Вентцель Е.С., Овчаров Л.А. Теория вероятностей и ее инженерные приложения: учеб. пособие для вузов. 2-е изд., стер. // М.: Высш. шк. 2000. 480 с.
41. Скороход А.В. Случайные линейные операторы // Киев: Наукова думка. 1978. 200 с.

References

1. Wozencraft, J.M., Jacobs I.M. Principles of communication engineering. New York, NY: Wiley, 1965. (Russ. ed.: Wozenkraft Dzh.M. Teoreticheskie osnovy tehniki svyazi: per. s angl. Dzh.M. Wozenkraft, I.M. Dzhekobc ; pod red. R. L. Dobrushina. Moscow. Mir. 1969. 640 p.).
2. Batenkov K.A. [Discrete displays of a continuous channel of communication on the basis of the generalized Fourier's number]. *Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. Rjazan' – Vestnik of Ryazan State Radio Engineering University*, 2013. vol. 1(43). pp. 12–20. (In Russ.).
3. Kazovsky L.G., Cheng N., Shaw W., et al. Broadband optical access networks. Hoboken. New Jersey : John Wiley & Sons, Inc. 2011. 283 p.
4. Horak R. Telecommunications and data communications handbook. Hoboken. New Jersey : John Wiley & Sons, Inc. 2007. 791 p.
5. Zjuko A.G., Klovskij D.D., Korzhik V.I., Nazarov M.V. *Teorija jelektricheskoj svyazi: ucheb. dlja vuzov. Pod red. D. D. Klovskogo* [Theory of electric communication. Edited by D.D. Klovskogo]. M.: Radio i svjaz', 1999. 432 p. (In Russ.).
6. Klovskij D.D., Kontorovich V.Ja., Shirokov S.M. *Modeli neprerynyh kanalov svyazi na osnove stohasticheskikh differencial'nyh uravnenij. Pod red. D.D. Klovskij* [Models of continuous channels of communication on the basis of the stochastic differential equations. Edited by D.D. Klovskij]. M. Radio i svjaz'. 1984. 247 p. (In Russ.).
7. Cioffi J.M. Advanced Digital Communication. Class reader EE379C. Stanford University. 2005. Available at: <http://www.stanford.edu/class/ee379c/>. (accessed: 12.09.2012).

8. Elahmadi S., Srinath M.D., Rajan D., Haberman R. Capacity and Modeling of Nonlinear Fiber Optic Communications as a Frequency-Selective Fading Channel. International Conference on Computing, Networking and Communications, Signal Processing for Communications Symposium. 2012. pp. 922–928.
9. Zjablov V.V., Korobkov D.L., Portnoj S.L. *Vysokoskorostnaja peredacha soobshhenij v real'nyh kanalakh* [High-speed transmission of messages in real channels]. M. Radio i svjaz'. 1991. 288 p. (In Russ.).
10. Cherubini G., Eleftheriou E., Olcer S., Cioffi J.M. Filter Bank Modulation Techniques for very High-speed Digital Subscriber Lines. *IEEE communication magazine*. 2000. pp. 98–104.
11. Regalia P. Filter Banks for Next Generation Multicarrier Wireless Communications. Editor-in-Chief. *EURASIP Journal on Advances in Signal Processing*. 2010. 147 p.
12. Zhang J., Djordjevic I.B. Optimum Signal Constellation Design for Rotationally Symmetric Optical Channel with Coherent Detection. OSA/OFC/NFOEC. 2011.
13. Kahn J.M., Ho K. Spectral Efficiency Limits and Modulation/Detection Techniques for DWDM Systems. International Conference on Computing, Networking and Communications, Signal Processing for Communications Symposium. 2012. pp. 922–928.
14. Tavassoli V. High Capacity Phase/Amplitude Modulated Optical Communication Systems and Nonlinear Inter-Channel Impairments : diss. ... PhD in the Dep. of El. and Comp. Eng. University of Victoria. 2012. 104 p.
15. Ip E., Lau A.P.T., Barros D.J.F., Kahn J.M. Coherent detection in optical fiber systems. *Optics express*. 2008. vol. 16. no. 2. pp. 753–791.
16. Louchet H. Top-Down Analysis of High-Capacity Fiber-Optic Transmission : diss. ... Doktor der Ingenieurwissenschaften Dr.-Ing. genehmigte. Berlin. Technischen Universität Berlin. 2006. 123 p.
17. Cao Ya., Musslimani Z.H., Titi E.S. Spectral Efficiency Limits and Modulation/Detection Techniques for DWDM Systems. *Numerical Functional Analysis and Optimization*. 2009. vol. 30(1–2). pp. 46–69.
18. Baskakov S.I. *Radiotekhnicheskie cepi i signaly: uchebnik* [Radio engineering chains and signals]. M. Vyssh. Shkola. 1983. 536 p. (In Russ.).
19. Franks L.E. Signal theory. Englewood Cliffs, NJ: Prentice- Hall. 1969. (Russ. ed.: Frenks L. Teorija signalov, per. s angl. L. Frenks. M. Sovetskoe radio. 1974. 344 p.).
20. Batenkov K. A. [Mathematical modeling of continuous multiple parameter communication channels in an operator form]. *Telekommunikacii – Telecommunications*. 2013. no 10. pp. 2–4. (In Russ.).
21. Batenkov K.A. [Modeling of continuous channels of communication in the form of operators of transformation of some spaces]. *Trudy SPIIRAN – SPIIRAS Proceedings*. 2014. vol 1(32). pp. 171–198. (In Russ.).
22. *Metody klassicheskij i sovremennoj teorii avtomaticheskogo upravlenija: uchebnik. V 5 t. Matematicheskie modeli, dinamičeskie harakteristiki i analiz sistem avtomaticheskogo upravlenija. Pod red. K.A. Pupkova, N.D. Egupova. 2-e izd. pererab. i dop.* [Methods of the classical and modern theory of automatic control. Edited by K.A. Pupkova, N.D. Egupova. 2nd edition: rev and exp.]. M. Izdatel'stvo MGTU im. N. Je. Bauman. 2004. Issue 1. 656 p. (In Russ.).
23. Solodovnikov V.V., Dmitriev A.N., Egupov N.D. *Spektral'nye metody rasčeta i proektirovanija sistem upravlenija* [Spectral methods of calculation and design of control systems]. M. Mashinostroenie. 1986. 440 p. (In Russ.).
24. Pupkov K.A., Egupov N.D., Makarenkov A.M., Trofimov A.I. *Teorija i komp'juternye metody issledovanija stohasticheskijh sistem* [Theory and computer methods of research of stochastic systems]. M. FIZMATLIT. 2003. 400 p. (In Russ.).
25. Zalmanzon L.A. *Preobrazovanie Fur'e, Uolsha, Haara i ih primenenie v upravlenii, svjazi i drugih oblastjah* [Fourier, Walsh, Haar's transformation and their application

- in management, communication and other areas]. M. Nauka. Gl. red. fiz.-mat. lit. 1989. 496 p. (In Russ.).
26. Kollatc L. *Zadachi na sobstvennye znachenija : per. s nem.* [Tasks on own values]. M. Nauka. 1968. 504 p. (In Russ.).
 27. Varakin L.E. *Sistemy svyazi s shumopodobnymi signalami* [Communication systems with noise-type signals]. M. Radio i svjaz'. 1985. 384 p. (In Russ.).
 28. Pugachev B.C. *Teorija sluchajnyh funkcij i ee primenenie k zadacham avtomaticheskogo upravlenija* [The theory of stochastic functions and its application to problems of automatic control]. M. Fizmatgiz. 1962. 883 p. (In Russ.).
 29. Dragan Ja.P. *Modeli signalov v linejnyh sistemah* [Models of signals in linear systems]. AN USSR. Fiz.–meh. in-t. Kiev, Naukova dumka, 1972. 302 p. (In Russ.).
 30. Batenkov K.A. [Approaches to the solution of a problem of optimum discrete display of a continuous channel of communication on the basis of the generalized penalty area of function]. *Informacionnye sistemy i tehnologii – Information Systems and Technologies*. 2014. vol. 2(82). pp. 78–83. (In Russ.).
 31. Rytov S.M., Kravcov Ju.A., Tatarskij V.I. *Vvedenie v statisticheskiju radiofiziku. Ch. 2. [Introduction to statistical radiophysics. Part 2]*. M. Nauka. 1978. 464 p. (In Russ.).
 32. Ventcel' E.S., Ovcharov L.A. *Teorija sluchajnyh processov i ee inzhenernye prilozhenija : ucheb. posobie dlja vuzov. 2-e izd. ster.* [Theory of casual processes and its engineering appendices. 2nd edition]. M. : Vyssh. shkola. 2000. 383 p. (In Russ.).
 33. Tihonov V.I. *Statisticheskaja radiotekhnika* [Statistical radio engineering]. M. Sovetskoe radio. 1966. 681 p. (In Russ.).
 34. Vasil'eva A.B., Tihonov N.A. *Integral'nye uravnenija. 2-e izd. stereot.* [Integral equation. 2nd edition]. M. Fizmatlit. 2002. 160 p. (In Russ.).
 35. Van Tris G. *Teorija obnaruzhenija, ocenok i moduljacji : per. s angl. V 3 t. Teorija obnaruzhenija, ocenok i linejnoj moduljacji* [Theory of detection, estimates and modulation]. M. Sovetskoe radio, 1977. Issue 1. 744 p. (In Russ.).
 36. Tihonov V.I., Harisov V.N. *Statisticheskij analiz i sintez radiotekhnicheskikh ustojchivyh sistem : ucheb. posobie dlja vuzov* [Statistical analysis and synthesis of radio engineering steady systems]. M. Radio i svjaz'. 1991. 608 p. (In Russ.).
 37. Fu K. *Posledovatel'nye metody v raspoznovanii obrazov i obuchenie mashin* [Consecutive methods in recognition of images and training of cars]. M. Nauka. 1971. 320 p. (In Russ.).
 38. Sojfer V.A. *Prikladnaja teorija informacii: uchebnoe posobie* [Applied theory of information]. Kujbyshev, KuAI. 1985. 93 p. (In Russ.).
 39. Batenkov K. A. [Problem of synthesis of functional knots of a discrete communication channel by information criteria]. *Trudy SPIIRAN – SPIIRAS Proceedings*. 2014. vol. 2(33). pp. 5–23. (In Russ.).
 40. Ventcel' E.S., Ovcharov L.A. *Teorija verojatnostej i ee inzhenernye prilozhenija : ucheb. posobie dlja vuzov. 2-e izd., ster.* [Probability theory and its engineering appendices. 2nd edition]. M. Vyssh. shk., 2000. 480 p. (In Russ.).
 41. Skorohod A.V. *Sluchajnye linejnye operatory* [Casual linear operators]. Kiev. Naukova dumka. 1978. 200 p. (In Russ.).

Батенков Кирилл Александрович — к-т техн. наук, сотрудник, Академия Федеральной службы охраны Российской Федерации. Область научных интересов: статистическая теория связи, модели и методы обработки сигналов. Число научных публикаций — 120. pustur@yandex.ru; Приборостроительная, 35, Орел, 302034; р.т.: +7(486)254-13-25.

Batenkov Kirill Aleksandrovich — Ph.D., researcher, The Academy of Federal Security Guard Service of the Russian Federation. Research interests: statistical communication theory, models and methods of signal processing. The number of publications — 120. pustur@yandex.ru; 35, Priborostroitelnaya Street, Orel, 302034, Russia; office phone: +7(486)254-13-25.

РЕФЕРАТ

Батенков К. А. **Моделирование и синтез линейных дискретных отображений непрерывных каналов связи.**

Рассматриваемая проблема моделирования и синтеза дискретных каналов связи имеет актуальный характер для задач, связанных с проектированием информационных систем, функционально предназначенных для передачи данных.

Формализованы дискретные отображения непрерывных каналов связи в виде операторов преобразования метрических конечномерных пространств общего вида.

Показана их связь с представлением в виде интегральных преобразований на основе предельного перехода от представления в виде рядов.

Сигнал на выходе линейного канала связи представим как в виде линейной комбинации системной характеристики канала и коэффициентов разложения входного сигнала, так и в форме дискретной свертки матричной системной характеристики и коэффициентов разложения входного сигнала, вне зависимости от выбора формы базисов.

В результате выходной сигнал можно рассматривать в виде дискретного разложения в базисе дискретной системной характеристики.

Нестационарный канал связи, в том числе и стохастический, описывается в виде матричной системной характеристики, называемой спектральной, или проекционной. Данное обстоятельство устанавливает наличие алгебраической связи между коэффициентами разложения входных и выходных сигналов.

Формализованы дискретные отображения непрерывных каналов связи в виде операторов преобразования метрических пространств с заданными базисами.

Аналогично интегральным преобразованиям, в качестве базисных функций может использоваться произвольный ряд линейно независимых функций, причем не обязательно ортогональных.

Формализованы дискретные отображения непрерывных каналов связи в виде операторов преобразования метрических пространств с базисом в виде координатных функций интегрального канонического представления В. С. Пугачева.

Главной особенностью данных отображений является задание форм базисных ядер на основе статистических свойств входных и выходных сигналов, которые задаются в общем случае в виде дискретного белого некоррелированного шума.

Операторы аналого-дискретных преобразований дискретных отображений непрерывных каналов связи на основе канонического разложения В. С. Пугачева, вычисляются путем итерационной процедуры, последовательно определяющей базисные функции на выходе канала в виде рекуррентного операторного преобразования входных.

SUMMARY

Batenkov K. A. Modeling and Synthesis of Discrete Mappings of Linear Continuous Channel.

Considered problem of modeling and synthesis of discrete mappings of linear continuous channel is actual for tasks associated with the design of an information system functionally intended for data transmission.

Discrete mappings of linear continuous channel in the form of general metric finite-dimensional space operators are formalized.

Its coupling with representation in the form of integral transforms on the basis of limiting process from series view is shown.

Signal on linear channel output is offered in system characteristic and input signal expansion coefficient linear combination form and matrix system characteristic and input signal expansion coefficient discrete convolution.

Consequently, output signal should be approached as discrete expansion in basis of discrete system characteristic.

Time-varying channel including stochastic is described in matrix system characteristic form called spectral or projecting. This circumstance determines algebraic bond between input and output signal expansion coefficients.

Discrete mappings of linear continuous channel in metric finite-dimensional space operator form with adjusted basis are formalized.

In similar integral conversions, arbitrary linearly independent functional series can be used as basis.

Discrete mappings of linear continuous channel in the form of metric finite-dimensional space operators with Pugachev integral canonical presentation of coordinate function basis are formalized.

Its principal singularity is basis kernel assignment in terms of input and output signal statistical properties set as, in general, in the form of discrete white uncorrelated noise.

Semi-digital operators of discrete mappings of linear continuous channel on the basis of Pugachev canonical presentation are calculated by force of iterative procedure consistently conditioning basis functions on input and output channel in terms of recurrence input transformation.

С.Н. МИХАЙЛОВ, О.И. МАЛАШЕНКО, А.А. ЗАЙЦЕВА
**МЕТОДИКА ИНФОЛОГИЧЕСКОГО АНАЛИЗА
СЕМАНТИЧЕСКОГО СОДЕРЖАНИЯ ОБРАЩЕНИЙ
ПАЦИЕНТОВ ДЛЯ ОРГАНИЗАЦИИ ЭЛЕКТРОННОЙ ЗАПИСИ**

Михайлов С.Н., Малашенко О.И., Зайцева А.А. Методика инфологического анализа семантического содержания обращений пациентов для организации электронной записи.

Аннотация. В рамках инфологического подхода, в целях автоматизации процесса записи больных в медицинских учреждениях, предложена методика обработки обращений пациентов на основе применения инфологической системы, позволяющая организовать электронную запись больных к специалистам медицинского учреждения путем выявления семантического содержания жалоб на состояние здоровья.

Ключевые слова: инфологический подход, тематическая антология, семантическое содержание, структурная декомпозиция текста, понятийный граф текста.

Mikhailov S.N., Malashenko O.I., Zaytseva A.A. The Method for the Infology Analysis of Patients Complaints Semantic Content in Order to Organize the Electronic Appointments.

Abstract. The method for patient reception processing based on infological system is proposed in the context of infological approach. This method allows organizing electronic queue for specialist attendance in health care facilities by semantic evaluation of patient health complaints.

Keywords: infological approach, thematic anthology, semantic content, structural decomposition of text, semantic environment of text.

1. Введение. В соответствии с общими положениями концепции развития системы здравоохранения в Российской Федерации до 2020 года среди приоритетов государственной политики особо выделяется направление повышения доступности и качества медицинской помощи [1]. Одним из путей повышения доступности медицинской помощи в рамках информатизации отрасли здравоохранения в РФ [2] выступает организация электронной записи пациентов на прием к требуемым специалистам. Такой подход показал достаточно высокую эффективность при реализации государственных услуг по технологии «единого окна» на основе создания многофункциональных центров государственных и муниципальных услуг.

В настоящее время процедура обращения за медицинской помощью в поликлинику сопряжена с существенной потерей времени и сил до момента непосредственного осмотра специалистом. Это связано с транспортными затратами, ожиданием в очереди как в регистратуру, так и непосредственно у кабинета врача. Кроме того, зачастую первым консультантом на пути получения медицинской помощи выступает регистратор, имеющий, в лучшем случае, среднее медицинское обра-

зование, что неизбежно может привести к ошибочному выбору профильного специалиста. С целью исключения подобных ошибочных обращений во многих лечебных учреждениях направление к «узким» специалистам (неврологу, хирургу, эндокринологу и т.д.) производится только после осмотра терапевта. Что также сопряжено с потерей времени и ухудшением самочувствия, особенно в условиях стремительного развития болезни. Часто возникают ситуации, когда возникшие у человека жалобы на состояние здоровья (симптомы заболевания), являются основанием для вызова бригады «скорой медицинской помощи» и экстренного направления в стационар, а в этом случае временной фактор может «дорого» обойтись больному (возможны серьезные осложнения, вплоть до летального исхода).

Перечисленные обстоятельства показывают, что разработка методики инфологической обработки обращений пациентов, позволяющей выделить смысловое содержание жалоб пациентов, определить профиль обращения и реализовать электронную запись к специалисту в удобное время является актуальной и представляет несомненный практический интерес.

2. Инфологический подход к обработке текстовых документов. В основу предлагаемой методики положен инфологический подход к обработке текстовых документов [3–6], суть которого состоит в итерационном процессе формирования тематических знаний посредством формирования тематических антологий — т.е. на основе предметно-ориентированных корпусов, их тезаурусов и глоссариев производится уточнение области и оценка сходства с ними новых текстовых документов.

Понятие «инфологии» дается согласно [3, 7] и понимается как проблема записи различного контента в виде, доступном для пользователя информационной системой.

В самом общем виде реализация методики обработки обращений пациентов предполагает наличие инфологической системы, ориентированной на хранение и совместную обработку тематических антологий, сформированных медицинскими специалистами в виде списков возможных жалоб (рисунок 1), и перечня жалоб пациента на свое состояние здоровья. Процесс инфологической обработки хранящихся тематических антологий врачей-специалистов и поступающих перечней жалоб предполагает выполнение структурной декомпозиции текстов и построение их понятийных графов (семантического окружения) — формирование онтологий [8–10]. В большинстве существующих в настоящее время систем интеллектуального анализа текстов, таких как [11], производится выделение синтаксических и семантиче-

ских дефиниций с использованием predeterminedенных заранее классификаторов. Также существует ряд подходов, специализирующихся на сравнении онтологий [12–15]. Отличительной особенностью таких методов является построение predeterminedенных онтологий, с последующим их сравнением [16].

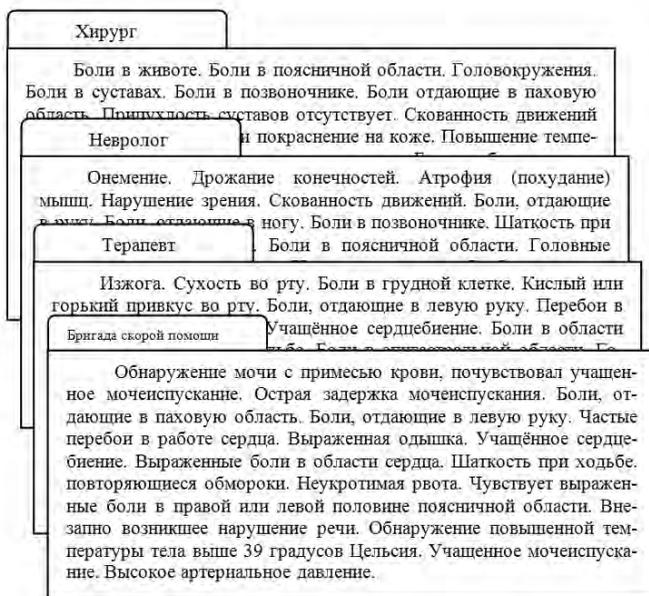


Рис. 1. Примеры тематических антологий специалистов

В отличие от них, в разрабатываемой методике производится ассоциативное формирование онтологий в процессе работы системы, а также использование в качестве понятийного графа — графа ассоциативных связей, формируемого непосредственно на самих текстах тематической антологии [8]. Результаты структурной декомпозиции текста сохраняются с использованием XML-разметки. Сохранению подлежит ранжирование по значимости используемых смысловых единиц (терминов или понятий) и их связей друг с другом в самих текстах тематической антологии. В дальнейшем полученный XML-файл используется для построения семантического окружения текста в виде визуального графа. На рисунке 2 показан пример онтологии специалиста в виде XML-файла, а на рисунке 3а — соответствующий ему графический вид семантического окружения.

```

<data count="12" nodes="7-3-(7),3-1-(3),7-4-(3),7-8-(3),7-1-(3),7-2-(2),4-5-(2),4-2-(2),7-5-(2),7-6-(2),2-5-(2),7-9-(2),"
mode="vis">
<node id="1" title="область"/>
<node id="2" title="после"/>
<node id="3" title="боль"/>
<node id="4" title="нога"/>
<node id="5" title="ходьба"/>
<node id="6" title="слабость"/>
<node id="7" title="хирург"/>
<node id="8" title="сустав"/>
<node id="9" title="притупленность"/>
</data>

```

Рис. 2. Пример XML-файла онтологии хирурга

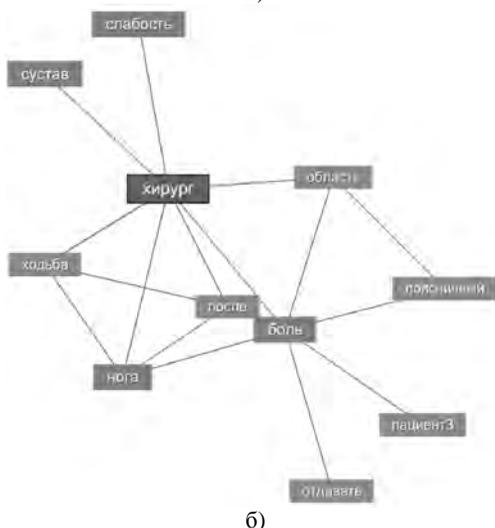
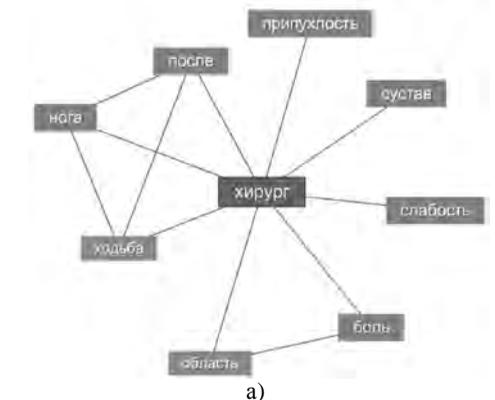


Рис. 3. Визуальный граф исходного (а) и расширенного (б) семантических окружений (онтологий) хирурга

Инфолингвистическая обработка перечней поступающих жалоб конкретного пациента также позволяет получить семантическое окружение этих жалоб и сохранить его в виде соответствующего нового XML-файла. По результатам сравнения полученных семантических окружений имеющихся антологий и поступающих жалоб конкретного пациента принимается решение об их отнесении к области компетенции соответствующего специалиста (хирурга, невролога, терапевта, эндокринолога и т. д.), т. е. выполняется процедура тематической кластеризации жалоб пациентов [9]. На рисунке 4 приведена блок-схема предложенной методики обработки обращений пациентов на основе применения инфолингвистической системы.

Отличительной особенностью предложенной методики является принятие решения о направлении пациента к соответствующему специалисту не по результатам непосредственного сравнения слов в его жалобе с предметной областью специалиста, а на основе анализа семантического окружения симптомов заболевания и его отнесения к области компетенции требуемого специалиста.

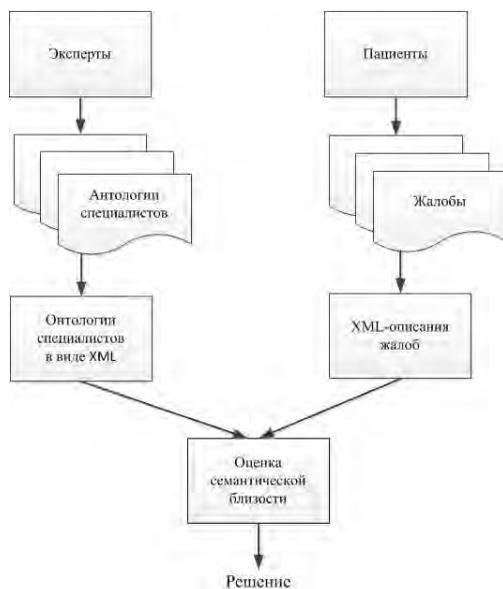


Рис. 4. Блок-схема методики обработки обращений пациентов в рамках применения инфолингвистической системы

Такой подход позволяет реализовать рутинные операции регистрации и определения требуемого специалиста в элементах информационно-коммуникационной инфраструктуры медицинского учрежде-

ния, обеспечивающих формирование электронной записи пациентов к соответствующим специалистам.

Реализация методики инфологической обработки показаний пациентов предполагает последовательное выполнение следующих взаимосвязанных этапов обработки текстов антологий и жалоб больного на состояние своего здоровья:

1) формирование каждым специалистом антологии по сфере собственной компетенции, объединяющий наиболее полный перечень возможных жалоб, характерных для определенных патологий (A_1, A_2, \dots, A_n , где n — количество «узких» специалистов лечебного учреждения);

2) хранение и инфологическая обработка каждой A_i антологии с формированием соответствующего XML-файла семантических окружений ($X_{A_1}, X_{A_2}, \dots, X_{A_n}$);

3) получение и инфологическая обработка нового текстового документа T , составленного пациентом и содержащего перечисление его жалоб на состояние собственного здоровья;

4) формирование XML-файла нового текстового документа жалобы (X_T);

5) количественная оценка семантической близости [17] имеющихся семантических окружений и нового документа жалобы путем обработки их XML-файлов ($X_{A_1}, X_{A_2}, \dots, X_{A_n}$ и X_T);

6) выработка варианта решения об отнесении жалоб больного к определенной области компетенций конкретного специалиста;

7) оценка загруженности требуемого специалиста по имеющемуся графику приема и формирование предложения о дате и времени посещения лечебного учреждения;

8) заблаговременное представление специалисту данных о планируемых к посещению пациентах с перечнем проявляющихся симптомов;

9) добавление специалистом, при необходимости, нового текстового документа T в состав соответствующей антологии по результатам осмотра и проведенной консультации.

На первом этапе предполагается непосредственное участие специалистов в процессе формирования каждым из них перечня потенциальных жалоб, характеризующих проявление определенных патологий в области собственной компетенции (хирург, невролог, терапевт, эндокринолог и др.).

Второй этап ориентирован на получение и хранение отражения семантического содержания потенциальных жалоб пациентов в сферах компетенций различных специалистов.

Третий и четвертый этапы направлены на формирование и сохранение отражения семантического содержания жалоб конкретного пациента на состояние собственного здоровья.

В ходе реализации пятого этапа проводится оценка близости семантического окружения жалобы конкретного пациента и возможных проявлений определенных патологий. Это позволяет эффективно транслировать естественно-языковые формулировки жалоб пациентов с использованием обиходной разговорной речи в признаки проявления определенной патологии.

На шестом этапе фиксируется вариант решения с целью запуска либо алгоритма реализации электронной записи на прием к специалисту либо уточнения этого варианта решения у компетентного консультанта (в случае отсутствия признаков явной семантической близости нового текстового документа жалобы к имеющимся антологиям специалистов).

Седьмой и восьмой этапы отражают выполнение известных функций, реализованных в информационных системах электронной очереди.

Заключительный, девятый этап позволяет осуществить итерационный процесс расширения содержания имеющихся антологий в случаях развития знаний и подтвержденных практических результатов в определенной области компетенций специалистов.

С целью подтверждения практической реализации предложенной методики проведены экспериментальные исследования процессов обработки составленных специалистами антологий и списков обращений четырех пациентов на макете инфологической системы, созданном коллективом Научно-образовательного центра инфокоммуникационных технологий и систем в ходе выполнения проекта 13-07-00137 «Исследование и разработка научно-технических путей создания инфологической системы информационно-аналитического обеспечения научных исследований ВУ-За» при поддержке Российского Фонда Фундаментальных Исследований.

На рисунках 3а, 5а, 6а представлены визуальные графы исходных семантических окружений (онтологий) для различных медицинских специалистов.

В таблице 1 представлено содержание обращений четырех пациентов на состояние собственного здоровья.

Таблица 1. Содержание жалоб на состояние здоровья

№ пациента	Содержание жалоб на состояние здоровья
Пациент 1	Боли в поясничной области; боли, отдающие в ногу
Пациент 2	Боли в поясничной области; учащенное мочеиспускание; повышенная температура тела
Пациент 3	Боли в поясничной области; боли, отдающие в паховую область
Пациент 4	Выраженные боли в правой половине поясничной области; учащенное мочеиспускание; повышенная температура тела; моча с примесью крови

Данные таблицы 1 показывают, что пациенты формулируют частично одинаковые жалобы, но у каждого из них разное их сочетание. Анализ содержания жалоб пациентов специалистами показал, что пациент 1 должен получить направление на прием к неврологу по поводу пояснично-крестцовой радикулопатии, пациент 2 — консультацию терапевта (признаки обострения хронического гломерулонефрита), пациент 3 — консультацию хирурга (симптомы мочекаменной болезни), а пациенту 4 рекомендован вызов бригады скорой медицинской помощи по поводу «почечной колики».

В ходе экспериментальных исследований тексты с перечислением жалоб четырех пациентов последовательно включались в сформированные антологии и проводилась инфологическая обработка полученных расширенных антологий. Визуальные графы семантических окружений, представленные на рисунках 3б, 5б, 6б показывают, что инфологическая система после оценки семантической близости текста пациента 3 отнесла его жалобы к компетенции хирурга, жалобы пациента 4 к компетенциям врача бригады скорой медицинской помощи, а пациента 1 — к неврологу, что совпадает с результатами предварительной оценки перечня жалоб пациентов соответствующими специалистами.

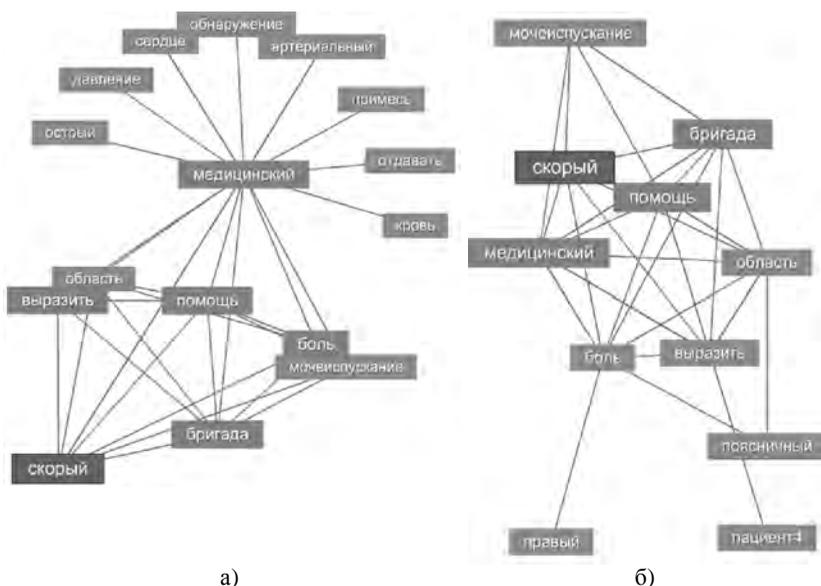


Рис. 5. Визуальный граф исходного (а) и расширенного (б) семантических окружений врача скорой медицинской помощи



а)



б)

Рис. 6. Визуальный граф исходного (а) и расширенного (б) семантических окружений невролога

Таким образом, апробация применения инфологической обработки перечня обращений различных пациентов показала практическую возможность реализации предложенной методики для организации электронной записи больных на прием к специалисту территориального лечебного учреждения.

Практическое применение методики инфологической обработки показаний пациентов заключается в следующем. Посредством сайта территориального лечебного учреждения обеспечивается информационная связь с инфологической системой, содержащей антологию, раз-

работанные штатными врачами специалистами соответствующие им XML-файлы.

Заболевший человек посредством ПК через Интернет (самостоятельно, если не в состоянии, то с помощью родственников, соседей или друзей) выходит на сайт своего территориального лечебного учреждения, авторизуется и перечисляет имеющиеся у него жалобы на состояние здоровья. При этом автоматически создается новый текстовый документ, подлежащий дальнейшей инфологической обработке и оценке его близости к одному или нескольким имеющимся семантическим окружениям, отражающим компетенции штатных врачей-специалистов данного лечебного учреждения. В случае уверенного отнесения жалоб пациента к компетенциям специалиста вырабатывается вариант решения на прием определенному врачу. В противном случае организуется on-line помощь опытного консультанта. В итоге обращения будет проведена либо запись на прием к определенному врачу, либо рекомендован немедленный вызов бригады «скорой медицинской помощи». В первом случае пациенту по электронной почте доводится: специальность врача и его фамилия имя и отчество; № талона или очередности; время приема; № кабинета врача. Во втором случае производится немедленное информирование дежурного специалиста по станции «скорой медицинской помощи» о наличии и местоположении пациента с определенным перечнем жалоб. Всю эту же информацию получит больной за несколько минут на свой ПК или в виде SMS-сообщения, на сотовый телефон, указанный при авторизации. Пациенту останется только прибыть в назначенное время в указанный кабинет. Эта же информация будет отражена и в компьютере регистратуры данного лечебного учреждения и регистратору останется только найти (оформить) медицинскую карту данного пациента и доставить её к определенному часу в кабинет указанного врача.

3. Заключение. Предложенная в настоящей работе методика инфологической обработки обращений пациентов направлена на обеспечение качественно нового уровня доступности медицинских услуг в территориальных учреждениях здравоохранения и позволяет функционально расширить сервис электронной записи больных на прием к врачам-специалистам. При этом может быть существенно повышена оперативность и упорядоченность обращений за медицинской помощью, что уже доказано практикой организации «единого окна» в многофункциональных центрах оказания государственных и муниципальных услуг.

В работе показывается принципиальная возможность использования инфологического подхода для повышения оперативности и доступности оказания медицинских услуг.

Практическая реализация предложенной методики требует применения инфологической системы в качестве функционального элемента либо системы информационно-аналитического обеспечения медицинского учреждения, либо региональной информационно-телекоммуникационной инфраструктуры здравоохранения.

