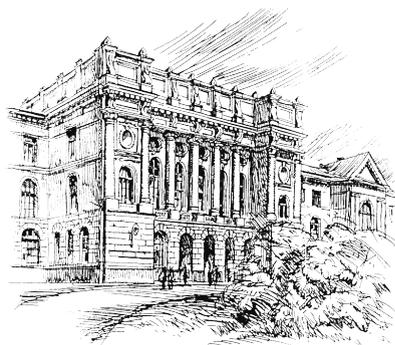


4(152)/2012



Научно-технические ведомости СПбГУ

ИНФОРМАТИКА.
ТЕЛЕКОММУНИКАЦИИ.
УПРАВЛЕНИЕ

Санкт-Петербург. Издательство Политехнического университета

Министерство образования и науки РФ

Санкт-Петербургский государственный политехнический университет

НАУЧНО-ТЕХНИЧЕСКИЕ ВЕДОМОСТИ СПбГПУ

РЕДАКЦИОННЫЙ СОВЕТ ЖУРНАЛА

Алферов Ж.И., академик РАН; *Васильев Ю.С.*, академик РАН (председатель);
Костюк В.В., академик РАН; *Лопота В.А.*, чл.-кор. РАН;
Окрепилов В.В., академик РАН; *Рудской А.И.*, чл.-кор. РАН;
Патон Б.Е., академик НАН Украины и РАН; *Федоров М.П.*, академик РАН;
Фортов В.Е., академик РАН.

РЕДАКЦИОННАЯ КОЛЛЕГИЯ ЖУРНАЛА

Васильев Ю.С., академик РАН (главный редактор); *Арсеньев Д.Г.*, д-р техн. наук, профессор;
Бабкин А.В., д-р экон. наук, профессор (зам. гл. редактора);
Боронин В.Н., д-р техн. наук, профессор; *Глухов В.В.*, д-р экон. наук, профессор;
Десярева Р.В., д-р ист. наук, профессор; *Иванов А.В.*, д-р техн. наук, профессор;
Иванов В.К., д-р физ.-мат. наук, профессор; *Козловский В.В.*, д-р физ.-мат. наук, профессор;
Рудской А.И., чл.-кор. РАН (зам. гл. редактора); *Юсупов Р.М.*, чл.-кор. РАН.

СЕРИЯ “ИНФОРМАТИКА. ТЕЛЕКОММУНИКАЦИИ. УПРАВЛЕНИЕ”

РЕДАКЦИОННЫЙ СОВЕТ СЕРИИ

Юсупов Р.М., чл.-кор. РАН – председатель;
Абрамов С.М., чл.-кор. РАН;
Воеводин В.В., чл.-кор. РАН;
Заборовский В.С., д-р техн. наук, профессор;
Козлов В.Н., д-р техн. наук, профессор;
Фотиади А.Э., д-р физ.-мат. наук, профессор;
Черноруцкий И.Г., д-р техн. наук, профессор.

РЕДАКЦИОННАЯ КОЛЛЕГИЯ СЕРИИ

Юсупов Р.М., чл.-кор. РАН – председатель;
Арсеньев Д.Г., д-р техн. наук, профессор – зам. председателя;
Бабкин А.В., д-р экон. наук, профессор – зам. председателя;
Пышкина Г.А., технический секретарь;
Антонов В.И., д-р техн. наук, профессор;
Голландцев Ю.А., д-р техн. наук, профессор;
Карпов Ю.Г., д-р техн. наук, профессор;
Коротков А.С., д-р техн. наук, профессор;
Макаров С.Б., д-р техн. наук, профессор;
Устинов С.М., д-р техн. наук, профессор;
Цикин И.А., д-р техн. наук, профессор;
Шкодыврев В.П., д-р техн. наук, профессор;
Клавдиев В.Е., канд. техн. наук, доцент.

Журнал с 1995 года издается под научно-методическим руководством Российской академии наук.

Журнал с 2002 года входит в Перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные результаты диссертаций на соискание ученой степени доктора и кандидата наук.

Сведения о публикациях представлены в Реферативном журнале ВИНИТИ РАН, в международной справочной системе «Ulrich`s Periodical Digestory».

Журнал зарегистрирован в Госкомпечати РФ. Свидетельство № 013165 от 23.12.94.

Подписной индекс **18390** в каталоге “Газеты. Журналы” Агентства “Роспечать”.

Журнал включен в базу данных “Российский индекс научного цитирования” (РИНЦ), размещенную на платформе Национальной электронной библиотеки на сайте <http://www.elibrary.ru>.

При перепечатке материалов ссылка на журнал обязательна.

Точка зрения редакции может не совпадать с мнением авторов статей.

Адрес редакции и издательства: Россия, 195251, Санкт-Петербург, ул. Политехническая, д. 29.
Тел. редакции серии (812) 552-62-16.

Содержание

Инфокоммуникационные технологии

Заборовский В.С., Лукашин А.А. Система контроля доступа в среде облачных вычислений	7
Осипов Н.А., Зудилова Т.В., Хоружников С.Э. Облачные технологии при обработке данных большого объема	12

Телекоммуникационные системы и компьютерные сети

Дмитриев В.М., Ганджа В.В. Алгоритм анализа и сжатия данных по топологической структуре исследуемого объекта	17
Борисевич А.В. Синтез алгоритмов стабилизации нелинейных объектов на основе схемы продолжения решения по параметру	20

Проблемы передачи и обработки информации

Волвенко С.В., Завьялов С.В., Макаров С.Б. Имитационная модель системы передачи спектрально-эффективных сигналов с OFDM	29
Бондарчук С.С., Додохов А.Л., Сабанов А.Г. Технология защиты персональных данных с использованием СУБД ORACLE	36
Семенов П.К. Практические аспекты алгоритма Тала–Варди списочного декодирования полярных кодов	40

Радиотехника, антенны, СВЧ-устройства

Акимов В.П., Глыбовский С.Б., Щесняк С.С. Коррекция фазовых ошибок параболического зеркала с помощью вторичного сетчатого рефлектора	45
---	----

Системный анализ и управление

Ляшенко А.Л. Разработка методики синтеза распределенного П-регулятора по показателю колебательности	51
Ростов Н.В. Многокритериальное нормирование эталонных моделей в задачах синтеза систем автоматического управления	55
Лычагин Н.И., Вершинина В.П., Дмитриев В.П. Управление знаниями в системе менеджмента качества предприятия – разработчика программного обеспечения	61

Математическое моделирование: методы, алгоритмы, технологии

Голуб Ю.А. Методы и модели исследования и разработки автоматизированных информационных систем избирательного распространения информации и инициативного обслуживания	67
Деркаченко В.Н., Зубков А.Ф., Ковалерова Н.В., Бармин М.А. Математические модели прогнозирования динамики развития рынка жилой недвижимости	71

Кривошейкин А.В., Молчанов А.П., Чесноков М.А. Динамическая фильтрация элементов речевых сигналов и сигналов с непериодическим линейчатым спектром	75
Морева С.Л., Золотов О.И., Ляшенко А.Л. Разработка математической модели температурных полей активной зоны реактора РБМК-1000	79
Дармахеев В.В. Эффективность применения ряда генетических операторов селекции	84
Ченцов А.Г., Ченцов П.А. Маршрутизация перемещений с ограничениями и нестационарными функциями стоимости	88
Зикратов И.А., Спивак А.И., Разумовский А.В. Метод разграничения доступа к объектам в системе контроля версий	94
Конференция «Технологии Microsoft в теории и практике программирования»	
Григорьева А.В., Васильев В.А., Сафонов В.О. Применение элементов аспектно-ориентированного подхода ASPECT.NET при разработке геолокационной системы управления ресурсами для облачной платформы MICROSOFT WINDOWS AZURE	98
Воинов Н.В., Котлярова Л.П., Поддубный И.Е. Методика разработки мобильных приложений для платформы WINDOWS PHONE 7	102
Малыгин Л.Л., Царев В.А., Черкас П.С. Автоматическое управление параметрами средств формирования изображений в системах распознавания текстовых меток реального времени	106
Иванов А.С., Котляров В.П., Летичевский А.А. Статический анализ требований реактивных систем	109
Никифоров И.В., Петров А.В., Котляров В.П. Статический метод отладки тестовых сценариев, сгенерированных с использованием эвристик	114
Тютин Б.В., Никифоров И.В., Котляров В.П. Построение системы автоматизации статической и динамической проверки требований к программному продукту	119
Дробинцев П.Д., Котляров В.П., Чернолуцкий И.Г. Автоматизация тестирования на основе покрытия пользовательских сценариев	123
Сведения об авторах	129
Аннотации	133