Литература

1. Приказ от 28 апреля 2011 г. N 364. Об утверждении концепции создания единой государственной информационной системы в сфере здравоохранения (в ред. Приказа Минздравсоцразвития России от 12.04.2012 N 348). 2011.
2. Дюк В.А., Эмануэль В.Л. Информационные технологии в медико-биологических исследованиях // СПб.: Питер. 2003.
3. Александров В.В., Андреева Н.А., Кулешов С.В. Методы построения информационно-логических систем // СПб: Из-во Политех. ун-та. 2006. 96 с.
4. Андреева Н.А., Кокорин П.П. Система построения понятийной иерархии для ассоциативного поиска по текстам. // Информационно-измерительные и управляющие системы. 2008. Т.6. №4. С. 9–13.
5. Александров В.В., Кулешов С.В. Аналитический мониторинг Internet контента. Инфологический подход. // Качество. Инновации. Образование. 2008. № 3. с. 68–70.
6. Кокорин П.П. Инфологическая система аналитического мониторинга научно-технических фондов библиотек // Информационно-измерительные и управляющие системы. 2009. №4, т.7. С. 11–15.
7. Långfors B., Dahlbom B. Essays on Infology: Summing up and Planning for the Future // Studentlitteratur, Lund. 1993. 187 p.
8. Кулешов С.В. Разработка автоматизированной системы семантического анализа и построения визуальных динамических глоссариев // Диссертация на соискание ученой степени кандидата технических наук. Санкт-Петербург. 2005. 100 с.
9. Александров В.В., Кулешов С.В. Семиологические информационные системы — аналитическое самореферирование // Материалы X Международной конференции и Российской научной школы «Системные проблемы надежности, качества, информационных технологий (Инноватика-2005)». Москва. Радио и связь. 2005. Ч. 6. С. 9–14.
10. Михайлов С.Н. Способ тематической кластеризации текстовых документов на основе их инфологической обработки // Научные технологии. 2012. Т. 13. № 9. С. 48–51.
11. Кузнецова Ю.М., Осипов Г.С., Чудова Н.В. Изучение положения дел в науке с помощью методов интеллектуального анализа текстов // Управление большими системами: сборник трудов. 2013. № 44. С. 106–138.
12. Smirnov A.V., Pashkin M., Chilov N., Levashova T. Intelligent Manufacturing Cover image Agent-based support of mass customization for corporate knowledge management. // Engineering Applications of Artificial Intelligence. 2003. vol. 16. Issue 4. pp. 349–364.
13. Smirnov A., Levashova T., Shilov N. Patterns for context-based knowledge fusion in decision support systems // Information Fusion. 2015. vol. 21. pp. 114–129.
14. Zenuni X. Raufi B., Ismaili F., Ajdari J. State of the Art of Semantic Web for Healthcare // Procedia — Social and Behavioral Sciences. 2015. vol. 195. pp. 1990–1998.
15. Kim H., Kang S., Oh S. Ontology-based quantitative similarity metric for event matching in publish/subscribe system // Neurocomputing. 2015. vol. 152. pp. 77–84.

16. Khan S., Safyan M. Semantic matching in hierarchical ontologies // Journal of King Saud University - Computer and Information Sciences. 2014. Issue. 3. vol. 26. pp. 247–257.
17. Зайцева А.А., Кулешов С.В., Михайлов С.Н. Метод оценки качества текстов в задачах аналитического мониторинга информационных ресурсов // Труды СПИИРАН. 2014. Вып. 37. С. 144–155.

References

1. Prikaz ot 28 aprelya 2011 g. N 364. Ob utverzhdenii koncepcii sozdaniya edinoj gosudarstvennoj informacionnoj sistemy v sfere zdravoohraneniya (v red. Prikaza Minzdravsocrazvitiya Rossii ot 12.04.2012 N 348). [On approval of the concept of a unified state information system in the health sector]. 2011. (In Russ.).
2. Duk V.A., Jemanujel' V.L. *Informacionnye tehnologii v mediko-biologicheskikh issledovanijah* [Information technologies in medicine and biology researches]. SPb.: Piter. 2003. (In Russ.).
3. Alexandrov V.V., Andreeva N.A., Kuleshov S.V. *Metody postroyeniya informatsionno-logicheskikh system* [Methods of construction for information and logistic systems]. SPb: Iz-vo Politekh. un-ta. 2006. 96 p. (In Russ.).
4. Andreeva N.A., Kokorin P.P. [The System of conceptual hierarchy construction for associative text search]. *Informacionno-izmeritel'nye i upravljajushhie sistemy – Information-measuring and control systems*. 2008. Issue 6. vol.4. pp. 9–13. (In Russ.).
5. Alexandrov V.V., Kuleshov S.V. Analytical monitoring of Internet content. Infological approach. *Kachestvo. Innovacii. Obrazovanie Quality – Innovations. Education*. 2008. vol. 3. pp. 68–70.
6. Kokorin P.P. [Infological system for analytical monitoring of sci-tech fonds of libraries]. *Informacionno-izmeritel'nye i upravljajushhie sistemy – Information-measuring and control systems*. 2009. Issue 7. vol. 4. pp. 11–15. (In Russ.).
7. Langefors B., Dahlbom B. Essays on Infology: Summing up and Planning for the Future. Studentlitteratur. Lund. 1993. 187 p.
8. Kuleshov S.V. *Razrabotka avtomatizirovannoj sistemy semanticheskogo analiza i postroyeniya vizual'nyh dinamicheskikh glossarijev* [The development of automatic semantic analysis system and visual dynamic glossaries]. Ph.D. (Tech) dissertation. Sankt-Peterburg. 2005. 100 p. (In Russ.).
9. Alexandrov V.V., Kuleshov S.V. [Semiological information systems — analytical self-referring]. *Materialy X Mezhdunarodnoj konferencii i Rossijskoj nauchnoj shkoly «Sistemnye problemy nadezhnosti, kachestva, informacionnyh tehnologij (Innovatika-2005)»* [Materials of X International Conference and Russian Scientific School “INNOVATICA-2005”] Moskva. Radio i svjaz'. 2005. vol. 6. pp. 9–14. (In Russ.).
10. Mikhailov S.N. [The method of thematic clustering of text documents based on their infological processing]. *Naukoyemkiye tekhnologii – Science Intensive Technologies*. 2012. Issue. 9. vol. 13. pp. 48–51. (In Russ.).
11. Kuznecova Ju.M., Osipov G.S., Chudova N.V. [Intellectual analysis of scientific publications and the current state of science]. *Upravlenie bol'shimi sistemami: sbornik trudov – Large-scale Systems Control*. 2013. vol. 44. pp. 106–138. (In Russ.).
12. A.V. Smirnov, M. Pashkin, N. Chilov, T. Levashova. Intelligent Manufacturing Cover image Agent-based support of mass customization for corporate knowledge management. *Engineering Applications of Artificial Intelligence*. 2003. Vol. 16, Is. 4, pp. 349–364.
13. Smirnov A., Levashova T., Shilov N. Patterns for context-based knowledge fusion in decision support systems. *Information Fusion*. 2015. vol. 21. pp. 114–129.

14. Zenuni X. Raufi B., Ismaili F., Ajdari J. State of the Art of Semantic Web for Healthcare. *Procedia – Social and Behavioral Sciences*. 2015. vol. 195. pp. 1990–1998.
15. Kim H., Kang S., Oh S. Ontology-based quantitative similarity metric for event matching in publish/subscribe system. *Neurocomputing*. 2015. vol. 152. pp. 77–84.
16. Khan S., Safyan M. Semantic matching in hierarchical ontologies. *Journal of King Saud University – Computer and Information Sciences*. 2014. Issue. 3. vol. 26. pp. 247–257.
17. Zaytseva A.A., Kuleshov S.V., Mikhailov S.N. [The method for the text quality estimation in the task of analytical monitoring of information resources]. *Trudy SPIIRAN – SPIIRAS Proceedings*. 2014. vol. 37. pp. 144–155. (In Russ.).

Михайлов Сергей Николаевич — к-т техн. наук, доцент, кафедра защиты информации и систем связи Юго-Западного Государственного университета. Область научных интересов: средства диагностирования систем управления, моделирование процессов функционирования средств диагностирования. Число научных публикаций — 170. tk_kursk@mail.ru; ул. 50 лет Октября, 94, Курск, 305040; р.т.: +7 (4712) 57 55 53.

Mikhailov Sergey Nikolaevich — Ph.D., associate professor, department of protection of information and communication systems of the Southwestern State University. Research interests: means of diagnosing control systems, modeling of the functioning of diagnostics tools. The number of publications — 170. tk_kursk@mail.ru; 94, 50 let Oktyabrya, Kursk, 305040, Russia; office phone: +7 (4712) 57 55 53.

Малашенко Олег Иванович — к-т мед. наук, заведующий неврологическим отделением, филиал №1 Федерального казенного государственного учреждения "416 Военный госпиталь" МО РФ. Область научных интересов: неврология, психиатрия. Число научных публикаций — 11. tkkaf@inbox.ru; ул. Бойцов 9-й дивизии, д. 84, кв. 90, Курск, 305003; р.т.: 8 905 041 96 40.

Malashenko Oleg Ivanovich — Ph.D., head of the neurological department, branch №1 Federal state public institution "416 Military Hospital. Research interests: neurology, psychiatry. The number of publications — 11. tkkaf@inbox.ru; 84, Fighters 9th Division street, apt. 90, Kursk, 305003; office phone: 8 905 041 96 40.

Зайцева Александра Алексеевна — к-т техн. наук, старший научный сотрудник, лаборатория автоматизации научных исследований Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН). Область научных интересов: обработка данных, цифровые технологии когнитивного программирования, методы 3D-сканирования и 3D-прототипирования пространственных объектов. Число научных публикаций — 30. cher@iias.spb.su; 14-я линия, д. 39, Санкт-Петербург, 199178; р.т.: +7(812)3235139.

Zaytseva Alexandra Alexeevna — Ph.D., senior researcher, laboratory of research automation of St. Petersburg institute for informatics and automation of Russian academy of sciences (SPIIRAS). Research interests: data processing, 3D rapid prototyping technology. The number of publications — 30. cher@iias.spb.su; 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone: +7(812)3235139.

Поддержка исследований. Работа выполнена при финансовой поддержке РФФИ (проект № 13-07-00137).

Acknowledgements. This research is supported by RFBR (grant 13-07-00137).

РЕФЕРАТ

Михайлов С.Н., Малащенко О.И., Зайцева А.А. **Методика инфологического анализа семантического содержания обращений пациентов для организации электронной записи.**

Одним из путей повышения доступности медицинской помощи выступает организация электронной записи пациентов на прием к требуемым специалистам. Для реализации такой системы в статье предлагается использовать инфологический подход.

В самом общем виде реализация методики обработки обращений пациентов предполагает наличие инфологической системы, ориентированной на хранение и совместную обработку антологий, сформированных специалистами в виде списков возможных жалоб, и перечня жалоб пациента на свое состояние здоровья.

Инфологическая обработка перечней поступающих жалоб конкретного пациента также позволит получить семантическое окружение этих жалоб и сохранить его в виде соответствующего нового XML-файла.

По результатам сравнения полученных отражений семантического содержания имеющихся антологий и поступающих жалоб конкретного пациента принимается решение об их отнесении к области компетенции соответствующего специалиста.

Отличительной особенностью предложенной методики является принятие решения о направлении пациента к соответствующему специалисту не по результатам сравнения значений слов в его жалобе, а на основе анализа семантического окружения симптомов заболевания и его отнесения к области компетенции требуемого специалиста.

Предложенная в настоящей работе методика инфологической обработки обращений пациентов направлена на обеспечение качественно нового уровня доступности медицинских услуг в территориальных учреждениях здравоохранения и позволяет сформировать электронную запись больных на прием к штатным врачам–специалистам.

SUMMARY

Mikhailov S.N., Malashenko O.I., Zaytseva A.A. **The Method for the Infology Analysis of Patients Complains Semantic Content in Order to Organize the Electronic Appointments.**

One way to improve access to health care availability is the organization of electronic patient records to the required specialists. For the realization of such system, the infological approach is proposed.

In the most general form, the realization of the infological processing method for the organization of patients cue presupposes the availability of the infological system oriented on the storage and processing of anthologies, formed by experts in the form of lists of possible complaints and a list of the patient's complaints on state of health.

Infological handling of complaint list of a current patient also allows receiving the reflection of the context of these complaints and to keep it in the form of new XML-file.

By the results of a comparison of available anthologies semantic content and complaint list of a current patient the decision on its assignment to the appropriate specialist areas is made.

The distinctive feature of proposed method is a decision making on which specialist patient should attend based not on words comparison containing in his complaint, but on semantics assessment of revealed symptoms and referring them to corresponding specialist.

The proposed method of infological patient complaints processing is aimed to supply qualitatively new level of medical services availability in territorial health care facilities and allows organizing electronic queue for state specialist attendance.

Д.Н. БИРЮКОВ, А.Г. ЛОМАКО
**ДЕНОТАЦИОННАЯ СЕМАНТИКА КОНТЕКСТОВ ЗНАНИЙ
ПРИ ОНТОЛОГИЧЕСКОМ МОДЕЛИРОВАНИИ ПРЕДМЕТНЫХ
ОБЛАСТЕЙ КОНФЛИКТА**

Бирюков Д.Н., Ломако А.Г. **Денотационная семантика контекстов знаний при онтологическом моделировании предметных областей конфликта.**

Аннотация. Предлагается подход к онтологическому описанию произвольной предметной области, основанный на применении концептов трёх типов: “Объекты”, “Свойства” и “Действия”. При этом различные аспекты представлений, используемые для описания знаний, предлагается частично упорядочивать свойством аппроксимации в функциональные слои, сегменты и области. Это должно позволить моделировать семантические особенности контекстно-зависимых знаний предметных областей, учитывать их изменения и уточнения при порождении решений. Это открывает возможности прогнозирования намерений и предотвращения реализации киберугроз критической информационной инфраструктуре.

Ключевые слова: знания, контекст, онтология, аппроксимация, память, денотационная семантика, конфликт, киберсистема.

Biryukov D.N., Lomako A.G. **Denotational Semantics of Knowledge Contexts in Ontological Modeling of Subject Domains of the Conflict.**

Abstract. An approach to the ontological description of any subject domain based on application of concepts of three types is offered: "Objects", "Properties" and "Actions". Thus, it is proposed to partially arrange various aspects of representations used for the description of knowledge by approximation property in functional layers, segments and areas. This makes possible to model semantic features of context-dependent knowledge of subject domains, to consider their changes and specifications at generation of decisions. Moreover, this opens the possibilities of predicting intentions and preventing realization of cyber threats to critical information infrastructure.

Keywords: knowledge, context, ontology, approximation, memory, denotational semantics, conflict, cybersystem.

1. Введение. Информационные и коммуникационные технологии используются для усиления человеческого интеллекта, дополняя его творческие способности в работе с информацией. Исследователи предсказывают, что по мере развития данной области будет происходить формирование «внешней коры» мозга («экзокортекса»), то есть, системы программ, дополняющих и расширяющих мыслительные процессы человека [1].

Очевидно, что чем более интеллектуальными будут искомые программы, тем большее количество задач в поиске решений сможет переложить на них человек, особенно в области информационной безопасности при защите критической информационной инфраструктуры от атакующих воздействий. Также можно предположить, что интеллектуальность проектируемых и разрабатываемых программ суще-

ственно зависит от их возможности осуществлять смысловую, связанную с контекстом, обработку знаний и данных. Приоритетной является разработка систем, способных осуществлять семантическую обработку поступающих и накопленных данных и знаний и синтезировать в начале возможные, а затем всё более точные и точные, контекстно-зависимые от условий задач, решения. Искомая система должна быть способной синтезировать сценарии упреждающего поведения в конфликте, что должно способствовать прогнозированию намерений и предотвращению кибервоздействий на КИИ.

2. Подход к моделированию контекстно-зависимых онтологий. Учитывая особенности организации данных в семантической памяти человека [2-5], целесообразно представлять знания о предметной области (ПрО) в виде иерархии структурированных объектов, связанных между собой отношениями. На этой идее базируются такие модели представлений, как фреймы, семантические сети, UML и т.д. К сожалению, все эти языки хоть и являются достаточно удобным средством для представления знаний о ПрО конфликта, но лишены возможностей отражения семантики, а выражаемая в них информация предназначена скорее для человеческого, а не машинного восприятия. Также, большое развитие получила логика предикатов (FOL), в которой присутствуют механизмы семантической обработки, но нет удобных средств представления знаний.

В 1985 году Р.Брахман предложил объединить семантические сети и FOL. То, что получилось, было названо терминологической логикой, а далее, в процессе развития, превратилось в семейство дескрипционных логик [6], которые нашли своё практическое применение при описании онтологий в развивающемся в настоящий момент семантическом Web-е.

Онтология – формальная спецификация концептуализации, которая имеет место в некотором контексте предметной области (Gruber, 1993). Онтология описывает основные концепции (положения) предметной области и определяет отношения между ними.

Концептуализация представляет собой описание понятий, а также всю информацию, имеющую отношение к понятиям (свойства, отношения, ограничения, аксиомы, утверждения), необходимую для описания и решения задач в избранной предметной области. Таким образом, онтология состоит из примитивов представления знаний о предметной области (определений основных понятий, таких как имена индивидуумов, классы, функции и другие сущности), а также различного рода семантических связей, поддерживаемых между ними.

Кроме того, под онтологией можно понимать формальное описание результатов концептуального моделирования предметной области, представленное в форме, удобной для восприятия человеком и при реализации в компьютерной системе [7].

Для описания онтологий используются соответствующие языки, например – OWL (Web Ontology Language). OWL обеспечивает три различных по выразительности диалекта (см. рисунок 1) [8]. Применение OWL может позволить распределённое хранение данных в структурированном виде.

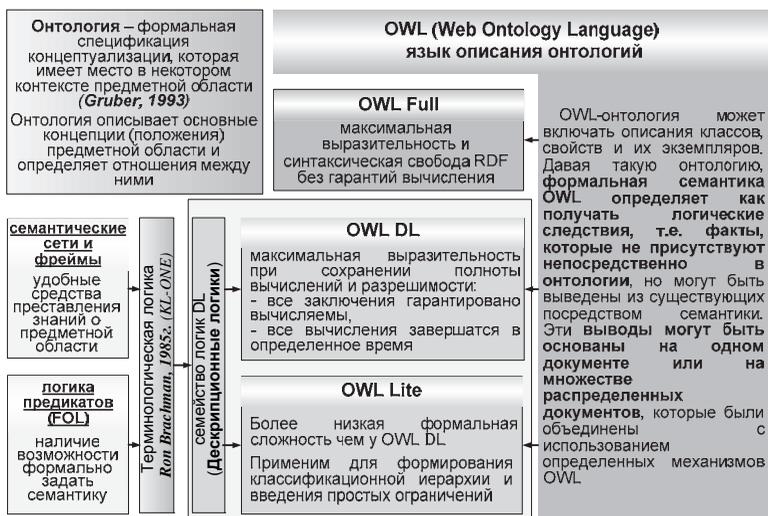


Рис. 1. Схема взаимосвязи семантических сетей, логики предикатов и OWL

На начальном этапе, в ходе построения онтологии, необходимо задать перечень Концептов (унарных предикатов) и Ролей (бинарных предикатов), использование которых позволило бы в дальнейшем описать Про конфликта с учётом возможных контекстов. При задании конкретных ролей, используемых при онтологическом моделировании, описание каждой роли может быть уточнено путём задания определённых характеристик. Основные характеристики (свойства) ролей: симметричность, функциональность, рефлексивность, транзитивность, антисимметричность, обратная функциональность, иррефлексивность. В зависимости от того, какие характеристики введены (используются), может применяться та или иная (с позиций выбранной логики) машина логического вывода.

Следует отдельно отметить, что сам онтологический подход к описанию произвольных ПрО не вносит определённых ограничений на порядок выбора и использования тех или иных типов Концептов и Ролей (кроме требований относительно правильного использования ролей с приведёнными выше характеристиками). Данный факт на практике приводит к тому, что инженеры по знаниям для описания разных ПрО используют большое количество различных Ролей, несущих различную семантику, что в лучшем случае не способствует, а чаще и вообще делает невозможным объединение различных онтологий (онтологий, описывающих различные ПрО и созданные различными специалистами). В тех же случаях, в которых такое объединение возможно, зачастую следует изменять машину логического вывода, используемую для порождения новых знаний над новой (объединённой) онтологией. Наличие же возможности объединения онтологий, описывающих различные ПрО конфликта, просто необходимо для того, чтобы проектируемая интеллектуальная система (ИС) имела потенциальную возможность к обучению, состоящему в переносе знаний о конфликтах и их разрешениях из одной ПрО в другую (например: из “биосферы” и “военного искусства” в “кибер-сферу”).

Для того, чтобы проектируемая система имела возможность синтезировать сценарии упреждающего поведения в конфликтах, она должна иметь возможность представлять и обрабатывать знания о причинах конфликтов, процессах их протекания и следствиях конфликтов, а для этого необходимо, чтобы она обладала возможностью представления знаний об объектах, их свойствах и процессах взаимодействия различных объектов и субъектов. При этом следует учесть, что сами рассматриваемые предметные области не должны вносить ограничений, приводящих к невозможности описания знаний из этих ПрО в единой онтологии. Так как в противном случае, это может привести к принципиальной невозможности обогащения ИС знаниями о поведении в конфликтах, имеющих место в других ПрО.

Достаточно важным вопросом при манипулировании знаниями, представленными посредством единой онтологии в Базе Знаний (БЗ) ИС, является вопрос, связанный с выделением из онтологии тех её фрагментов, знания из которых должны быть доступны системе в тот или иной момент времени при решении тех или иных задач, т.е. при различных контекстах – см. рисунок 2 (на рисунке каждый концепт представляет собой определённый тип действий (А); роль, отображённую пунктирной линией, следует интерпретировать как “является подклассом”, а роль, отображённую сплошной линией, - “следует за”). Если проводить аналогию с механизмами, протекающими в памяти человека, то данную задачу можно переформулировать как задачу управления Фокусом Внимания [9] и задачу помещения части данных в Эпизодический Буфер [10] для их дальнейшей обработки.

С учётом этого можно сформулировать ряд требований, которые должны быть учтены при создании системы метамоделирования, представления и манипулирования знаниями. Система метамоделирования знаний должна быть способной:

- представлять и обрабатывать знания из различных предметных областей об объектах, их свойствах и процессах (для возможности обогащения БЗ описаниями различных конфликтов, которые могут наблюдаться в различных ПрО);

- содержать ограниченное количество типов Концептов и Ролей, достаточное для описания произвольных ПрО (для задания единых правил построения онтологии, способствующих дальнейшему их объединению, а также применению единых логических правил для порождения новых знаний вне зависимости от специфики ПрО и рассматриваемых конфликтов);

- иметь разрешимые алгоритмы направленного поиска данных (для реализации возможности определения необходимого фрагмента онтологии исходя из задач и контекстов);

- осуществлять поиск подобия фрагментов онтологии по аналогии (для осуществления поиска по аналогии подобия возникающих задач и их решений в ходе конфликтов в разных ПрО);

- изменять доступность знаний для их дальнейшей обработки (для упорядочивания получаемых решений, определения наиболее ассоциируемых знаний, а также “забывания” ложных и мало используемых / мало подтверждённых знаний).

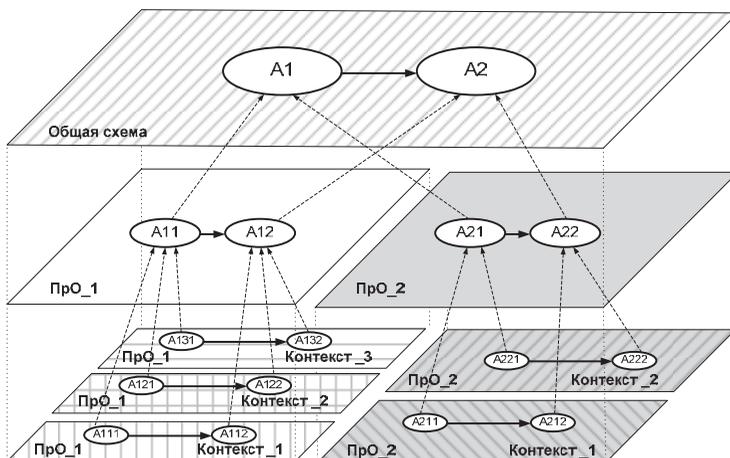


Рис. 2. Пример обобщенного представления контекстно-зависимых знаний предметных областей в единой онтологии

Искомые способности требуют формального определения денотационной семантики контекстов знаний при онтологическом моделировании предметных областей конфликта.

Вопрос, связанный с наделением языков программирования математической семантикой, был поднят ещё в 1971 году Д.Скоттом [11], однако и по сей день, предложенная им математическая теория не развита до уровня, позволяющего применить её для решения конкретных практических задач на должном уровне.

В качестве центрального элемента математической семантики, введённой Д. Скоттом, можно указать на абстрактные функции от элементов того типа данных, который связан с входными переменными, и принимающие значения элементов того типа данных, который связан с выходом. Очевидно, что при переходе к более “абстрактному” описанию решаемых задач на уровне функций и входных/выходных данных на уровне типов, невозможно полностью уйти от операциональных аспектов, поскольку в конечном итоге функции должны быть реализованы и выполняться на вычислительных средствах, способных манипулировать только конечными конфигурациями. В то же время, математическая семантика должна позволить осуществлять операциональное моделирование абстрактных сущностей.

Для того чтобы стало возможным манипулирование абстракциями, необходимо иметь соответствующий математически определённый язык. В работе [11] Д.Скотт предложил заслуживающий внимания взгляд на природу типов данных и функций (отображений) из одного типа данных в другой. Он также указал на то, что в математике недостаточно проработаны вопросы, связанные с описанием функций, определённых на всех допустимых функциях как аргументах и применимых даже к самим себе как к аргументу, и предложил математическую теорию функций, которая может быть использована как проект, дающий “корректный” подход к семантике.

Далее, в работе [12], Д.Скотт с помощью решёток в общих терминах описал абстрактную теорию конечной аппроксимации и бесконечных пределов. Эти понятия он применил для эффективного построения некоторого класса пространств (типов данных), в том числе и функциональных, что позволило использовать их как пространство математических значений в семантических интерпретациях языков программирования высокого уровня.

В статье [13] А.Шамир и У.Уэйдж, опираясь на формализмы, введённые Д.Скоттом, изложили новый подход к семантике типов данных, при котором типы сами включаются как элементы в область объектов. Этот подход позволяет типам иметь подтипы, позволяет рас-

смагивать подлинно полиморфные функции и дает точную семантику для рекурсивных определений типов (включая определения с параметрами). Кроме того, данный подход дает простые и прямые методы для доказательства типовых свойств рекурсивных определений.

Некоторые учёные [14-18] выступали за алгебраическое рассмотрение типов данных, при котором подчёркивался тот факт, что тип данных состоит не только из множества (иногда частично упорядоченного), но и из операций, удовлетворяющих некоторым равенствам.

А.Демерс, Дж.Донахью, Р.Гейтельбаум и Дж.Уильямс также указали [19] на то, что типы данных необходимо рассматривать неразрывно с операциями, реализуемыми на них: “Если два типа отличаются смыслом перечисленных операций, который задается определениями этих типов, то их следует считать различными”.

Таким образом, следует указать на наличие двух точек зрения на сами функции, а именно: экстенциональной и интенциональной точки зрения. При экстенциональном взгляде на функции они рассматриваются как абстрактные объекты, которые полностью определяются своими парами (аргумент, значение), т.е. своими графиками. Т.е. подразумевается выполнимость аксиомы экстенциональности, утверждающей, что если $f(x) = g(x)$ для всех x , то $f = g$.

Этой точки зрения противостоит интенциональная точка зрения, которая представляется очень существенной [20] в контексте вычислимости и программирования, согласно которой функция – это «операция», правило «преобразования» или «вычисления», какое-то определение такого правила.

Следует также отдельно остановиться на теории типов Ч.Хоара [21], в основе которой лежит понятие типа, отличительными особенностями которого являются следующие:

- 1) тип определяет класс значений, которые могут принимать переменная или выражение;
- 2) каждое значение принадлежит одному и только одному типу;
- 3) тип значения константы, переменной или выражения можно вывести либо из контекста, либо из вида самого операнда, не обращаясь к значениям, вычисляемым во время работы программы;
- 4) каждой операции соответствует некоторый фиксированный тип ее операндов и некоторый фиксированный тип результата;
- 5) для каждого типа свойства значений и элементарных операций над значениями задаются с помощью аксиом;
- 6) при работе с языком высокого уровня знание типа позволяет обнаруживать в программе бессмысленные конструкции и решать вопрос о методе представления данных и преобразования их в вычисли-

тельной машине;

7) интересующие нас типы - это типы, хорошо знакомые математикам: прямые произведения, размеченные объединения, множества, функции, последовательности и рекурсивные структуры.

Ряд положений, указанных Ч.Хоаром, спорны (например, положение 2, так как ряд авторов считают, что некоторые типы могут быть подмножествами других типов [22-24] или что объект может принадлежать конечному числу типов [25]). Этот вопрос обсуждается в работах [13, 26, 27] и является частью более общего вопроса о взаимоотношении между типами), однако положение, в котором указывается, что для каждого типа свойства значений и элементарных операций над значениями задаются с помощью аксиом – заслуживает отдельного внимания.

3. Предложения по представлению знаний для интеллектуальной системы. В работе [11] Д.Скотт пришёл к выводу о том, что пространство функций также должно рассматриваться как имеющее тип данных. Так как функция есть, вообще говоря, сама по себе бесконечный объект, то Д.Скотт предложил идею конечного приближения (аппроксимации). Он утверждал [28], что существует общая теория конечной аппроксимации и имеется много типов объектов, которые могут быть получены как пределы аппроксимации.

Математическая теория, предложенная Д.Скоттом, основана на идее, что *отношение аппроксимации* (\sqsubseteq) должно быть *частичным порядком*, и годится не любая структура с частичным порядком, а такая, в которой можно брать пределы.

Запись $x \sqsubseteq y$ означает, что x аппроксимирует y . Это качественное отношение, из которого должно следовать, что x совместимо с y . Можно утверждать, что x хуже, а y лучше (или наоборот), но нельзя сказать, как близко x находится по отношению к y .

В дальнейших рассуждениях принимается, что типы (по меньшей мере) структурированы отношениями \sqsubseteq . В соответствии с интуитивным пониманием следует считать, что отношение \sqsubseteq : рефлексивно ($\forall a(a\sqsubseteq a)$), транзитивно ($\forall a,b,c (a\sqsubseteq b) \wedge (b\sqsubseteq c) \Rightarrow a\sqsubseteq c$) и антисимметрично ($\forall a,b (a\sqsubseteq b) \wedge (b\sqsubseteq a) \Rightarrow a = b$).

Д.Скотт ввёл аксиому, утверждающую, что: “Тип данных есть частично упорядоченное множество”. Фактически можно предполагать, что всё множество элементов ограничено сверху элементом \top , т.е. $x \sqsubseteq \top$ всегда имеет место. Так же можно ввести и объект \perp , для которого всегда выполняется $\perp \sqsubseteq x$. Кроме того, можно предположить, что любые два элемента x и y имеют *наименьшую верхнюю грань*

– н.в.г. $(x \sqcup y)$ и *наибольшую нижнюю грань* – н.н.г. $(x \sqcap y)$. Пример графического отображения частично упорядоченного множества типов данных приведен на рисунке 3.

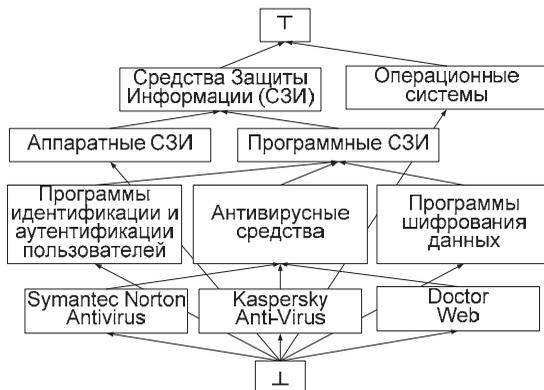


Рис. 3. Пример графического отображения частично упорядоченного множества типов данных

После того, как предположили, что тип данных допускает пределы, необходимо пересмотреть взгляд на функции. Если функция вычислима в каком-либо интуитивном смысле, то для извлечения “конечного” объема информации из какого-либо значения функции, необходимо запросить лишь “конечный” объем информации об аргументах (тут понятие информации скорее качественное, чем количественное). Однако ещё возможно выразить это фундаментальное ограничение на функции: а именно, функции должны сохранять пределы. Отображения же между типами данных *непрерывны*.

Во всей общности это можно точно выразить в терминах направленных множеств. Подмножество $X \subseteq D$ *направлено*, если каждое подмножество X имеет по крайней мере одну верхнюю грань в X . Следует заметить, что направленное множество не пусто. Функция $f : D \rightarrow D'$ непрерывна в том и только том случае, если для любого направленного множества $X \subseteq D$: $f(\sqcup X) = \sqcup \{f(x) : x \in X\}$.

Не все н.в.г. следует рассматривать как пределы, а только н.в.г. направленных множеств. Следует заметить, что понятие непрерывности легко распространяется на функции многих переменных. Для непрерывности функции по всем переменным одновременно достаточно требовать её непрерывности по каждой переменной в отдельности [12].

Таким образом, в более абстрактных терминах по любым двум

полным решёткам D и D' можно образовать функциональное пространство $[D \rightarrow D']$ всех непрерывных функций из первой во вторую.

Далее Д.Скотт [28] предположил, что частичный порядок является *решёткой*, а тип данных – *полной решёткой* по своему частичному порядку (это ещё одна важная аксиома, введённая Д.Скоттом).

Тип данных, удовлетворяющий двум приведённым выше аксиомам, введённым Д.Скоттом, можно рассматривать как топологическое пространство [28], а в подобных пространствах открытое множество содержит вместе с каждым элементом всё, что его аппроксимирует (см. рисунок 4).

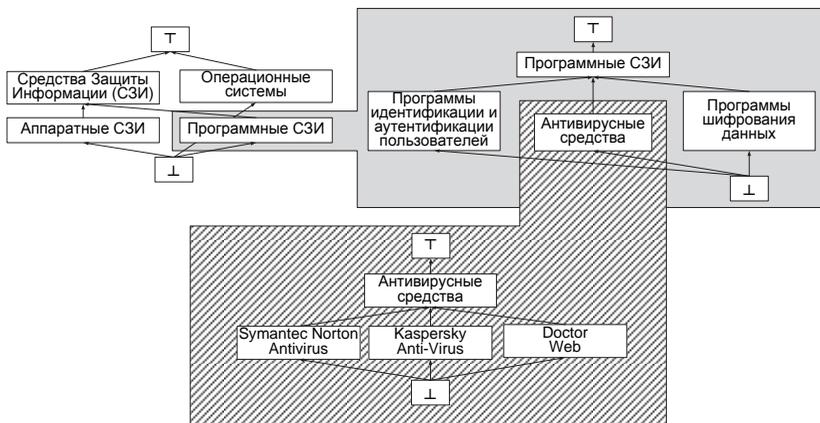


Рис. 4. Пример дуальности Концептов онтологии (Концепт может рассматриваться как объект и как тип)

Данное заключение развито в работе [13], в которой авторы предложили объединить все объекты и типы данных вместе в унифицированную область. Любой элемент полученной таким образом области выступает тогда в двух ролях:

- 1) он является *объектом*, на котором могут быть определены функции, включая функции, являющиеся наименьшими неподвижными точками рекурсивных определений;
- 2) он является *типом* всех объектов, которые аппроксимируют его; следовательно, утверждения $x \sqsubseteq y$ “ x имеет тип y ” и “любой объект типа x является объектом типа y ” эквивалентны.

По произвольной исходной области “ O ” объектов и совокупности интуитивных типов данных можно образовать новую область “ \hat{O} ” (которую называют типовым расширением области “ O ”), добавив типы как

новые объекты в смысле (1). Расширенное отношение \sqsubseteq на “ \hat{O} ” определяется эквивалентностью из (2): тип-объект помещается над объектами этого типа и над его подтипами (так что \sqsubseteq одновременно упорядочивает типы по включению и объекты по аппроксимации).

Важным понятием при описании любой ПрО, является понятие самой области. Под *областью* подразумевается [13] частично упорядоченное множество D , такое, что D имеет наименьший элемент и любое направленное множество элементов D имеет н.в.г. (такие области принято называть полными частичными порядками).

Под *типом данных* над D подразумевается подмножество x универсума D , такое, что x :

(1) замкнуто вниз, т. е. если d_0 и d_1 лежат в D и если $d_0 \sqsubseteq d_1$, то $d_1 \in x$ влечет $d_0 \in x$;

(2) замкнуто относительно взятия верхних граней, т. е. если s есть направленное подмножество x , то $\sqcup s \in x$.

Множества с этими свойствами называются также *идеалами*.

С каждым элементом d из D ассоциируется тип \tilde{d} , определяемый как: $\tilde{d} = \{d' \mid d' \sqsubseteq d\}$, т.е. \tilde{d} есть множество всех элементов, аппроксимирующих d .

Расширенная область \hat{D} образуется (фактически) добавлением к D некоторой совокупности типов.

Под *структурой типов* над D подразумевают совокупность T типов над D , удовлетворяющую следующим условиям [13]:

(1) $\tilde{d} \in T$ для всех $d \in D$;

(2) универсальный тип U (множество всех элементов из D) принадлежит T ;

(3) теоретико-множественное пересечение типов из любой непустой подколлекции, взятой из T , снова лежит в T .

Область \hat{D} – это множество T , упорядоченное по теоретико-множественному включению. Таким образом, \hat{D} содержит изоморфную копию D и поэтому может рассматриваться как расширение D .