Contents

Infocommunication technologies

Zaborovsky V.S., Lukashin A.A. <i>Access control system in cloud computing environment</i>	7
Osipov N.A., Zudilova T.V., Khoruzhnikov S.E. <i>Cloud technologies while working with big data</i>	12

Telecommunication systems and computer networks

Dmitriev V.M., Gandzha V.V. <i>Algorithm analysis and data compression the topological structure of examined object</i>	17
Borisevich A.V. <i>Synthesis of algorithms for nonlinear stabilization based on numerical continuation approach</i>	20

Information transfer and processing

Volvenko S.V., Zavialov S.V., Makarov S.B. <i>Simulation model of system of transfer of spectral-effective signals with OFDM</i>	29
Bondarchuk S.S., Dodokhov A.L., Sabanov A.G. <i>Technology for protection of personal data using ORACLE DBMS</i>	36
Semenov P.K. <i>Practical issues of Tal-Vardy algorithm of polar code list decoding</i>	40

Radio engineering, aerials, SHF-devices

Akimov V.P., Glybovskiy S.B., Shchesnyak S.S. <i>A correction of parabolic mirror's phase errors by means of secondary meshed reflector</i>	45
--	----

System analysis and control

Lyashenko A.L. <i>Method of distributed P-regulator synthesis by using index of oscillation</i>	51
Rostov N.V. <i>Multiobjective normalization of reference models in synthesis of automatic control systems ..</i>	55
Lychagin N.I., Vershinina V.P., Dmitriev V.P. <i>Knowledge management in software developing company quality management system</i>	61

Mathematical modelling: methods, algorithms, technologies

Golub Yu.A. <i>Methods and models of research and development of automated information systems selective dissemination of information and services initiative</i>	67
Derkachenko V.N., Zubkov A.F., Kovalerova N.V., Barmin M.A. <i>Mathematical models of the real estate market development prognostics</i>	71
Krivosheykin A.V., Molchanov, A.P., Chesnokov M.A. <i>Dynamic filtering of elements of speech signals and signals with non-periodic spectrum</i>	75
Moreva S.L., Zolotov O.I., Lyashenko A.L. <i>Development of mathematical model of temperature fields of the active zone of the RBMK-1000 reactor</i>	79
Darmakheev V.V. <i>Efficiency of usage of genetic operator selection of sequence</i>	84

Chentsov A.G., Chentsov P.A. <i>Routing of permutations with constraints and nonstationary cost functions</i>	88
Zikratov I.A., Spivak A.I., Razumovskiy A.V. <i>Method of differentiating access to objects in version control system</i>	94
Conference «Technologies Microsoft in the theory and programming practice»	
Grigorjeva A.V., Vasilev V.A., Safonov V.O. <i>Use element aspect-oriented approach in the drafting ASPECT.NET geolocation resource management system to cloud platforms MICROSOFT WINDOWS AZURE</i>	98
Voinov N.V., Kotlyarova L.P., Poddubnyy I.E. <i>Methodic of WINDOWS PHONE 7 mobile applications development</i>	102
Malygin L.L., Tsarev V.A., Cherkas P.S. <i>Automatic parameters control of cameras, lenses and illumination in real-time text marker recognition systems</i>	106
Ivanov A.S., Kotlyarov V.P., Letichevsky A.A. <i>Verification of requirements for reactive systems</i>	109
Nikiforov I.V., Petrov A.V., Kotlyarov V.P. <i>Static approach for debugging of test scenarios generated with usage of heuristics</i>	114
Tyutin B.V., Nikiforov I.V., Kotlyarov V.P. <i>Elaboration of the toolsuite for automation of the static and dynamic software requirement checking</i>	119
Drobintsev P.D., Kotliarov V.P., Chernorutsky I.G. <i>Approach for testing automation based on user scenarios</i>	123
<i>About the authors</i>	129
<i>Abstracts</i>	133

СИСТЕМА КОНТРОЛЯ ДОСТУПА В СРЕДЕ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ

Развитие современной информационной инфраструктуры идет по пути создания высокопроизводительных центров обработки данных (ЦОД), применения технологии виртуализации и организации на ее основе сервисов облачных вычислений (ОВ). Использование технологии ОВ обеспечивает повышение эффективности функционирования ЦОД, однако предъявляет повышенные требования к системе защиты информации и средствам контроля доступа (КД) [1]. Эти требования должны учитывать динамический характер использования информационно-вычислительных ресурсов, которые в среде ОВ разделяются между пользователями и сервисами в соответствии с требованиями политики информационной безопасности (ПИБ).

В статье развивается сетевцентрический подход к решению задачи контроля доступа в среде ОВ, в рамках которого осуществимость операций доступа, например чтения или записи, контролируется с помощью управления величиной пропускной способности виртуальных соединений (ВС), участвующих в реализации этих операций [2, 3]. Таким образом, в предложенной модели условию запрета соответствует использование для операции доступа ВС с нулевой пропускной способностью. Для формализации описания данной модели доступа предлагается аппарат теории категорий, а в качестве средства реализации ПИБ и устройств управления пропускной способностью ВС – межсетевые экраны (МСЭ), функционирующие в режиме полной скрытной фильтрации и представляющие собой новый класс сервисов среды ОВ.

Особенности системы КД в среде ОВ

Системы КД, реализуемые в рамках подхода AAA (Authentication, Authorization, Accounting), оправдано применять тогда, когда условия доступа к информационным ресурсам описываются

с помощью предикатных отношений или булевых функций (полиномов) со свободными параметрами. Однако в условиях динамически изменяющейся конфигурации информационно-сетевых ресурсов реализация требований мандатных, дискреционных и ролевых ПИБ не позволяет контролировать такие аспекты безопасности, которые являются следствием информационной связанности субъектов и объектов сетевого взаимодействия. Эти аспекты проявляются тогда, когда необходимо учитывать:

- условия ситуационной осведомленности, связанные с необходимостью непрерывной актуализации состояния субъектов и объектов информационного взаимодействия;
- риски нарушения ПИБ, являющиеся следствием динамического изменения параметров и состояния ВС (технологических и информационных);
- нелокальный характер хранения информационных ресурсов пиринговых одноранговых (P2P) и гибридных файлообменных сетей.