При построении структуры типов над D может оказаться необходимым добавить в T некоторые множества с одной лишь целью – пополнить структуру так, чтобы получить необходимые пересечения (см. рисунок 5). Наличие самой структуры типов, представленной в виде решётки, позволяет говорить о возможности нахождения классов эквивалентности (о наличии в некотором роде эквивалентных типов данных, представляющих понятия из различных ПрО).

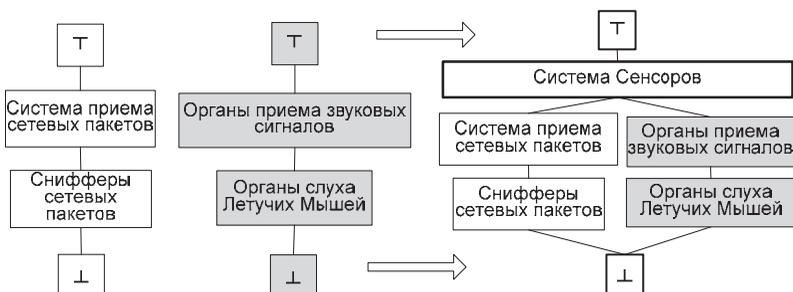


Рис. 5. Пример построения структуры типов, путём внесения множества “Система сенсоров”

При моделировании семантических вычислений весьма перспективным видится использование функциональных типов. Д.Скотт считал [29] очень важной также ту часть своих исследований, в которой он рассматривал функции, в качестве аргументов которых выступают также функции, т.е. функции высших порядков. Именно Д.Скотт одним из первых предложил в качестве семантической структуры использовать *функциональное пространство*, но многие тогда не понимали его, а ведь именно функциональные пространства должны учитываться при выборе и реализации операторов (исходя из семантики моделируемых процессов), осуществляющих отображения на функциональных пространствах. Такие операторы есть не что иное, как комбинаторы – в терминах Х.Карри и А.Чёрча (или – функциональные формы у Дж.Бэкуса).

Для упрощения восприятия и использования функциональных типов на практике, полезно иметь возможность явного задания их частично упорядоченного множества в виде Концептов соответствующего типа, а также отношений между ними (см. рисунок 6). Это должно облегчить процесс онтологического моделирования предметных областей конфликта.



Рис. 6. Пример явного задания функциональных типов и отношений между ними

Исходя из результатов проведённого анализа можно предположить, что любой тип данных, используемый при онтологическом моделировании предметных областей конфликта, можно описать через перечень (1) *объектов*, которые ему соответствуют; (2) *свойств* этих

объектов и функций (а для общего случая – (3) *действий*), которые могут рассматриваемые объекты выполнять [30].

Если рассматривать абстрактные классы объектов “*O*”, свойств “*P*” и действий “*A*”, то их в совокупности можно представить в виде решётки, содержащей один нижний, один верхний элемент и три дискретно разделённых элемента, находящихся на одной “горизонтالي”.

Следует отметить, что если учесть тот факт, что объект может состоять из объектов более низкой иерархии (более простых объектов) и быть частью объектов более высокой иерархии (структурно более сложных объектов), то в общем случае можно говорить о структурах, а не об объектах. Однако далее для простоты изложения предлагается использовать термин “объекты”, тем самым обозначая некий “срез наблюдения” в условиях определённых контекстов.

Таким образом, пусть: “*O*” – решётка “Объектов”, “*P*” – решётка “Свойств”, “*A*” – решётка “Действий”.

Исходя из анализа порядка интеллектуальной деятельности человека можно предположить, что поиск решений осуществляется в плоскости “Свойств” объектов.

При построении решётки “Свойства” предлагается воспользоваться отношением “Аппроксимирует” (“Является подклассом”).

$F1_1: [P \rightarrow P]$ Свойство аппроксимирует Свойство (Свойство является подклассом Свойства)

Пример такого построения приведён на рисунке 7.



Рис. 7. Пример отношения Аппроксимирует (“Является подклассом”)

Для возможности иерархического представления данных об объектах и их возможных действиях предлагается ввести отображения $F1_2$ и $F1_3$:

$F1_2: [O \rightarrow O]$ Объект аппроксимирует Объект

$F1_3: [A \rightarrow A]$ Действие аппроксимирует Действие

Очевидно, что об объектах судят через их свойства, которые следует рассматривать как атрибуты объектов. Следовательно, можно определить отображение $F2$:

$F2: [O \rightarrow P]$ Объект обладает Свойством

Следует заметить, что разбиение “Объектов” на подклассы может осуществляться различными способами (в зависимости от того, какое свойство выбирается за признак классификации). Возможность различной классификации свойств (а следовательно объектов и действий) порождает возможность стратификации знаний.

Реализация решений осуществляется через выполнение “Действий”, суть которых может сводиться к изменению объектов (или к измерению и/или оцениванию их свойств).

Для того, чтобы взаимодействие между двумя объектами стало потенциально возможным, они должны обладать соответствующими свойствами. Иными словами, наличие у объекта определённого свойства порождает способность к выполнению определённого действия (определённого класса действий). Само же действие, осуществляемое первым объектом, может быть пригодно для воздействия на определённое свойство второго объекта. Таким образом, можно определить ещё две функции-стрелки на решётках “Свойства” и “Действия”:

F3: $[P \rightarrow A]$ Свойство порождает способность к Действию

F4: $[A \rightarrow P]$ Действие пригодно для воздействия на Свойство

Множества данных функциональных типов (**F3** и **F4**) также образуют решётки.

Взаимодействие двух объектов (O_1 и O_2) можно формально описать следующим образом: $[[O_1 \rightarrow P_1] \rightarrow A_1] \rightarrow [O_2 \rightarrow P_2]$. Что означает: “Объект O_1 , обладающий свойством P_1 , осуществляет воздействие A_1 на объект O_2 , поскольку последний обладает свойством P_2 ”.

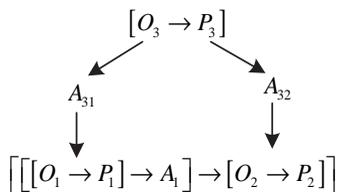
Приведённая конструкция, позволяющая строить функциональные пространства, весьма важна, так как её анализ (как инженером по знаниям, так и самой ИС) позволяет ответить на вопросы:

- “кто (что) взаимодействует” – (O_1 и O_2),
- “как взаимодействует” – (A_1),
- “почему взаимодействие возможно” – ($O_1 \rightarrow P_1$ и $O_2 \rightarrow P_2$).

Так как O_1 , O_2 , P_1 , P_2 и A_1 являются элементами соответствующих частично упорядоченных множеств (“ O ”, “ P ” и “ A ”), то это даёт предпосылки к возможности системного рассмотрения описываемого процесса “взаимодействия” и, в последующем, индуктивного и дедуктивного вывода “подобных” процессов.

В подавляющем же большинстве случаев на практике при построении онтологий используют конструкции типа: $O_1 \xrightarrow{A_1} O_2$, где O_1 и O_2 – Концепты, описывающие взаимодействующие объекты, а A_1 – Роль, носящая семантически окрашенное имя. Применение подобных конструкций не всегда приемлемо, так как в ряде случаев для интеллектуальной системы, обрабатывающей знания, представленные в онтологии, теряется семантика Роли. Если же Роль представлять конструкцией типа: $P_1 \rightarrow A_1 \rightarrow P_2$, то данное построение позволяет учитывать дополнительный контекст, извлекаемый из местоположения P_1 , P_2 и A_1 в соответствующих решётках.

В 1960-х годах Д.Маккарти [31] было предложено исчисление ситуаций, целью которого было найти способ описания результатов действий (операций) вне зависимости от проблемной области. Основной формализм ситуационного исчисления – это выражение вида $s' = result(e, s)$, где s' – ситуация, возникающая, когда происходит событие e в ситуации s . Иными словами, был обозначен вопрос, связанный с необходимостью формализовать понятие контекста. Очевидно, что проектируемая ИС просто обязана принимать те или иные решения с учётом контекста, так как именно контекст оказывает влияние на оценку значения измеряемых киберсистемой свойств. Тогда взаимодействие двух объектов в условиях определённого контекста можно в общем виде формализовано представить следующим образом:



где $[O_3 \rightarrow P_3]$ – обобщённое описание контекста (которое при необходимости можно “развернуть” в сколь угодно сложное взаимодействие объектов окружающей систему действительности), а A_{31} и A_{32} – оказываемые воздействия со стороны окружающей действительности (контекста) на первый и второй объекты соответственно.

Для того, чтобы стало возможным описывать (выводить) процессы, необходимо ввести отображение на решётке действий:

F5: $[A \rightarrow A]$ Действие следует за Действием (апликативное применение)

Бесспорно, важным элементом в познании является декомпозиция, т.о., предлагается ввести отображение на решётке объектов:

F6: $[O \rightarrow O]$ Объект состоит из Объекта

Полезно также ввести подобное отображение над решётками “Действия”, что при необходимости позволит осуществить декомпозицию рассматриваемого процесса на составляющие его действия:

F7: $[A \rightarrow A]$ Действие состоит из Действия

Основные концепты и роли (отображения между ними), используемые для построения Базы Знаний проектируемой интеллектуальной системы, можно представить в виде единой схемы: см. рисунок 8. Предложенная модель представления знаний (см. рисунок 8) может быть использована для организации семантической памяти [2] Гиromата [32], в которой будет храниться онтология, построенная им.

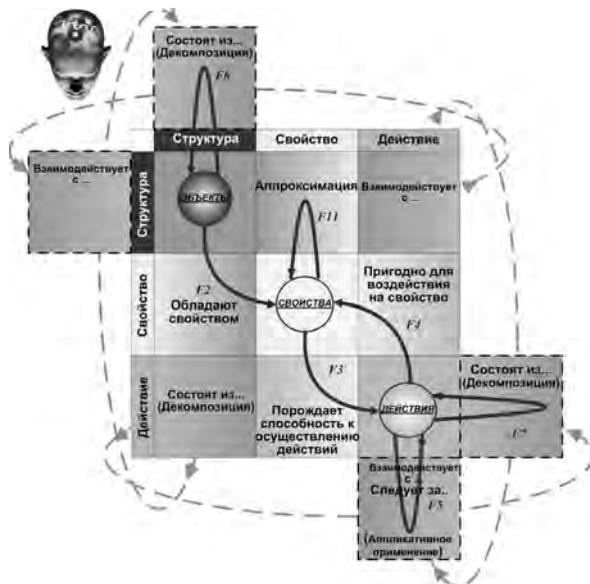


Рис. 8. Основные Концепты и Роли, которые предлагается использовать для построения Базы Знаний (семантическая память)

Перечень предложенных отображений при необходимости может быть расширен. Так, например, можно дополнительно ввести Роль между двумя объектами – “Продуцирует”. Применение указанной Ро-

ли позволит более просто описывать переход объекта из одного состояния в другое под воздействием второго объекта. Грамотное использование подобных отображений (Ролей) позволит обогатить БЗ интеллектуальной системы, что может привести к облегчению осуществления семантического вывода на имеющихся знаниях и к упрощению интерпретации полученных результатов.

Следует заметить, что предложенный порядок структурной организации данных в памяти Гиромата, соответствует порядку организации и обработки данных человеком, так как доказано, что он отдельно хранит данные (1) *о событиях и последовательностях событий* (сценариях) и (2) *об объектах и их свойствах* (каркасах). Также доказано [4], что в семантической памяти информация об объектах организована на основе различий между сенсорными или зрительными образами и функциональными свойствами, что вполне соответствует предложенному подходу по построению решёток “Объекты” и “Действия” через построение решётки “Свойства”.

4. Семантические звенья и иерархическое упорядочение концептов в онтологии. В конечном итоге знания, представленные в онтологии, предлагается хранить в виде совокупности взаимосвязанных “Семантических Звеньев” (СЗ – два Концепта, связанных Ролью), типы которых представлены выше. Однако, для того чтобы проектируемую систему можно было считать интеллектуальной, она должна быть способной не только структурировано хранить данные, но и “логически мыслить”, а также выполнять сложные когнитивные действия с целью порождения новых знаний, не закладываемых в неё непосредственно при создании. Для этого ИС должна быть способной выполнять рассуждения. При этом полезно, чтобы при необходимости эксперт, работающих с ИС, мог проверить порядок проведённых ею рассуждений.

Рассуждением называется процедура обоснования некоторого высказывания путем пошагового выведения его из других высказываний. Простейшим видом рассуждения является умозаключение (умозаключение – это непосредственный переход от одного или нескольких высказываний $A_1 A_2, \dots, A_n$ к некоторому высказыванию B). Любое высказывание, исходя из предложенной модели представления знаний [32], можно рассматривать как факт, представленный в Базе Знаний интеллектуальной системы в виде СЗ. При этом все Концепты, входящие в различные СЗ иерархически упорядочены и, как было указано выше, являются элементами частично упорядоченных множеств.

Иерархическое упорядочивание Концептов является результатом классификации, которую осуществляют при пополнении онтологии ИС. Под классификацией понимается результат последовательного деления некоторого понятия на его виды, видов на подвиды и т. д. Любая классификация может быть представлена в форме дерева понятий. “Понятия” используются при классификации объектов и явлений, ок-

ружающих ИС в ходе её функционирования. Значимость же построенных в результате классификации иерархий понятий весьма существенна как для представления знаний в памяти ИС и последующего взаимодействия с экспертами и инженерами по знаниям, так и для осуществления умозаключений самой системой. Поэтому важно указать (напомнить) лицам, заносщим в онтологию ИС новые данные, на то, каким образом следует осуществлять классификацию. В то же время следует отметить, что весьма полезно стремление к тому, чтобы ИС была способной манипулировать терминами, близкими к терминам естественного языка, так как это должно значительно облегчить процедуру человеко-машинного взаимодействия.

Термины естественного языка имеют две важнейшие характеристики: *значение* (предмет, знаком которого является термин – экстенционал) и *смысл* (информация о значении термина, которую содержит сам термин или которая связывается с ним – интенционал).

Понимать термин – значит знать, какие именно предметы попадают под него, то есть по любому предъявленному предмету уметь решить вопрос, можно ли данный предмет обозначить данным термином.

Понятие есть мысль, которая посредством указания на некоторый признак выделяет из универсума и собирает в класс (обобщает) предметы, обладающие этим признаком.

Одним из атрибутов интеллектуальной обработки информации является возможность осуществлять манипулирование не только терминами, но и пониманиями терминов.

Пусть U - универсум (*род*), тогда с синтаксической точки зрения понятия можно выражать конструкцией вида: $\alpha A(\alpha)$, которая читается как: “предмет α из универсума U , такой, что α обладает признаком $A(\alpha)$ ”. Признак $A(\alpha)$ в этом случае является *видовым отличием*. Таким образом, всякое понятие выделяет в универсуме (роде) U те и только те предметы, которые обладают видовым отличием $A(\alpha)$.

С семантической точки зрения каждое понятие обладает двумя важнейшими характеристиками – *содержанием* и *объемом*.

Содержанием понятия, выраженного универсалией $\alpha A(\alpha)$, называется признак $A(\alpha)$, на основании которого обобщаются и выделяются предметы в данном понятии.

Объемом понятия, выраженного универсалией $\alpha A(\alpha)$, называется класс всех тех предметов из универсума, которые обладают признаком $A(\alpha)$.

Объем понятия является экстенсиональной характеристикой понятия и его представляют выражением $W\alpha A(\alpha)$, которое читается: “множество тех предметов α из универсума U , для которых верно

$A(\alpha)$ ”. Те предметы, которые входят в объем понятий, называют *элементами* их объемов.

Процесс образования понятия естественно описывается в терминах гомоморфизма. Разбивая интересующее множество объектов на классы «эквивалентных» в каком-либо отношении элементов (то есть, игнорируя все различия между элементами одного класса, не интересующие исследователя в данный момент), можно получить новое множество, гомоморфное исходному (т.н. фактормножество), по выделенному отношению эквивалентности. Элементы этого нового множества (классы эквивалентности) можно воспринимать теперь как единые, нерасчленимые объекты, полученные в результате «склеивания» всех неразличимых в фиксированных отношениях исходных объектов в один «комок». Эти «комки» отождествлённых между собой образов исходных объектов и есть то, что называют понятиями, полученными в результате мысленной замены класса близких между собой представлений одним «родовым» понятием. Понятия в онтологии как раз и представляются отдельными Концептами.

Операции над понятиями бывают двух типов: операции с объемами понятий и операции с их содержаниями [33]. Взаимосвязь между объемными характеристиками понятий и их содержательными характеристиками явно отражается в *законе обратного отношения между объемами и содержаниями понятий*: “Объем понятия $\alpha A(\alpha)$ составляет часть объема понятия $\alpha B(\alpha)$, если и только если содержание $\alpha B(\alpha)$ является частью содержания $\alpha A(\alpha)$ ”.

Следует отметить, что сравнить объемы рассматриваемых понятий можно лишь при наличии некоторого дополнительного знания, описывающего область предметов U .

Учитывая указанное замечание можно более точно сформулировать закон обратного отношения, приведённый выше: $W\alpha A(\alpha) \subseteq W\alpha B(\alpha)$, если и только если $T \vdash \forall \alpha(A(\alpha) \supset B(\alpha))$ и $T \vdash \forall \alpha(B(\alpha) \supset A(\alpha))$, где левая часть записи говорит о том, что класс реально существующих предметов, образующих объем понятия $\alpha A(\alpha)$, составляет часть объема $\alpha B(\alpha)$, а правая часть записи означает, что утверждение о соответствующем отношении содержаний данных понятий имеет место в теории T (здесь под теорией следует понимать определённую “модель мира”, конкретную “предметную область”, которую ещё называют “интерпретацией”).

Над понятиями можно осуществлять целый ряд операций, необходимых для реализации “навигации” по частично упорядоченному множеству понятий (Концептов онтологии), одной из которых является *операция ограничения понятий*.

Ограничить понятие $\alpha B(\alpha)$ - это значит указать понятие $\alpha A(\alpha)$, такое что для объемов A и B будет справедливо отношение $A \subset B$. Операция ограничения $\alpha B(\alpha)$, таким образом, состоит в переходе к *видовому* понятию $\alpha A(\alpha)$. Само $\alpha B(\alpha)$ при этом считается *родовым*. Для непустых понятий пределом их ограничения считается единичное понятие.

Операцию ограничения объема понятия нельзя путать с операцией *членения* предмета. Первая является действием с понятиями, т. е. с мыслительными конструкциями, вторая же осуществляется с элементами понятий, т. е. с самими предметами.

Осуществить операцию *обобщения* понятия $\alpha A(\alpha)$ – это значит указать понятие $\alpha B(\alpha)$ такое, что будет верно отношение $A \subset B$. Процедура обобщения, таким образом, состоит в переходе от видового понятия к родовому (т.е. вверх по частично упорядоченному множеству Концептов, представленных в онтологии [32])

Пределом обобщения является универсальное понятие, т. е. некоторое понятие $\alpha D(\alpha)$ такое, что $W\alpha D(\alpha) = U$ (т.е. $\alpha D(\alpha)$ является верхом решётки).

Еще одной операцией является операция *деления* понятий. Под делением непустого понятия $\alpha B(\alpha)$ понимают построение системы понятий $S = \{\alpha A_1(\alpha), \alpha A_2(\alpha), \dots, \alpha A_n(\alpha)\}$ на основании какой-либо характеристики Ψ элементов этого понятия. При осуществлении операции деления используется следующая терминология: $\alpha B(\alpha)$ называется *делимым понятием*; понятия, входящие в систему S , называются *членами деления*, а характеристика Ψ - *основанием деления*.

Деление считается правильным, если:

1) деление осуществляется по одному основанию, т. е. в качестве Ψ используется в точности одна (простая или сложная) характеристика предметов;

2) $\forall i (W\alpha A_i(\alpha) \subset W\alpha B(\alpha))$ – члены деления, входящие в S , являются видами по отношению к понятию $\alpha B(\alpha)$;

3) $\forall i (W\alpha A_i(\alpha) \neq \emptyset)$ – члены деления не пусты;

4) $\forall i \forall j (i \neq j \supset W\alpha A_i(\alpha) \cap W\alpha A_j(\alpha) = \emptyset)$ – члены деления попарно несовместимы;

5) $W\alpha A_1(\alpha) \cup W\alpha A_2(\alpha) \cup \dots \cup W\alpha A_n(\alpha) = W\alpha B(\alpha)$ – объединение объемов всех членов деления из S совпадает с объемом $\alpha B(\alpha)$.

Невыполнение лицами, вносящими данные в ИС, какого-либо из

выше указанных требований может привести к неправильному делению и неверному составлению классификаций (построению частично упорядоченных множеств Концептов), что в свою очередь может сказаться на самой возможности “правдоподобных” умозаключений.

5. Заключение. Для того чтобы проектируемая киберсистема могла стать поистине интеллектуальной, она должна быть способной структурировано хранить данные, знания и их контекстуальную привязку к задачам и решениям, а также выполнять сложные когнитивные действия (“логически мыслить”) с целью порождения новых знаний, не закладываемых в неё непосредственно, а приобретённых ею в процессах постановки и решения возможных задач. Предложенный подход к онтологическому представлению данных в памяти интеллектуальной системы, основанный на теории типов и теории конечной аппроксимации, открывает возможность формального представления контекстно-зависимых многослойных знаний о предметных областях, потенциально содержащих знания о стратегиях поведения конфликтующих сторон. Данный подход должен позволить в дальнейшем организовать корректную процедуру осуществления правдоподобных умозаключений над контекстно-зависимыми знаниями, представленными в памяти систем предотвращения киберугроз [34].

Литература

1. *Прайд В., Медведев Д.А.* Феномен NBIC-конвергенции: Реальность и ожидания // URL: <http://www.nanonewsnet.ru/articles/2010/fenomen-nbic-konvergensii-realnost-ozhidaniya> (дата обращения: 20.12.2014).
2. *Tulving E.* Episodic and semantic memory. Organization of Memory // New York: Academic Press, 1972. pp. 381–403.
3. *Martin A., Chao L.L.* Semantic memory and the brain: Structure and Processes // Current Opinion in Neurobiology. 2001. vol. 11, pp. 194–201.
4. *Marques J.F., Canessa N., Siri S., Catricala E., Cappa S.* Conceptual knowledge in the brain: fMRI evidence for a featural organization // Brain Research. 2008. vol. 1194. pp. 90–99.
5. *Collins A.M., Quillian M.R.* Retrieval time from semantic memory // Journal of Verbal Learning and Verbal Behavior. 1969. vol. 8. pp. 240–247.
6. About formal bases of OWL // URL: http://semanticfuture.net/index.php/O_формальных_основах_OWL (дата обращения: 20.12.2014).
7. Ontology // URL: <http://www.aiportal.ru/articles/other/ontology.html> (дата обращения: 20.12.2014).
8. OWL, язык веб-онтологий. Краткий обзор. Рекомендация W3C от 10 февраля 2004 г. // URL: <http://www.thalion.kiev.ua/idx.php/7/009/article/#s4> (дата обращения: 20.12.2014).
9. *Norman D.A., Shallice T.* Attention to action: Willed and automatic control of behavior // Consciousness and Self-regulation. Advances in Research and Theory. New York: Plenum Press. 1986. vol. 4. pp. 1–18.
10. *Badddeley A.D.* The episodic buffer: A new component of working memory? // Trends in Cognitive Sciences, 2000. vol. 4(11). pp. 417–423.
11. *Scott D. S.* Models for various type-free calculi // Logic, Methodology and Philosophy of Science IV (Proc. Int. Congress 1971), North-Holland. 1973. pp. 157–188.

12. *Scott D.S.* Outline of mathematical theory // 4th Annual Princeton Conf. on Information Sciences and Systems, Princeton University. 1970. pp. 169–176.
13. *Shamir A.* Data types as objects // Springer Berlin Heidelberg. 1977. pp. 465–479.
14. *Burstall R.* Programs and their proofs: an algebraic approach // Machine Intelligence. Edinburgh Univ. Press, 1969. no. 4.
15. *Burstall R.* The algebraic theory of recursive program schemes // Category Theory Applied to Computation and Control. 1974. no. 25. pp. 126–131.
16. *Goguen J.* Abstract data types as initial algebras and the correctness of data representation // Current Trends in Progr. Methodology, IV. Data Structuring. Prentice-Hall. 1978. pp. 80–144.
17. *Gutttag J.* Abstract data types and software validation // Communications of the ACM. 1978. vol. 21(12). pp. 1048–1064.
18. *Zilles S.* Algebraic specifications for data types // IBM Research Laboratory, San Jose, California, 1975.
19. *Demers A.J.* Incapsulated Data Types and Generic Procedures // Design and Implementation of Programming Languages, LNCS 1977. no. 54. pp. 174–214.
20. *Church. A.* The calculi of lambda-conversion // Annals of Math. Studies. 1951. no. 6.
21. *Hoare C.A.R.* Notes on data structuring // Structured Programming. Academic Press. 1972. pp. 98–197.
22. *Parnas D.L.* Abstract types defined as classes of variables // Proceedings Conference on Data: Abstraction, Definition, and Structure, Salt Lake City. 1976. pp. 149–154.
23. *Kaplan M.A.* A general scheme for the automatic inference of variable types // Proceedings of the 5th ACM SIGACT-SIGPLAN symposium on Principles of programming languages. 1978. pp. 60–75.
24. *Лаэров С.С.* Основные понятия и конструкции языков программирования // М.: Статистика. 1982. 80 с.
25. *Ledgard H. F.* Ten mini-languages: a study of topical issues in programming languages // Corp. Surveys. 1971. no. 3. pp. 115–147.
26. *Lewis C.H.* Recursively defined data types // Proceedings of the 1st annual ACM SIGACT-SIGPLAN symposium on Principles of programming languages. 1973. pp. 125–138.
27. *Berry D.M.* Type equivalence in strongly typed languages: one more look // SIGPLAN Notices, 1979. no. 9. pp. 35–41.
28. *Scott D.S.* Lattice Theory. Data Types and Semantics // Formal Semantics of Programming Languages. Prentice-Hall, Englewood Cliffs, N.J. 1972.
29. *Скотт Д.С.* Логика и языки программирования // Лекции лауреатов премии Тьюринга (ред.: Эшенхерст Р.). М.: Мир, 1993. С. 65–83.
30. *Бирюков Д.Н., Ломако А.Г.* Формализация семантики для представления знаний о поведении конфликтующих сторон // Материалы 22-й научно-практической конференции “Методы и технические средства обеспечения безопасности информации”. Спб.: Издательство Политехнического университета, 2013. С. 8–11.
31. *McCarthy J., Hayes P.J.* Some philosophical problems from the standpoint of artificial intelligence // Machine Intelligence, American Elsevier, New York, 1969. no. 4.
32. *Бирюков Д. Н., Ломако А. Г.* Подход к построению ИБ-систем, способных синтезировать сценарии упрещающего поведения в информационном конфликте // INSIDE. 2014. №6. С. 42–50.
33. *Бочаров В.А., Маркин В.И.* Основы логики // М.: МГУ, 2008. 336с.
34. *Бирюков Д. Н., Ломако А. Г.* Подход к построению системы предотвращения киберугроз // Проблемы информационной безопасности. Компьютерные системы. 2013. №2. С. 13–19.

References

1. Pride V., Medvedev D.A. [NBIC convergence phenomenon: Reality and expectations]. Available at: <http://www.nanonewsnet.ru/articles/2010/fenomen-nbic-konvergensii-realnost-ozhidaniya> (accessed 20.12.2014).
2. Tulving E. Episodic and semantic memory. *Organization of Memory*. New York: Academic Press. 1972. pp. 381–403.
3. Martin A., Chao L.L. Semantic memory and the brain: Structure and Processes. *Current Opinion in Neurobiology*, 2001. vol. 11, pp. 194–201.
4. Marques J.F., Canessa N., Siri S., Catricala E., & Cappa S. Conceptual knowledge in the brain: fMRI evidence for a featural organization. *Brain Research*. 2008. vol. 1194. pp. 90–99.
5. Collins A.M., Quillian M.R. Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*. 1969. vol. 8. pp. 240–247.
6. About formal bases of OWL. Available at: <http://semanticfuture.net/index.php> (accessed 20.12.2014).
7. Ontology. Available at: <http://www.aiportal.ru/articles/other/ontology.html> (accessed 20.12.2014).
8. OWL, language of web ontologies. Short review. Recommendation of W3C. 2004. Available at: <http://www.thalion.kiev.ua/idx.php/7/009/article/#s4> (accessed 20.12.2014).
9. Norman D.A., Shallice T. Attention to action: Willed and automatic control of behavior. *Consciousness and Self-regulation. Advances in Research and Theory*. New York: Plenum Press. 1986. vol. 4. pp. 1–18.
10. Baddeley A.D. The episodic buffer: A new component of working memory?. *Trends in Cognitive Sciences*, 2000. vol. 4(11). pp. 417–423.
11. Scott D. S. Models for various type-free calculi. *Logic, Methodology and Philosophy of Science IV* (Proc. Int. Congress 1971), North-Holland. 1973. pp. 157–188.
12. Scott D.S. Outline of mathematical theory. 4th Annual Princeton Conf. on Information Sciences and Systems, Princeton University. 1970. pp. 169–176.
13. Shamir A. Data types as objects. Springer Berlin Heidelberg. 1977. p. 465–479.
14. Burstall R. Programs and their proofs: an algebraic approach. *Machine Intelligence*. Edinburgh Univ. Press, 1969. no. 4.
15. Burstall R. The algebraic theory of recursive program schemes. *Category Theory Applied to Computation and Control*. 1975. no. 25. pp. 126–131.
16. Goguen J. Abstract data types as initial algebras and the correctness of data representation. *Current Trends in Progr. Methodology, IV. Data Structuring*. Prentice-Hall. 1978. pp. 80–144.
17. Guttag J. Abstract data types and software validation. *Communications of the ACM*. 1978. vol. 21(12). pp. 1048–1064.
18. Zilles S. Algebraic specifications for data types. IBM Research Laboratory, San Jose, California, 1975.
19. Demers A.J. Incapsulated Data Types and Generic Procedures. *Design and Implementation of Programming Languages*. LNCS 1977. no. 54. pp. 174–214.
20. Church A. The calculi of lambda-conversion. *Annals of Math. Studies*. 1951. no. 6.
21. Hoare C.A.R. Notes on data structuring. *Structured Programming*. Academic Press. 1972. pp. 98–197.
22. Parnas D.L. Abstract types defined as classes of variables. *Proceedings Conference on Data: Abstraction, Definition, and Structure*, Salt Lake City. 1976. pp. 149–154.
23. Kaplan M.A. A general scheme for the automatic inference of variable types. *Proceedings of the 5th ACM SIGACT-SIGPLAN symposium on Principles of programming languages*. 1978. pp. 60–75.
24. Lavrov S.S. *Osnovnye ponjatiya i konstrukcii jazykov programmirovaniya* [Basic concepts and designs of programming languages] M.: Statistica. 1981. (In Russ).

25. Ledgard H. F. Ten mini-languages: a study of topical issues in programming languages. *Corp. Surveys*. 1971. no. 3. pp. 115–147.
26. Lewis C.H. Recursively defined data types. Proceedings of the 1st annual ACM SIGACT-SIGPLAN symposium on Principles of programming languages. 1973. pp. 125–138.
27. Berry D.M. Type equivalence in strongly typed languages: one more look. *SIGPLAN Notices*, 1979. no. 9. pp. 35–41.
28. Scott D.S. Lattice Theory. Data Types and Semantics. Formal Semantics of Programming Languages. Prentice-Hall, Englewood Cliffs, N.J. 1972.
29. Scott D.S. [Logic and programming languages]. *Lektsii laureatov premii T'yuringa – Lectures of winners of an award of Turing*. M.: Mir. 1993. pp. 65–83. (In Russ).
30. Biryukov D.N., Lomako A.G. [Formalization of semantics for representation of knowledge of behavior of conflicting parties]. *Materialy 22 nauchno-prakticheskoy konferentsii "Metody i tehnicheskiye sredstva obespecheniya bezopasnosti informacii"* [Materials of the 22nd scientific and practical conference "Methods and Technical Means of Safety of Information". SPB: St. Petersburg Polytechnical University]. 2013. pp. 8–11. (In Russ).
31. McCarthy J. Some philosophical problems from the standpoint of artificial. *Machine Intelligence*, American Elsevier, New York, 1969. no. 4.
32. Biryukov D.N., Lomako A.G. [Approach to creation of the IB-systems capable to synthesize scenarios of anticipatory behavior in the information conflict]. *INSIDE*. 2014. no. 6. pp. 42–50. (In Russ).
33. Bocharov V.A., Markin V.I. *Osnovy logiki* [Logic bases]. M.: MSU. 2008. 336 p.
34. Biryukov D.N., Lomako A.G. [Approach to creation of system of cyber-threats preventing]. *Problemy informatsionnoy bezopasnosti. Kompyuternie sistemy – Problems of information security. Computer systems*. SPB: St. Petersburg Polytechnical University. 2013. no. 2. pp. 13–19. (In Russ).

Бирюков Денис Николаевич — к-т техн. наук, профессор кафедры систем сбора и обработки информации Военно-космической академии имени А.Ф. Можайского. Область научных интересов: системный анализ, защита информации, интеллектуальная поддержка принятия решений. Число научных публикаций — 70. Biryukov.D.N@yandex.ru; 197198, г. Санкт-Петербург, ул. Ждановская 13.; p.t.: +7(812) 237-19-60.

Biryukov Denis Nikolaevich — Ph.D., professor of system for collecting and processing information department, Mozhaisky Military Space Academy. Research interests: system analyses, IT-Security, intelligent decision support. The number of publications — 70. Biryukov.D.N@yandex.ru; 13, Zhdanovskaya street, St.-Petersburg, 197198, Russia; office phone: +7(812) 237-19-60.

Ломako Александр Григорьевич — д-р техн. наук, профессор, профессор кафедры систем сбора и обработки информации, Военно-космическая академия имени А.Ф. Можайского. Область научных интересов: информационная безопасность, теоретическое и системное программирование, синтез и верификация корректности моделей программ. Число научных публикаций — 250. lomako_ag@mail.ru; ул. Ждановская 13, 197198, Санкт-Петербург; p.t.: +7(812) 237-19-60.

Lomako Aleksandr Grigor'evich — Ph.D., Dr. Sci., professor, professor of system for collecting and processing information department, Mozhaisky Military Space Academy. Research interests: information security, theoretical and system programming, synthesis and verification of program models. The number of publications — 250. lomako_ag@mail.ru; 13, Zhdanovskaya street, St.-Petersburg, 197198, Russia; office phone: +7(812) 237-19-60.

РЕФЕРАТ

Бирюков Д.Н., Ломако А.Г. **Денотационная семантика контекстов знаний при онтологическом моделировании предметных областей конфликта.**

Интеллектуальность проектируемых и разрабатываемых систем упреждения и пресечения атакующих воздействий (компьютерных атак) существенно зависит от их возможности осуществлять смысловую, контекстуальную и наконец, семантическую обработку данных и знаний. При этом искомая система должна иметь возможность представлять и обрабатывать знания о причинах конфликтов, процессах их протекания и следствиях конфликтов. Для этого необходимо, чтобы она обладала возможностью представлять знания об объектах (субъектах), их свойствах и процессах их взаимодействия в различных контекстах. В связи с этим, в статье предлагается подход к онтологическому представлению данных в памяти интеллектуальной системы, основанный на теории типов и теории конечной аппроксимации, который открывает возможность формального представления контекстно-зависимых многослойных знаний о поведении конфликтующих сторон. Применение предложенного подхода должно позволить моделировать семантические особенности контекстно-зависимых знаний предметных областей, учитывать их изменения и уточнения при порождении решений по предотвращению кибервоздействий на критическую информационную инфраструктуру.

SUMMARY

Biryukov D.N., Lomako A.G. **Denotational Semantics of Knowledge Contexts in Ontological Modeling of Subject Domains of the Conflict.**

Intellectuality of the designed and developed systems of anticipation and suppression of the attacking influences (computer attacks) significantly depends on their ability to carry out semantic and contextual data and knowledge processing. Thus, the required system must be able to represent and process knowledge of the causes of conflicts, processes of their course and consequences. For this purpose, it is necessary for the system to be able to represent knowledge of objects (subjects), their properties and processes of their interaction in various contexts. In this regard, the article proposes an approach to ontological data presentation in memory of an intellectual system based on the theory of types and the theory of final approximation, which opens the possibility of formal representation of context-dependent multilayered knowledge of behavior of the conflicting parties. Application of the offered approach should allow one to model semantic features of context-dependent knowledge of subject domains, to consider their changes and specifications at generation of decisions on prevention of cyber impacts on critical information infrastructure.