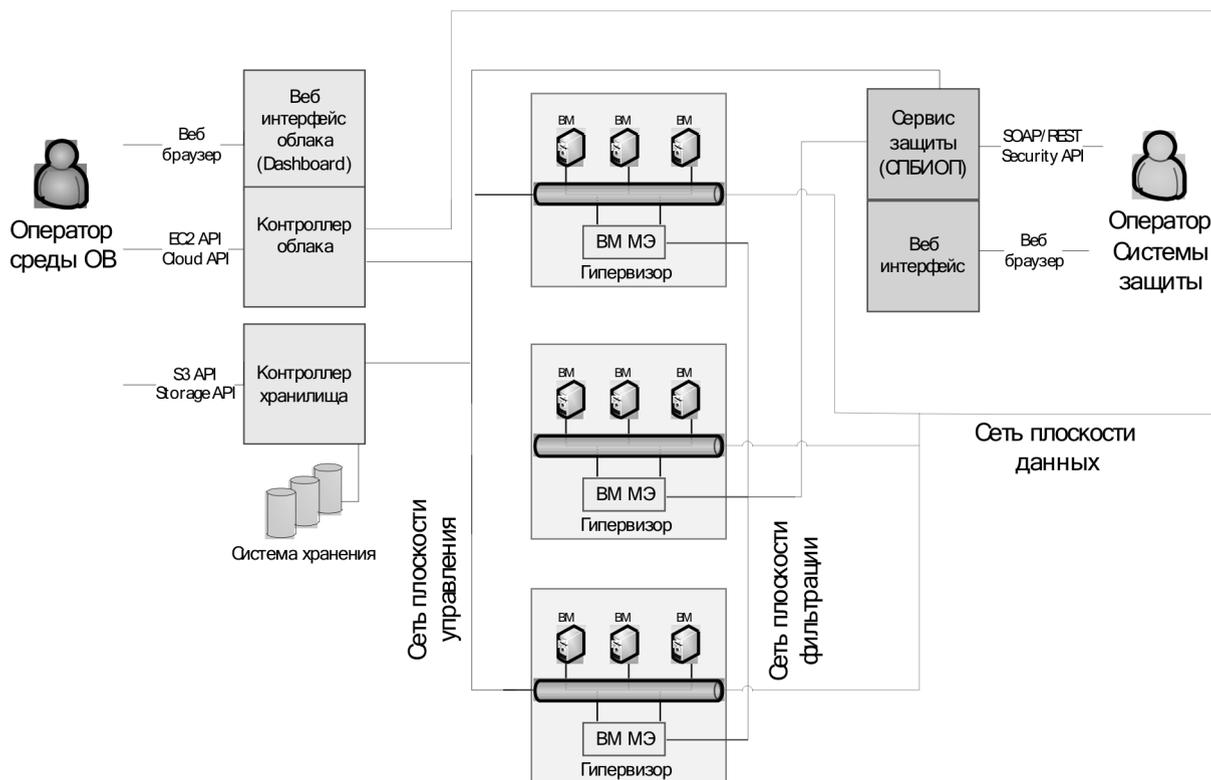
Среда ОВ характеризуется динамически изменяющимся набором вычислительных ресурсов – виртуальных машин, контейнеров приложений и программных сервисов, находящихся под управлением большого числа (сравнимого с количеством ресурсов) пользователей. Важной особенностью среды ОВ является то, что ресурсы виртуализованы и разделяют между собой аппаратную платформу, управление ими доступно только в удаленном режиме, по сети, без возможности непосредственного физического доступа к ним. Система контроля доступа в среде ОВ должна учитывать особенности таких систем, уметь реконфигурироваться в процессе функционирования – постоянно актуализировать ПИБ, запрашивать дополнительные аппаратные средства при высокой нагрузке и обладать сетью связи для

координированных действий в распределенной среде ОВ.

Для решения перечисленных выше задач и проблем является обоснованным использование сетецентрической системы защиты [2]. Применительно к задаче контроля доступа под такой системой понимается связанная сеть межсетевых экранов, осуществляющих фильтрацию трафика в узлах облака на основе правил фильтрации. Каждый межсетевой экран системы принимает локальное решение о возможности доступа субъекта к объекту на основе имеющихся у него данных. Данные в виде правил фильтрации непрерывно актуализируются путем информационного обмена между МСЭ и сервисом защиты, располагающем данными о состоянии системы в целом и текущей ПИБ. Процесс преобразования ПИБ, выраженной, например, в модели OgVAC, в правила фильтрации (ПФ), назовем *трансляцией ПИБ*. Сформулируем свойства системы защиты, которые служат носителями сетецентричности:

- 1) локальность действий компонент системы;
- 2) информационная связанность компонент;
- 3) целостность решаемой задачи.

В рассматриваемом случае компонентом системы является межсетевой экран, который на основе имеющихся у него сведений принимает решение о разрешении или запрете информационного взаимодействия в форме контроля ВС. Решение принимается локально, в конкретной зоне влияния МСЭ. Информационная связанность – важнейший аспект сетецентрической системы, все компоненты должны быть объединены в общую сеть, и информация о состоянии защищаемой сетевой среды должна непрерывно актуализироваться. Информационная связанность компонент позволяет построить самоорганизующуюся систему, обладающую возможностью реконфигурации и перераспределения нагрузки между компонентами. Решение задачи контроля доступа должно осуществляться согласованно, и совокупность локальных решений должна быть целостной с точки зрения рассматриваемой ПИБ. Свойство целостности можно также рассматривать с точки зрения эмерджентности системы защиты, каждый МСЭ решает задачи, которые не обеспечивают полное выполнение ПИБ, а лишь ее часть, однако совокупность принятых решений



Среда ОВ с интегрированной системой КД



обеспечивает полное выполнение ПИБ и является эмерджентной характеристикой.

На рисунке представлена высокоуровневая архитектура рассматриваемой в статье системы контроля доступа в среде ОВ. Контроль информационного взаимодействия осуществляется виртуальными машинами с программным обеспечением межсетевых экранов (ВМ МЭ), которые запущены внутри гипервизоров среды ОВ. Виртуализованный МСЭ подключается к сетевой подсистеме гипервизора и осуществляет фильтрацию ВС между виртуальными ресурсами и внешними по отношению к гипервизору соединениями. Технические характеристики системы контроля доступа и исследование производительности виртуализованных МСЭ представлены в работе [4]. Исследования производительности показали, что МСЭ могут эффективно применяться в виртуальной среде, при этом применяемые алгоритмы фильтрации и сетевая подсистема Netgraph позволяют организовать параллельную обработку сетевого трафика. При использовании паравиртуализованных драйверов в операционной системе виртуализованного МСЭ спад производительности незначителен, около 10 %, по сравнению с программно-аппаратным решением. В то же время описываемая система контроля доступа использует вычислительные ресурсы гипервизора, у каждого гипервизора среды ОВ будет дополнительно занято 1-2 Гб ОЗУ и от одного до восьми вычислительных ядер, что приводит к снижению максимально возможного количества запущенных облачных ресурсов.

Описание процессов взаимодействия в среде ОВ

Следуя [5], для формализации требований ПИБ будем использовать параметрические модели описания субъекта s , действия d и объекта o информационного взаимодействия. Параметры этих моделей являются несущими множествами состояний ВС, которые с точки зрения требований ПИБ относятся к категории разрешенных или запрещенных состояний. Параметрического описания достаточно, чтобы представить различные ПИБ, в которых операциям доступа к информационным ресурсам сопоставляются предикатные условия, составленные относительно множества свободных переменных, входящих в описание упомянутых выше моделей. Однако отмеченный выше динамический характер процессов инфор-

мационного взаимодействия требует рассмотрения их свойств с точки зрения не только параметрического соответствия требованиям ПИБ, но и процессов перехода ВС из одних возможных состояний в другие. Для этого перейдем к использованию моделей, основанных на формализме теории категорий. Введем категорию ВС $catIC$, для каждой пары объектов (A, B) которой определим множество морфизмов $Hom(A, B)$ или отношений, свойства которых не зависят от внутренней структуры самих этих объектов. Объекты категории ВС назовем *информационными виртуальными соединениями* (ИВС), для которых морфизмы задаются совокупностью транспортных виртуальных соединений (ТВС), реализуемых с помощью транспортных протоколов, таких, как TCP и UDP. Объекты категории ВС, или множество ИВС, задают соответствующий класс информационных сервисов, для которых заданы требования ПИБ. Примером ИВС может служить сессия пользователя веб ресурса gmail, или загрузка файла по протоколу ftp.

С точки зрения ПИБ объекты категории ВС могут принадлежать классу, в который входят три макросостояния:

- 1) разрешенные ВС, РВС;
- 2) запрещенные ВС, ЗВС;
- 3) неопределенные ВС, НВС.

Объекты категории ВС, или ИВС, принадлежащие к одному макросостоянию, обладают общим признаком, который является вычислимой функцией параметров, образующих ИВС различных ТВС. Заметим, что хотя объекты категории ВС имеют различные структуры, связанные с особенностями сервисов среды ОВ, но их свойства, влияющие на требования ПИБ, определяются параметрами заголовков или тела пакетов ТВС. Эти параметры должны удовлетворять свойствам связанности, целостности и устойчивости по отношению к сигнатуре используемых протоколов, поэтому могут контролироваться межсетевыми экранами, функционирующими в режиме скрытой фильтрации.

Сетецентрическая модель разграничения доступа

Введенная выше категория ВС в дальнейшем будет применяться для формализации сетецентрической модели КД. Основу модели составляет несущее множество макросостояний, объекты которого входят в класс понятий, используемых для

формализации описания ПИБ. Задача, решаемая в системе КД с помощью межсетевых экранов, состоит в том, чтобы вычислить конкретное макросостояние, в котором находятся объекты категории ВС данный момент времени, используя для этого информацию о ТВС и ПИБ, а также данные о структуре сети и характеристиках субъектов информационного взаимодействия. Переход ИВС в одно из возможных макросостояний определяет динамику информационных процессов и может контролироваться с МСЭ с помощью применения правил фильтрации к образующим информационное соединение ТВС.

Таким образом, сетевая модель КД представляется с помощью категории ВС, объединяющей класс объектов (ИВС) и класс морфизмов (ТВС). Переходы между макросостояниями контролируются МСЭ с помощью правил фильтрации, полученных в результате трансляции заданной ПИБ. Для того чтобы представить отношения между ИВС и ПФ введем категорию правил фильтрации $CatF$, объектами которой являются подмножества ПФ, каждое из которых определяет макросостояние ИВС. Введем функтор $funcICF : catIC \rightarrow catF$, который осуществляет перевод объекта категории ВС в объект категории ПФ. Контроль макросостояния ИВС сводится к контролю объекта категории ПФ, т. е. к контролю набора правил фильтрации. На программном уровне указанный функтор представляет собой ассоциативный массив или хеш-таблицу связей между ПФ и ИВС.

Рассмотрим среду ОБ как совокупность взаимодействующих объектов. Разделим эту совокупность на три плоскости – плоскость данных (data plane), плоскость управления (control plane) и плоскость фильтрации (security plane), каждая из которых представлена выделенной сетью связи.

Плоскость данных. Все внутренние ресурсы среды ОБ представим множеством объектов:

$$O = \{o_i\}, i = 1..n,$$

где n – число сущностей, используемых для формирования ПИБ. Субъектами считаются те объекты, которые иницируют обмен данными или инициировали такой обмен ранее, но в рассматриваемый момент времени t_i находятся в незавершенной стадии информационного обмена. Множество объектов среды ОБ можно разбить на три составляющих:

$$O(t_i) = S(t_i) \cup P(t_i) \cup N(t_i),$$

где S – множество активных объектов или субъектов (Subjects); P – множество объектов – источников данных (Providers); N – множество объектов, не участвующих в информационном обмене (Nonparticipating). Каждый объект среды ОБ принадлежит одному ресурсу $r_i \in R$, где R – множество ресурсов или сервисов среды ОБ. Под ресурсом среды ОБ понимается виртуальная машина (ВМ), запущенная на одном из серверов облака и находящаяся под контролем гипервизора.

Соответствие объектов и ресурсов формально задается ассоциативным массивом $arrR$:

$$\forall o \in O, \exists r \in R : arrR(o) = r.$$

Для МСЭ в среде ОБ ресурс идентифицируется сетевыми адресами $A_r = \{a_r, e_r\}$, где $a_r \in A$, A – множество адресов сетевых сегментов, входящих в среду ОБ, $e_r \in E$, E – множество адресов, с помощью которых объект идентифицируется для сегментов, не входящих в среду ОБ. Другими словами, это внутренний и внешний IP адрес ресурса. ПИБ представляет множество ПФ $Rules(PB, O) = \{rul_k\}, k = 1..s$. Пусть некоторое ИВС с индексом j состоит из множества ТВС: $ИВС_j = \{TBC\}_{ИВС_j}$. Подмножество ПФ, задающих макросостояние заданного ИВС, является объектом категории ПФ и доступно через функтор $funcICF$. Функция контроля ИВС выглядит следующим образом: $F(Rules, \{TBC\}_{ИВС_j}, O) = \{1, 0, *\}$. Значение «1» соответствует макросостоянию разрешенных соединений, «0» – макросостоянию запрещенных соединений, «*» – макросостоянию неопределенных, фоновых соединений. Если функция контроля j -го ИВС в предыдущие моменты времени t_i имела значение «1» или «*», но в результате изменения состояния облака, изменения множества правил фильтрации или открытия запрещенного ТВС принимает значение «0», то ИВС переходит в макросостояние запрещенных соединений, пропускная способность ТВС уменьшается до нуля, сетевое взаимодействие прекращается.

Плоскость управления. В плоскость управления входят программно-аппаратные системы, управляющие ресурсами среды ОБ, к которым относятся гипервизоры, средства хранения образов, управляющий сервис облака, хранилище пользовательских объектов и вспомогательные сервера (базы данных, очередей заявок, авторизации и аутентификации). Все системы функционируют в управляющей сети, а взаимодействие осуществляется путем вызова веб-сервисов.

По управляющей сети передаются следующие основные команды:

1) управление жизненным циклом виртуальной машины

$$vm_{cmd} = \{start_{vm}, stop_{vm}, pause_{vm}\};$$

2) управление пользовательскими данными

$$data_{cmd} = \{create_{obj}, rm_{obj}, get_{obj}, update_{obj}\};$$

3) управление образами ВМ

$$image_{cmd} = \{upload_{img}, send_{img}, delete_{img}\}.$$

Команды реализуются в рамках ИВС, принадлежащих категории «разрешенные». ИВС, принадлежащие плоскости управления, порождают ИВС в плоскости данных и по команде, например, запуска новой виртуальной машины, осуществляют генерацию правил фильтрации на основе метаданных запроса и информации о других запущенных в облаке ресурсах. Таким образом достигается согласованность и целостность системы защиты, локальное событие порождает изменение правил фильтрации во всех компонентах системы. Важным аспектом защищенности облачной среды является наличие выделенной сети, изолированной от плоскости данных. Но при этом передаваемые по данной сети управляющие сообщения должны находиться под контролем системы защиты.

Плоскость фильтрации. В плоскость фильтрации входит группировка МСЭ и сервис политики безопасности и оперативной обстановки (СПБИОП). Фильтрующую функцию осуществляют виртуальные межсетевые экраны, функционирующие в скрытном режиме, подключенные к виртуальным коммутаторам гипервизоров, и программно-аппаратные межсетевые экраны, установленные на границе сегментов облака. Виртуализованная сущность межсетевых экранов позволяет осуществить контроль трафика между виртуальными машинами. Управляющую функцию выполняет СПБИОП, который взаимодействует с компонентами облака и межсетевыми экранами, при изменении политики безопасности генерирует правила фильтрации и отправляет в МСЭ по управляющему каналу плоскости защиты [6]. Управляющий канал плоскости фильтрации представляет собой отдельную подсеть, недоступную для подсетей управления и ресурсов облака. Управление доступом осуществляется на

основе контроля виртуальных соединений (ВС) между субъектами и объектами. Интеграция МСЭ в облачную среду достигается путем запуска программного обеспечения экрана в виртуальной среде, подключенного к виртуальному коммутатору, который интегрируется с гипервизором. Виртуальный сетевой трафик коммутируется на порт коммутатора, к которому подключен входной интерфейс МСЭ, осуществляющий контроль сетевого трафика между ресурсами среды ОВ. Важной особенностью плоскости фильтрации является то, что МСЭ осуществляют контроль информационного взаимодействия в плоскости управления и в плоскости данных. Сеть плоскости фильтрации недоступна для других участников информационного взаимодействия. Получение информации об изменении состава облачных ресурсов можно реализовать двумя способами: координацией действий с использованием сервиса СПБИОП, который получает данные от управляющих серверов облака, а также осуществляет прослушивание управляющих сообщений в плоскости управления и информированием локально установленным МСЭ остальных компонент системы разграничения доступа.

В статье представлена формализация среды ОВ и системы контроля доступа. Модель системы контроля доступа с сетевым подходом обобщена как категория, объединяющая класс объектов (класс состояний) и класс морфизмов (переходов между состояниями). КД осуществляется путем управления пропускной способностью ВС. Сетевым подходом к организации системы разграничения доступом обеспечивает эффективность функционирования средств защиты за счет проверки в МСЭ только тех правил фильтрации, которые необходимы для РД между существующими в облаке ресурсами, а также согласованность функционирования всех компонент системы. Прозрачная интеграция средств контроля доступа обеспечивается благодаря возможности функционирования МСЭ в режиме «стелс», что освобождает от необходимости изменения топологии сетевой подсистемы облака. При этом средства контроля доступа остаются невидимыми для участников информационного обмена.

СПИСОК ЛИТЕРАТУРЫ

1. Cloud Security Alliance. Top Threats to Cloud Computing. Март 2010 [Электронный ресурс] /Режим

доступа: <https://cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf> (Дата обращения 09.07.2012)

2. **Заборовский, В.С.** Сетецентрическая модель и методы контроля доступа к информационным ресурсам в среде облачных вычислений [Текст] / В.С. Заборовский, А.А. Лукашин // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2012. –№2 (145). –183 с.

3. **Zaborovsky, V.** Access Control in a Form of Active Queuing Management in Congested Network Environment [Text] / V. Zaborovsky, V. Mulukha // Proc. of the Tenth International Conf. on Networks, ICN 2011. –P. 12–17.