П.А. ГЛЫБОВСКИЙ, А.П. ГЛУХОВ, Ю.А. ПОНОМАРЕВ, М.В. ШИЛЕНКОВ
**ПОДХОД К ОЦЕНИВАНИЮ И ПРОГНОЗИРОВАНИЮ УРОВНЯ
ЗАЩИЩЕННОСТИ ИНФОРМАЦИОННЫХ И
ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМ**

Глыбовский П.А., Глухов А.П., Пономарев Ю.А., Шиленков М.В. Подход к оцениванию и прогнозированию уровня защищенности информационных и телекоммуникационных систем.

Аннотация. В статье рассматриваются подходы к долгосрочному прогнозированию количественных и качественных показателей подсистемы защиты информационных и телекоммуникационных систем, оценивается возможность их применения для анализа защищенности систем от несанкционированного доступа.

Ключевые слова: система оценки защищенности, политика безопасности, модель безопасности, подсистема защиты, методы самоорганизации.

Glybovsky P.A., Gluhov A.P., Ponomarev Yu.A., Shilenkov M.V. Approach to the Evaluation and Prediction of the Level of Security of Information and Telecommunications Systems.

Abstract. The article discusses approaches to long-term forecast of quantitative and qualitative indices of the security subsystem of information and telecommunication systems. The possibility of their use for the analysis of protection of systems against unauthorized access is assessed.

Keywords: system security assessment; security policy; security model; security subsystem; methods of self-organization.

1. Введение. В настоящее время при оценке уровня безопасности корпоративной информационной инфраструктуры применяются как методики отечественных и международных стандартов, ведомственных руководящих документов, так и методы прогнозирования ситуаций и оценки требуемых значений показателей уровня защищенности в совокупности с оценками экономической эффективности инвестиций в обеспечение безопасности и защиту информации [1,2].

К основным принципам создания системы оценки защищенности информационной инфраструктуры и анализа прогнозов показателей уровня защищенности относятся следующие:

- комплексное оценивание соответствия типовым требованиям руководящих документов Российской Федерации, международных стандартов ISO;

- количественное оценивание текущего уровня безопасности, задание допустимых уровней рисков;

- выявление и блокирование наиболее опасных уязвимостей до осуществления атак на уязвимые ресурсы;

- определение функциональных отношений и зон ответственности при взаимодействии подразделений и лиц по обеспечению информационной безопасности;

- обеспечение поддержания внедренного комплекса защиты, регулярные доработки организационно-распорядительной документации, модификация технологических процессов и модернизация технических средств защиты.

Следует отметить, что количественное оценивание подразумевает проведение инструментальных исследований информационной инфраструктуры на наличие уязвимостей [2].

2. Процесс оценки уровня защищенности ИТКС. Процесс анализа, контроля и оценки защищенности компонент информационных и телекоммуникационных систем (ИТКС) представляет собой процесс выполнения во времени некоторой совокупности отдельных процедур, связанных с получением информации об объекте оценки и проведением собственно требуемых оценок защищенности на предмет удовлетворения состоянием соответствующей компоненты ИТКС каким-либо требованиям по защищенности (безопасности) данной компоненты (рисунок 1). Каждая из процедур выполняется по соответствующей методике, и в результате чего формируются показатели уровня защищенности, которые подвергаются прогнозированию.

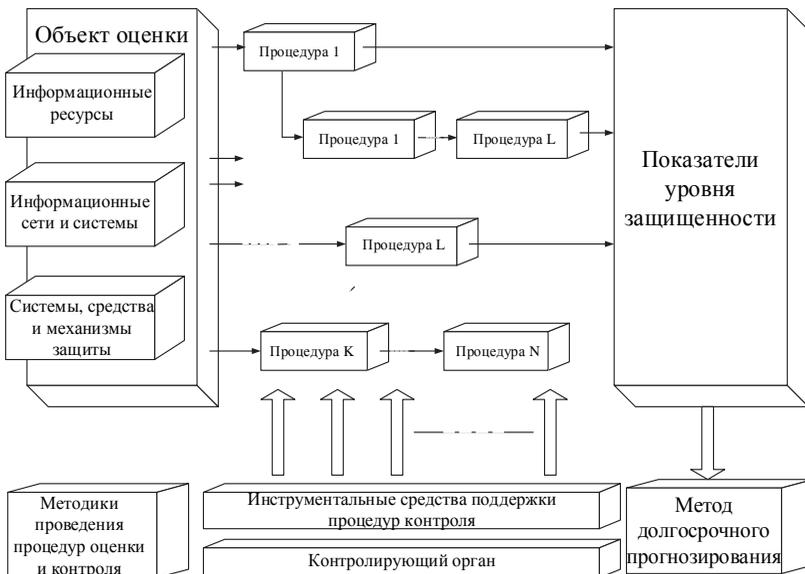


Рисунок 1 – Процесс оценки уровня защищенности ИТКС

Прогнозирующей моделью называют уравнение (или систему уравнений), описывающее предполагаемый ход моделируемого про-

цесса в будущем [3]. Соответствующие уравнения зависимости, представленные в виде таблиц или графиков, являются прогнозом [4].

При использовании методов самоорганизации прогнозирующие модели строятся на основе опытных данных (по предыдущим наблюдениям процесса или, иначе, по его предыстории) [4]. Обычно такая информация представляется в таблицах через равные отрезки времени (дискретные интервалы или шаги). Задачей прогнозирования процесса (или временного ряда) является определение его систематической составляющей, называемой трендом [5,6]. Общие тенденции изменения временных рядов могут иметь регулярный (например, возрастающий) или циклический характер и описываться, соответственно, как полиномиальными, так и циклическими трендами. В ряде случаев тренд может являться функцией не только времени, но и ряда других наблюдаемых величин и параметров (при этом он применяется для построения так называемых нормативных прогнозов), флуктуации и нерегулярные составляющие временных рядов включают в себя ошибки измерения, а также отклонения от тренда, вызываемые воздействием не учитываемых, изменчивых факторов.

3. Критерии селекции. При синтезе трендов оптимальной сложности организуется перебор большого числа трендов-pretендентов по критериям [7,8]:

- регулярности;
- минимума смещения;
- баланса переменных;
- комбинированным критериям.

Критерий регулярности. Проверочная последовательность используется в методе группового учета аргументов (МГУА) для выбора числа членов и степени уравнения регрессии: это позволяет получить наиболее регулярное решение (оптимальной сложности), малочувствительное к небольшим изменениям исходных данных. Такое решение дает небольшую ошибку на новых точках, в чем и состоит основная цель МГУА в задачах однократного (краткосрочного) прогноза и, иногда, идентификации. Величина критериев регулярности вычисляется по данным отдельной проверочной последовательности.

Разделение исходной выборки на две последовательности - обучающую и проверочную - производится по одному из следующих приемов:

Опытные точки ранжируются в ряд по дисперсии (квадрат расстояния от центра «тяжести» пространственной фигуры, охватывающей все представляющие точки):

$$D_i^2 = \sum_{j=1}^n (x_{ij} - \bar{x}_i)^2, \quad (1)$$

где $j=1, 2, \dots, N$; N – общее число точек; n – число аргументов;

$$\bar{x}_i = \frac{1}{N} \sum_{j=1}^N x_{ij}.$$

В этом случае для определения коэффициентов частных описаний используются возможно более удаленные точки.

После ранжировки по дисперсии можно произвести дополнительное разделение точек на нечетные - четные. В таком случае первая и вторая части последовательности имеют близкие статистические свойства.

Если исходные данные снимаются в строгой временной последовательности (например, опытные данные физического поля или технологического процесса), то ранжировка точек в ряд недопустима [9]. В этом случае можно применить деление точек по четным и нечетным индексам [10]. После разделения по тому или другому способу первые N_A точек составляют обучающую последовательность, следующие N_B точек - проверочную.

В качестве критериев регулярности используются: относительная среднеквадратичная ошибка:

$$\Delta^2(B) = \frac{\sum_{i=1}^{N_B} (q_i^* - q_i)^2}{\sum_{i=1}^{N_B} (q_i^*)^2} \rightarrow \min, \quad (2)$$

коэффициент корреляции:

$$K_{qq}^* = \frac{\sum_{i=1}^{N_B} q_i^* q_i}{\sqrt{\sum_{i=1}^{N_B} (q_i^*)^2 \sum_{i=1}^{N_B} q_i^2}} \rightarrow 1, \quad (3)$$

индекс корреляции:

$$\rho = \sqrt{1 - \frac{\sum_{i=1}^{N_B} (q_i^* - q_i)^2}{\sum_{i=1}^{N_B} (q_i^* - \bar{q}_i)^2}} \rightarrow 1, \quad (4)$$

где $\Delta(B)$ - относительная среднеквадратичная ошибка; q_i, q_i^* - полученное по модели и действительное значение выходной величины в i -й точке ($t = 1, 2, \dots, N_B$); N_B - число точек проверочной последовательности:

$$\bar{q}_i = \frac{1}{N_B} \sum_{i=1}^{N_B} q_i^* .$$

Величины $\Delta^2(B), K_{qq^*}, \rho$ вычисляются на проверочной последовательности B . Чем меньше величина относительной среднеквадратичной ошибки или чем больше величина коэффициента корреляции или индекса корреляции, тем модель более регулярна.

Значения коэффициента корреляции и индекса корреляции в меньшей степени зависят от конкретного процесса и более подходят для сравнения результатов моделирования, чем значение среднеквадратической ошибки на проверочной последовательности.

Критерий минимума смещения. Модели, найденные по минимуму критерия регулярности, хорошо решают задачу краткосрочного прогноза при малом уровне шумов в исходных данных. При увеличении помех минимум критерия регулярности смещается влево, в сторону выбора более простых моделей, и при большой интенсивности шумов оптимальными оказываются линейные прогнозирующие модели.

Более устойчивыми к помехам являются различные формы критерия минимума смещения, так как, хотя при увеличении помех минимальное значение показателя смещения тоже увеличивается, минимум получается при одной и той же сложности модели, отвечающей истинной модели (это объясняет, почему такой критерий рекомендуется для решения задачи открытия законов). При малых шумах в исходных данных критерий регулярности и критерий несмещенности равноценны и приводят обычно к одинаковым результатам.

Общая, основная идея построения критерия минимума смещения состоит в том, что модели, получаемые при использовании различных частей таблицы исходных данных, должны по возможности мало отличаться друг от друга, а при полном отсутствии смещения -

совпадать. Это означает, что если исходные данные разбиты на две части, A и B , то модель, полученная на множестве точек A , должна возможно меньше отличаться от модели, полученной на множестве точек B .

Для расчета показателя смещения все имеющиеся экспериментальные данные делятся на две части A и B . Сначала первая последовательность данных является обучающей, а вторая проверочной. Уравнения регрессии (частные модели), получаемые при этом, обозначим $q(A) = f(x_i, x_j)$. Далее первая последовательность служит проверочной, а вторая — обучающей, что выражается уравнением регрессии (частной моделью) $q(B) = f(x_i, x_j)$. Близость этих уравнений оценивается по величине среднеквадратического расхождения их выходов, рассчитанных по всем табличным точкам:

$$n_{см}^2 = \frac{\sum_{i=1}^N [q_i(A) - q_i(B)]^2}{\sum_{i=1}^N q_{iтабл}^2} \quad (5)$$

В многорядных алгоритмах МГУА эта величина, называемая показателем смещения, используется для выбора F лучших частных описаний, пропускаемых на последующие ряды селекции. В качестве суммарной характеристики смещения частных моделей ряда используется среднее значение показателей смещения самых несмещенных уравнений:

$$N_{см} = \frac{1}{F} \sum_{i=1}^F n_{см_i}^2 \quad (6)$$

В некоторых программах вместо $N_{см}$ используется одно, лучшее значение $n_{см_i}^2$. Ряды селекции наращиваются до тех пор, пока величина $N_{см}$ уменьшается.

В комбинированных алгоритмах МГУА значения $n_{см}^2$ рассчитываются для всех моделей, заданных в таблице постепенного усложнения, и в результате выбирается самая несмещенная модель.

В некоторых алгоритмах применяются следующие разновидности показателя смещения, основанные на анализе решений:

$$n_{cm}^2 = \frac{1}{N} \sum_{i=1}^N [q_i(A) - q_i(B)]^2 \rightarrow 0, \quad (7)$$

$$n_{cm}^2 = 2 \frac{\sum_{i=1}^N q_i(A)q_i(B)}{\sum_{i=1}^N (q_i^2(A) + q_i^2(B))} \rightarrow 1. \quad (8)$$

В том случае, когда все особым образом нормированные аргументы частных моделей МГУА имеют примерно одинаковые максимальные и минимальные значения, для оценивания частных описаний можно воспользоваться критерием минимума смещения коэффициентов:

$$n_{cm}^2 = \frac{\sum_{i=1}^N (a_i - b_i)^2}{\sum_{i=1}^N (a_i^2 - b_i^2)}. \quad (9)$$

Возможны и другие формы этого критерия, однако соответствующий показатель смещения, как функция параметров модели, должен быть безразмерным, симметричным и малочувствительным к различию в малых коэффициентах, например,

$$n_{cm}^2 = 2 \frac{\sum_{i=1}^N a_i b_i}{\sum_{i=1}^N (a_i^2 + b_i^2)} \rightarrow 1. \quad (10)$$

В ортогонализированных алгоритмах МГУА критерий минимума смещения коэффициентов упрощается. Действительно, в этом случае частные модели содержат всего один коэффициент, смещение которого может быть вычислено по формуле:

$$n_{cm}^2 = \frac{(a-b)^2}{a^2 + b^2} \rightarrow 1. \quad (11)$$

Пусть $f(q_1(t), q_2(t), \dots, q_S(t)) = 0$ – функция баланса (т. е. «закон», связывающий переменные $q_1(t), q_2(t), \dots, q_S(t)$). Из множества всех прогнозирующих моделей для переменных $q(t)$ нужно выбрать такую

систему, для которой на интервале экстраполяции (в районе точки прогноза) эта связь выполняется наилучшим образом. Нарушение баланса переменных можно характеризовать величинами небаланса

$$b_i = f(q_1(t), q_2(t), \dots, q_s(t)), \quad (12)$$

которые рассчитываются для f , принадлежащих интервалу экстраполяции, по различным прогнозирующим моделям для каждой переменной $q_i(t)$.

В результате выбирается наилучшая комбинация моделей. Их усложнение удобнее всего задавать в виде специальных таблиц постепенного усложнения.

Критерий баланса позволяет выбрать наилучший прогноз системы переменных набора так называемых «вееров» прогнозов (трендов) для каждой переменной.

В некоторых задачах функция связи между переменными ясна из физических соображений и использование ее в алгоритме является вполне естественным. В других случаях из физики процесса можно судить о существовании функциональной зависимости между переменными, однако вид ее неизвестен. При этом рекомендуется предварительно восстановить функцию баланса, используя один из алгоритмов МГУА на основе критерия минимума смещения.

В том случае, когда прогнозируемые переменные признаны несвязанными, возможно искусственное расширение множества переменных за счет дополнения их временными рядами, образованными из исходных по выбранным законам. Эти зависимости и выступают в дальнейшем в качестве функций баланса. При этом линейные функции баланса приводят к неоднозначности выбора модели, в то время как нелинейные – обеспечивают более надежную селекцию.

Если взаимосвязь между переменными существует, то на участке интерполяции могут быть построены несмещенные законы:

$$\begin{aligned} q_1 &= f_1(q_2, q_3, \dots, q_s); \\ q_2 &= f_2(q_1, q_3, \dots, q_s); \\ &\vdots \\ q_s &= f_s(q_1, q_2, \dots, q_{s-1}); \end{aligned} \quad (13)$$

Допустим, что функции (13), называемые прямыми, достаточно просты и допускают «обращение», т. е. выражение некоторых аргу-

ментов, входящих в правую часть, через остальные. Тогда из системы уравнений (13) можно получить:

$$\begin{aligned} q_1 &= f_{1o\bar{o}p}(q_2, q_3, \dots, q_s); \\ q_2 &= f_{2o\bar{o}p}(q_1, q_3, \dots, q_s); \\ &\vdots \\ q_s &= f_{so\bar{o}p}(q_1, q_2, \dots, q_{s-1}); \end{aligned} \quad (14)$$

причем t -я обратная функция находится из прямой ($i \neq t$).

Из всех возможных вариантов построения обратной функции следует предпочесть тот, при котором она находится из самой несмещенной прямой (исходной) функции. Величины $b_{1i}=(f_1 - f_{1o\bar{o}p})^2_i$, $b_{2i}=(f_2 - f_{2o\bar{o}p})^2_i, \dots, b_{si}=(f_s - f_{so\bar{o}p})^2_i$ - рассчитанные на интервале экстраполяции (t_i принадлежит интервалу прогнозирования) по прогнозирующим моделям (трендам) для каждой переменной, характеризуют небаланс системы прогнозов в точке, а величина:

$$B = \sum_{i=1}^m \left(\frac{\sum_i b_{ij}}{\sum_i f_i^2(i)} \right), \quad (15)$$

используется для выбора системы моделей.

Критерий баланса. Среди комбинаций трендов (по одному из «веера» каждой переменной), дающих малое, близкое к нулю значение критерия баланса переменных, обязательно находится одна комбинация, отвечающая действительному ходу процесса, так как закон, связывающий переменные, постоянен во времени. Однако среди таких комбинаций могут быть и ложные, не отвечающие ходу процесса, поэтому необходимо выделить «истинный» баланс среди нескольких «ложных». Для решения этой задачи рекомендуется использовать дополнительные ограничения (критерий «физической реализуемости прогноза» и селекцию модели по двум целесообразно выбранным критериям).

При реализации критерия баланса в виде «кольца разностей прямых и обратных функций» необходимо иметь в виду, что в случае тождественности f_j и $f_{jo\bar{o}p}$. Величина B не позволяет сделать однозначный выбор модели, поэтому рекомендуется переход к показателю:

$$B^N = \sum_{i=1}^m \left(\frac{\sum_i [(q_i(i) - f_i(i))^2 + (q_i(i) - f_{ioop}(i))^2]}{\sum_i x_i^2(i)} \right). \quad (16)$$

В случае дифференциальных прогнозирующих моделей выход за пределы интервала интерполяции при формировании критерия баланса осуществляется при многошаговом интегрировании уравнений динамики. Наиболее точные прогнозы получаются при использовании в критерии истинных функций баланса, определяемых из физических соображений.

Комбинированный критерий селекции, его разновидности и свойства. Выбор критерия отбора (селекции) определяется типом решаемой задачи, принадлежит автору модели и отражает его цели. По рекомендациям, приведенным в работе, можно целесообразно выбрать один из эвристических критериев. Однако в некоторых ситуациях требования к моделям оказываются противоречивыми и не могут быть выражены в виде одного минимизируемого критерия. Кроме того, иногда выбранный критерий селекции не обеспечивает единственности выбора структуры модели или же требует слишком больших вычислений, так что желательно сокращение объема перебора. В ряде случаев может оказаться недостаточной «плавность» критерия, и требуется предварительно сократить число оцениваемых моделей, чтобы попасть в область надежной селекции.

В том случае, когда необходимо согласование нескольких противоречивых свойств (качеств) модели, рекомендуется перейти к комбинированным критериям, которые обычно формируются из составляющих критериев $K(1)$ и $K(2)$:

$$K_{комб} = \sqrt{\lambda K_1^2 + (1 - \lambda) K_2^2}, 0 < \lambda < 1, \quad (17)$$

и находить модели, не улучшаемые по обоим критериям, а при изменении $\lambda_{во}$ всем интервале $[0, 1]$ построить множество паретовских структур моделей.

Кроме согласования противоречивых требований к модели (фактически, задача многокритериальной оптимизации), применение комбинированных критериев позволяет обеспечить единственность выбора структуры в том случае, когда один из критериев приводит к многозначности модели. Оба критерия должны быть выбраны целесообразно, т. е. с учетом физического смысла решаемой задачи моде-

лирования. Модели, равноценные по одному из критериев, будут отличаться по другому, что и определяет однозначный выбор.

Комбинированный критерий, учитывающий исходные критерии, может формироваться и в виде:

$$K_{\text{комб}} = \sqrt{\lambda \left(\frac{K_1}{K_{1\max}}\right)^2 + (1-\lambda) \left(\frac{K_2}{K_{2\max}}\right)^2}, 0 < \lambda < 1, \quad (18)$$

приведенном к безразмерным величинам, что необходимо в том случае, когда K_1 и K_2 имеют различные единицы измерения и различный физический смысл.

Комбинированные критерии селекции. Теория самоорганизации подчеркивает, что целесообразный выбор критериев селекции и порядка их использования принадлежит человеку. В качестве составляющих K_1 и K_2 в комбинированных критериях могут быть использованы;

$\Delta(B)$ - критерий регулярности, $n_{\text{см}}$ - критерий минимума смещения, B_i - различные виды критериев баланса переменных, i_2 - критерий устойчивости прогнозов.

Ориентировочно можно сформулировать следующие правила применения комбинированных критериев селекции [11]:

1. Для алгебраических (полиномиальных) моделей, не содержащих среди аргументов время (т. е. для моделей, не предназначенных для долгосрочного прогноза), следует применять комбинированный критерий «смещение плюс регулярность»:

$$K_1 = \sqrt{n_{\text{см}}^2 + \Delta^2(B)} \rightarrow \min. \quad (19)$$

2. Для гармонических и алгебраических моделей, использующих функции времени (т. е. предназначенных для долгосрочного прогноза), в случаях, когда известен или может быть открыт закон физического баланса переменных, рекомендуется критерий «смещение плюс баланс переменных» [12]:

$$K_2 = \sqrt{n_{\text{см}}^2 + b^2} \rightarrow \min. \quad (20)$$

3. Для дифференциальных моделей (в виде конечно-разностных уравнений), предназначенных для многократного долгосрочного про-

гноза при помощи пошагового интегрирования, рекомендуется комбинированный критерий «несмещенность плюс устойчивость прогноза»:

$$K_3 = \sqrt{n_{cm}^2 + i^2(N)} \rightarrow \min . \quad (21)$$

При использовании дифференциальных моделей результаты долгосрочного прогноза могут быть неудовлетворительными, если вычислительная процедура не обладает достаточной устойчивостью, т. е. приводит к накоплению ошибок. Существует два способа достижения устойчивых результатов:

усреднение некоторого числа лучших частных моделей на каждом ряду селекции или только в конце синтеза модели;

использование комбинированного критерия K_3 , содержащего требования устойчивости. Величина $i_2(N)$ формируется как сумма квадратов разностей табличных значений выхода модели и рассчитанных в процессе многократного интегрирования уравнения динамики.

Алгоритмы с усреднением также позволяют получать более устойчивые модели; истинная же модель всегда устойчива [13].

Применяются также комбинированные критерии, учитывающие три или четыре составляющих, например, вида:

$$K_4 = \sqrt{n_{cm}^2 + b^2 + i^2(N)} \rightarrow \min . \quad (22)$$

Последний этап построения математических моделей заключается в пересчете (адаптации) их параметров, необходимом для минимизации какого-либо (обычно, комбинированного) критерия, и позволяющем улучшить точность прогноза и добиться более полного соответствия свойств процесса и модели.

Адаптация может производиться для одной наилучшей модели (после селекции) и для всех структур моделей (до селекции), так что отбор производится по улучшенным значениям критериев селекции.

Среди составляющих комбинированного критерия могут быть такие, которые характеризуют только структуру модели, независимо от ее коэффициентов (например, показатели смещения). При этом адаптация сводится к минимизации подбором коэффициентов модели только второй составляющей комбинированного критерия (если она является характеристикой и структуры, и параметров), что упрощает расчеты и допускает применение градиентных процедур и методов случайного поиска [14].

4. Заключение. Выводы по статье:

1. Метод самоорганизации моделей основан на применении внешних критериев. Критерий называется внешним, если его определение основано на новой информации, «свежих» точках, не использованных при синтезе модели выбора: регулярности, минимума смещения, баланса переменных и комбинированных. Структура модели оптимальной сложности соответствует минимуму внешнего критерия. Принцип самоорганизации состоит в том, что при постепенном усложнении моделей некоторые критерии (имеющие свойства «внешнего дополнения») проходят через свой минимум. При помощи перебора моделей находят этот минимум и указывают единственную модель оптимальной сложности.

2. Для прогноза показателей уровня защищенности АИТС разработан нечеткий групповой метод обработки данных. Данный прогноз имеет важное значение для оценки требуемых показателей уровня защищенности АИТС и планирования защищенности систем от несанкционированного доступа в будущем.

Литература

1. *Стоянова О.В., Бояринов Ю.Г.* Моделирование социально-экономических систем на основе самоорганизации нейро-нечетких сетей // Вестник МЭИ. 2014 №4.
2. *Zelinka I., Oplatkova Z., Nolle L.* Analytic programming – symbolic regression by means of arbitrary evolutionary algorithm // I. J. of Simulation. 2013. vol. 6, no. 9. pp. 44–56.
3. *Дрейнер Н., Смит Г.* Прикладной регрессионный анализ // М.: Издательский дом “Вильямс”. 2007. 912 с.
4. *Anastasakis L., Mort N.* The development of selforganization techniques in modeling: A review of the Group Method of Data Handling (GMDH) // Research Report № 823. Sheffield, United Kingdom, The University of Sheffield. 2015.
5. *Орлов А.А.* Принципы построения архитектуры программной платформы для реализации алгоритмов метода группового учета аргументов (МГУА) // Управляющие системы и машины: Международный журнал. 2013. № 2. С. 65–71.
6. *Краснощечков П. С., Петров А. А.* Принципы построения моделей // М.: Фазис. 2010. 264 с.
7. *Booch G., Maksimchuk R.A., Engel M.W., Young B.J., Conallen J., Houston K.A.* Object Oriented Analysis and Design with Applications // NY. Addison_Wesley Professional. 2013. 720 p.
8. *Бояринов Ю.Г., Стоянова О.В., Дли М.И.* Нейро-нечеткий метод группового учета аргументов для поддержки принятия решений по управлению региональными социально-экономическими системами // М.: Физматлит. 2005.
9. *Larose D.T.* Discovering knowledge in Data: An Introduction to Data Mining // New Jersey: Wiley & Sons. 2005. 336 p.
10. *Olson D.L.* Advanced Data Mining Techniques // Berlin: Springer-Verlag. 2008. 180 p.
11. *Kordik P.* Fully Automated Knowledge Extraction using Group of Adaptive Models Evolution // PhD thesis. Prague. 2014. 150 p.
12. *Стрижов В.В., Крымова Е.А.* Методы выбора регрессионных моделей // М.: ВЦ РАН. 2010. Т. 60. С. 2.

13. Guyon I., Gunn S. Feature extraction: foundation and applications // Springer. 2011. 778 p.
14. Efron B., Hastie T., Johnstone I., Tibshirani R. Least angle regression // The Annals of Statistics. 2014. vol. 32. no. 2. pp. 407–499.

References

1. Stojanova O.V., Bojarinov Ju.G. [Modeling of socio-economic systems based on self-organizing neural-fuzzy networks]. *Vestnik MJeI – Bulletin of the MEI*. 2014. no. 4. (In Russ).
2. Zelinka I., Oplatkova Z., Nolle L. Analytic programming – symbolic regression by means of arbitrary evolutionary algorithm. *I. J. of Simulation*. 2013. vol. 6, no. 9. pp. 44–56.
3. Drejper N., Smit G. *Prikladnoj regressionnyj analiz* [Applied Regression Analysis]. M.: Publishing house "Williams". 2007. 912 p. (In Russ).
4. Anastasakis L., Mort N. The development of selforganization techniques in modeling: A review of the Group Method of Data Handling (GMDH). Research Report № 823. Sheffield, United Kingdom, The University of Sheffield. 2015.
5. Orlov A.A. [Architecture principles of a software platform for the implementation of algorithms group method of data handling (GMDH)]. *Upravljajushhie sistemy i mashiny: Mezhdunarodnyj zhurnal – Control systems and machines: An International Journal*. 2013. no. 2. pp. 65–71. (In Russ).
6. Krasnoshhekov P. S., Petrov A. A. *Principy postroenija modelej* [Principles of constructing models]. Phasis. 2010. 264 p. (In Russ).
7. Booch G., Maksimchuk R.A., Engel M.W., Young B.J., Conallen J., Houston K.A. Object Oriented Analysis and Design with Applications. NY. Addison_Wesley Professional. 2013. 720 p.
8. Boyarinov U.G., Stoyanova O.V., Dli M.I. *Nejro-nechetkij metod gruppovogo ucheta argumentov dlja podderzhki prinjatija reshenij po upravleniju regional'nymi social'no-jekonomicheskimi sistemami* [Neuro-fuzzy group method of data to support management decision-making regional socio-economic systems]. M.: Fizmatlit. 2005. (In Russ).
9. Larose D.T. *Discovering knowledge in Data: An Introduction to Data Mining*. New Jersey: Wiley & Sons. 2005. 336 p.
10. Olson D.L. *Advanced Data Mining Techniques*. Berlin: Springer-Verlag. 2008. 180 p.
11. Kordik P. Fully Automated Knowledge Extraction using Group of Adaptive Models Evolution. PhD thesis. Prague, 2014. 150 p.
12. Strizhov V.V., Krymova E.A. [Methods selection of regression models]. *Vychislitel'nyj centr RAN – Computing Centre of RAS*. 2010. Issue 60, p. 2. (In Russ).
13. Guyon I., Gunn S. Feature extraction: foundation and applications. Springer. 2011. 778 p.
14. Efron B., Hastie T., Johnstone I., Tibshirani R. Least angle regression. *The Annals of Statistics*. 2014. vol. 32. no. 2. pp. 407–499.

Глыбовский Павел Анатольевич — к-т техн. наук, доцент, доцент кафедры систем сбора и обработки информации, Военно-космическая академия имени А.Ф. Можайского. Область научных интересов: теория распознавания образов, теория информации. Число научных публикаций — 45. p_glybovsky@mail.ru; ул. Ждановская, д. 13, Санкт-Петербург, 197198; p.t.: +7(812) 237-19-60.

Glybovsky Pavel Anatolievich — Ph.D., associate professor, associate professor of system for collecting and processing information department, Mozhaisky Military Space Academy. Research interests: theory of pattern recognition, information theory. The number of publica-

tions — 45. p_glybovsky@mail.ru; 13, Zhdanovskaya street, St.-Petersburg, 197198, Russia; office phone: +7(812) 237-19-60.

Глухов Александр Петрович — к-т техн. наук, начальник департамента информационной безопасности, ОАО «РЖД». Область научных интересов: информационная безопасность, охрана объектов железнодорожного транспорта. Число научных публикаций — 40. gie76@yandex.ru; ул. Ждановская, д. 13, Санкт-Петербург, 197198; р.т.: +7(812) 347-96-87.

Gluhov Aleksandr Petrovich — Ph.D., head of department of information security, JSC «RZHD». Research interests: information security, protection of railways. The number of publications — 40. gie76@yandex.ru; 13, Zhdanovskaya street, St.-Petersburg, 197198, Russia; office phone: +7(812) 347-96-87.

Пономарев Юрий Александрович — к-т техн. наук, доцент, заместитель начальника кафедры систем сбора и обработки информации, Военно-космическая академия имени А.Ф. Можайского. Область научных интересов: методы анализа социально-психологических характеристик, теория нечетких множеств. Число научных публикаций — 35. yurij_1969_2011@mail.ru; ул. Ждановская, д. 13, Санкт-Петербург, 197198; р.т.: +7(812) 237-19-60.

Ponomarev Yuri Aleksandrovich — Ph.D., associate professor, deputy head of system for collecting and processing information department, Mozhaisky Military Space Academy. Research interests: methods of analysis of the socio-psychological characteristics, fuzzy set theory. The number of publications — 35. yurij_1969_2011@mail.ru; 13, Zhdanovskaya street, St.-Petersburg, 197198, Russia; office phone: +7(812) 237-19-60.

Шиленков Максим Викторович — к-т техн. наук, заместитель генерального директора, ЗАО "Эврика". Область научных интересов: информационные технологии, информационная безопасность. Число научных публикаций — 40. ema5@yandex.ru; ул. Ждановская, д. 13, Санкт-Петербург, 197198; р.т.: +7(911)2231431.

Shilenkov Maxim Viktorovich — Ph.D., deputy general director, JSC "Jevrika". Research interests: information technology, information security. The number of publications — 40. ema5@yandex.ru; 13, Zhdanovskaya street, St.-Petersburg, 197198, Russia; office phone: +7(911)2231431.

РЕФЕРАТ

Глыбовский П.А., Глухов А.П., Пономарев Ю.А., Шиленков М.В.
Подход к оцениванию и прогнозированию уровня защищенности информационных и телекоммуникационных систем.

В настоящее время при оценке уровня безопасности корпоративной информационной инфраструктуры применяются методы прогнозирования ситуаций и оценки требуемых значений показателей уровня защищенности в совокупности с оценками экономической эффективности инвестиций в обеспечение безопасности и защиту информации.

Для повышения эффективности прогнозирования уровня защищенности информационных и телекоммуникационных систем необходимо выполнение прогнозирования совокупности отдельных процедур. Каждая из процедур выполняется по соответствующей методике, и в результате чего формируются показатели уровня защищенности, которые подвергаются прогнозированию.

Одной из особенностей, свойственных задачам математического моделирования уровня защищенности информационных и телекоммуникационных систем, является наличие большого числа факторов, влияющих на исследуемый показатель, и ограниченного объема наблюдений данных факторов, используемых при построении модели. Для решения задачи прогнозирования уровня защищенности был предложен метод группового учета аргументов (МГУА), позволяющий обеспечить приемлемое качество модели в условиях многофакторности и ограниченности объема обучающей выборки.

SUMMARY

Glybovsky P.A., Gluhov A.P., Ponomarev Yu.A., Shilenkov M.V. **Approach to the Evaluation and Prediction of the Level of Security of Information and Telecommunications Systems.**

Currently, in the evaluation of the safety of corporate information infrastructure, techniques are applied for forecasting situations and assessing the required indicator values for security level in conjunction with the assessment of the economic efficiency of investment in safety and protection of information.

To improve the efficiency of predicting the level of security of information and telecommunication systems it is necessary to perform forecast of the aggregate of separate procedures. Each of the procedures is performed by the corresponding method, resulting in the formation of indicators of the level of protection, which are subject to forecasting.

One of the features, peculiar to the problems of mathematical modeling of the level of protection of information and telecommunication systems, is the presence of a large number of factors influencing the studied parameters. At the same time, there is a limited amount of observations of these factors used in the construction of the model. To solve the problem of predicting the level of protection we propose a group method of data handling (GMDH), which allows one to ensure an acceptable quality of the model in the conditions of multivariate models and the limited amount of training sample.

А.А. Бойко
**СПОСОБ АНАЛИТИЧЕСКОГО МОДЕЛИРОВАНИЯ
ПРОЦЕССА РАСПРОСТРАНЕНИЯ ВИРУСОВ В
КОМПЬЮТЕРНЫХ СЕТЯХ РАЗЛИЧНОЙ СТРУКТУРЫ**

Бойко А.А. Способ аналитического моделирования процесса распространения вирусов в компьютерных сетях различной структуры.

Аннотация. Предложен способ аналитического моделирования процесса распространения вирусов в компьютерной сети. Он учитывает особенности сетевой структуры, поведенческие характеристики вирусов и подсистем защиты информации узлов и возможность исходного заражения множества узлов различными вирусами. Способ основан на представлении сети в виде модели с дискретными состояниями и временем переходов, которое распределено по обобщенному закону Эрланга n -го порядка.

Ключевые слова: распространение вирусов, компьютерная сеть, модель с дискретными состояниями и непрерывным временем, структура сети.

Boyko A.A. Method of Analytical Modeling of Spread of Viruses in Computer Networks with Different Structures.

Abstract. The article proposes a method of analytical modeling of viruses propagation process in computer network, which takes into account the characteristics of its topology, behavioral characteristics of viruses and information security sub-systems of nodes and likelihood of source infection of a plurality of nodes with different viruses. The method is based on performance of a network topology in the form of a model with discrete states and times of branches, distributed by the generalized Erlang law of n -th order.

Keywords: the spread of virus, computer network, model with discrete states and continuum time, network structure.

1. Введение. При исследовании конфликта современных организационно-технических систем (например, конкурирующих промышленных предприятий или противоборствующих воинских формирований [1]) важно уметь оценивать их конфликтную устойчивость в условиях деструктивных информационно-технических воздействий [2] различного рода. Частной задачей такого исследования является оценка защищенности проектируемых или эксплуатируемых компьютерных сетей (далее – сетей) организационно-технических систем (ОТС) от воздействия вирусов на узловые информационно-технические средства (ИТС), образующие эти сети (далее – узлы). Информационно-техническими называются такие технические средства, которые участвуют в процессе создания и выполнения операций с данными, из которых может быть получена информация. ИТС является любое радиоэлектронное средство или средство вычислительной техники, комбинация таких средств друг с другом и (или) с другими видами технических средств [3].