4. **Lukashin, A.** Dynamic Access Control Using Virtual Multicore Firewalls [Text] / A. Lukashin,

V. Zabo-rovsky // The Fourth International Conf. on Evolving Internet INTERNET 2012, Venice. –2012. –P. 37–43.

5. **Гайдамакин, Н.А.** Разграничение доступа к информации в компьютерных системах [Текст] / Н.А. Гайдамакин. –Екатеринбург: Изд-во Уральского ун-та, 2003. –328 с.

6. **Zaborovsky, V.** Dynamic Access Control in Cloud Services [Text] / V. Zaborovsky, A. Lukashin, S. Kupreenko, V. Mulukha // International Transactions on Systems Science and Applications. –2011. –Vol. 7. –№ 3/4. –P. 264–277.

УДК 004.94

Н.А. Осипов, Т.В. Зудилова, С.Э. Хоружников

ОБЛАЧНЫЕ ТЕХНОЛОГИИ ПРИ ОБРАБОТКЕ ДАННЫХ БОЛЬШОГО ОБЪЕМА

По мере роста производительности вычислительных систем исследователи получили дополнительные возможности при решении сложных задач, требующие в основном мощные аппаратные средства. Современный вычислительный эксперимент ориентирован в большей степени на данные и основан на неограниченных возможностях объема этих данных.

Для примера рассмотрим работу страховых компаний, в которых при выработке оптимальной стратегии страхования водителей собирается и оценивается разного рода статистика, в т. ч. пол, возраст, стаж водителя и множество других параметров, характеризующих человека за рулем, но все это косвенные оценки, не привязанные к конкретной личности [1]. С развитием современных средств связи появилась возможность устанавливать, например, в транспортном средстве блок с регистраторами и передатчиками, который ежесекундно, в режиме реального времени информирует о манере езды водителя, ускорениях при разгоне и торможении, скорости прохождения поворотов и др. параметрах. Обработка информации с регистраторов позволяет получать большие объемы данных, характеризующие поведение водителя за рулем. Анализ таких данных является более сложной задачей ввиду необходимости обработки больших объемов данных.

Применение аналогичных методов и средств в других предметных областях позволяет исследователям получать информацию в больших объемах, что неизбежно требует новых подходов к ее обработке и анализу. В литературе содержится описание подобных задач [1, 2], но не описаны методы их реализации на основе современных ИКТ.

Подобные исследовательские задачи решаются студентами факультета ИКТ НИУ ИТМО на практических занятиях по дисциплине «Моделирование инфокоммуникационных систем». Суммарный объем данных результатов моделирования сложных систем ввиду большого числа студентов и проведенных ими исследований превышает, как правило, несколько миллионов строк таблиц Excel. При применении стандартных подходов к обработке и анализу полученных данных требуются существенные вычислительные и временные ресурсы, которые нередко малоэффективны в решении актуальных задач моделирования из-за отсутствия надежности, разделения ресурсов, безопасности, использования старых вычислительных алгоритмов.

В статье рассматривается новый подход к решению задач обработки больших данных с использованием технологии облачных вычислений, при котором реализуется образовательная плат-



форма, позволяющая студентам получать качественно новый уровень знаний с использованием практически неограниченных вычислительных сервисов.

Основные характеристики больших данных

Экспоненциальный рост объема данных кардинально изменил природу исследований. Первой парадигмой науки был эксперимент, затем появилась теория, формализующая экспериментальные знания. В XX веке новой парадигмой стало вычислительное моделирование, позволившее проверять теории в тех областях, где экспериментирование было либо крайне дорого, либо невозможно. Созданные объемы данных привели к четвертой парадигме науки [3]: вместо того, чтобы использовать данные для проверки теории, теперь можно, опираясь на большие объемы данных, формулировать теории. Эти данные принимают форму достаточно точных моделей, которые было бы невозможно создать, основываясь на небольших наборах информации. Например, сейчас созданы точные системы автоматического перевода на основе моделей, построенных в результате байесовского статистического анализа больших совокупностей переведенных текстов.

Данные принято считать большими при наличии одного из трех признаков [2]:

- объем – количество строк в таблицах данных превышает десятки миллионов;
- разнородность данных – структурированные данные и информация из самых разнообразных источников данных хранятся совместно;
- скорость поступления данных – быстрый, плотный поток информации, поступающей с датчиков.

Считается, что если присутствует один из указанных признаков, то это задача из области больших данных. При этом признаки обычно оказываются взаимосвязаны, разнородные данные, как правило, имеют большие объемы, а информация с датчиков поступает с огромной скоростью.

Задача обработки результатов практических работ студентов, к которым относятся лабораторные и контрольные работы, домашние задания и результаты тестирования по дисциплине «Моделирование инфокоммуникационных систем» в полной мере обладает первым из перечисленных признаков.

Достоинства облачных вычислений в учебном процессе

Отметим основные преимущества и достоинства технологий облачных вычислений применительно к решаемой задаче [4, 5]:

- обеспечение при решении задач обработки и анализа больших данных высокой масштабируемости, надежности, разделения ресурсов, гибкости подключаемых сервисов, безопасности, использования старых вычислительных мощностей, а также легкости администрирования и лицензионной чистоты;

- эффективное использование учебных площадей для студентов, т. к. отпадает необходимость выделять отдельные и специально оборудованные помещения под традиционные компьютерные классы, что приводит к сокращению затрат, требующихся на создание и поддержание компьютерных классов;

- качественно новый уровень получения современных знаний по специальности, т. к. студенты получают доступ к учебным материалам и сервисам для решения задач моделирования в любое время и в любом месте, где есть Интернет;

- возможность быстрого создания аналитических отчетов и использования их результатов в ходе учебного процесса;

- возможность осуществления для студентов интерактивного общения с преподавателем в процессе решения возникающих вопросов;

- централизованное администрирование обслуживающим персоналом и преподавателем программных и информационных ресурсов, используемых в учебном процессе.

По сравнению с персональным компьютером вычислительная мощность, доступная студентам «облачных» компьютеров, практически ограничена лишь размером «облака», т. е. общим количеством удаленных серверов. Студенты могут запускать более сложные задачи, с большим количеством необходимой памяти, места для хранения данных, тогда, когда это необходимо. Возможность запуска множества копий приложения на виртуальных машинах представляет преимущества масштабируемости: количество экземпляров приложения способно практически мгновенно увеличиваться по требованию, в зависимости от нагрузок.

Таким образом, можно сделать вывод о целесообразности внедрения в учебный процесс при проведении дисциплин, подобных «Моделирование инфокоммуникационных систем», технологий облачных вычислений.

Реализация обработки данных большого объема с помощью PowerPivot

Для обработки результатов практических работ студентов базовые элементы бизнес-аналитики Microsoft Excel явно недостаточны. Здесь необходимо работать с большими данными в режиме реального времени. Применение существующей надстройки PowerPivot для Excel 2010 обеспечивает возможность добавления и интеграции больших объемов данных в книги Excel. PowerPivot преодолевает существующие ограничения анализа больших объемов данных на настольном компьютере с помощью эффективных алгоритмов сжатия для загрузки даже самых больших наборов данных в память.

Выбор PowerPivot для решения задачи обработки практических работ студентов обусловлен следующими причинами. Во-первых, PowerPivot представляет собой набор приложений и сервисов, позволяющих студентам и преподавателям самостоятельно создавать аналитические решения, поддерживает связывание между собой и расширение показателей в больших объемах данных, загруженных из гетерогенных источников, например, таких, как Microsoft SQL Server, Access, Excel, SQL Azure, SSAS, Oracle, текстовые файлы [6]. Во-вторых, на основании этих данных PowerPivot позволяет создавать таблицы и графики (PivotTables и PivotCharts), управляемые с помощью обычных и визуальных фильтров. И, в-третьих, файл PowerPivot, созданный с помощью Excel, можно опубликовать на портале SharePoint, использовавшемся при решении поставленной задачи для реализации облачной технологии. Ниже представлены этапы обработки данных, полученных студентами при решении исследовательских задач моделирования инфокоммуникационных систем с помощью PowerPivot (рис. 1).

Раскроем каждый этап подробнее.

Формирование исходных данных для обработки можно считать предварительным этапом. Исходными данными являются результаты лабораторных работ, тестов и домашних заданий 100 студентов, сведенные в отчет в таблицах Excel.

Загрузка данных выполняется в окно PowerPivot, которое может содержать несколько таблиц, каждая из которых находится на отдельной вкладке. Таблицы вместе со столбцами образуют базу данных, хранящуюся в памяти.

Список сданных отчетов сводится в книгу Excel. Он содержит идентификатор студента (StudentID), идентификатор работы (LabrabID) и идентификатор преподавателя (InstructorID).

На этапе *Преобразования данных* список представляется в виде связанной таблицы PowerPivot, а затем создаются связи с таблицами Student и Instructor, сопоставив StudentID с полем StudentKey, а InstructorID – с полем Instructor.

Для отображения данных в связанной таблице добавляются два новых вычисляемых столбца с помощью формулы RELATED().

Для определения количества сданных работ каждым студентом в таблице Student создается новый вычисляемый столбец, содержащий формулу, которая находит записи для каждого студента в таблице StudentLabrab и определяет количество сданных работ по каждому студенту с помощью формулы COUNTROWS().

На заключительном этапе *Создания и форматирования отчетов PowerPivot* формируются отчеты для анализа данных. Отчеты PowerPivot создавались на основании построенной модели PowerPivot с помощью одного из трех инструментов: PivotTables (для создания сводных таблиц), PivotCharts (для создания сводных диаграмм) и двух функций Excel [7] «CUBE» CUBEMEMBER

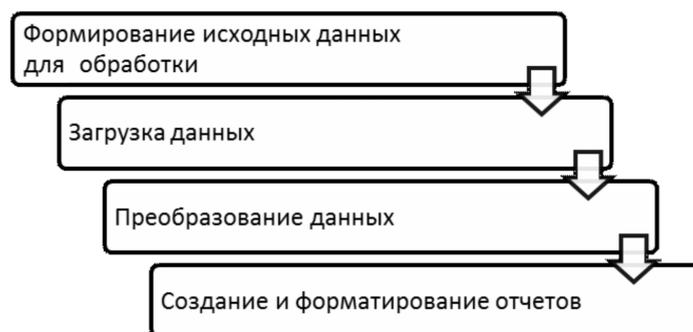


Рис. 1. Обработка данных

(возвращает элемент) и CUBEVALUE (возвращает агрегированное значение).

Для наглядности отчеты PivotTables и PivotCharts отформатированы с помощью стандартных стилей и настроек форматирования.

Отчеты PowerPivot размещены на сервере, что позволило расширить следующие функции PowerPivot:

1) импортировать миллионы строк данных из различных источников в одну книгу Excel, создавать отношения между разнородными данными, создавать вычисляемые столбцы и показатели с помощью формул, строить сводные таблицы и сводные диаграммы с последующим анализом данных;

2) создавать быстрые расчеты и анализировать большие объемы данных при максимально эффективном использовании возможностей многоядерных процессоров для быстрых вычислений.

Данные, с которыми студенты работают в окне PowerPivot, сохраняются в аналитической базе данных в книге Excel. Мощная система выполняет загрузку, запросы и обновление данных в базе данных. Поскольку данные PowerPivot внедряются в книгу Excel, они сразу становятся доступными для сводных таблиц, сводных диаграмм и других функций Excel, используемых для агрегатной обработки и взаимодействия с данными. Данные PowerPivot и объекты представления Excel хранятся внутри одного файла книги.

Облачные сервисы для обработки результатов данных реализованы через центр администрирования SharePoint 2010 [8]. Службы Excel включены на сервере SharePoint и студенты могут просматривать и взаимодействовать с книгами в окне браузера без необходимости установки какого-либо дополнительного клиентского ПО.

Развертывание клиентских и серверных приложений PowerPivot включает несколько компонентов, которые интегрируются с Excel и службами Excel в ферме SharePoint. На рис. 2 показаны клиентские и серверные компоненты PowerPivot, позволяющие реализовать облачные технологии.

Отметим следующие преимущества PowerPivot при решении задачи обработки практических результатов данных 100 студентов по курсу «Моделирование инфокоммуникационных систем»:

- применение стандартных возможностей и функций Excel;
- практически неограниченная поддержка источников данных. PowerPivot дает возможность импортировать и комбинировать источники данных из любого расположения для анализа больших объемов данных;
- удобная информационная панель управления PowerPivot, позволяющая отслеживать общие приложения и управлять ими для обеспечения безопасности, высокой доступности и производительности;
- новый язык формул – язык выражений анализа данных (DAX), расширивший имеющиеся в Excel возможности работы с данными и позволивший выполнять более сложное группирование, вычисления и анализ;
- публикация файла на сервере SharePoint с размещением сервисов PowerPivot для обработки данных. Студенты могут просматривать результаты обработки своих данных и оперативно реагировать на них с целью, например, повышения оценки и продолжения работ.

Таким образом, применение PowerPivot совместно с SharePoint позволили повысить производительность и эффективность работы студентов и преподавателя при обработке больших объемов

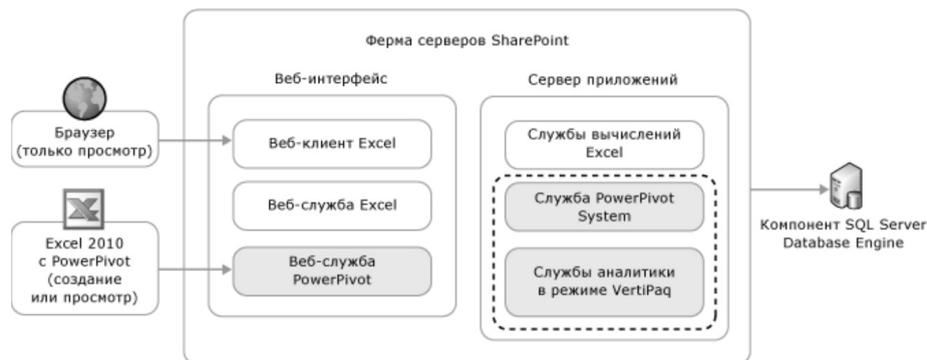


Рис. 2. Компоненты PowerPivot

аналитических данных студентов с использованием облачных технологий.

Благодаря ресурсам облака удобные инструментальные средства, с которыми студенты и преподаватели ежедневно работают, стали во много раз более мощными. Появилась возможность для преподавателей работать с большим количеством данных и применять более сложные вычисления, используя при этом привычные инструменты.

Проведенные исследования показали, что применение PowerPivot, особенно совместно с SharePoint, за счет реализации доступа к данным и их обработки повысило производительность и

эффективность работы при решении аналитических задач студентами с одной стороны и обработки их результатов преподавателями с другой. Возможности реализованных облачных технологий обеспечили обработку требуемого объема данных с существенным запасом, а время обработки снизилось до нескольких минут.

Поставленная задача реализована в рамках выполнения пилотного проекта «Разработка и создание сегмента корпоративной облачной инфраструктуры для формирования системы воспроизводства высококвалифицированных кадров» в Центре Авторизованного обучения ИТ НИУ ИТМО.

СПИСОК ЛИТЕРАТУРЫ

1. **Черняк, Л.** О Больших Данных с четырех сторон [Электронный ресурс] / Л. Черняк. –Режим доступа: <http://www.osp.ru/os/2011/10/13012226/> (Дата обращения 21.04.2012)

2. **Дубова, Н.** Большие Данные – комплексный подход [Электронный ресурс] / Н. Дубова. –Режим доступа: <http://www.osp.ru/news/articles/2012/13/13014203/> (Дата обращения 22.04.2012).

3. **Геннон, Д.** Облака: демократизация научных вычислений [Электронный ресурс] / Д. Геннон, Д. Рид, Р. Барга. –Режим доступа: <http://www.osp.ru/os/2011/02/13007709/> (Дата обращения 22.04.2012)

4. **Риз, Д.** Облачные вычисления [Текст] / Д. Риз –СПб.: БХВ-Петербург, 2011. –288 с.

5. **Фингар, П.** Облачные вычисления –

бизнес-платформа XXI века [Текст] / П. Фингар. –М.: Книга, 2011. –256с.