Основными характерными чертами сетей конфликтующих ОТС с позиции распространения в них вирусов являются:

- 1) априорная ограниченность информации о составе программного и технического обеспечения узлов сетей;
- 2) фиксированная структура сетей, коррелирующая с иерархической структурой ОТС и включающая от десятков до тысяч узлов;
- 3) среднестатистические временные характеристики функционирования конкретных вирусов в узлах сетей ОТС, соответствующие вектору целей и возможностям этих вирусов;
- 4) среднестатистические временные характеристики функционирования подсистем защиты информации (ПЗИ) сетевых узлов, полученные из "инсайдерских" и открытых источников.

Очевидно, что оценка защищенности сетей ОТС от воздействия вирусов с применением натуральных методов моделирования в подавляющем большинстве случаев является крайне дорогостоящей. Поэтому в настоящей статье уделяется внимание математическим моделям распространения вирусов в сетях ОТС.

2. Анализ существующих работ в области моделирования процесса распространения вирусов. На сегодняшний день известен весьма широкий спектр результатов отечественных и зарубежных исследований в области моделирования процесса распространения вирусов. В целях определения возможности их применения для анализа процесса распространения вирусов в сетях ОТС ограничимся упоминанием наиболее типовых из них.

Работы, посвященные анализу процесса распространения вирусов, традиционно делятся на два основных направления [4, 5]: аналитическое и имитационное.

Модели аналитического направления в свою очередь можно разделить на две группы. Модели первой группы (например, [4-14]) не учитывают структуру сетей, но предоставляют возможность для анализа важных с точки зрения вирусной угрозы состояний узлов с учетом времени. Исторически сложилось, что такие модели явились пионерскими в рассматриваемой области и были заимствованы из математических основ эпидемиологии. В этих моделях все множество объектов в зоне риска разделялось на несколько подмножеств "инфицированных", "уязвимых к заражению", "излеченных" и т.д., а динамика численности этих подмножеств описывалась дифференциальными уравнениями. Модели второй группы учитывают структуру сетей. Но они либо ограничены использованием заведомо недостаточного количества состояний узлов по причине высокой вычислительной сложности применяемых методов (например, [15-17]), либо не дают информа-

ции о состоянии защищенности каждого конкретного узла сети в заданный момент времени (например, [18]).

Одна из наиболее удачных классификаций моделей и систем имитационного направления представлена в [19]. Имитационные модели получили широкое применение в условиях появления высокопроизводительных систем имитационного моделирования, в том числе объектно-ориентированных (пример такой системы показан в [20]). Они обеспечивают высокую точность моделирования при большом количестве сетевых узлов. Однако такие модели требуют детального знания алгоритмов информационного взаимодействия узлов сети, которые зачастую являются недоступными для исследователя.

Для наиболее вероятных на практике исходных данных только о структуре сетей взаимодействующих ОТС и среднестатистических временных характеристиках функционирования вирусов и ПЗИ [21, 22] их узлов приоритет имеют аналитические модели. Они отличаются высокой скоростью моделирования и возможностью получения решения «в общем виде» [5]. Однако попытки применения известных аналитических моделей процесса распространения вирусов или их комбинаций для потенциально доступного набора исходных данных о сетях ОТС оказались безуспешными, поскольку преимущества этих моделей с лихвой перекрывались их недостатками. В результате возникла потребность в разработке нового подхода, парирующего недостатки известных аналитических моделей в рассматриваемой области.

Цель работы – *разработка способа аналитического моделирования процесса распространения вирусов в сетях различной структуры, позволяющего в различные моменты времени оценить вероятность заражения несколькими вирусами каждого узла этих сетей с учетом поведенческих характеристик вирусов и подсистем защиты информации узлов.*

3. Математическая модель процесса распространения вирусов в компьютерных сетях различной структуры. Пусть сеть состоит из N узлов и в различные моменты времени заражается V типами вирусов по q_g экземпляров с различными вероятностями изначального заражения μ_g ($g=1..q_v$). Структура сети задается среднестатистическими временными интервалами (далее – интервалами) τ_{ij} с момента штатного инициирования i -м узлом сеанса связи с j -м узлом до момента окончания этого сеанса. Вирусы могут обеспечивать и не обеспечивать скрытность своего распространения (далее – скрытные и не скрытные вирусы, соответственно). Скрытному вирусу, находящемуся в i -м узле, для заражения j -го узла необходимо дожидаться штатного сеанса связи между i -м и j -м узлами. Не скрытный вирус, попав в i -й узел, присту-

пает к инициированию сеанса связи с j -м узлом независимо от штатных сеансов связи.

Тогда временная диаграмма заражения j -го узла вирусом v -го типа, находящимся в i -м узле, имеет вид, показанный на рисунке 1. В точке 1 на временной оси вирус внедрился в i -й узел и начинает подготовку к размножению. Вирус готов к размножению из i -го узла в точке 3. От точки 3 до точки 5 вирус внедряется из i -го в j -й узел. ПЗИ i -го узла стремится излечить вирус. Излечение начинается с точки 1 и, в зависимости от возможностей ПЗИ, заканчивается либо в точках 2 (вариант 1) или 4 (вариант 2) и тогда вирус не сможет размножиться, либо в точке 6 (вариант 3) и тогда вирус размножится. По причине информационной несовместимости вирус может не внедриться в узел. Например, вирус "не знает" протоколы взаимодействия узлов или их уязвимости.

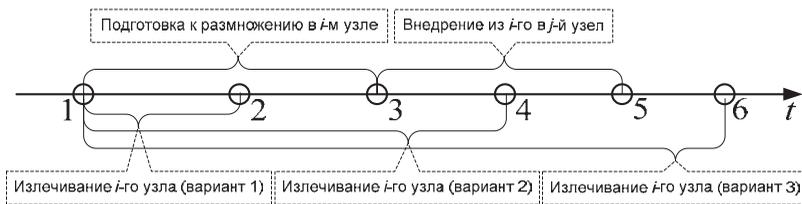


Рис. 1. Временная диаграмма заражения вирусом узла сети

Значение интервала передачи вируса v -го типа между связанными i -м и j -м узлами предлагается вычислять по формуле:

$$m_{vij} = \begin{cases} \alpha_{vij} + \beta_{vi}, & \text{если } \gamma_{vi} > \alpha_{vij} + \beta_{vi}; \\ \infty, & \text{если вирус информационно не совместим} \\ c \text{ узлом или } \gamma_{vi} \leq \alpha_{vij} + \beta_{vi}, \end{cases} \quad (1)$$

где α_{vij} – интервал с момента готовности вируса v -го типа к размножению в i -м узле до момента его внедрения в j -й узел; β_{vi} – интервал с момента внедрения вируса v -го типа в i -й узел до момента, когда этот вирус будет готов к размножению; γ_{vi} – интервал с момента внедрения вируса v -го типа в i -й узел до момента, когда данный узел от этого вируса будет излечен.

Рассмотрим, каким образом можно получить численные значения показателей α_{vij} , β_{vi} и γ_{vi} в формуле (1).

Сеансы связи между парой связанных узлов сети могут устанавливаться либо по инициативе только одного из них (например, в большинстве пар "клиент-сервер" сеансы связи инициируются только кли-

ентами), либо по инициативе любого узла из этой пары (например, в паре связанных сетевых терминалов сеанс связи может инициировать любой из них). Поскольку скрытый вирус может передаваться независимо от того, какой из двух связанных узлов является инициатором сеанса связи, значение интервала α_{vij} предлагается вычислять как:

$$\alpha_{vij} = \begin{cases} \frac{1}{\frac{1}{\tau_{ij}} + \frac{1}{\tau_{ji}}} = \frac{\tau_{ij} \cdot \tau_{ji}}{\tau_{ij} + \tau_{ji}}, & \text{если связь инициируется} \\ & \text{обоими узлами и вирус скрытый;} \\ \tau_{ij}, & \text{если связь инициируется только одним узлом} \\ & \text{и вирус скрытый;} \\ \tau_v, & \text{если вирус не скрытый,} \end{cases} \quad (2)$$

где τ_v – интервал внедрения не скрытого вируса v -го типа, одинаковый для всех узлов сети и являющийся исходной поведенческой характеристикой вирусов этого типа.

Поведенческие характеристики вирусов рассмотрены в работах [21, 22]. В этих работах анализируется информационный конфликт ПЗИ ИТС со специальными программными средствами (СПС), частным случаем которых являются вирусы. В таком конфликте СПС находится в трех основных режимах функционирования: подготовка к применению, скрытое и открытое применение, а ПЗИ выполняет типовые задачи по предупреждению, обнаружению и устранению последствий воздействия СПС. Полагая отлаженными процессы сопряжения узлов в сети друг с другом, этот конфликт с позиции распространения вирусов в сетях сводится к частному конфликту вируса и ПЗИ узла, граф состояний которого представлен на рисунке 2.

Состояния на рисунке 2 характеризуют поведенческие характеристики вируса и ПЗИ узла и имеют следующее описание:

A_1 – вирус внедрился в программную среду узла и анализирует его текущее состояние;

A_2 – вирус производит отказ в обслуживании узла, приводящий к невозможности выполнения любых задач;

A_3 – вирус использует функции узла в своих целях;

A_4 – вирус осуществляет вывод узла из строя;

A_5 – вирус размножается;

A_6 – вирус дезинформирует узел;

A_7 – вирус производит разведку;

A_8 – ПЗИ проводит принудительный поиск вирусов;

A_9 – вирус осуществляет самомодификацию;
 A_{10} – вирус перешел в режим ожидания;
 A_{11} – вирус излечен.

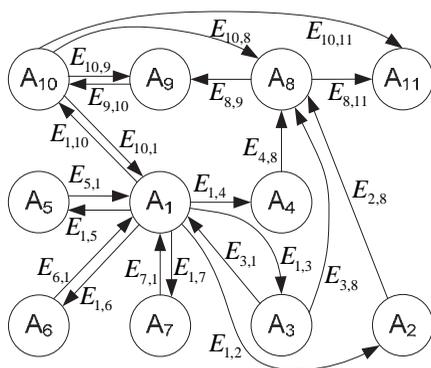


Рис. 2. Граф состояний динамики конфликта вируса и ПЗИ узла

Значения интервалов переходов конфликта вируса и ПЗИ для каждого узла могут быть получены натурным методом, задаваться экспертно или вычисляться с использованием специальных методик. Значение показателя β_{vi} по существу равно значению интервала $E_{1,5}$ перехода i -го узла из состояния A_i в состояние A_5 и является исходной поведенческой характеристикой вируса v -го типа. Интервал γ_{vi} соответствует интервалу перехода рассматриваемого конфликта i -го узла из состояния A_i в состояние A_{11} при воздействии вируса v -го типа. Поэтому численное значение этого интервала предлагается вычислять следующим образом.

Интервалы переходов конфликта вируса и ПЗИ узла из одного состояния в другое не имеют экспоненциальный характер, а коррелируют с некоторой средней величиной. Для анализа такого конфликта применим метод аналитического описания процессов с дискретным множеством состояний и не показательными распределениями времен переходов [23]. Сущность данного метода состоит в преобразовании структуры дискретного процесса с распределенными по обобщенному закону Эрланга n -го порядка временем переходов в непрерывную марковскую цепь путем введения псевдосостояний. Полученный процесс конфликтного взаимодействия описывается в виде системы обыкновенных линейных дифференциальных уравнений. Данный метод оперирует тем фактом, что произвольная плотность распределения времени нахождения системы в некотором состоянии с достаточной степенью точности аппроксимируется с помощью обобщенного закона Эр-

ланга n -го порядка [24]. На практике 2-й или 3-й порядки этого закона являются достаточными для исследования моделей рассматриваемого класса [21-23]. Полученный с применением метода [23] граф марковской цепи конфликта вируса и ПЗИ узла при $n=2$ имеет 165 псевдосостояний и 990 переходов, а при $n=3$ – 2 305 псевдосостояний и 15 689 переходов. Поэтому в настоящей статье эти графы не приводятся.

С учетом изложенного значение показателя γ_{vi} может быть получено с применением следующего алгоритма:

1) решение задачи Коши для марковской цепи конфликта вируса v -го типа и ПЗИ в j -м узле численным методом. Начальным условием является равенство единице вероятности сохранения процесса в состоянии A_i и нулевые вероятности других состояний;

2) вычисление значения показателя γ_{vi} как интервала времени с момента, когда конфликт вируса и ПЗИ вышел из состояния A_i , до момента, когда этот конфликт перешел в состояние A_{j1} , с учетом того, что изображенный на рисунке 2 процесс не является стационарным:

$$\gamma_{vi} = t\{P_{i11} = \xi_{i11} - \Delta\} - t\{P_{i1} = \xi_{i1} - \Delta\}, \quad (3)$$

где $t\{P_{i11} = \xi_{i11} - \Delta\}$ – момент времени установления в i -м узле процесса конфликта вируса и ПЗИ в состоянии A_{j1} с вероятностью $\xi_{i11} - \Delta$; $t\{P_{i1} = \xi_{i1} - \Delta\}$ – момент времени, после которого процесс конфликта вируса и ПЗИ для i -го узла находится в состоянии A_i с вероятностью меньше $\xi_{i1} - \Delta$; Δ – погрешность численного метода решения задачи Коши; ξ_{i1} и ξ_{i11} – достаточные вероятности нахождения процесса конфликта вируса и ПЗИ i -го узла в состояниях A_i и A_{j1} , соответственно.

Поскольку рассматриваемая математическая модель оперирует среднестатистическими временными интервалами, то показатели ξ_{i1} и ξ_{i11} в формуле (3) предлагается определять как вероятности нахождения процесса конфликта вируса и ПЗИ в состояниях A_i и A_{j1} в т.н. "медианный" момент времени. "Медианным" будем называть момент времени, в котором площадь, ограниченная кривой распределения вероятности соответствующего состояния, делится пополам.

Полученные таким образом значения интервалов передачи вирусов между связанными узлами предлагается использовать для анализа процесса распространения вирусов в сетях различной структуры с применением указанного выше универсального метода [23]. Для этого состояния процесса распространения вирусов будем отождествлять с узлами сети, а переходы между состояниями – со связями между этими узлами. Такое отождествление уместно для наиболее часто встречающегося на практике случая, когда ПЗИ узлов напоминают излеченные ими вирусы, приобретая тем самым иммунитет к ним.

4. Содержание способа аналитического моделирования процесса распространения вирусов в компьютерных сетях различной структуры. Исходными данными для способа являются:

- количество узлов сети (N);
- интервалы с момента инициирования узлами соединений до момента окончания этих соединений (τ_{ij});
- количество типов вирусов (V);
- количество экземпляров вируса каждого типа (q_v);
- вероятности изначального заражения сети различными экземплярами вирусов (μ_g);
- интервалы внедрения не скрытых вирусов (τ_v);
- поведенческие характеристики вирусов и ПЗИ в каждом узле сети, обуславливающие интервалы переходов (E_{vjsl} , где s и l – индексы переходов модели конфликта вируса v -го типа и ПЗИ j -го узла);
- интервал моделирования (T_0);
- точность численного метода решения задачи Коши (Δ).

Предлагаемый способ состоит в выполнении следующих шагов.

Шаг 1. Вычисление γ_{vi} с применением вышеуказанного алгоритма для каждого узла сети и каждого типа вируса.

Шаг 2. Представление сети в виде ориентированного графа, в котором переходы характеризуются интервалами m_{vij} , вычисляемыми по формуле (1).

Шаг 3. Преобразование построенного на шаге 2 графа в марковскую цепь с использованием метода [23].

Шаг 4. Вычисление вероятностно-временных характеристик заражения всех узлов сети всеми экземплярами вирусов всех типов. Для этого для построенной на шаге 3 марковской цепи решается задача Коши для каждого экземпляра каждого вируса.

Начальные условия для решения задачи Коши:

- единичная вероятность нахождения экземпляра вируса в том узле, который этим вирусом заражен изначально, и нулевые вероятности нахождения этого экземпляра вируса в остальных узлах;
- начальное время соответствует времени начала действия экземпляра вируса.

Шаг 5. Определение вероятностно-временной характеристики заражения каждого i -го узла вирусом каждого v -го типа по формуле:

$$P_{vi}(t) = \max_{g=1..q_v} \{ P_{gi}(t) \cdot \mu_g \}, \quad (4)$$

где $P_{gi}(t)$ – вероятностно-временная характеристика заражения i -го узла g -м экземпляром вируса v -го типа; μ_g – вероятность изначального заражения сети g -м экземпляром вируса v -го типа.

5. Пример применения способа. Рассмотрим пример сети с 7 узлами, граф которой представлен на рисунке 3.

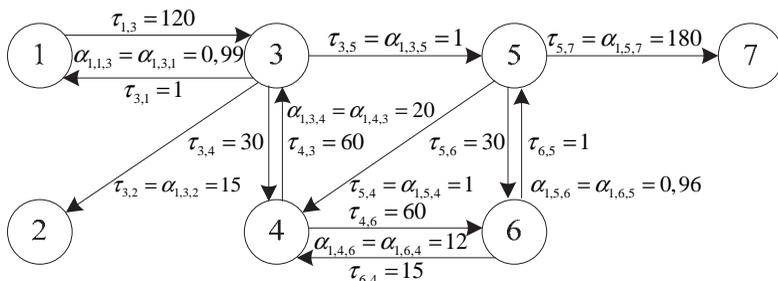


Рис. 3. Граф сети

Сеть поражается четырьмя вирусами двух типов. Узел 1 в начальный момент времени и узел 5 через сутки после этого заражаются скрытым вирусом (1-й тип). Узел 3 через 6 часов после начального момента и узел 6 через 12 часов заражаются не скрытым вирусом (2-й тип). Значения интервалов τ_{ij} и α_{1ij} для вирусов 1-го типа приведены на рисунке 3 в минутах. Для вирусов 2-го типа $\tau_v = 1$ мин. Все узлы кроме узла 4 имеют одинаковый состав программного и информационного обеспечения. Вирусы 1-го и 2-го типов "знают" уязвимости в протоколах информационного взаимодействия всех узлов сети кроме узла 4. Исходные данные о конфликте вирусов 1-го типа и ПЗИ имеют вид: $E_{1,i,1,2}=E_{1,i,1,3}=E_{1,i,1,5}=E_{1,i,1,6}=E_{1,i,1,7} = 3$ часа; $E_{1,i,1,4} = 2$ нед; $E_{1,i,1,10}=E_{1,i,6,1} = E_{1,i,7,1}=E_{1,i,9,10}=1$ мин; $E_{1,i,2,8}=E_{1,i,10,1}=30$ мин; $E_{1,i,3,1}=E_{1,i,8,11}=1$ час; $E_{1,i,3,8}=E_{1,i,4,8} = 15$ мин; $E_{1,i,5,1} = 12$ час; $E_{1,i,8,9}=E_{1,i,10,8}=E_{1,i,10,9}=E_{1,i,10,11} = 1$ день. Исходные данные о конфликте вирусов 2-го типа и ПЗИ имеют вид: $E_{2,i,1,2}=E_{2,i,1,3}=E_{2,i,1,5}=E_{2,i,1,6}=E_{2,i,1,7}=E_{2,i,3,1}=E_{2,i,5,1}=E_{2,i,8,11}=E_{2,i,10,1} = 15$ мин; $E_{2,i,1,4} = 1$ нед; $E_{2,i,1,10}=E_{2,i,9,10} = 1$ мин; $E_{2,i,2,8}=E_{2,i,3,8}=E_{2,i,4,8} = 10$ мин; $E_{2,i,6,1} = E_{2,i,10,8} = 60$ мин; $E_{2,i,7,1} = 30$ мин; $E_{2,i,8,9}=E_{2,i,10,9} = 1$ день; $E_{2,i,10,11} = 3$ часа. Поведенческие характеристики ПЗИ учитывают не скрытый характер вирусов 2-го типа. Интервал моделирования T_0 равен 3 месяцам.

Из исходных данных видно, что $\beta_{1i}=3$ часа, а $\beta_{2i}=15$ мин. Полученные для них значения показателей γ_{1i} и γ_{2i} составляют десятки дней и практически не оказывают влияние на процесс распространения вирусов. Для обобщенного закона Эрланга при $n=2$ анализируемая сеть имеет 25 псевдосостояний и 58 переходов, а при $n=3$ она имеет 67 псевдосостояний и 176 переходов. Результирующие вероятностно-временные характеристики заражения узлов сети совокупностью вирусов одного типа показаны на рисунке 4.

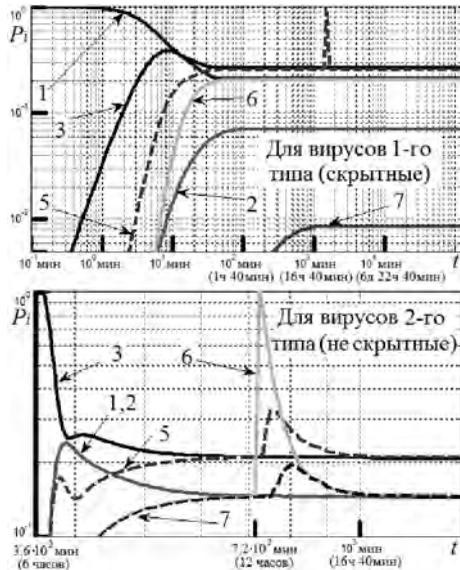


Рис. 4. Вероятностно-временные характеристики заражения узлов сети совокупностью вирусов одного типа

Полученные с использованием предложенного способа вероятностно-временные характеристики могут применяться, в частности, для оценки количества узлов сети, зараженных вирусом каждого типа. Для этого вводится пороговое значение, относительно которого принимается решение о том, заражен ли вирусом узел или нет. Пример таких зависимостей приведен на рисунке 5. В нем пороговое значение равно усредненной на интервале моделирования вероятности заражения вирусом соответствующего типа наименее зараженного узла сети.

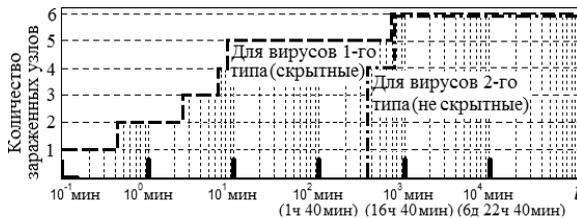


Рис. 5. Зависимость количества зараженных вирусами узлов от времени

Зависимости на рисунке 5 свидетельствуют о том, что для указанных исходных данных ПЗИ вирусы 1-го типа уже через 8 минут

способны заразить 5 из 7 узлов сети, а вирусы 2-го типа в течение 1 часа могут заразить все 6 доступных для них узлов.

6. Заключение. Предложенный способ предоставляет возможность для аналитического решения задачи оценки вероятности заражения каждого узлового информационно-технического средства компьютерной сети несколькими экземплярами вирусов одного или нескольких типов в различные моменты времени. Способ основан на представлении сети в виде модели с дискретными состояниями и распределенным по обобщенному закону Эрланга n -го порядка временем переходов. Он учитывает особенности структуры компьютерных сетей и известные из практики поведенческие характеристики вирусов и подсистем защиты информации узловых информационно-технических средств. Способ может быть применен для исследования конфликтной устойчивости сложных организационно-технических систем.

Литература

1. *Бойко А.А., Храмов В.Ю.* Модель информационного конфликта информационно-технических и специальных программных средств в вооруженном противоборстве группировок со статичными характеристиками // Радиотехника. 2013. №7. С. 5-10.
2. *Бойко А.А., Дьякова А.В.* Способ разработки тестовых удаленных информационно-технических воздействий на пространственно распределенные системы информационно-технических средств // Информационно-управляющие системы. 2014. № 3. С. 84-92.
3. *Бальбин В.А., Донсков Ю.Е., Бойко А.А.* О терминологии в области радиоэлектронной борьбы в условиях современного информационного противоборства // Военная Мысль. 2013. № 9. С. 28–32.
4. *Кондратьев М.А.* Методы прогнозирования и модели распространения заболеваний // Компьютерные исследования и моделирование. 2013. Т. 5. № 5. С. 863-882.
5. *Котенко И.В., Воронцов В.В.* Аналитические модели распространения сетевых червей // Труды СПИИРАН. 2007. № 4. С. 208-224.
6. *Zhang C., Feng T., Zhao Y., Jiang G.* A New Model for Capturing the Spread of Computer Viruses on Complex-Networks // Discrete Dynamics in Nature and Society. 2013. vol. 2013. 9 p.
7. *Mishra B.K., Jha N.* SEIQRS model for the transmission of malicious objects in computer network // Applied Mathematical Modelling. 2010. vol. 34. no. 3. pp. 710–715.
8. *Gan C., Yang X., Liu W., Zhu Q.* A propagation model of computer virus with nonlinear vaccination probability // Communications in Nonlinear Science and Numerical Simulation. 2014. vol. 19. no. 1. pp. 92–100.
9. *Wang Y., Jin Z., Yang Z., Zhang Z.-K., Zhou T., Sun G.-Q.* Global analysis of an SIS model with an infective vector on complex networks // Nonlinear Analysis: Real World Applications. 2012. vol.13. no.2. pp.543–557.
10. *Piqueira J.R.C., Navarro B.F., Monteiro L.H.A.* Epidemiological Models Applied to Viruses in Computer Networks // Journal of Computer Science. 2005. no. 1(1). pp. 31-34.

11. Толстых Н.Н., Остапенко А.Г., Толстых И.О., Ахромеев М.В. Распространение вирусов в кластеризованной сети мобильной связи // Информация и безопасность. 2008. № 3. С. 441-444.
12. Гусаров А.Н., Жуков Д.О., Косарева А.В. Описание динамики распространения компьютерных угроз в информационно-вычислительных сетях с запаздыванием действия антивирусов // Вестник МГТУ им. Н.Э. Баумана. Сер. "Приборостроение". 2010. № 1. С. 112-120.
13. Климентьев К.Е. Моделирования распространения и взаимодействия самовоспроизводящихся объектов // Известия Самарского научного центра Российской академии наук, 2014. т. 16. № 4(2), С. 313-317.
14. Madar N., Kalisky T., Cohen R., ben-Avraham D., Havlin S. Immunization and epidemic dynamics in complex networks // The European Physical Journal B. 2004. № 38. pp. 269-276.
15. Новиков С.В. Модель распространения вирусных атак в сетях передачи данных общего пользования на основе расчета длины гамильтонова пути: автореф... дис. канд. техн. наук // СПб: СПб ГУ ИТМО, 2007.
16. Далингер Я.М., Бабанин Д.В., Бурков С.М. Математические модели распространения вирусов в компьютерных сетях различной структуры // Информатика и системы управления. 2012. № 3(33). С. 25-33.
17. Semenov S.G., Davydov V.V. A Mathematical Model for Technology for Spreading Malicious Software across Heterogeneous Networks based on Markov Chains // European researcher. 2014. № 1-1 (66). pp. 21-30.
18. Утакаева И.Х., Кунижева Л.А. Математическая модель распространения вирусов в сети на предфрактальных графах // Информационное противодействие угрозам терроризма. 2012. №18. С. 64-70.
19. Котенко И.В., Воронцов В.В., Уланов А.В. Модели и системы имитационного моделирования распространения сетевых червей // Труды СПИИРАН. 2007. № 4. С. 225-238.
20. Котенко И.В., Шоров А.В. Механизмы защиты компьютерных сетей от инфроструктурных атак на основе биоинспирированного подхода «нервная система сети» // Вопросы защиты информации. 2013. № 2. С. 57-66.
21. Бойко А.А. Способ стратифицированного аналитического описания процесса функционирования информационно-технических средств // Информационные технологии. 2015. № 1. С. 35-42.
22. Бойко А.А., Будников С.А. Модель информационного конфликта специального программного средства и подсистемы защиты информации информационно-технического средства // Радиотехника. 2015. №4. С. 136-141.
23. Чикин М.Г. Метод аналитического описания процессов с дискретным множеством состояний и не показательными распределениями времен переходов // Информационно-измерительные и управляющие системы. 2004. №5. С. 8-11.
24. Кокс Д., Смит В. Теория восстановления // М.: Сов. радио, 1967. 300 с.

References

1. Boyko A.A., Khramov V.U. [Model of Information Conflict between Special Software and Information-Technical Tools in Military Warfare with Static Characteristics]. *Radiotekhnika - Radioengineering*. 2013. no. 7. pp. 5-10. (In Russ.).
2. Boyko A.A., Djakova A.V. [Method of Developing Test Remote Information-Technical Impacts on Spatially Distributed Systems of Information-Technical Tools] . *Informatsionno-Upravliaiushchie Sistemy - Information and Control Systems*. 2014. no. 3. pp. 84-92. (In Russ.).

3. Balybin V.A., Donskov Ju.E., Boyko A.A. [About the terminology in the field of electronic warfare in modern information warfare]. *Voenmaja Mysl' - Military Thought*. 2013. no. 9. pp. 28–32. (In Russ.).
4. Kondrat'ev M.A. [Forecasting methods and models of disease]. *Komp'juternye issledovaniya i modelirovanie - Computer research and modeling*. 2013. vol. 5. no. 5. pp. 863–882. (In Russ.).
5. Kotenko I.V., Voronov V.V. [Analytical models of network worm propagation]. *Trudy SPIIRAN - SPIIRAS Proceedings*. 2007. no. 4. pp. 208-224. (In Russ.).
6. Zhang C., Feng T., Zhao Y., Jiang G. A New Model for Capturing the Spread of Computer Viruses on Complex-Networks. *Discrete Dynamics in Nature and Society*. 2013. vol. 2013. 9 p.
7. Mishra B.K., Jha N. SEIQRS model for the transmission of malicious objects in computer network. *Applied Mathematical Modelling*. 2010. vol. 34. no. 3. pp. 710–715.
8. Gan C., Yang X., Liu W., Zhu Q. A propagation model of computer virus with non-linear vaccination probability. *Communications in Nonlinear Science and Numerical Simulation*. 2014. vol. 19. no. 1. pp. 92–100.
9. Wang Y., Jin Z., Yang Z., Zhang Z.-K., Zhou T., Sun G.-Q. Global analysis of an SIS model with an infective vector on complex networks. *Nonlinear Analysis: Real World Applications*. 2012. vol.13. no.2. pp.543–557.
10. Piqueira J.R.C., Navarro B.F., Monteiro L.H.A. Epidemiological Models Applied to Viruses in Computer Networks. *Journal of Computer Science*. 2005. no. 1(1). pp. 31-34.
11. Tolstyh N.N., Ostapenko A.G., Tolstyh I.O., Ahromeev M.V. [The spread of viruses in a clustered mobile network]. *Informacija i bezopasnost' - Information and security*. 2008. no. 3. pp. 441-444. (In Russ.).
12. Gusarov A.N., Zhukov D.O., Kosareva A.V. [Description of the dynamics of the spread of computer threats in the information and computer networks with delay action antivirus]. *Vestnik MGTU im. N.Je. Baumana. Ser. "Priborostroenie" - Herald of the Bauman Moscow State Technical University. Instrument Engineering*. 2010. no. 1. pp. 112-120. (In Russ.).
13. Kliment'ev K.E. [Modeling of the propagation and interaction of self-replicating objects]. *Izvestija Samarskogo Nauchnogo Centra Rossijskoj Akademii Nauk - Izvestiya of Samara scientific center of the Russian Academy of Sciences*. 2014. vol. 16. no. 4(2). pp. 313-317. (In Russ.).
14. Madar N., Kalisky T., Cohen R., ben-Avraham D., Havlin S. Immunization and epidemic dynamics in complex networks. *The European Physical Journal B*. 2004. no. 38. pp. 269-276.
15. Novikov S.V. *Model' rasprostraneniya virusnyh atak v setjah peredachi dannyh obshhego pol'zovanija na osnovе rascheta dliny gamil'tonova puti* [Model the spread of viral attacks in networks of public data by calculating the length of a Hamiltonian path]. *synopsis... dis. cand. tech. sciences*. St. Petersburg. GU ITMO. 2007. 17 p. (In Russ.).
16. Dalinger Ja.M., Babanin D.V., Burkov S.M. [Mathematical models of virus spreading in multicomputer systems with different structures]. *Informatika i sistemy upravlenija - Information science and control systems*. 2012. no. 3(33). pp. 25-33. (In Russ.).
17. Semenov S.G., Davydov V.V. A Mathematical Model for Technology for Spreading Malicious Software across Heterogeneous Networks based on Markov Chains. *European researcher*. 2014. no. 1-1 (66). pp. 21-30. (In Russ.).
18. Utakaeva I.H., Kunizheva L.A. [A mathematical model of the spread of viruses on the network on prefractal graphs]. *Informacionnoe protivodejstvie ugrozam terrorizma - Counteracting the threats of terrorism*. 2012. no. 18. pp. 64-70. (In Russ.).

19. Kotenko I.V., Voroncov V.V., Ulanov A.V. [Models and systems of network worm propagation simulation]. *Trudy SPIIRAN - SPIIRAS Proceedings*. 2007. no. 4. pp. 225-238. (In Russ.).
20. Kotenko I.V., Shorov A.V. [Infrastructure attack protection mechanisms based on bio-inspired approach "Network Nervous System"]. *Voprosy zashchity informacii - Information security questions*. 2013. no. 2. pp. 57-66. (In Russ.).
21. Boyko A.A. [Method of Stratified Analytical Description of the Process of Functioning of Information-Technical Tools]. *Informacionnye tehnologii - Information Technologies*. 2015. no. 1. pp. 35-42. (In Russ.).
22. Boyko A.A., Budnikov S.A. [Model of information conflict between special software and information security subsystem of information-technical tool]. *Radiotekhnika - Radioengineering*. 2015. no. 4. pp. 136-141. (In Russ.).
23. Chikin M.G. [Method of analytical description of processes with a discrete set of states and exponential distributions of times of transitions]. *Informacionno-izmeritel'nye i upravljajushhie sistemy - Information-measuring and Control Systems*. 2004. no. 5. pp. 8-11. (In Russ.).
24. Koks D., Smit V. *Teoriya vosstanovlenija* [The theory of recovery]. M.: Sov. radio, 1967. 300 p. (In Russ.).

Бойко Алексей Александрович — к-т тех. наук, доцент, зам. начальника отдела, Военный учебно-научный центр Военно-воздушных сил «Военно-воздушная академия имени профессора Н.Е. Жуковского и Ю.А. Гагарина» (г. Воронеж). Область научных интересов: методы и системы защиты информации, методы оценки качества сложных систем. Число научных публикаций — 100. algeminy@mail.ru; ВУНЦ ВВС "ВВА", ул. Ст. Большеви хъков, д. 54А, г. Воронеж, 394064, РФ; р.т. +7(473)236-5228, факс +7(473)244-7860.

Boyko Aleksey Aleksandrovich — Ph.D., Tech., associate professor, associate chief of department, Military education-science center of Military aviation forces "Military aviation academy named for prof. N.E. Zhukovsky and J.A. Gagarin" (Voronezh, Russian Federation). Research interests: methods and systems of information protection, methods of assessing the quality of complex systems. The number of publications — 100. algeminy@mail.ru; MESC MAF "MAA", 54A, Old Bolsheviks street, Voronezh, 394064, Russia; office phone +7(473)236-5228, fax +(473)244-7860.

РЕФЕРАТ

Бойко А.А. Способ аналитического моделирования процесса распространения вирусов в компьютерных сетях различной структуры.

Предложен способ аналитического моделирования процесса распространения вирусов в компьютерной сети. Он учитывает особенности сетевой структуры, поведенческие характеристики вирусов и подсистем защиты информации узлов и возможность исходного заражения множества узлов различными вирусами.

Способ основан на представлении сети в виде модели с дискретными состояниями и временем переходов, распределенным по обобщенному закону Эрланга n -го порядка. В модели состояния процесса распространения вирусов отождествляются с узлами сети, а переходы между состояниями – со связями между этими узлами. Полученная модель процесса распространения вирусов преобразуется в непрерывную марковскую цепь путем введения псевдосостояний. Результирующая марковская цепь описывается в виде системы обыкновенных линейных дифференциальных уравнений.

Поведенческие характеристики вирусов и подсистем защиты информации узлов определяются следующими состояниями: вирус производит отказ в обслуживании, приводящий к невозможности выполнения любых задач; вирус использует функции узла в своих целях; вирус осуществляет вывод узла из строя; вирус размножается; вирус дезинформирует узел; вирус производит разведку; подсистема защиты информации узла проводит принудительный поиск вирусов; вирус осуществляет самомодификацию; вирус перешел в режим ожидания; вирус излечен.

Способ применим для случая, когда подсистемы защиты информации узлов запоминают излеченные ими вирусы, приобретая тем самым иммунитет к ним. Он может быть использован при исследовании конфликтной устойчивости сложных организационно-технических систем в условиях деструктивных информационно-технических воздействий.

SUMMARY

***Boyko A.A.* Method of Analytical Modeling of Spread of Viruses in Computer Networks with Different Structures.**

The article proposes a method of analytical modeling of the spread of viruses in computer networks. It takes into account the characteristics of the network structure, behavioral characteristics of viruses and information security subsystems of hosts and the possible source of infection of many hosts with different viruses.

The method is based on representing the network in the form of a model with discrete states and transitions, distributed by the generalized Erlang law of n-th order. In the model, the states of the process of viruses spreading are identified with the hosts of the network, and transitions between states - with connections between these hosts. The resulting model of the spread of viruses is converted to a continuous Markov chain by introducing pseudostates. The resulting Markov chain is described as a system of ordinary linear differential equations.

Behavioral characteristics of viruses and information security subsystems of hosts are determined by the following conditions: the virus produces a denial of service, resulting in the inability of the host to perform any task; the virus uses the functions of the host for its own purposes; the virus disables the host; the virus replicates; the virus misinforms the host; the virus produces intelligence; the information security subsystem of host conducts forcible virus search; the virus carries out self-modification; the virus has passed in the standby mode; the virus is cured.

The method is applicable for the case when information security subsystems of hosts remember cured viruses, thereby acquiring immunity to them. It can be used in the study of conflict stability of complex organizational-technical systems in terms of destructive information-technical influences.

М.О. ШУДРАК, В.В. ЗОЛОТАРЕВ
**МОДЕЛЬ, АЛГОРИТМЫ И ПРОГРАММНЫЙ КОМПЛЕКС
АВТОМАТИЗИРОВАННОГО ПОИСКА УЯЗВИМОСТЕЙ В
ИСПОЛНЯЕМОМ КОДЕ**

Шудрак М.О., Золотарев В.В. Модель, алгоритмы и программный комплекс автоматизированного поиска уязвимостей в исполняемом коде.

Аннотация. В настоящей работе рассматривается проблема автоматизированного поиска уязвимостей в исполняемом коде. В работе проводится анализ проблематики, и выделяются недостатки существующих решений, в части отсутствия возможности обнаружения уязвимостей с учётом тех угроз, которые они несут для защищаемой информации, которая обрабатывается в ПО. Для решения этой проблемы предлагается оригинальная модель автоматизированного поиска уязвимостей в трассе программы, её алгоритмическое обеспечение и программная реализация. В рамках модели приводятся формальные критерии отнесения ошибки к уязвимости с учётом распределения защищаемой информации в памяти программы. Для выделения участков памяти с защищаемой информацией в работе используется методика анализа помеченных данных. Кроме того приводится экспериментальная оценка эффективности разработанного программного комплекса, которая показала, что разработанное решение позволяет детектировать на 5 типов уязвимостей больше в ОС Windows и на 4 типа уязвимостей больше в Linux по сравнению с существующими аналогами. Все модули разработанного комплекса были опубликованы как ПО с открытым исходным кодом, могут свободно использоваться в других проектах и доступны для скачивания в Интернете.

Ключевые слова: уязвимость, исполняемый код, динамическое тестирование, ошибка, критерий.

Shudrak M.O., Zolotarev V.V. A Model, Algorithms and Software Tool for Vulnerabilities Detection in Machine Code.

Abstract. In the article we consider the problem of vulnerabilities detection in machine code. In this paper, disadvantages of current solutions in case of possibility to detect vulnerabilities in view of threats to confidential information that is processed in vulnerable software are highlighted. To solve this problem, we propose original model of vulnerabilities detection in program trace, its algorithmic support and software implementation. The model provides formal criteria to distinct bug from vulnerability taking into account distribution of protected information in the memory of software under test. We use tainted data analysis technique to highlight such memory regions. In addition, we conduct experimental evaluation of developed system efficiency which demonstrates that our solution allows detecting 5 types of Windows software vulnerabilities more and 4 types Linux software vulnerabilities more than existing analogs.

Keywords: vulnerability; machine code; dynamic analysis; bug; criteria.

1. Введение. Процесс разработки программного обеспечения невозможен без тех или иных ошибок, допускаемых программистами или архитекторами в ходе своей работы. В свою очередь такие ошибки могут приводить к уязвимостям в программном обеспечении, которые могут использоваться злоумышленниками для компрометации конфиденциальной информации обрабатываемой уязвимым ПО. Ошибки в

программном обеспечении (ПО) в свою очередь являются следствием развития современных информационных технологий в части увеличения объема, сложности и требований к масштабируемости программного кода.

В данной работе вопрос анализа безопасности ПО будет рассматриваться в условии отсутствия исходного текста программы. Актуальность анализа программ без исходного кода во многом обусловлена высоким уровнем распространения проприетарного ПО, исходный код которого представляет собой интеллектуальную собственность той или иной компании, которые, как правило, не заинтересованы в публикации или передачи своих разработок третьим лицам. Также даже при наличии исходного кода существует проблема того, что преобразования выполняемые компилятором и различными оптимизаторами могут значительно изменить поведение приложения, сделав результаты аудита исходного кода недостоверными. В зарубежной литературе эта проблема носит название «What You See Is Not What You eXecute» [1].

Структурно данная работа будет построена следующим образом: во втором разделе будет рассмотрена проблема поиска уязвимостей в исполняемом коде. Будут выделены критерии отнесения ошибки к уязвимости, рассмотрена нормативно-методическая база, а также выделены недостатки существующих методик и программных систем. В третьем разделе будет описана новая модель автоматизированного поиска уязвимостей в трассе программы, будут формально определены критерии возникновения ошибки в исполняемом коде и их отнесения у уязвимости с учётом распределения защищаемой информации в памяти программы. Кроме того будет приведено алгоритмическое обеспечение модели (раздел 4), а также программная реализация этих алгоритмов (раздел 5). В 6 разделе будут приведены результаты сравнительной оценки эффективности разработанного комплекса с аналогами, а в заключительном разделе будут сделаны выводы и дальнейшие перспективы работы.

2. Описание проблемы. Согласно ГОСТам и другим нормативно – правовым документам РФ отдельного понятия для *программной ошибки, программного дефекта* или *уязвимости в программном коде* не существует, однако в [2] *дефект* определяется как каждое отдельное несоответствие продукции установленным к ней требованиям. В [3] автор подчеркивает, что дефект является более общим понятием и объединяет в себе понятия ошибки и уязвимости. Таким образом, под *уязвимостью в ПО* будем понимать дефект безопасности ПО, обуславливающий возможность реализации угроз безопасности обрабатываемой в ПО защищаемой информации.

Выше было отмечено, что причиной уязвимости является ошибка, приводящая к нарушению безопасности обрабатываемой в ПО информации. Опишем в таблице 1 все возможные типы ошибок, приводящие к нарушению безопасности защищаемой информации. Отметим, что там, где в классе угроз находится нарушение безопасности, следует понимать, что тип ошибок потенциально может приводить к нарушению целостности, доступности или конфиденциальности защищаемой информации.

Таблица 1. Описание классов угроз нарушения безопасности в зависимости от типа ошибки и критерии отнесения ошибки к уязвимости

Тип ошибки	Класс угроз	Критерии отнесения ошибки к уязвимости
Переполнение кучи	Нарушение целостности, доступности или конфиденциальности обрабатываемой в ПО защищаемой информации	Есть возможность перезаписать другой фрагмент кучи, содержащий защищаемую информацию или перезаписи информации, управляющей потоком управления программы. Есть возможность вызова необработанного исключения.
Переполнение стека		Есть возможность перезаписать другой фрагмент стека, содержащий защищаемую информацию или перезаписи информации, управляющей потоком управления программы. Есть возможность вызова необработанного исключения.
Ошибки форматирования строк		Есть возможность перезаписать другой фрагмент стека, содержащий защищаемую информацию или перезаписи информации, управляющей потоком управления программы. Есть возможность вызова необработанного исключения.
Использование неинициализированной памяти	Нарушение доступности обрабатываемой информации	Есть возможность вызова необработанного исключения в тестируемой программе
Использование освобожденной памяти	Нарушение конфиденциальности или доступности обрабатываемой в ПО защищаемой информации	Есть возможность вызова необработанного исключения или чтения защищаемой информации, которая ранее была освобождена
Обращение по недопустимому адресу	Нарушение доступности обрабатываемой в ПО защищаемой информации	Есть возможность вызова необработанного исключения в тестируемой программе

Отдельно необходимо остановиться на понятии защищаемой информации. Согласно [4] к защищаемой информации относится информация подлежащая защите в соответствии с требованиями правовых документов или требованиями, устанавливаемыми собственником этой информации. В рамках тестирования часто нет информации о том, какие конкретно защищаемые данные будут обрабатываться в программе, однако для выполнения поиска уязвимостей достаточно иметь адреса памяти, в которых эта информация будет храниться. В этом случае, до начала тестирования достаточно сформировать и пометить тестовые защищаемые данные (которые будут обрабатываться в ПО, как защищаемая информация), а затем проанализировать, то, как эти данные будут обрабатываться, и сохранить информацию о том, какие участки памяти участвовали в этом процессе. Подробный алгоритм поиска участков памяти с защищаемой информацией будет приведен далее в работе. Кроме того к защищаемым участкам памяти могут относиться те участки памяти, которые отвечают за управление программой и её бесперебойное функционирование.

На сегодняшний день для поиска уязвимостей в условиях отсутствия исходного кода используется динамический и статический анализ бинарного кода. Так в рамках автоматизированного статического анализа могут использоваться методики поиска потенциальных уязвимостей по шаблонам [5] или символическое исполнение программы [6]. Однако фундаментальные проблемы, связанные с декомпиляцией и анализом бинарного кода не позволяют полностью восстановить высокоуровневый аналог скомпилированного модуля, что создает серьезные трудности для статического анализа, а рост числа потенциальных путей исполнения в рамках символического исполнения ведёт к проблеме «экспоненциального взрыва» [6].

Для решения этих проблем применяются методики динамического анализа, отчасти решающие проблемы статического анализа, однако не решающие вопрос с покрытием кода при тестировании, а также порождающие проблемы, связанные с влиянием анализатора на тестируемое приложение. При этом для некоторых инструментов тестирования может быть характерен высокий уровень потребления вычислительных ресурсов [7]. Также важно отметить, что многие методики динамического анализа характеризуются направленностью на обнаружение уязвимостей, связанных с нарушением доступности обрабатываемой информации, которые не позволяют детектировать некоторые типы уязвимости, не приводящие к вызову исключения в тестируемом приложении. Примером может служить ситуация, продемонстрированная в листинге 1.

```

1  int main(){
2      char buffer[10];
3      char overflow_string[9];
4      int a = 0;
5      fgets(buffer, sizeof buffer, stdin);
6      strcpy(overflow_string, buffer);
7      if (a == 0)
8          printf("expected behavior");
9      else
10         printf("overflow occurred");
11     return 0;
12 }

```

Листинг 1. Пример переполнения 1 байта в стеке

Согласно листингу 1, в случае передачи этому приложению строки длиной в 10 байт, произойдет переполнение (строка 6), так как принимающий буфер на 1 байт меньше, чем исходный буфер. Произойдет переполнение одного байта, что может привести к перезаписи следующих за ним данных. Такая ситуация может и не привести к отказу в обслуживании, и приложение продолжит свою работу штатно. Например, в вышеизложенном примере может произойти вывод строки «overflow occurred» в зависимости от того, как будут располагаться данные в стеке.

Отметим, что для отслеживания ситуаций таких, как в вышеизложенном примере, необходимо отслеживать ошибочные обращения к памяти в тестируемом приложении. Для решения этой проблемы существует методика, в рамках которой осуществляется поиск уязвимостей в программе путем учета взаимодействия программы с памятью в ходе обработки тестовых данных (таким образом, генерация тестовых данных является задачей эксперта или может выполняться автоматически с помощью фаззера) [8, 9]. Для реализации этого подхода учитывается каждый байт памяти тестируемого приложения, который может иметь три состояния:

- недоступная память. Память, которая недоступна приложению;
- неинициализированная память. Память, которая адресуема, но доступ к которой не должен быть доступен приложению до тех пор, пока эта память не будет выделена;
- инициализированная память. Память, которая адресуема и доступна на запись или чтение приложению.

Информация о состоянии памяти сохраняется в так называемой «теневого памяти» - специальная область в памяти, которая описывает совокупность состояний памяти в тестируемом приложении. Тестирование программы заключается в проверке корректности работы программы с каждым байтом памяти и сигнализации об ошибках в случае

некорректного обращения к ней. На сегодняшний день существует только два инструмента, реализующих эту методику без доступа к исходному коду. Это системы DrMemory [10] в основе которой лежит среда динамической бинарной инструментации (ДБИ) DynamoRIO [8] и Memcheck в основе которой лежит среда ДБИ Valgrind [9].

Крайне важно отметить, что данные системы направлены преимущественно на обнаружение ошибок в программном обеспечении и не проводят оценку тех угроз, которые они несут для защищаемой информации, обрабатываемой в программе (отметим, что данное утверждение будет подтверждено экспериментальными данными далее в работе). Эти недостатки порождают необходимость поиска новых подходов и решений в этой области.

3. Модель автоматизированного поиска уязвимостей. Рассмотрим работу программы как систему с конечным множеством состояний, в которые она последовательно переходит в ходе своей работы. Будем выполнять поиск уязвимостей в трассе по завершению работы программы. Под трассой программы следует понимать упорядоченную последовательность выполненных инструкций, значений регистров и состояний памяти после выполнения каждой инструкции.

Представим состояние регистров программы после выполнения каждой инструкции в трассе в виде вектора $s = (eax, ebx, ecx, edx, esi, edi, ebp, esp, eip)$, описывающего значения регистров процессора, где $eax, ebx, ecx, edx, esi, edi, ebp, esp, eip$ – рассматриваемые в работе регистры процессора. Представим трассу программы в виде системы упорядоченных векторов: s_0, s_1, \dots, s_n , где n – количество выполненных инструкций, s_0 – начальное состояние, s_n – заключительное состояние.

Для поиска уязвимостей определим L как множество всех адресов в памяти доступных программе для обращения. Также определим $P_i = f(s_i)$ как множество фактических указателей в память из состояния s_i и $P_i \subseteq L, i \in [0; n]$. Отметим, что функция f будет задана алгоритмически в разделе 3.

В свою очередь пусть имеется M_i – множество доступных адресов в памяти из состояния s_i и $M_i \subseteq L$. Алгоритм определения этого множества будет описан позднее в разделе 4.

Пусть $Lprot_i \subseteq L$ некоторое подмножество фрагментов памяти, хранящих защищаемую информацию в рамках состояния s_i . Алгоритм автоматизированного поиска фрагментов памяти, хранящих защищаемую информацию, будет приведен позднее в разделе 4.1.

Определив $Lprot_i, M_i$ и P_i , можно записать вектор состояния s_i следующим образом:

$$s_i = (eax_i, ebx_i, ecx_i, edx_i, esi_i, edi_i, ebp_i, esp_i, eip_i, P_i, M_i, Lprot_i)$$

Таким образом, трасса программы также может быть представлена с помощью матрицы состояний в следующем виде:

$$SM = \begin{pmatrix} eax_0 & ebx_0 & ecx_0 & edx_0 & edi_0 & esi_0 & esp_0 & ebp_0 & eip_0 & P_0 & M_0 & Lprot_0 \\ eax_1 & ebx_1 & ecx_1 & edx_1 & edi_1 & esi_1 & esp_1 & ebp_1 & eip_1 & P_1 & M_1 & Lprot_1 \\ \vdots & \vdots \\ eax_i & ebx_i & ecx_i & edx_i & edi_i & esi_i & esp_i & ebp_i & eip_i & P_i & M_i & Lprot_i \\ \vdots & \vdots \\ eax_n & ebx_n & ecx_n & edx_n & edi_n & esi_n & esp_n & ebp_n & eip_n & P_n & M_n & Lprot_n \end{pmatrix}$$

Обозначим через γ функцию, которая позволяет определить, является ли состояние s_i допустимым или не допустимым следующим образом:

$$\gamma(s_i, L) = \begin{cases} 2, & \text{если } P_i \cap (Lprot_i \setminus M_i) \neq \emptyset \\ 1, & \text{если } ((P_i \cap Lprot_i = \emptyset) \wedge (P_i \cap (L \setminus M_i) \neq \emptyset)) \vee (P_i \notin L) \\ 0, & \text{во всех других случаях} \end{cases} \quad (1)$$

2 – недопустимое состояние и есть уязвимость;

1 – недопустимое состояние и есть ошибка;

0 – допустимое состояние.

Рассмотрим детально каждый критерий из выражения 1 в таблице 2.

Таблица 2. Описание каждого критерия из выражения 1.

Условие	Описание критерия	Тип ошибки/уязвимости
$P_i \cap (Lprot_i \setminus M_i) \neq \emptyset$	Как минимум один из элементов множества фактических указателей принадлежит множеству защищаемых участков памяти и не принадлежит множеству адресов памяти доступных из состояния s_i	Переполнение буфера, переполнение кучи, использование ранее освобожденной памяти, ошибки форматирования строк
$(P_i \cap Lprot_i = \emptyset) \wedge (P_i \cap (L \setminus M_i) \neq \emptyset)$	Ни один из элементов множества фактических указателей не принадлежит множеству защищаемых участков памяти и как минимум один из элементов множества фактических указателей принадлежит множеству выделенных участков памяти и этот участок не доступен из s_i	Переполнение буфера, переполнение кучи, использование ранее освобожденной памяти, использование неинициализированной памяти, ошибки форматирования строк
$P_i \notin L$	Как минимум один из элементов множества фактических указателей не принадлежит доступному для обращения множеству адресов памяти	Обращение по недопустимому адресу

4. Алгоритм автоматизированного поиска уязвимостей в трассе. Рассмотрим алгоритм автоматизированного поиска уязвимостей в трассе на рисунке 1.

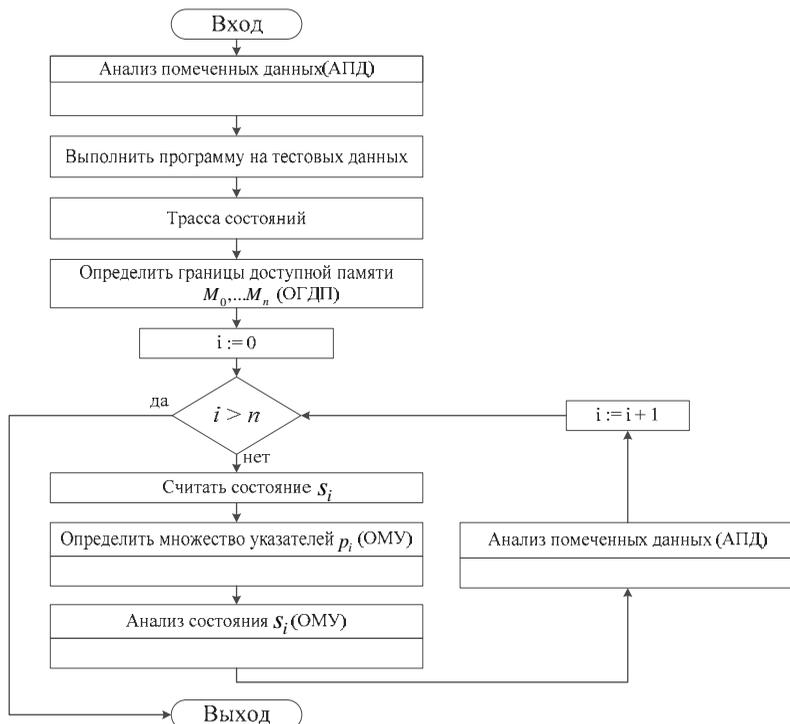


Рис. 1. Алгоритм автоматизированного поиска уязвимостей в трассе

На первом этапе алгоритма производится определение участков памяти, которые задействованы в обработке защищаемой информации (алгоритм определения таких участков будет приведен далее в работе). Затем программа выполняется на тестовых данных, в результате чего имеем трассу размером n инструкций.

На следующем этапе вступает в работу алгоритм определения границ доступной памяти (ОГДП). Рассмотрим его детально на рисунке 2. В качестве параметра передается итератор i равный нулю. На первом шаге производится вход в цикл считывания инструкций и выполняется дизассемблирование инструкции по адресу eip_i . Затем, если выполняется выделение или освобождение памяти, определяется множество адресов, которые затем добавляются или исключаются из M_{i+1} . Затем, если размер выделенного участка стека увеличивается или

уменьшается, то в множество M_{i+1} , добавляется или исключается множество новых доступных/не доступных адресов. В том случае если инструкция является вызовом процедуры, то производится определение множества передаваемых адресов памяти в качестве аргументов процедуры, а затем производится рекурсивный вызов алгоритма ОГДП с параметром $i = i+1$.

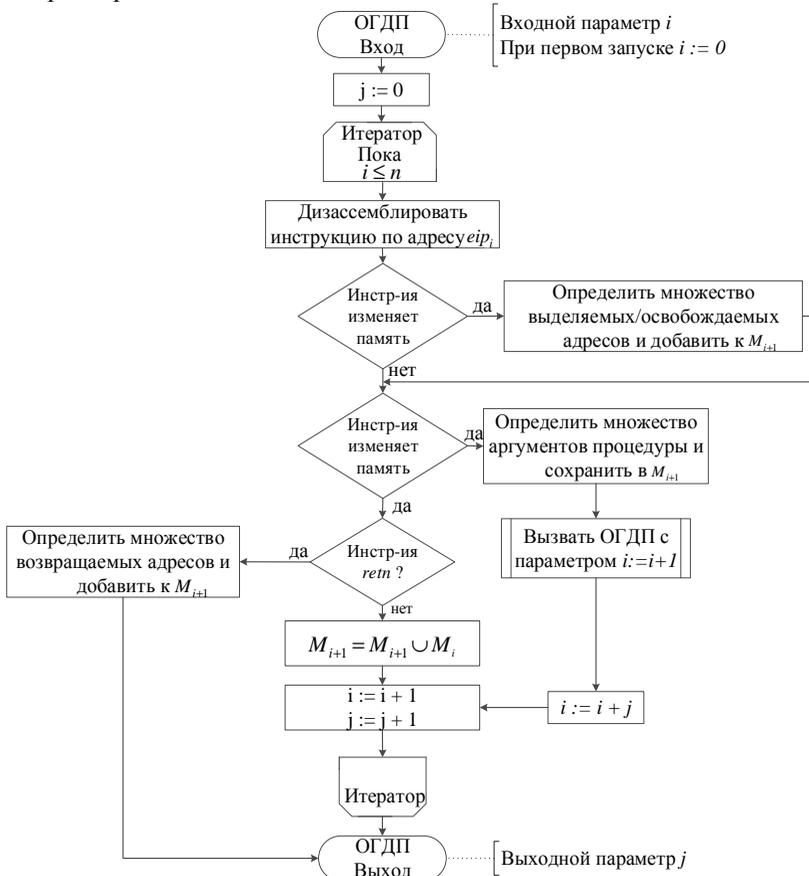


Рис. 2. Алгоритм определения границ доступной памяти (ОГДП)

Таким образом, описанный на рисунке 2 алгоритм является рекурсивным. Возврат из рекурсии происходит при достижении инструкции `getn`, для которой определяется множество возвращаемых адресов памяти, которые сохраняются в M_{i+1} . Переменная j определяет

количество проанализированных инструкций в рекурсии и позволяет выполнить переход к последующему после вызова call состоянию.

На следующем этапе согласно алгоритму выполняется последовательное считывание состояний s_i . Для каждого состояния последовательно выполняются два алгоритма: определение множества указателей P_i (ОМУ) и анализ состояния s_i (АС) на основе M_i и P_i с помощью выражения 1.

Для описания алгоритма определения множества фактических указателей рассмотрим блок – схему на рисунке 3.

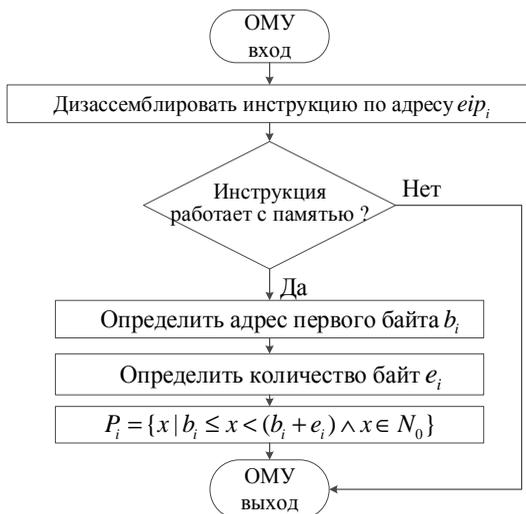
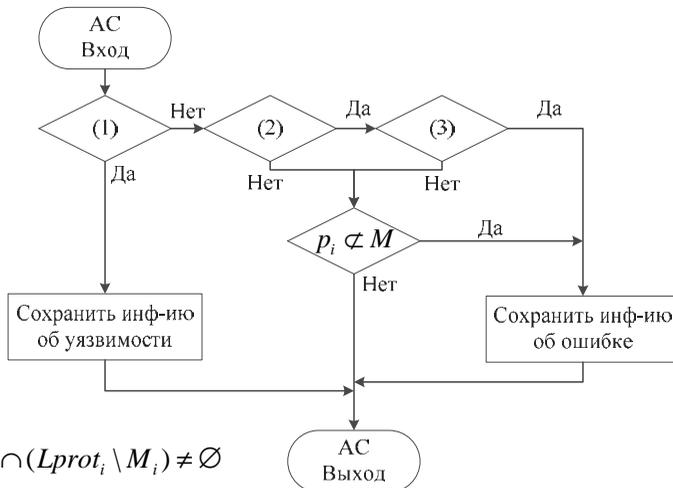


Рис. 3. Алгоритм определения множества указателей P_i (ОМУ)

На первом шаге алгоритма ОМУ выполняется считывание и дизассемблирование инструкции по адресу eip_i . Затем в том случае, если инструкция работает с памятью выполняется определение адреса в памяти первого байта, к которому осуществляется доступ (обозначим его как b_i), а затем определяется количество считываемых или записываемых байт (обозначим это количество через e_i). Тогда элементы множества P_i будут определены следующим образом: $P_i \in \{x | b_i \leq x < (b_i + e_i) \wedge x \in N_0\}$.

Получив множества P_i и M_i , алгоритм переходит к анализу состояния (АС). Рассмотрим этот алгоритм на рисунке 4. На вход алгоритму поступают множества M_i и P_i . Затем последовательно выполняется проверка принадлежности множества фактических указателей в соответствии с критериями из выражения 1. В том случае, если один из критериев определяет факт наличия ошибки или уязвимости, инфор-

мация об этом сохраняется, и анализ продолжается для следующего состояния до тех пор, пока не будут считаны все состояния в трассе. Отметим, что после определения допустимости состояния s_i выполняется анализ помеченных данных и расчёт $Lprot_{i+1}$ (при $i = n$ не выполняется) для того, чтобы учесть возможные изменения во множестве защищаемых участков памяти при переходе в следующее состояние.



- (1) – $P_i \cap (Lprot_i \setminus M_i) \neq \emptyset$
- (2) – $P_i \cap Lprot_i = \emptyset$
- (3) – $P_i \cap (L \setminus M_i) \neq \emptyset$

Рис. 4. Алгоритм анализа состояния s_i (AC)

4.1. Алгоритм анализа помеченных данных. Разработав модель и алгоритм автоматизированного поиска уязвимостей в трассе программы, рассмотрим алгоритм анализа помеченных данных. Согласно модели для выполнения поиска уязвимостей необходимо знать адреса памяти, в которых будет храниться защищаемая информация. Учитывая недостатки методов статического анализа, было принято решение выполнять анализ памяти в процессе исполнения программы. Для этого на подготовительном этапе тестирования необходимо подать в программу тестовые защищаемые данные, которые могут быть сформированы как разработчиком будущего программного продукта, так и владельцем защищаемой информации. В ходе обработки этих данных вступает в работу алгоритм анализа помеченных данных. Ключевым моментом для понимания работы алгоритма является понятия помеченных и не помеченных данных, а также помеченных и не помеченных участков памяти.

Под помеченными данными понимаются защищаемые данные, а под непомеченными соответственно любые другие данные используемые в программе. Рассмотрим алгоритм анализа помеченных и не помеченных данных на рисунке 5.

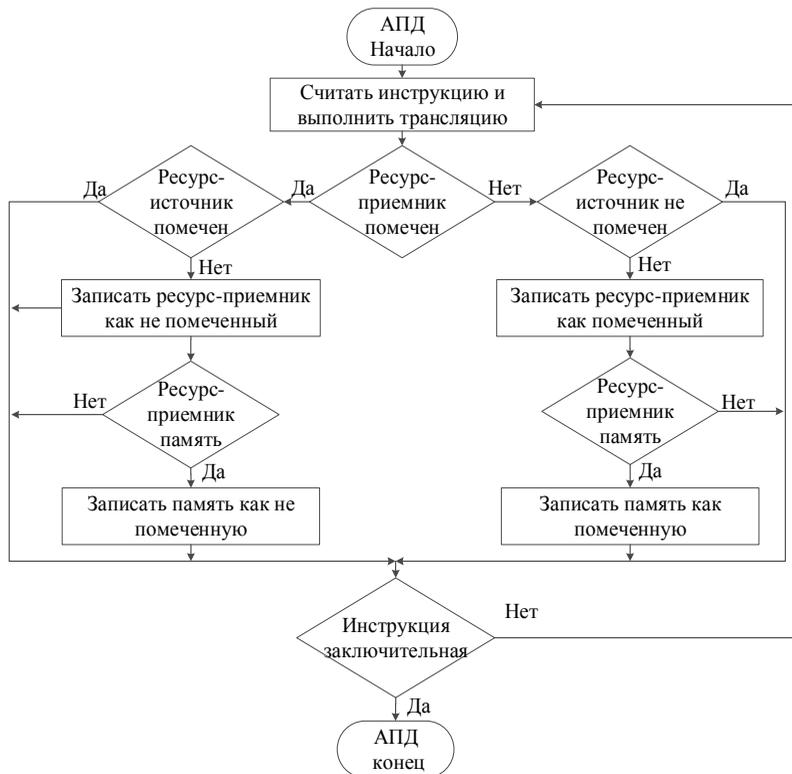


Рис. 5. Алгоритм анализа помеченных данных

В алгоритме реализуются два ключевых правила анализа продвижения помеченных данных в программе:

1) Если помеченные данные записываются или считываются в не помеченный ресурс (регистр или ячейка памяти), то этот ресурс становится помеченным.

2) Если не помеченные данные записываются или считываются в помеченный ресурс, этот ресурс также становится помеченным.

Кроме того для анализа того, какую операцию выполняет инструкция, в алгоритме производится трансляция инструкции в набор высокоуровневых операций. Методика подробно описана в данной работе [11].

5. Описание программного комплекса. Рассмотрим общую архитектуру программного комплекса на рисунке 6. Отметим, что отладчик используется для мониторинга исключительных ситуаций при работе приложения. Согласно рисунку для сбора трассы и анализа помеченных данных используется система ДБИ Intel PIN [12] ввиду её превосходящей производительности над аналогами [13]. В рамках этой системы была реализована специальная утилита, реализующая возможность выполнения анализа помеченных данных и сбора трассы для последующего анализа и поиска уязвимостей.

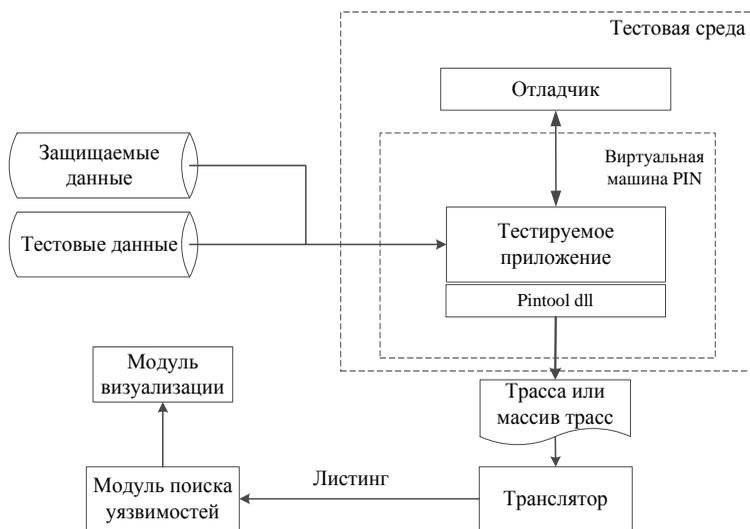


Рис. 6. Архитектура программного комплекса

Для анализа того, как помеченные данные обрабатываются в тестируемом приложении, прежде всего, необходимо учитывать потенциальные точки ввода этих данных в приложение. В качестве входных данных в работе рассматриваются сетевые данные, файлы или аргументы командной строки, а в качестве точек входа системные вызовы ОС (например, `fgets` или `recv`). Кроме того, необходимо отметить, что согласно алгоритму на рисунке 5 в качестве помеченных данных также выступают и адреса в памяти, задействованные в управлении программой (адреса возврата на стеке, таблица смещений виртуальных адресов процедур и т.п.) Программа выполняется на тестовых данных и параллельно осуществляется анализ продвижения помеченных данных в памяти и регистрах ПО.

На следующем этапе программа последовательно выполняется на каждом экземпляре тестовых данных, в результате чего имеем трассу или массив трасс (если программа требует перезапуск после каждой тестовой итерации). Затем осуществляется трансляция трассы, выделение инструкций оперирующих с памятью, определение множества фактических указателей, множества доступных участков памяти и проверка условий согласно алгоритму на рисунке 4 для каждого состояния в каждой из полученных трасс.

6. Экспериментальная оценка эффективности программного комплекса. Для сравнительной оценки эффективности системы поиска уязвимостей в трассе с аналогами, случайным образом было отобрано 24 уязвимых приложений из международной базы уязвимостей exploit-db на каждый тип уязвимости, порождающий каждый класс угроз к защищаемой информации (таблица 3). По 12 приложений для ОС Windows и Linux.

Таблица 3. Описание типов отобранных уязвимостей на каждый класс угроз

Класс угроз	Тип уязвимости	Описание уязвимости
Нарушение целостности защищаемой информации	Переполнение кучи	Запись за границами кучи
	Переполнение стека	Запись за границами стека
	Ошибка форматирования строк	Запись за границами стека
Нарушение доступности защищаемой информации	Использование неинициализированной памяти	Обращение к неинициализированной памяти
	Переполнение кучи	Перезапись управляющей информации, находящейся за границами кучи
	Переполнение стека	Перезапись адреса возврата из стека
	Ошибка форматирования строк	Перезапись управляющей информации за границами выделенной памяти
	Обращение по недопустимому адресу	Обращение по недопустимому адресу в памяти
Нарушение конфиденциальности защищаемой информации	Использование освобожденной памяти	Чтение ранее освобожденной памяти в куче, содержащей защищаемую информацию
	Переполнение стека	Чтение защищаемой информации за границами стека
	Ошибка форматирования строк	Чтение защищаемой информации за границами стека
	Переполнение кучи	Чтение защищаемой информации за границами кучи

Каждый программный продукт разворачивался в собственной виртуальной среде на базе ОС Ubuntu Linux 12.10 и Windows 7 в следующей аппаратной конфигурации: x86 2 ГБ ОЗУ, Intel Core i7-3630QM, 2.4 GHz. Результаты работы систем приведены в таблице 4.

Таблица 4. Результаты экспериментального анализа систем детектирования уязвимостей (нет – уязвимость не обнаружена, да – уязвимость обнаружена)

Тип ошибки, приводящей к уязвимости	Наименование ПО и ОС. Идентификатор уязвимости	DrMemory	MemCheck	Разра- бот. система
Ошибки форматирования строк (конфиденциальность)	Dpkg 1.16.1.2 CVE-2014-8625	Да (Lin.)	Да (Lin.)	Да (Lin.)
	Mshhtml.dll 6.0.2900.2853 Дата публикации: 03.09.2013	Нет (Win.)	Нет под- держки	Да (Win.)
Ошибки форматирования строк (целостность)	Yelp 2.23.1 CVE-2008-3533	Да (Lin.)	Да (Lin.)	Да (Lin.)
	ZipItFast 3.0 Дата публикации: 08.07.2011	Да (Win.)	Нет под- держки	Да (Win.)
Ошибки форматирования строк (доступность)	Dia 0.94 CVE-2006-2480	Да (Lin.)	Да (Lin.)	Да (Lin.)
	Opera 2.15 Дата публикации: 02.07.2013	Да (Win.)	Нет под- держки	Да (Win.)
Обращение по недопустимому адресу (доступность)	Squid 3.3.5 CVE-2013-4123	Нет (Lin.)	Да (Lin.)	Да (Lin.)
	KNeT Web Server 1.04 CVE-2005-0575	Да (Win.)	Нет под- держки	Да (Win.)
Переполнение стека (конфиденциальность)	Mozilla Firefox 3.0 CVE-2009-3382	Нет (Lin.)	Нет (Lin.)	Да (Lin.)
	Ezhometech EZ Server 6.4 CVE-2012-0618	Нет (Win.)	Нет под- держки	Да (Win.)
Переполнение стека (целостность)	Aireplay-ng 1.2 b3 CVE-2014-8322	Да (Lin.)	Да (Lin.)	Да (Lin.)
	Qualcomm Imapd 9.0.333.0 CVE-2014-10031	Да (Win.)	Нет под- держки	Да (Win.)