6. Справка по PowerPivot: Справочная документация [Электронный ресурс]. –Режим доступа: <http://technet.microsoft.com/ru-ru/library/hh597808/> (Дата обращения 02.04.2012)

7. Функции PowerPivot: Справочная документация. [Электронный ресурс]. –Режим доступа: <http://msdn.microsoft.com/ru-ru/library/ee210639%28v=sql.105%29.aspx/> (Дата обращения 25.04.2012)

8. Компоненты и средства PowerPivot: Справочная документация. [Электронный ресурс]. –Режим доступа: http://msdn.microsoft.com/ru-ru/library/ee210653%28v=sql.105%29.aspx (Дата обращения 02.04.2012)

АЛГОРИТМ АНАЛИЗА И СЖАТИЯ ДАННЫХ ПО ТОПОЛОГИЧЕСКОЙ СТРУКТУРЕ ИССЛЕДУЕМОГО ОБЪЕКТА

Назначение и структура Активного редактора

В системах компьютерного моделирования технических и технико-экономических систем [1] редакторы входных данных, заносимых в основном в виде схем, сетей или цепей, обладают только функциями восприятия и трансляции этой информации к вычислительной части системы моделирования. В то же время во многих практических ситуациях входная информация в силу топологических особенностей объекта исследования и его формализованного представления требует предварительной обработки с целью ее сжатия, редуцирования или определенной интерпретации.

Для решения этой задачи предлагается ввести понятие «Активный редактор».

Активный редактор (АР) – многорежимный редактор, способный не только воспринимать и передавать информацию, заносимую в него пользователем, но и преобразовывать полученную информацию к заданному виду.

Такой редактор имеет два слоя:

структурный – редактор схем в основном режиме работы и генератор моделей компонентов (ГМК);

операционный, включающий в себя функциональные модули, задающие различные режимы работы АР.

Режимами работы активного редактора являются следующие:

- выделение подцепей в структуре сложной цепи – *Reductor subcircuits*;
- преобразование (редукция) схем полученных подцепей – *Reductor circuits*;
- работа с векторными связями у компонентов схемы – *Vector-branches*;
- автоматическая генерация регулярных (циклических) структур – *Cycle circuits*;
- установление эквивалентности схем для поиска прототипов в БД – *Ecivalence subcircuits*;
- генератор моделей компонентов – *Generator of models*.

В настоящей статье рассмотрены алгоритмы, реализующие первые два из перечисленных выше режимов.

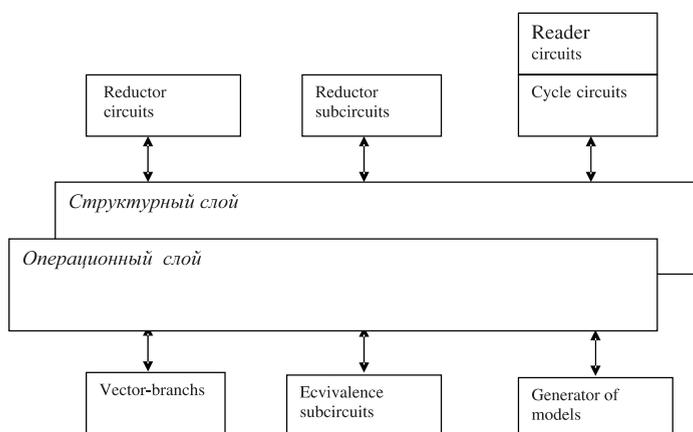


Рис. 1. Структура Активного редактора

Специальные режимы редактирования

Все топологические процедуры, связанные с вводом, редактированием, обработкой и исследованием топологической информации, базируются на понятии компонентной цепи (КЦ) [1], представляемой тройкой объектов $C = (K, B, N)$. В этой же работе определены понятия компонента (K), связей компонента, ветвей цепи (B) и узлов цепи (N). Дана классификация компонентов и связей, определены типы связей, включая векторные. Классификация строится на основе использования топологических, физических и математических предикатов (признаков). Основными для данной работы являются *топологические предикаты* моделей компонентов и цепей, однако участвуют и остальные предикаты.

Автоматическое разбиение исходной цепи большой размерности на подцепи. Для систем большой размерности актуальным становится поблочный ввод информации. Для этого желательно иметь возможность автоматического разбиения исходной цепи на подцепи [2]. *Подцепью* (ПЦ) будем называть произвольную совокупность компонентов цепи. Полагаем, что ПЦ описывается набором внешних и внутренних переменных, относящихся соответственно к внешним и внутренним ветвям и узлам. Внешним является узел подцепи, в который ведет хотя бы одна связь компонента, не принадлежащего данной ПЦ, и всякая ветвь, ведущая во внешний узел. Внешние ветви ПЦ выполняют роль, аналогичную связям компонентов.

Путь цепь C содержит Q компонентов. Необходимо представить C в виде совокупности ПЦ, т. е. $C = \cup P_i$ для которых $\sum_{i \in I} S_i$ минимальна (S_i – число внешних ветвей подцепи). Каждая ПЦ содержит не более $q < Q$ компонентов. Для решения поставленной задачи предлагается алгоритм выращивания ПЦ вокруг заданных компонентов. Вводим расстояние $\rho_{\alpha\beta}$ между компонентами K_α и K_β как минимальное число ветвей цепи, которое нужно пройти при построении пути от одного компонента к другому. Если пути между K_α и K_β не существует, полагаем $\rho_{\alpha\beta} = \infty$. Полагаем также, что $\rho_{\alpha\beta} = \rho_{\beta\alpha}$ и $\rho_{\alpha\alpha} = 0$, а $\rho_{\alpha\beta} = 1$, если K_α и K_β имеют общий узел.

Пусть имеется группа компонентов $K_i, i \in I$, заданная индексным множеством I и компонент K_α . Расстояние от K_α до группы K_i

$$\rho_{\alpha I} = \sum_{i \in I} \rho_{\alpha i}.$$

Алгоритм поиска минимального пути в графе путем выращивания подцепей реализуется следующими шагами.

Шаг 1. Фиксируем компоненты $K_j, j \in J$ в качестве исходных подцепей. Индексное множество J задает пользователь.

Шаг 2. Для каждой подцепи ищем ближайший компонент на основании введенного понятия расстояния $\rho_{\alpha I}$, которое определяем следующим образом.

Помечаем единицами все компоненты, смежные с K_α . Далее помечаем двойкой все непомеченные компоненты, смежные с помеченными ранее, и т. д. Как только все компоненты $K_i, i \in I$ окажутся помеченными, определяем расстояние $\rho_{\alpha I}$ как сумму пометок, соответствующих компонентам $K_i, i \in I$.

Шаг 3. Присоединяем выбранные компоненты к ПЦ. Требуем, чтобы подцепи не пересекались и их размерность не превышала заданную.

Шаг 4. Уточняем структуру ПЦ обменом компонентов между подцепями. Обмен считается целесообразным, если он ведет к уменьшению числа внешних ветвей подцепей.

Шаг 5. Графическое описание ПЦ преобразуется в текстовый файл и далее эта информация больше не хранится.

Шаг 6. Переход к описанию КЦ из ПЦ. Завершение алгоритма.

Алгоритм редукции подцепей. После того как большая цепь свернута до уровня подцепей, в дальнейшем может быть произведена следующая упаковка топологической информации, которая осуществляется методом редукции [3, 4].

Для реализации этого метода не нужно строить никаких матриц. Основой метода редукции является замена двух параллельно или последовательно соединенных элементов одним эквивалентным. Это наиболее простой вариант соединения. Наиболее сложным является замена звездообразного соединения многоугольником.

Основные этапы алгоритма редукции представлены последовательностью шагов.

Шаг 1. Выбрать два элемента.

Шаг 2. Проверить, возможно ли выполнение свертки параллельного соединения. Если да, то перейти на шаг 3, иначе – на шаг 5.

Шаг 3. Выполнить свертку двух параллельно соединенных элементов.

Шаг 4. Перейти на шаг 13.

Шаг 5. Проверить, возможно ли выполнение

свертки последовательного соединения. Если да – перейти к шагу 6, иначе – к шагу 8.

Шаг 6. Выполнить свертку двух последовательно соединенных элементов.

Шаг 7. Перейти к шагу 13.

Шаг 8. Проверить, имеют ли эти элементы общий узел. Если да, то перейти на шаг 9, в противном случае – на шаг 13.

Шаг 9. Определить число подключенных к общему узлу элементов. Если число элементов равно трем, перейти на шаг 10, если больше трех – перейти на шаг 12.