Продолжение таблицы 4.

Тип ошибки, приводящей к уязвимости	Наименование ПО и ОС. Идентификатор уязвимости	DrMemory	MemCheck	Разра- бот. система
Переполнение стека (доступность)	Unrar 3.9.3 Дата публикации: 05.08.2011	Да (Lin.)	Да (Lin.)	Да (Lin.)
	Httpdx 1.4.5 CVE-2009-4769	Да (Win.)	Нет под- держки	Да (Win.)
Переполнение кучи (конфиденциальность)	OpenSSL 1.0.1 CVE-2014-0160	Нет (Lin.)	Нет (Lin.)	Да (Lin.)
	XM Easy FTP Server 5.3.0 CVE-2007-1195	Нет (Win.)	Нет под- держки	Да (Win.)
Переполнение кучи (целостность)	inetutils 1.8.1 (ftp) Дата публикации: 07.12.2010	Нет (Lin.)	Нет (Lin.)	Да (Lin.)
	UltraVNC 1.0.1 CVE-2007-1195	Нет (Win.)	Нет под- держки	Да (Win.)
Переполнение кучи (доступность)	make 3.81. Дата публикации: 24.07.2014	Нет (Lin.)	Нет (Lin.)	Да (Lin.)
	Opera 2.12 CVE-2013-16380	Нет (Win.)	Нет под- держки	Да (Win.)
Использование освобожденной памяти (конфиденциальность)	OpenSSL 1.0.0a CVE-2010-2939	Нет (Lin.)	Нет (Lin.)	Нет (Lin.)
	ESTSoft ALPPlayer 2.0 Дата публикации: 06.07.2011	Да (Win.)	Нет под- держки	Да (Win.)
Использование неинициализированной памяти (доступность)	OpenLiteSpeed 1.3.9 Дата публикации 18.05.2015	Да (Lin.)	Да (Lin.)	Да (Lin.)
	UMPlayer 0.95 (QTGui4.dll) Дата публикации: 29.11.2012	Да (Win.)	Нет под- держки	Да (Win.)
Все	Итого	6/12 (Lin.) 7/12(Win.)	7/12 (Lin.)	11/12 (Lin.) 12/12 (Win.)

Дадим пояснения к таблице и эксперименту. В рамках эксперимента каждое приложение запускалось с использованием системы детектирования уязвимостей, а затем в приложение подавались тестовые

данные, которые приводили к срабатыванию уязвимости, после чего проводился анализ результата работы системы. Отметим, что система MemCheck не поддерживает ОС Windows, поэтому возможность её тестирования в этой ОС отсутствует.

Таким образом, разработанная система выполнила обнаружения 23 из 24 уязвимостей в приложениях для ОС Windows и Linux, что на 4 типа уязвимостей больше, чем обнаружили системы Memcheck и DrMemory в Linux и на 5 типов уязвимостей больше системы DrMemory в Windows.

Данный результат обусловлен тем, что разработанная система позволяет выполнять детектирование тех типов уязвимостей, которые не поддерживаются другими системами. Система выполнила обнаружение следующих типов уязвимостей, которые не были обнаружены другими системами:

- нарушение конфиденциальности защищаемой информации вследствие реализации уязвимости типа переполнение стека, переполнения кучи и использования освобожденной памяти (только Windows);
- переполнение кучи, приводящее к нарушению целостности и доступности защищаемой информации.

Отметим, что пропущенная уязвимость в Linux обусловлена внутренней ошибкой в среде ДБИ, а не разработанной системой.

7. Заключение. В данной работе была предложена оригинальная модель автоматизированного поиска уязвимостей в исполняемом коде и её алгоритмическое обеспечение.

Разработанная модель автоматизированного поиска уязвимостей в трассе программы позволила формально определить критерии возникновения уязвимости в исполняемом коде и осуществлять поиск уязвимостей для каждого выполненного состояния в трассе с учетом распределения защищаемой информации в памяти программы. В свою очередь алгоритмическое обеспечение модели позволило реализовать модель в виде программного комплекса.

Проведенная экспериментальная оценка программного комплекса программы показала, что разработанное решение позволяет детектировать на 5 типов уязвимостей больше в ОС Windows и на 4 типа уязвимостей больше в Linux по сравнению с существующими аналогами. Все модули разработанного комплекса были опубликованы как ПО с открытым исходным кодом, могут свободно использоваться в других проектах и доступны для скачивания в Интернете [14].

Дальнейшие перспективы видятся в использовании методики поиска уязвимостей в памяти анализируемого приложения в обход стандартных интерфейсов обработки пользовательских данных, а также расширение программного комплекса для тестирования программ отличных от x86/64 архитектур процессора.

Литература

1. Gogul B., Repts T., Melski D., Teitelbaum T. WYSINWYX: What you see is not what you execute // ACM Transactions on Programming Languages and Systems. 2010. vol. 32. no. 6. pp. 202–213.
2. Исследование международной базы уязвимостей CVE. URL: <http://opennet.ru/opennews/art.shtml?num=36287> (дата обращения 17.08.2015).
3. Аветисян А.И., Белеванцев А.А., Чукуляев И.И. Технологии статического и динамического анализа уязвимостей программного обеспечения // Вопросы кибербезопасности. 2014. № 3 (4). С. 20–28.
4. ГОСТ Р 50922-2006 Защита информации. Основные термины и определения // М.: Стандартинформ. 2008.
5. Dowd M., McDonald J., Schuh J. The art of software security assessment: Identifying and preventing software vulnerabilities // Boston. USA: Addison-Wesley Professional. 2006. 1244 p.
6. Каушан В.В., Маркин Ю.В., Падалян В.А., Тихонов А.Ю. Методы поиска ошибок в бинарном коде // Технический отчет Института системного программирования РАН. 2013. № 2013-1. 42 с.
7. Clarke T. Fuzzing for software vulnerability discovery: technical report // Department of Mathematic. Royal Holloway University of London. 2009. 178 p.
8. Bruening D., Zhao O. Practical memory checking with Dr. Memory // In Proceedings of the 9th Annual IEEE/ACM International Symposium on Code Generation and Optimization. 2011. pp. 213–223.
9. Nethercote N., Seward J. Valgrind: a framework for heavyweight dynamic binary instrumentation // ACM Sigplan notices. 2007. vol. 42(6). pp. 89–100.
10. Программный комплекс DrMemory. URL: www.drmemory.org (дата обращения 17.08.2015).
11. Шудрак М.О., Лубкин И.А., Золотарев В.В. Методика декомпиляции бинарного кода и её применение в сфере информационной безопасности // Безопасность информационных технологий НИЯУ МИФИ. Москва. 2012. №3. С. 75–80.
12. Luk C.K., Cohn R., Muth R., Patil H., Klauser A., Lowney G., Hazelwood K. Pin: building customized program analysis tools with dynamic instrumentation // In Acm Sigplan Notices. 2005. vol. 40(6). pp. 190–200.
13. Uh G. R., Cohn R., Yadavalli B., Peri R., Ayupari R. Analyzing dynamic binary instrumentation overhead // In WBIA Workshop at ASPLOS. 2006.
14. Репозиторий с исходным кодом разработанной системы. URL: <https://github.com/MShudrak/tvc> (дата обращения 19.08.2015).

References

1. Gogul B., Repts T., Melski D., Teitelbaum T. WYSINWYX: What you see is not what you execute. ACM Transactions on Programming Languages and Systems. 2010. vol. 32. no. 6. pp. 202–213.
2. Issledovanie mezhdunarodnoj bazy uyazvimostej CVE [A taxonomy of the international vulnerabilities database CVE]. Available at: <http://opennet.ru/opennews/art.shtml?num=36287> (accessed 17.08.2015). (In Russ.).
3. Avetisan A.I., Belevancev A.A., Chukliaev I.I. [The technologies of static and dynamic analyses for detecting software vulnerabilities] *Voprosy kiberbezopasnosti – CyberSecurity Issues*. 2014. vol. 3(4). pp. 20 – 28. (In Russ.).
4. GOST R 50922-2006 [Information Security. Terms and Definitions]. M.: Standartinform. 2008. (In Russ.).
5. Dowd M., McDonald J., Schuh J. The art of software security assessment: Identifying and preventing software vulnerabilities. Boston. USA: Addison-Wesley Professional. 2006. 1244 p.
6. Kaushan V.V., Markin U.V., Padarian V.A. Tihonov A.U. [Techniques of bugs detection in machine code] *Texnicheskij otchet Tehnicheskij otchet Instituta sistemnogo*

programmirovaniya RAN – Technical report of Institute of System Programming RAS. 2013. no. 2013-1. 42 p. (In Russ.).

7. Clarke T. Fuzzing for software vulnerability discovery: technical report. Department of Mathematic, Royal Holloway University of London. 2009. 178 p.
8. Bruening D., Zhao Q. Practical memory checking with Dr. Memory. In Proceedings of the 9th Annual IEEE/ACM International Symposium on Code Generation and Optimization. 2011. pp. 213–223.
9. Nethercote N., Seward J. Valgrind: a framework for heavyweight dynamic binary instrumentation. ACM Sigplan notices. 2007. vol. 42(6). pp. 89–100.
10. Программный комплекс DrMemory [DrMemory system]. Available at: www.drmemory.org (accessed 17.08.2015). (In Russ.).
11. Shudrak M.O., Lubkin I.A., Zolotarev V.V. [The technique of binary code decompilation and its application in cyber security sphere]. *Bezopasnost' informacionnyx tehnologij NIYaU MIFI – Security of Information Technologies MEPhI*. 2012. vol. 3. pp. 75–80. (In Russ.)
12. Luk C. K., Cohn R., Muth R., Patil H., Klauser A., Lowney G., Hazelwood, K. Pin: building customized program analysis tools with dynamic instrumentation. In *Acm Sigplan Notices*. 2005. vol. 40. no. 6. pp. 190–200.
13. Uh G. R., Cohn R., Yadavalli B., Peri R., Ayyagari R. Analyzing dynamic binary instrumentation overhead. In *WBIA Workshop at ASPLOS*. 2006.
14. Repozitorij s ishodnym kodom razrabotanoj sistemy [The system's source code repository]. Available at: <https://github.com/MShudrak/tvc> (accessed 19.08.2015). (In Russ.).

Шудрак Максим Олегович — аспирант кафедры безопасности информационных технологий, ФГБОУ ВПО Сибирский государственный аэрокосмический университет имени академика М.Ф. Решетнёва. Область научных интересов: безопасность информационных технологий, верификация программного обеспечения, тестирование безопасности программного обеспечения. Число научных публикаций — 14. slender.bit@mail.ru; пр. им. газеты «Красноярский рабочий» д.31, Красноярск, 660037; п.т.: +79233088703.

Shudrak Maksim Olegovich — Ph.D. student of information security department, Siberian State Aerospace University (SibSAU). Research interests: software verification, vulnerabilities analysis, malware analysis. The number of publications — 14. slender.bit@mail.ru; 31, Krasnoyarsky rabochiy Av., 660037, Russia; office phone: +79233088703.

Золотарев Вячеслав Владимирович — к-т техн. наук, доцент, доцент кафедры безопасности информационных технологий, ФГБОУ ВПО Сибирский государственный аэрокосмический университет имени академика М.Ф. Решетнёва. Область научных интересов: анализ рисков, верификация программного обеспечения, анализ безопасности программного кода, анализ вредоносного кода. Число научных публикаций — 29. imposer_89@mail.ru; пр. им. газеты «Красноярский рабочий» д.31, Красноярск, 660037; п.т.: +79233088703.

Zolotarev Vyacheslav Vladimirovich — Ph.D., associate professor, associated professor of information security department, Siberian State Aerospace University (SibSAU). Research interests: software verification, vulnerabilities analysis, malware analysis. The number of publications — 29. imposer_89@mail.ru; 31, Krasnoyarsky rabochiy Av., 660037, Russia; office phone: +79233088703.

Поддержка исследований. Работа выполнена при финансовой поддержке РФФИ (проект №14-07-31350-мол_а).

Acknowledgements. This research is supported by RFBR (grant №14-07-31350-mol_a).

РЕФЕРАТ

Шудрак М.О. Золотарев В.В. Модель, алгоритмы и программный комплекс автоматизированного поиска уязвимостей в исполняемом коде.

В статье рассматривается проблема поиска уязвимостей в исполняемом коде. В первом разделе работы рассматривается понятие ошибки и уязвимости, выделяются критерии отнесения ошибки к уязвимости, а также анализируются существующие недостатки методик динамического и статического анализа исполняемого кода. Отмечается, что существующие инструменты направлены на обнаружение ошибок без учёта тех угроз, которые они несут для защищаемой информации, которая обрабатывается в ПО.

Во втором разделе работы представлена оригинальная модель автоматизированного поиска уязвимостей в трассе программы, которая позволяет формально определить критерии возникновения ошибки в исполняемом коде и их отнесения к уязвимости с учётом распределения защищаемой информации в памяти программы на каждом шаге работы программы. Кроме того приводится алгоритмическое обеспечение модели, а также программная реализация этих алгоритмов на базе системы динамической бинарной инструментации Intel PIN.

В работе приводятся результаты экспериментального сравнение эффективности разработанной системы, которые позволяют заключить, что разработанное решение позволяет детектировать на 5 типов уязвимостей больше в ОС Windows и на 4 типа уязвимостей больше в Linux по сравнению с существующими аналогами. В заключительном разделе делаются выводы и описываются перспективные направления дальнейшей работы.

SUMMARY

Shudrak M.O., Zolotarev V.V. A Model, Algorithms and Software Tool for Vulnerabilities Detection in Machine Code.

In the article we consider the problem of vulnerabilities detection in machine code. In the first section of the paper we discuss the concept of bug and vulnerability, stand out criteria for distinct bug from vulnerability, as well as provide disadvantages of the existing methods for dynamic and static analysis of executable code. In addition, we describe disadvantages of current solutions in case of possibility to detect vulnerabilities in view of threats to confidential information that is processed in vulnerable software.

In the second part of the article we propose original model of vulnerabilities detection in program trace that provides formal criteria to distinct bug from vulnerability taking into account distribution of protected information in the memory of software at each step of the test. In addition, algorithmic support of the model and its software implementation based on dynamic binary instrumentation framework Intel PIN are provided.

We conduct experimental evaluation of software implementation efficiency which demonstrates that our solution allows detecting 5 types of Windows software vulnerabilities more and 4 types Linux software vulnerabilities more than existing analogs. The final section summarizes the conclusions and describes future directions of the research.

М.В. КОЛОМЕЕЦ, А.А. ЧЕЧУЛИН, И.В. КОТЕНКО
**ОБЗОР МЕТОДОЛОГИЧЕСКИХ ПРИМИТИВОВ ДЛЯ
ПОЭТАПНОГО ПОСТРОЕНИЯ МОДЕЛИ ВИЗУАЛИЗАЦИИ
ДАННЫХ**

Коломеец М.В., Чечулин А.А., Котенко И.В., Обзор методологических примитивов для поэтапного построения модели визуализации данных.

Аннотация. В статье рассматриваются основные методологические примитивы на примере поэтапного построения модели визуализации с заранее подготовленными данными, с целью сформировать комплексное видение процесса создания модели и влияющих на неё аспектов. Приводится классификация примитивов и их связи между собой в соответствии с этапами построения модели. Рассматриваются библиотеки визуализации на популярных языках программирования.

Ключевые слова: визуализация данных безопасности, обзор методик визуализации, модель визуализации.

Kolomeec M.V., Chechulin A.A., Kotenko I.V. Review of Methodological Primitives for the Phased Construction of Data Visualization Model.

Abstract. The paper considers common methodological primitives by the example of phased data visualization model construction with prepared data. The purpose of the article is to form a comprehensive vision of the visualization model creation. The primitives classification and communication between them are suggested. The paper also considers data visualization libraries in popular programming languages.

Keywords: security data visualization, overview, method, visualization model.

1. Введение. Построение модели визуализации это сложный процесс, который определяют различные аспекты [1-2]. При этом, сами аспекты могут принадлежать различным областям знаний, таким как компьютерная графика, математика, статистика, когнитивная психология и дизайн. В условиях постоянного увеличения объёма и размерности визуализируемых данных [3], весьма актуальна проблема формирования концептуально новых моделей визуализации. Для разработки новых методик визуализации, необходимо знать общие особенности процесса их построения, а также уметь ориентироваться в уже существующих методиках, в том числе тех, которые используются вне сферы информационной безопасности, в которой авторы проводят интенсивные исследования и разрабатывают подсистему визуализации.

Зачастую новые модели систем визуализации приходят из областей искусства, медиа-дизайна, маркетинга, биоинформатики и т.п. Несмотря на отличие области применения, сами концепции отображения данных и представления сложных отношений между ними остаются прежними [4–5] (рисунок 1): в первую очередь осуществляется сбор данных, их анализ, фильтрация и только потом преобразование данных в графические примитивы из которых складывается графическая мо-

дель, которая, вместе с инструментами управления и отображения, представляет модель визуализации.



Рис. 1. Этапы построения визуализации

Работы, рассматривающие методики визуализации данных, как правило, уделяют недостаточно внимания целостности данного процесса, а именно, не рассматривают требуемые для визуализации аспекты, не упоминают при помощи каких инструментов и библиотек реализуется графическая модель или какие концептуальные инструменты могут расширить возможности той или иной графической модели, а также не рассматривают новые и уникальные графические модели.

Например, в [4] большое количество внимания уделяется этапу подготовки данных, а сами модели визуализации рассматриваются на уровне уже готовых инструментов. Этап подготовки данных безусловно является важным, однако, именно выбор модели визуализации является ключевым моментом для эффективного взаимодействия данных и пользователя. В [3] представлен обзор графических моделей, однако не упоминаются основные аспекты, исходя из которых, необходимо совершать выбор модели или разрабатывать свою собственную. В [7] также представлен обзор классических методик визуализации, но отсутствуют новые и уникальные концепции, построение которых на текущий момент актуально. Отдельно можно выделить обзоры из областей, не касающихся информационной безопасности. Например, графические модели в области социологии [8] могут быть успешно перенесены в область информационной безопасности, но перед этим должны быть преобразованы в соответствии с особенностями данных, касающихся информационной безопасности.

При построении или выборе модели важно понимать, как различные этапы и элементы процесса визуализации влияют на модель комплексно. В этом обзоре рассматриваются некоторые методологические примитивы на примере поэтапного построения модели визуализации с уже подготовленными данными, с целью сформировать комплексное видение процесса создания модели и влияющих на неё аспектов. Рассматриваемые примитивы можно классифицировать как:

– примитивы процесса визуализации - аспекты, которые влияют на процесс построения модели визуализации и с использованием которых разрабатывается начальная мета модель;

– примитивы графических моделей - основные принципы построения модели визуализации; часто, именно выбор графической концепции определяет формат взаимодействия данных с пользователем, а также определяет ограничения и возможности расширения этого взаимодействия;

– дополнительные инструменты - компоненты инструментов, расширяющие возможности графических моделей.

В статье рассматриваются основные методологические примитивы на примере поэтапного построения модели визуализации с заранее подготовленными данными, с целью сформировать комплексное видение процесса создания модели и влияющих на неё аспектов. Приводится классификация примитивов и их связи между собой в соответствии с этапами построения модели. В конце обзора также приведены библиотеки визуализации на популярных языках программирования Java и JavaScript, которые можно использовать для реализации и дополнения построенных графических моделей.

2. Примитивы процесса визуализации. Перед разработкой собственной или выбором готовой модели визуализации, важно понимать аспекты в рамках которых эта модель должна функционировать. Эти аспекты можно использовать как в качестве элементов шаблона при построении, так и в качестве инструментов выбора для анализа уже существующих систем. Их можно классифицировать на два класса:

– аспекты информативности - элементы, в соответствии с которыми необходимо строить или выбирать модель визуализации; аспекты информативности задают требования к модели визуализации, влияя на информативность;

– аспекты эффективности - элементы, в соответствии с которыми желательно строить модель визуализации; аспекты эффективности задают требования к модели, влияя на эффективность представления информации пользователю.

Сами принципы и их наборы могут различаться в зависимости от цели визуализации и располагаемых ресурсов. Если предполагается что с механизмами визуализации будут работать пользователи, не обладающие необходимой квалификацией, то скорее всего, стоит уделить внимание аспектам, связанным с дизайном и, в целом, аспектам эффективного представления информации. В случае информационной безопасности, где точность и полнота информации имеют решающую роль, стоит уделить внимание аспектам информативности.

2.1. Примеры аспектов информативности. Аспекты информативности влияют на модель визуализации, задавая ограничения на количество информации, которую модель может отобразить. Рассмотрим несколько примеров аспектов, которые влияют на информативность модели.

2.1.1. Шаблон визуального поиска. Этот принцип был предложен Беном Шнейдерманом в 1996 году [9]. Согласно ему, визуальный поиск состоит из трёх этапов: (1) обзор ситуации в целом; (2) масштабирование и фильтрация; (3) детали по требованию. Соответственно, любая модель визуализации, цель которой не только представление, но и поиск, а также анализ информации, должна иметь инструменты для каждого из этапов.

На рисунке 2 приведен пример выполнения шаблона визуального поиска, на основе анализа перехваченных TCP-пакетов.



Рис. 2. Пример выполнения шаблона визуального поиска: а) обзор ситуации; б) масштабирование и фильтрация; в) детали по требованию

Изначально поиск осуществляется на верхнем уровне абстракции, позволяя видеть ситуацию по всем перехваченным пакетам в целом (рисунок 2а); на втором этапе пакеты фильтруются таким образом, что бы пользователь мог видеть пакеты в рамках конкретного соеди-

нения (рисунок 2б); на третьем этапе пользователь, по требованию, может перейти на нижний уровень абстракции - посмотреть подробности пакета (рисунок 2в).

2.1.2. Когнитивный аппарат. Эффективность модели визуализации целиком обусловлена когнитивным аппаратом пользователя (память, восприятие, ассоциации и т.д.). Модель визуализации должна учитывать эти особенности в целом и не выходить за пределы их возможностей [10]. Несмотря на то, что данный аспект целиком принадлежит области психологии, его базовые основы часто понятны на интуитивном уровне. Но чем больше и сложнее графическая модель, тем больше влияние данного аспекта. Особенно важно учитывать особенности когнитивного аппарата при разработке моделей, которые работают с большими или неоднородными данными. Примеры:

– с учётом психологических ассоциаций, красный (на рисунок 3а тёмный) цвет является “зарезервированным цветом” для отображения угроз;

– анализ отдельных элементов и метрик может быть затруднён, так как модель выходит за пределы возможностей когнитивного аппарата (пределы восприятия множества элементов) (рисунок 3б).

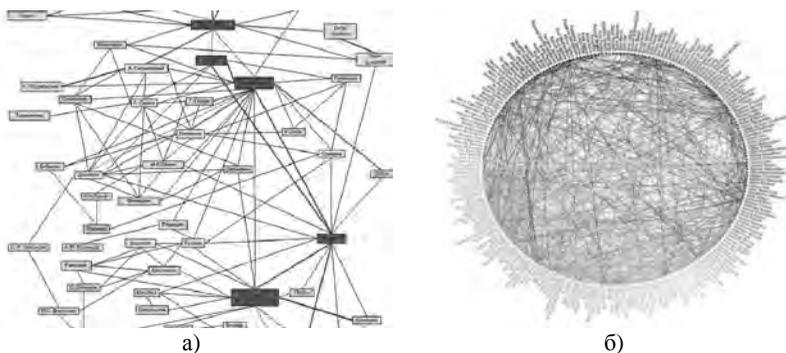


Рис. 3. Графы, учитывающие особенности когнитивного аппарата: а) граф с выделением элементов “зарезервированными цветами”; б) сильно перегруженный элементами радиальный граф

2.1.3. Соответствие данных и их представления. При анализе данных необходимо, чтобы их визуализация соответствовала источнику. В некоторых моделях (например, использующих трёхмерное пространство) пропорциональное отношение между визуализацией и данными может быть нарушено.

В [11] Эдвард Тафти предлагает оценивать степень соответствия данных и изображенной информации при помощи коэффициента

“фактор лжи”, который равен $\frac{\text{эффективность визуализации}}{\text{эффективность данных}}$, где $\text{эффективность} = \frac{\text{значение а} - \text{значение б}}{\text{значение а}}$. Для эффективности визуализации выбираются графические представления двух значений. Для эффективности данных – данные, по которым строятся графические представления.

Рассмотрим пример: в инфографике (рисунок 4), цель которой показать на сколько увеличился расход топлива на человека в США, значение для 1978 года равно 18 галлонам, а для 1985 года - 27.5 галлонам, т.е. эффективность данных равна 53%; ширина дороги, которая является визуализацией расхода топлива, для 1978 года равна 1 сантиметру, для 1985 равна 7.8 сантиметрам; эффективность визуализации равна 780%; подобное несоответствие эффективности визуализации и эффективности данных даёт фактор лжи = 17.8 [11].

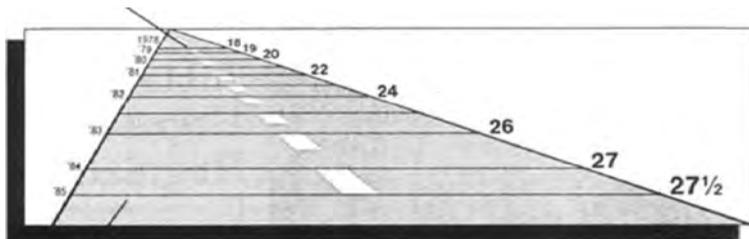


Рис. 4. Инфографика расхода топлива на человека в США

2.2. Примеры аспектов эффективности. Аспекты эффективности больше влияют на графическую модель, чем на модель визуализации. Они определяют эффективность диалога между механизмами визуализации и пользователем в рамках модели визуализации. Рассмотрим несколько аспектов, влияющих на эффективность.

2.2.1. Контроль информационного шума. Оптимальная модель не только та, в которую нечего добавить, но и из которой нечего убрать. Если элемент не входит в минимальный набор элементов визуализации, необходимых для решения поставленной задачи, он может являться источником информационного шума [12]. Как правило, подобные элементы являются элементами дизайна, который не был проработан в соответствии с особенностями модели визуализации.

На рисунке 5а модель становится более сложной для восприятия, за счёт необходимости анализировать данные в трёх измерениях. В данном случае информационным шумом является измерение, отображающее глубину.

На рисунке 5б пользователь может принять градиент внутри провинций в качестве параметра модели (например, плотность населения/температура и т.п.).

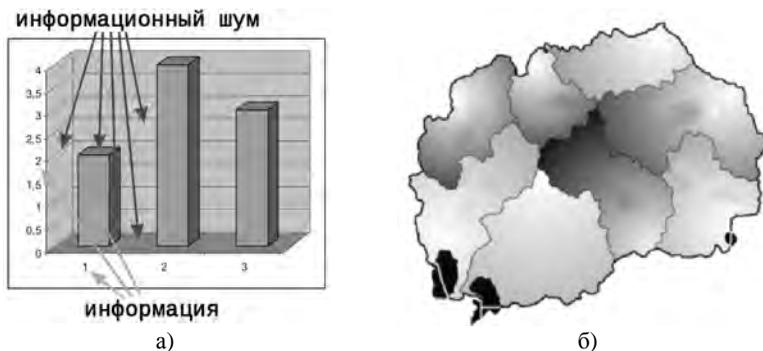


Рис. 5. Модели визуализации с информационным шумом: а) трёхмерный график с информационным шумом; б) карта с информационным шумом в виде градиентов

2.2.2. Прямые манипуляции. Инструменты для взаимодействия между информацией и пользователем должны иметь аналоги в реальности [13]. Подобный подход использует когнитивный аппарат человека и позволяет пользователю предугадать, что произойдёт после взаимодействия, а также выбирать инструменты и работать с ними на уровне интуиции.

Так, например, в концепции тактильных поверхностей Material Design [14–15] каждый контейнер аналогичен листу бумаги; выбранный контейнер за счёт тени располагается к наблюдателю ближе по сравнению с остальными, давая понять на интуитивном уровне о его активности (рисунок 6).



Рис. 6. Тактильные поверхности Material Design

2.2.3. Графический дизайн. Модель визуализации должна работать в рамках одной из моделей графического дизайна или же иметь

свою собственную. Модель графического дизайна также можно представить в качестве элементов, каждый из которых должен иметь обоснование своего присутствия в модели. Часто именно графический дизайн имеет наибольшее влияние на эффективность взаимодействия модели визуализации и пользователя. Пример графика в рамках модели графического дизайна Material Design [14–15] изображён на рисунке 7.

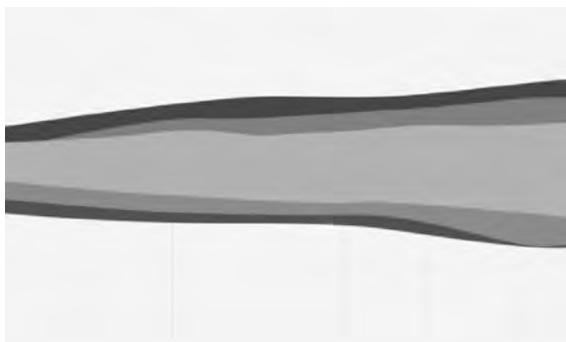


Рис. 7. График в рамках модели графического дизайна

3. Прimitives графических моделей. Модель визуализации является ядром процесса визуализации, определяя большую часть её ограничений и возможностей. Как правило, выбор модели напрямую зависит от цели процесса визуализации и типа информации. По степени использования, модели можно классифицировать как:

- широко распространённые;
- средне распространённые;
- специфичные.

3.1. Примеры широко распространённых моделей. К широко распространённым моделям, как правило, относят: графики, графы и их вариации и карты. Универсальность этих моделей обусловлена возможностью визуализации практически любых типов данных.

3.1.1. Графики. Графики получили широкое распространение как наиболее простая модель визуализации небольших данных [16]. При этом, степень успеха визуализации часто зависит от выбора графика, который, в свою очередь, зависит от типа данных [17-18]. На рисунке 8 изображены некоторые примеры простых графиков.

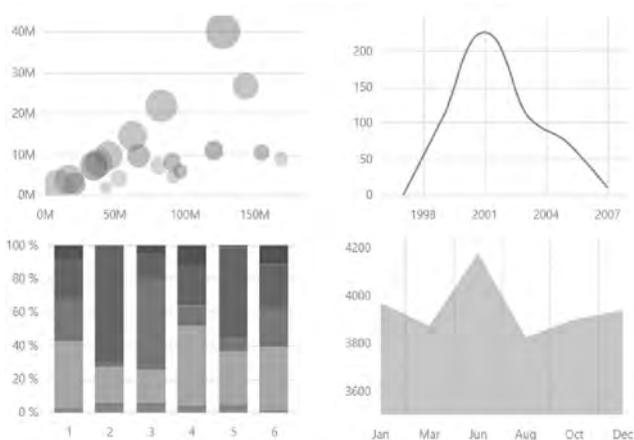


Рис. 8. Примеры простых графиков

3.1.2. Графы. Графы особенно популярны в информационной сфере: они интуитивно понятны, имеют множество вариаций и способны отображать большие объёмы разнородных данных [16]. На рисунок 9 изображены некоторые виды графов:

- классические - стандартные графы обычного представления (рисунок 9а);
- карты деревьев - иерархические графы на плоскости, вершины которых представлены прямоугольниками а отношения вложенностью (рисунок 9б);
- радиальные - иерархические графы, элементы которых расположены радиально (рисунок 9в).

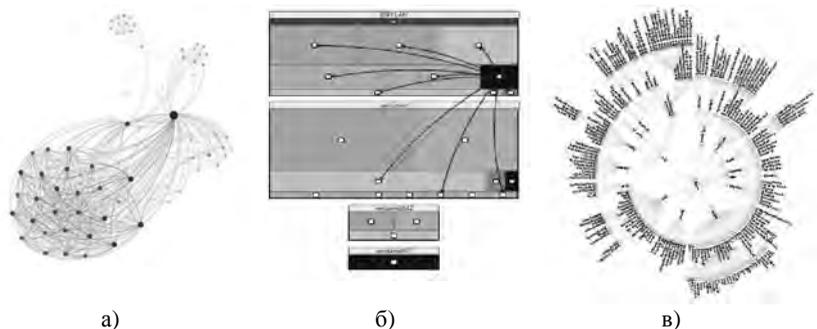


Рис. 9. Примеры различных графов: а) стандартный граф; б) карта дерева; в) радиальный граф

3.1.3. Карты. Карты используются в случаях, когда географические данные фигурируют в качестве ключевых [16]. При этом, остальные параметры выражаются в виде других графических моделей (графики, графы и т.д.), накладываясь на модель карт. На рисунке 10 представлена географическая карта атак [19–20] в реальном времени.

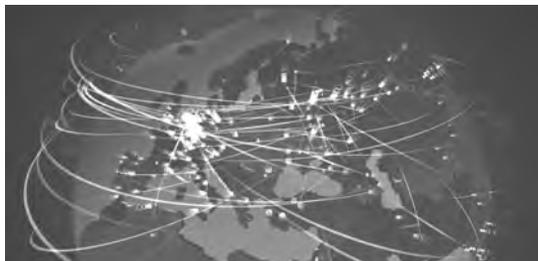


Рис. 10. Географическая карта атак

3.2. Примеры средне распространённых моделей. Средне распространённые модели - это модели, созданные в рамках определенной задачи, но имеющие потенциал в визуализации данных, выходящих за рамки цели их создания.

3.2.1. Матрицы. Для того что бы замаскировать атаку, атакующий обычно изменяет идентифицирующие его параметры, такие, например, как IP-адрес. Таким образом, чтобы идентифицировать нарушителя, необходимо опираться на другие параметры, например время прибытия пакета, которое зависит от типа операционной системы, поддержки маршрутизатора и прочие метрики, которые трудно изменить.

В [21] предлагается последовательный анализ на основе матричного представления.

На рисунке 11а в матрице слева, где время представлено вертикальной шкалой, один из всплесков фигурирует на протяжении нескольких часов. Матрица в центре показывает активность всех портов в выбранном временном диапазоне. Позиции всплеска на матрице слева соответствует всего один аномально активный порт в центральной матрице. Графики справа показывают, что большому количеству получателей соответствует небольшое количество источников, что, вероятнее всего, является признаком сканирования сети.

В [21] также предлагается использование данной модели в совокупности с классическими графами для выборочного анализа кластеров сети (рисунок 11б).

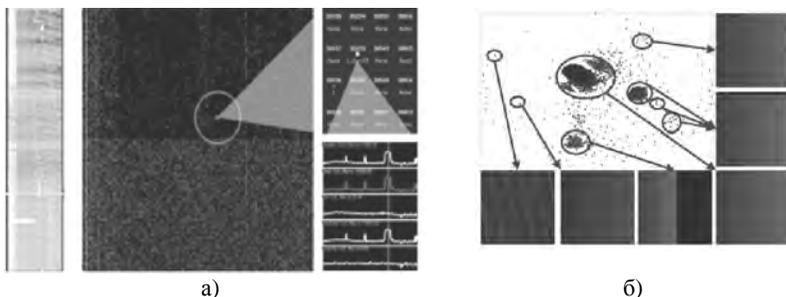


Рис. 11. Примеры моделей на основе матриц: а) модель на основе матричного представления; б) использование матриц для анализа кластеров

В [22] анализируется распределение TCP/IP-адресов с целью уменьшения рисков TCP/IP-спуфинга. Несмотря на то, что операционные системы используют для распределения генераторы случайных чисел, при их визуализации, несмотря на высокую или полную случайность, для каждой операционной системы можно выделить кластеры IP-адресов, наличие которых может помочь нарушителю.

На рисунке 12а изображен анализ последовательности адресов созданных операционной системой UNICOS 10.0.0.8. Несмотря на низкую оценку целесообразности атаки со стороны операционной системы (рисунок 12б), можно выделить 3 больших кластера, в пределах которых атака наверняка будет более успешной.

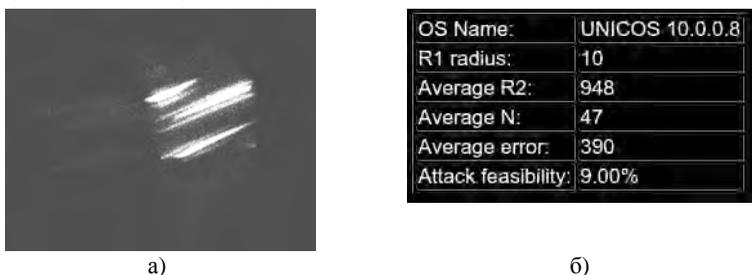


Рис. 12. Пример использования матриц для анализа распределения: а) матрица распределения TCP/IP-адресов; б) вероятностная оценка целесообразности атаки

3.2.2. Гистограммы. При необходимости сделать общие выводы, матричное представление можно преобразовать в гистограммы. Матрица на рисунке 13 получена с использованием параметра, рассчитанного на основе анализа посещений с уникальным адресом, где красный (на рисунке 13 - светлый) цвет соответствует максимуму количества, а черный - отсутствию посещений [21].

В гистограммах верхних матриц сравниваются похожие всплески на предмет различий. В гистограммах нижних матриц сравниваются различные всплески на предмет совпадений.

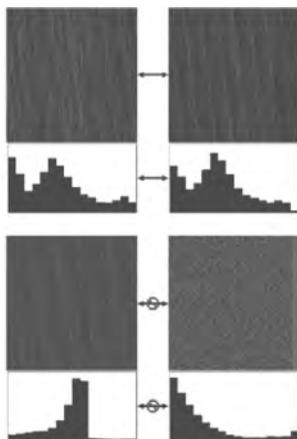


Рис. 13. Представление матриц в виде гистограмм

3.2.3. Трилинейные координаты. В [1] предлагается модель визуализации (рисунке 14б), разработанная на основе треугольной модели визуализации, которая была представлена геологическим департаментом США. Модель оперирует тремя метриками, каждая из которых соотносится с ребром треугольника и выражается в процентном соотношении по отношению друг к другу. В качестве примера, было взято процентное соотношение сообщений, переданных по различным протоколам (TCP, UDP, ICMP). Возможные положения источника в трилинейных координатах, а также положение источника с параметрами 30%, 40% и 30% приведены на рисунке 14а.

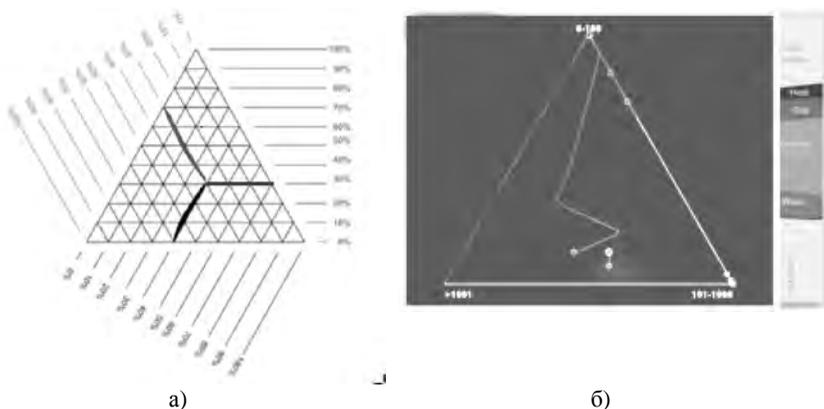


Рис. 14. Пример модели на основе трилинейных координат: а) возможные положения источника в трилинейных координатах; б) Модель на основе трилинейных координат

Для отображения изменений во времени, за каждым перемещением источника прокладывается трасса (рисунок 14б). Зоны, в которых источник проводит больше всего времени, подкрашиваются синим (на рисунок 14б - светлым) цветом.

Модель позволяет выявить аномальное поведение источника, когда он переходит в область, не типичную для его местонахождения, либо же когда движение для него само по себе аномально.

3.2.4. Параллельные координаты. Модель параллельных координат [23] является частным случаем графика. Она позволяет эффективно отобразить многомерные данные, располагая каждый тип данных вдоль одной из параллелей (рисунок 15).



Рис. 15. Модель на основе параллельных координат

В [24] предлагается похожая модель, оси которой расположены не параллельно, а радиально (рисунок 16). Предполагается, что использование полярных координат позволит лучше выявлять аномалии на когнитивном уровне, а также отображать метрики с большей эффективностью.

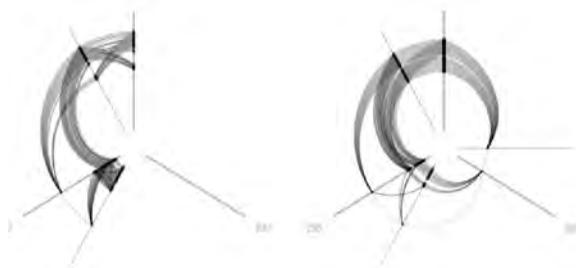


Рис. 16. Модель радиальных координат

3.3. Примеры специфичных моделей. Специфичные модели - это модели специализированного типа, созданные для решения конкретных задач, и которые, как правило, не могут быть использованы для решения задач другого вида.

3.3.1. Визуализация инструментов работы с лог-файлами.

В [25] предлагается визуализация инструментов для детальной работы с лог-файлами на основе регулярных выражений (рисунок 17). Преимуществами данного подхода являются: возможность работы с абстракцией (визуализацией) и самими лог-файлами одновременно, быстрая навигация без потери ориентации, анализ подмножеств записей с учётом контекста и исследовательский характер поиска, вместо стандартных поисковых запросов.

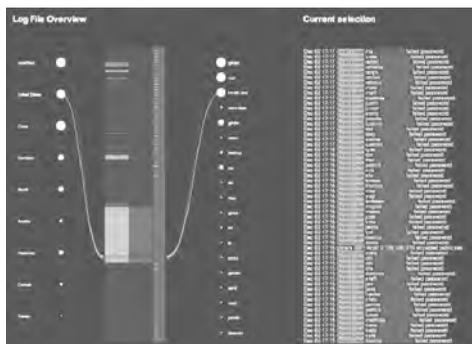
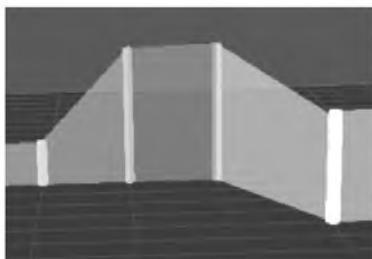


Рис. 17. Визуализация инструментов для работы с лог файлами

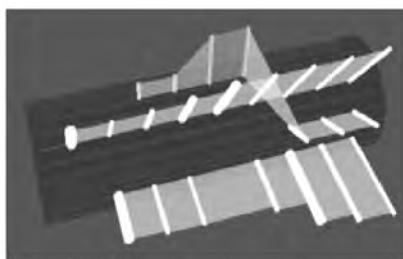
3.3.2. Визуализация сложных атак.

В [26] предлагается трехмерная модель визуализации сложных многоступенчатых атак (рисунок 18).

Каждый цилиндр представляет собой событие, тип которого задается цветом: зелёный - сканирование, фиолетовый - попытка удаленного доступа, красный - успешная попытка удаленного доступа, жёлтый - DoS (на рисунке 18 цвета заданы соответствующими оттенками серого).



а)



б)

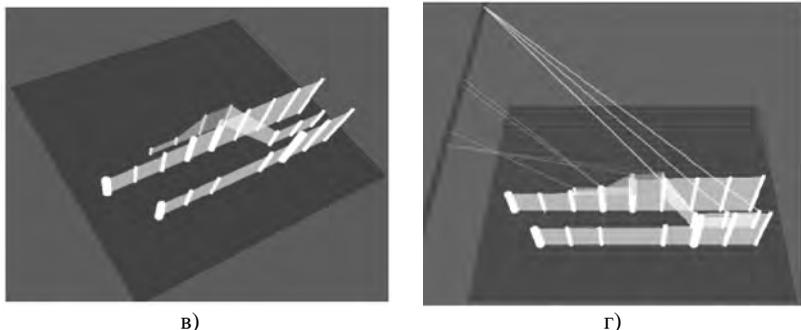


Рис. 18. Модель визуализации сложных атак: а) параметры разных видов атак; б) модель в полярных координатах; в) последовательное расположение атак; г) отображение источников атак в виде отдельной плоскости

Уровень серьёзности оповещений задаётся высотой цилиндра (рисунок 18а).

Ключевая особенность системы - это возможность постепенного отслеживания атаки как единого целого, зная какие события предшествовали ей и какие произошли после (рисунок 18в).

При этом система обладает рядом дополнительных инструментов, таких как переход от декартовых к полярным координатам (рисунок 18б), возможность отображения источников атак в соответствии с типом атаки (рисунок 18г), а также управление плоскостью источников для обеспечения лучшего угла обзора (рисунок 18г).

4. Дополнительные инструменты. Для некоторых типов данных или определенного набора данных, выбор графической модели может быть представлен всего несколькими видами моделей или вовсе отсутствовать.

Например, для представления топологии сети наглядным вариантом всегда является представление в виде графа.

При этом, если многомерность данных или их параметров сопровождается большим объёмом, модель визуализации в определенные моменты может выходить за ограничительные пределы процесса визуализации.

Наиболее остро данная проблема стоит при визуализации больших сетей, когда пользователь вынужден ещё на стадии обзора жертвовать отображением одних метрик, для отображения других.

На рисунке 19, для отображения отдельных элементов (рисунок 19б), пользователь, воспользовавшись инструментом масштабирования, вынужден отказаться от отображения сети целиком (рисунок 19а).

Предполагается, что пользователь способен выбрать область фокуса, в результате чего интересующие его элементы станут больше, в то время как остальные также останутся в поле зрения. В отличие от обычного масштабирования, пользователю не надо мысленно объединять изображения для полного обзора. В инструменте “рыбий глаз” кратности масштаба соответствует коэффициент искажения.

4.2. Инструмент “Множественный взгляд”. Для некоторых типов данных полезно использование сразу нескольких моделей визуализации одновременно [28]. Нахождение необходимых отношений или аномалий при данном подходе значительно упрощено. При многомерности метрик, подход, отображающий для каждого множества параметров соответствующую оптимальную графическую модель, может оказаться эффективнее стандартных средств, таких как переход между уровнями абстракции или исключение ряда параметров.

На рисунке 21 один и тот же набор данных представлен параллельными координатами, 2D графом и картой деревьев.

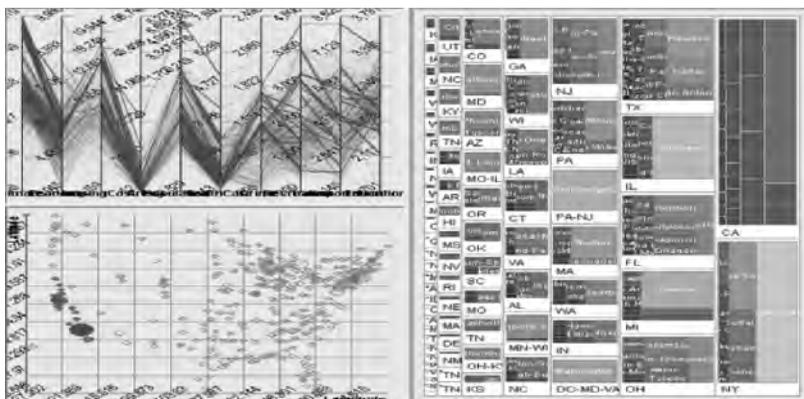


Рис. 21. Различные представления одного графа

4.3. Инструмент “Семантическое масштабирование”. Восприятие графических форм в некоторых случаях менее эффективно семантического восприятия. Таким образом, при работе с представлением в виде текста также необходимо иметь соответствующие инструменты. Масштабирование возможно не только на уровне абстракций или отдельно взятого изображения, но и на уровне семантики [29]. Классическим примером семантического масштабирования является текстовая каталогизация, когда элементы группируются не по функциональным, а по семантическим параметрам, таким как алфавитный порядок, лексическое значение и т.д.

4.4. Инструмент “Небольшие различия”. Иногда, при применении фильтров, необходимо видеть не только итог фильтрации, но и её предварительный результат, или несколько результатов сразу. Например, на рисунке 22 в левой части отображено множество результатов применений фильтра с разными параметрами. Маленькие, размером с почтовую марку компоненты, предназначенные для моментального ответа на вопрос “а что если?”, могут не только ускорить процесс поиска информации, но и избавить от необходимости однозначно выбирать модель визуализации, где в качестве одной из метрик будет предварительный результат [30–31].

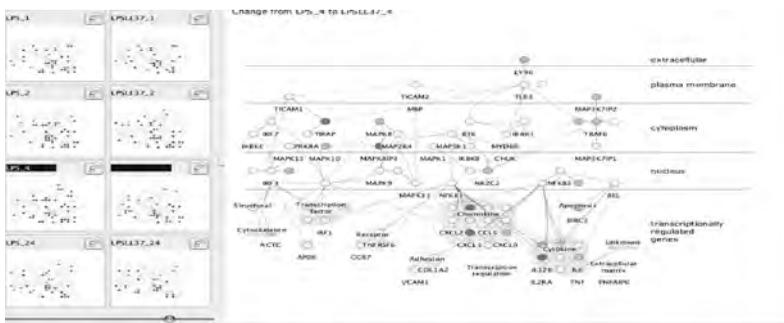


Рис. 22. Отображение вариаций применения фильтра

4. Библиотеки визуализации. Построение модели визуализации всегда зависит от метрик, некоторые из которых часто встречаются во многих уже существующих системах. Таким образом, может получиться, что разработчик будет реализовать то, что было разработано другими специалистами ранее.

Знание основных библиотек для визуализации данных и шаблонов реализации на их основе однозначно необходимо перед построением модели визуализации. Также библиотеки содержат уже готовые реализации некоторых компонентов системы и предоставляют различные фреймворки для упрощения процесса разработки.

Ниже приведено несколько популярных некоммерческих библиотек для языков Java и JavaScript.

4.1. Jung. Простая и стабильная библиотека Jung [32] разработана на Java и используется преимущественно для визуализации графов.

Среди недостатков этой библиотеки можно отметить, что разработчиками не была должным образом проработана документация, а также то, что проект приостановил своё развитие.

Таким образом, Jung может быть полезна только для визуализации простых графов (рисунок 23).

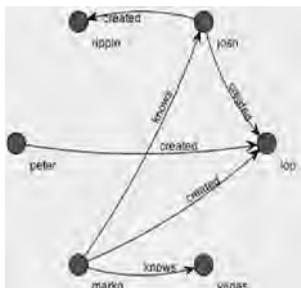


Рис. 23. Пример графа, реализованного на Jung

4.2. GraphViz. Визуализация в рамках пакета GraphViz [33] ведётся при помощи языка описания графов DOT [33]. GraphViz принимает файл на языке DOT и автоматически формирует изображение с заранее определенной моделью.

Из достоинств данной библиотеки можно отметить хорошо проработанную документацию, простоту использования, наличие возможности кластеризации и поддержку нескольких графических моделей. Из недостатков можно отметить, что GraphViz может использоваться только для визуализации графов.

Пример визуализации графа с разбиением на кластеры приведен на рисунке 24.



Рис. 24. Пример графа реализованного на GraphViz

4.3. Prefuse. Данная библиотека использует стандартную Java библиотеку Java2D, поэтому она легко интегрируется с приложениями, разработанными с использованием пакета Java Swing. Prefuse [34] содержит широкий выбор инструментов (в том числе подключение к базе

данных, наличие собственных структур данных, поддержка анимации, динамических запросов, поиска, таблиц, графиков и т.д.) и, как следствие, предназначена не только для визуализации графов. Prefuse является достаточно гибким средством, однако ее развитие приостановлено, вследствие чего документация проработана недостаточно хорошо. Пример графа с использованием Prefuse изображен на рисунке 25.

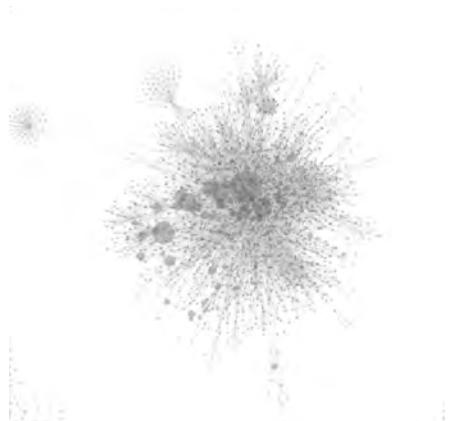


Рис. 25. Пример графа реализованного на Prefuse

4.4. D3. Эта библиотека [35] разработана на JavaScript и является одной из наиболее популярных библиотек для визуализации данных. Её можно использовать как основу для реализации модели визуализации в рамках фреймворка Data-Driven-Document, или же использовать одну из сотен готовых реализаций графических моделей, предлагаемых разработчиками. Данная библиотека имеет хорошо проработанную документацию, причем эта документация переводится на множество языков, в том числе и на русский. Примеры графических моделей, реализованных на D3, изображены на рисунке 26.



Рис. 26. Примеры различных моделей, реализованных на D3

5. Заключение. Можно сказать, что целью процесса визуализации является нахождение компромисса между показателями информативности данных и эффективности их представления. Каждый элемент модели визуализации - аспекты процесса визуализации, графическая модель, инструмент, а также библиотека реализации - по отдельности влияют на эти показатели. Но в конечном итоге, решение об использовании тех или иных элементов зависит именно от данных. Здесь важно понимать что не данные должны определяться моделью визуализации, а модель визуализации должна формироваться на основе данных. Порой, более правильным набором элементов будет простой и минимальный набор, чем сложная графическая модель с большим количеством инструментов под управлением множества примитивов.

В данном обзоре приведены основные элементы процесса визуализации. Знание этих элементов для анализа уже существующей модели и умение их применять на разных этапах разработки модели визуализации поможет, если не достигнуть компромисса информативности и эффективности, то максимально к нему приблизиться.

Описанные в данной статье методы успешно применяются авторами как для визуализации общих моделей атак и метрик защищенности [36], так и для визуализации отдельных элементов атак представленных в базе данных CAPEC [20].

Литература

1. *Bruce R.* Applying Information Visualization to Computer Security Applications // All Graduate Theses and Dissertations. 2010. 636 p.
2. *Ботя М.В.* Инфографика как объект информационного дизайна // Новые информационные технологии в образовании: материалы VIII международной научно-практической конференции. Екатеринбург, 2015. С. 411–414.
3. *Клышинский Э.С., Рысаков С.В. и Шихов А.И.* Обзор методов визуализации многомерных данных // Новые информационные технологии в автоматизированных системах. 2014. С. 519–530.
4. *Falschlunger L., Lehner O., Eisl C., Losbichler H.* Development of a Data Visualization Model based on Information Processing Theory // Proceedings of the 9th conference for Austrian universities of applied sciences. Hagenberg. Austria. 2015. pp. 1–7.
5. *Novikova E., Kotenko I.* Analytical Visualization Techniques for Security Information and Event Management // 21st Euromicro International Conference on Parallel, Distributed, and Network-Based Processing. 2013. pp. 519–525.
6. *Ferebee D., Dasgupta D.* Security Visualization Survey // Proceedings of the 12th Colloquium for Information Systems Security Education University of Texas. 2008. 124 p.
7. *Barrera D.* Towards Classifying And Selecting Appropriate Security Visualisation Techniques // A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master Of Science School of Computer Science at Carleton University Ottawa. Ontario. 2009. 117 p.
8. *Healy K., Moody J.* Data Visualization in Sociology // Annual Review of Ecology and Systematics 2014. 2014. pp. 105–128.

9. *Shneiderman B.* The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations // In Proceedings of the IEEE Symposium on Visual Languages. Washington. 2006. pp. 336–343.
10. *Goldstein B.* Cognitive Psychology // Thomson Wadsworth. 2005.
11. *Tufte E.* The Visual Display of Quantitative Information // Graphics Press. USA. 1991. pp. 57–69.
12. *Tufte E.* Visual Explanations // Graphics Press. Cheshire. Connecticut. 1997.
13. *Hutchins E., Hollan J., Norman D.* Direct Manipulation Interfaces // Lawrence Erlbaum Associates. 1985. vol. 1. pp. 311–338.
14. *Google inc.* Material-Design Introduction. URL: www.google.com/design (дата обращения 01.10.15).
15. Блог компании REDMADROBOT. URL: www.habrahabr.ru/company/redmadrobot (дата обращения 01.10.15).
16. *Marty R.* Applied Security Visualization // Addison Wesley Professional. 2009.
17. *Kotenko I., Novikova E.* Visualization of Security Metrics for Cyber Situation Awareness // The 1st International Software Assurance Workshop (SAW 2014). In conjunction with the 9th International Conference on Availability, Reliability and Security (ARES 2014). Fribourg, Switzerland. IEEE Computer Society. 2014. pp. 506–513.
18. Блог компании Devexpress. URL: www.habrahabr.ru/company/devexpress (дата обращения 01.10.15).
19. Сайт компании Kaspersky с презентацией карты атак реального времени. URL: www.cybermap.kaspersky.com (дата обращения 01.10.15).
20. *Котенко И.В., Дойникова Е.В., Чецулин А.А.* Общее перечисление и классификация шаблонов атак (САРЕС): описание и примеры применения // Защита информации. Инсайд. 2012. №4. С. 54–66.
21. *Kwan-Liu M.* Cyber Security Through Visualization // In Proc. Asia Pacific Symposium on Information Visualisation. Tokyo. Japan. 2006.
22. *Zalewski M.* TCP/IP Sequence Number Analysis. 2002. URL: www.lcamtuf.coredump.cx/newtcp. (дата обращения 01.10.15).
23. *Tricaud S.* Visualizing Network Activity using Parallel Coordinates // System Sciences. 44th Hawaii International Conference. 2011. pp. 1–8.
24. *Engle S., Whalen S.* Visualizing Distributed Memory Computations with Hive Plots // VizSec '12 Proceedings of the Ninth International Symposium on Visualization for Cyber Security. 2012.
25. *Stange J., Dörk J., Landstorfer J., Wettach R.* Visual Filter: Graphical Exploration of Network Security Log Files // VizSec '14 Proceedings of the Eleventh Workshop on Visualization for Cyber Security. 2014. pp. 41–48.
26. *Yelizarov A., Gamayunov D.* Visualization of Complex Attacks and State of Attacked Network // Visualization for Cyber Security. VizSec 2009: 6th International Workshop. 2009. pp. 1–9.
27. *Sarkar M., Brown M.* Graphical fisheye views // Communications of the ACM. 1994. vol. 37. no. 12. pp. 73–83.
28. *Wang M., Woodruff A., Kuchinsky A.* Guidelines for Using Multiple Views in Information Visualization // Proc. of Advanced Visual Interfaces. 2000. pp. 110–119.
29. *Watson G.* Lecture Lecture 15 - Visualisation of Abstract Information // Edinburgh Virtual Environment Centre. 2004.
30. *Wroblewski L.* Small Multiples Within a User Interface // Web Form Design. 2005.
31. *Tufte E.* Envisioning Information // Graphics Press. Cheshire. 1990.
32. Официальный сайт библиотеки Jung. URL: www.jung.sourceforge.net (дата обращения 01.10.15).
33. Официальный сайт библиотеки Graphviz. URL: www.graphviz.org (дата обращения 01.10.15).

34. Официальный сайт библиотеки Prefuse. URL: www.prefuse.org (дата обращения 01.10.15).
35. Официальный сайт библиотеки D3js. URL: www.d3js.org (дата обращения 01.10.15).
36. *Kotenko I.V., Chechulin A.A. A Cyber Attack Modeling and Impact Assessment Framework // Proceedings of the 5th International Conference on Cyber Conflict 2013 (CyCon 2013). IEEE and NATO COE Publications. Tallinn, Estonia. 2013. pp. 119–142.*

References

1. Bruce R. Applying Information Visualization to Computer Security Applications // All Graduate Theses and Dissertations. 2010. 636 p.
2. Botja M.V. [Infographic as Object of Information Design]. *Novye informacionnye tehnologii v obrazovanii: materialy VIII mezhdunarodnoj nauchno-prakticheskoy konferencii* [New information technologies in education: materials of VIII scientific-practice conference]. Ekaterinburg. 2015. pp. 411–414. (In Russ.).
3. Klyshinskij E.S., Rysakov S.V., Shihov A.I. [Review of the methods of multidimensional data visualization]. *Novye Informacionnoe Tehnologii v Avtomatizirovannyh Sistemah – New information technologies in automated systems*. 2014. pp. 519–530. (In Russ.).
4. Falschlunger L., Lehner O., Eisl C., Losbichler H. Development of a Data Visualization Model based on Information Processing Theory. Proceedings of the 9th conference for Austrian universities of applied sciences. Hagenberg, Austria. 2015. pp. 1–7.
5. Novikova E., Kotenko I. Analytical Visualization Techniques for Security Information and Event Management. 21st Euromicro International Conference on Parallel, Distributed, and Network-Based Processing. 2013. pp. 519–525.
6. Ferebee D. and Dasgupta D. Security Visualization Survey. Proceedings of the 12th Colloquium for Information Systems Security Education University of Texas. 2008. 124 p.
7. Barrera D. Towards Classifying And Selecting Appropriate Security Visualisation Techniques. A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master Of Science School of Computer Science at Carleton University Ottawa, Ontario. 2009. 117 p.
8. Healy K., Moody J. Data Visualization in Sociology. *Annual Review of Ecology and Systematics*. 2014. pp. 105–128.
9. Shneiderman B. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In Proceedings of the IEEE Symposium on Visual Languages. Washington. 2006. pp. 336–343.
10. Goldstein B. Cognitive Psychology. Thomson Wadsworth. 2005.
11. Tufte E. The Visual Display of Quantitative Information. *Graphics Press*. USA. 1991. pp. 57 – 69.
12. Edward R. Tufte, Visual Explanations. *Graphics Press*. Cheshire, Connecticut. 1997.
13. Hutchins E., Hollan J., Norman D. Direct Manipulation Interfaces. Lawrence Erlbaum Associates. 1985. vol. 1. pp. 311–338.
14. Google inc. Material-Design Introduction. Available at: www.google.com/design. (accessed 01.10.2015).
15. Blog kompanii REDMADROBOT [REDMADROBOT company blog]. Available at: www.habrahabr.ru/company/redmadrobot. (accessed 01.10.2015). (In Russ.).
16. Marty R. Applied Security Visualization. Addison Wesley Professional. 2009.
17. Kotenko I., Novikova E. Visualization of Security Metrics for Cyber Situation Awareness. The 1st International Software Assurance Workshop (SAW 2014). In con-

- junction with the 9th International Conference on Availability, Reliability and Security (ARES 2014). Fribourg. Switzerland. 2014. pp. 506–513.
18. Blog kompanii Devexpress [Devexpress company blog]. Available at: www.habrahabr.ru/company/devexpress. (accessed 01.10.2015). (In Russ.).
 19. Sajt kompanii Kaspersky s prezentaciej karty atak realnogo vremeni [Kaspersky web site with real time attack cybermap]. Available at: www.cybermap.kaspersky.com. (accessed 01.10.2015).
 20. Kotenko I.V., Dojnikova E.V., Chechulin A.A. [General enumeration and classification of attack patterns (CAPEC): description and application examples]. *Zashhita informacii. Insajd – Data protection. Inside*. 2012. vol. 4. pp. 54–66.
 21. Kwan-Liu M. Cyber Security Through Visualization, In Proc. Asia Pacific Symposium on Information Visualisation. Tokyo. Japan. 2006.
 22. Zalewski M. TCP/IP Sequence Number Analysis. 2002. Available at: www.lcamtuf.coredump.cx/newtcp. (accessed 01.10.2015).
 23. Tricaud S. Visualizing Network Activity using Parallel Coordinates, System Sciences. 44th Hawaii International Conference. 2011. pp. 1–8.
 24. Engle S., Whalen S. Visualizing Distributed Memory Computations with Hive Plots. VizSec '12 Proceedings of the Ninth International Symposium on Visualization for Cyber Security. 2012.
 25. Stange J., Dörk J., Landstorfer J., Wettach R. Visual Filter: Graphical Exploration of Network Security Log Files. VizSec '14 Proceedings of the Eleventh Workshop on Visualization for Cyber Security. 2014. pp. 41–48.
 26. Yelizarov A., Gamayunov D. Visualization of Complex Attacks and State of Attacked Network, Visualization for Cyber Security. VizSec 2009: 6th International Workshop. 2009. pp. 1–9.
 27. Sarkar M., Brown M. Graphical fisheye views. Communications of the ACM. 1994. vol. 37. no. 12. pp. 73–83.
 28. Wang M., Woodruff A., Kuchinsky A. Guidelines for Using Multiple Views in Information Visualization. Proc. of Advanced Visual Interfaces. 2000. pp. 110–119.
 29. Watson G. Lecture Lecture 15 - Visualisation of Abstract Information. Edinburgh Virtual Environment Centre. 2004.
 30. Wroblewski L. Small Multiples Within a User Interface, Web Form Design. 2005.
 31. Tuft E. Envisioning Information. *Graphics Press*. Cheshire. 1990.
 32. Oficial'nyj sajt biblioteki Jung [Official web site of Jung library]. Available at: www.jung.sourceforge.net. (accessed 01.10.2015).
 33. Oficial'nyj sajt biblioteki Graphviz [Official web site of Graphviz library]. Available at: www.graphviz.org. (accessed 01.10.2015).
 34. Oficial'nyj sajt biblioteki Prefuse library [Official web site of Prefuse library]. Available at: www.prefuse.org. (accessed 01.10.2015).
 35. Oficial'nyj sajt biblioteki D3js [Official web site of D3js library]. Available at: www.d3js.org. (accessed 01.10.2015).
 36. Kotenko I.V., Chechulin A.A. A Cyber Attack Modeling and Impact Assessment Framework. Proceedings of the 5th International Conference on Cyber Conflict 2013 (CyCon 2013). IEEE and NATO COE Publications. Tallinn. Estonia. 2013. pp. 119–142.

Коломеец Максим Вадимович — программист лаборатории проблем компьютерной безопасности, Федеральное государственное бюджетное учреждение науки Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН). Область научных интересов: безопасность распределенных систем, визуализация данных. Число научных публикаций — 0. guardeeccwalker@gmail.com; 14-я ли-

ния В.О., д. 39, ком. 205, Санкт-Петербург, 199178; р.т.: +7(812)328–2642, Факс: +7(812)328–4450.

Kolomeec Maxim Vadimovich — developer of computer security problems laboratory, St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences (SPIIRAS). Research interests: distributed system security, security visualization. The number of publications — 0. guardeecwalker@gmail.com; 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone: +7(812)328–2642, Fax: +7(812)328–4450.

Чечулин Андрей Алексеевич — к-т техн. наук, старший научный сотрудник лаборатории проблем компьютерной безопасности, Федеральное государственное бюджетное учреждение науки Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН). Область научных интересов: безопасность компьютерных сетей, обнаружение вторжений, анализ сетевого трафика, анализ уязвимостей. Число научных публикаций — 110. andreych@bk.ru, <http://comsec.spb.ru/ru/staff/chechulin>; 14-я линия В.О., д. 39, ком. 205, Санкт-Петербург, 199178; р.т.: +78123287181.

Chechulin Andrew Alexeevich — Ph.D., senior researcher of computer security problems laboratory, St. Petersburg Institute for Informatics and Automation of the Russian Academy of Science (SPIIRAS). Research interests: computer network security, intrusion detection, analysis of the network traffic, vulnerability analysis. The number of publications — 110. andreych@bk.ru, <http://comsec.spb.ru/ru/staff/chechulin>; 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone: +78123287181.

Котенко Игорь Витальевич — д-р техн. наук, профессор, заведующий лабораторией проблем компьютерной безопасности, Федеральное государственное бюджетное учреждение науки Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН). Область научных интересов: безопасность компьютерных сетей, в том числе управление политиками безопасности, разграничение доступа, аутентификация, анализ защищенности, обнаружение компьютерных атак, межсетевые экраны, защита от вирусов и сетевых червей, анализ и верификация протоколов безопасности и систем защиты информации, защита программного обеспечения от взлома и управление цифровыми правами, технологии моделирования и визуализации для противодействия кибер-терроризму. Число научных публикаций — 450. ivkote@comsec.spb.ru, <http://www.comsec.spb.ru>; 14-я линия В.О., д. 39, Санкт-Петербург, 199178; р.т.: +7(812)328–2642, Факс: +7(812)328–4450.

Kotenko Igor Vitalievich — Ph.D., Dr. Sci., professor, head of computer security problems Laboratory, St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences (SPIIRAS). Research interests: computer network security, including security policy management, access control, authentication, network security analysis, intrusion detection, firewalls, deception systems, malware protection, verification of security systems, digital right management, modeling, simulation and visualization technologies for counteraction to cyber terrorism. The number of publications — 450. ivkote@comsec.spb.ru, <http://www.comsec.spb.ru>; 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone: +7(812)328–2642, Fax: +7(812)328–4450.

Поддержка исследований. Работа выполнена при финансовой поддержке РФФИ (13-01-00843, 14-07-00697, 14-07-00417, 15-07-07451, 15-37-51126) и за счет гранта РНФ 15-11-30029 в СПИИРАН

Acknowledgements. This research is supported by RFBR (projects no. 13-01-00843, 14-07-00697, 14-07-00417, 15-07-07451, 15-37-51126) as well as by RSF project 15-11-30029.

РЕФЕРАТ

Коломеец М.В., Чечулин А.А., Котенко И.В. **Обзор методологических примитивов для поэтапного построения модели визуализации данных.**

В условиях постоянного увеличения объёма и размерности информации, весьма актуальна проблема разработки концептуально новых моделей визуализации. Построение модели визуализации данных это сложный процесс, на который влияют большое количество аспектов. Для разработки новых методик, необходимо знать общие особенности процесса их построения, а так же уметь ориентироваться в уже существующих методиках. При этом, важно понимать, как различные этапы и элементы процесса визуализации влияют на модель комплексно.

В статье рассматриваются основные методологические примитивы на примере поэтапного построения модели визуализации с заранее подготовленными данными. Приводится классификация примитивов и их связей в соответствии с этапами построения модели. Рассматриваются библиотеки визуализации на популярных языках программирования.

SUMMARY

Kolomeec M.V., Chechulin A.A., Kotenko I.V. **Review of Methodological Primitives for the Phased Construction of Data Visualization Model.**

Development of new conceptual visualization models is an actual problem, especially when information volume and its dimensions are always growing up. Construction of data visualization model is a difficult process, which is influenced by different aspects. It is important to know common features of processes and already existing methods for constructing new visualization techniques. In addition, it is necessary to understand how different phases and elements influence visualization model comprehensively.

The paper considers common methodological primitives by the example of phased data visualization model construction with prepared data. The primitives classification and communication between them are suggested. The paper also considers data visualization libraries in different programming languages.

РУКОВОДСТВО ДЛЯ АВТОРОВ



Взаимодействие автора с редакцией осуществляется через личный кабинет на сайте журнала «Труды СПИИРАН» <http://www.proceedings.spiiras.nw.ru>. При регистрации авторам рекомендуется заполнить все предложенные поля данных, так как это значительно ускорит процесс оформления метаданных к новым статьям.

Подготовка статьи ведется с помощью текстовых редакторов MS Word 2007 и выше. При подаче материала в редакцию сначала отправляется только статья в формате *.docx. Для обеспечения требований слепого рецензирования при представлении статьи в журнал авторам необходимо удалить персональные данные, содержащиеся в тексте файла и его свойствах.

Объем основного текста – от 5 до 20 страниц включительно. Формат страницы документа – А5 (148 мм ширина, 210 мм высота); ориентация – портретная; все поля – 20 мм. Верхний и нижний колонтитулы страницы – пустые. Основной шрифт документа – Times New Roman, основной кегль (размер) шрифта – 10 pt. Переносы разрешены. Абзацный отступ устанавливается размером в 10 мм. Межстрочный интервал – одинарный. Номера страниц не проставляются.

Не допускается использования цветных шрифтов, цветовых выделений и цветных рисунков. Статьи должны быть полностью готовы к черно-белой печати.

Основная часть текста статьи разбивается на разделы, среди которых являются обязательными: введение, хотя бы один «содержательный» раздел и заключение. Допускается также мотивированное содержанием и структурой материала выделение подразделов.

В основную часть допускается помещать рисунки, таблицы, листинги и формулы. Правила их оформления подробно рассмотрены на нашем сайте в разделе «Руководство для авторов».