Шаг 10. Проверить, не имеет ли третий элемент второго общего узла с первым и вторым элементами. Если не имеет – перейти на шаг 11, иначе – на шаг 13.

Шаг 11. Выполнить преобразование звезда-треугольник.

Шаг 12. Перейти на шаг 16.

Шаг 13. Выбрать все элементы, подходящие к узлу.

Шаг 14. Проверить, не имеют ли вновь выбранные элементы второго общего узла с первым и вторым элементами. Если не имеют, то перейти на шаг 15, иначе – на шаг 16.

Шаг 15. Выполнить преобразование n -лучевая звезда – многоугольник.

Шаг 16. Проверить, возможно ли дальнейшее упрощение, если да – перейти на шаг 1, если нет – перейти на шаг 14.

Шаг 17. Выход.

На рис. 2 приведен пример работы алгоритма редукции на основе цепи.

Пример тестовой схемы алгоритма редукции. Для схемы, представленной на рис. 2 а, опишем работу алгоритма последовательностью выполнения этапов – преобразований.

На первом этапе работы алгоритма выполняется преобразование звезда-треугольник, в котором участвуют резисторы R_1 , R_3 и R_5 . В результате выполнения преобразования получаются новые элементы R_6 , R_7 и R_8 (рис. 2 б).

На втором и третьем этапе работы алгоритма выполняется свертка двух параллельно стоящих резисторов R_2 , R_6 и R_4 и R_7 соответственно. В результате выполнения сверток получаем резисторы R_9 и R_{10} (рис. 2 в).

На четвертом этапе работы алгоритма выполняется свертка двух параллельно стоящих резисторов R_{11} и R_8 . Результат свертки – резистор R_{12} (рис. 2 д).

Поскольку дальнейшая свертка схемы невозможна, работа алгоритма завершается.

Таким образом, в данной статье введено понятие Активного редактора как дополнение обычного редактора операционным слоем, в котором реализуются все необходимые функциональные характеристики. Приведены алгоритмы, реализующие функции автоматического разбиения большой цепи на подцепи и алгоритм редукции подцепей.

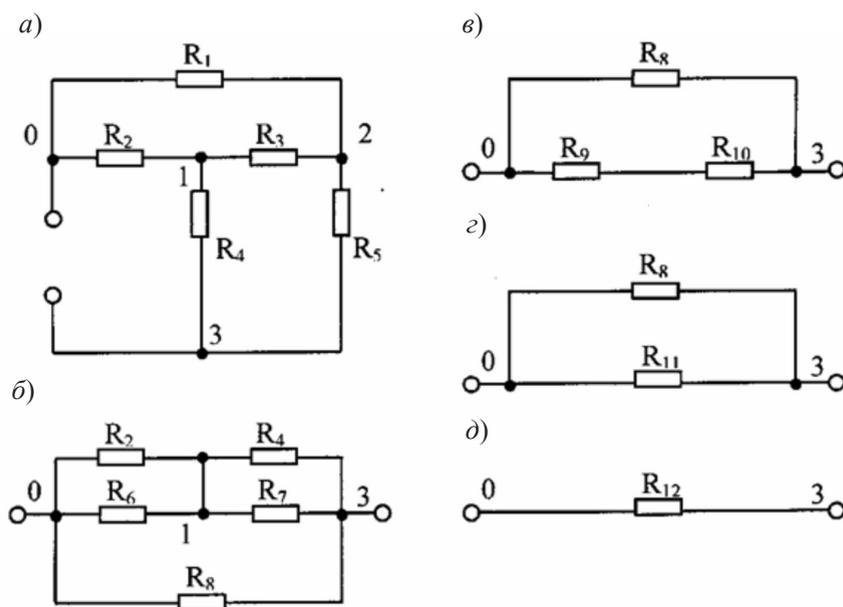


Рис. 2. Последовательность эквивалентных преобразований тестовой схемы

СПИСОК ЛИТЕРАТУРЫ

1. **Дмитриев, В.М.** МАРС – среда моделирования технических устройств и систем [Текст] / В.М. Дмитриев, А.В. Шутенков [и др.]. –Томск: Изд-во «В-Спектр», 2011. –278с.
2. **Арайс, Е.А.** Алгоритмы и программы анализа сложных цепей и систем [Текст] / Е.А. Арайс, В.М. Дмитриев. –Изд-во ТГУ, 1976. –167 с.
3. **Ахо, А.** Построение и анализ вычислительных алгоритмов [Текст] / А. Ахо, Дж. Хопкрофт. –М.: Мир, 1979. –527 с.
4. **Сешу, С.** Линейные графы и электрические цепи [Текст] / С. Сешу, М.Б. Рид. –М.: Высш. школа, 1971. –447 с.

УДК 681.5.013

А.В. Борисевич

СИНТЕЗ АЛГОРИТМОВ СТАБИЛИЗАЦИИ НЕЛИНЕЙНЫХ ОБЪЕКТОВ НА ОСНОВЕ СХЕМЫ ПРОДОЛЖЕНИЯ РЕШЕНИЯ ПО ПАРАМЕТРУ

Аффинные нелинейные системы – это особый класс моделей в теории автоматического управления, описывающих многие технические процессы [1–3]: динамику мобильных роботов, процессы в электродвигателях, пневмоприводе, гидроприводе и т. д.

Рассмотрим реализацию многомерной системы с m входами и m выходами в пространстве состояний размерности n :

$$\begin{aligned} \dot{x} &= f(x) + \sum_{i=1}^m g_i(x)u_i, \\ y &= h(x), \end{aligned} \quad (1)$$

где $x \in X \subseteq \mathbb{R}^n$, $y \in Y \subseteq \mathbb{R}^m$, $u \in U \subseteq \mathbb{R}^m$, отображения $f: \mathbb{R}^n \rightarrow \mathbb{R}^n$, $g_i: \mathbb{R}^n \rightarrow \mathbb{R}^n$, $h: \mathbb{R}^n \rightarrow \mathbb{R}^m$ – гладкие векторные поля $f, g, h \in C^\infty$. Функции $f(x)$ и $g(x)$ считаем ограниченными на X .

В статье рассматривается частная задача установки константного значения на выходе системы.

Определение 1. Задача установки постоянного значения на выходе (setpoint tracking) – синтез такого закона управления $u(t) = u(x)$, который асимптотически переводит выход y объекта управления в состояние $\bar{y} = \text{const}$: $\lim_{t \rightarrow \infty} y(t) = \bar{y}$. В частном случае, при $\bar{y} = 0$ управление называется *обнулением выхода* (output-zeroing problem).

Без потери общности будем рассматривать задачу обнуления выхода $\lim_{t \rightarrow \infty} y(t) = 0$. Эта задача может быть решена методом линеаризации по обратной связи [4, 5]. Главная идея подхода со-

стоит в трансформации с помощью нелинейной обратной связи системы $N: u(t) \mapsto y(t)$ вида (1) в систему с линейной динамикой $L: v(t) \mapsto y(t)$ и такими же выходами y , но новыми входами v . После трансформации полученная система L может управляться любыми известными методами линейной теории автоматического управления (модальное управление, линейный квадратичный регулятор и т. д.).

Технически предлагаемый подход основан на методе численного продолжения по параметру для решения систем нелинейных уравнений [6], состоящем в параметризованном комбинировании исходной задачи и некоторой очень простой с известным решением. В предлагаемом методе осуществляется непрерывная параметрическая деформация тривиального объекта управления (цепочки интеграторов) с известным управлением в исходную аффинную нелинейную систему. Следует заметить, что применение методов вариации параметра (продолжения по параметру) активно используется как для синтеза регуляторов [7], так и для поиска оптимальных траекторий [8].

Настоящий подход отличается тем, что алгоритм продолжения по параметру непосредственно встраивается в регулятор нелинейной системы, синтезированный на основе метода линеаризации по обратной связи. Таким образом, одновременно реализуется возможность компенсации неопределенностей модели внешним линейным контуром управления (как это обычно осуществляется на практике при использовании методов линеари-

зации [9]) вместе с возможностью преодоления особых точек (сингулярностей) линеаризующего преобразования.

1. Предварительные сведения

Линеаризация по обратной связи. В этом разделе мы будем следовать материалу работы [5] для описания сущности метода точной (полной) линеаризации по обратной связи (output feedback linearization).

Определение 2. Многомерная нелинейная система имеет относительную степень r_j по выходу y_j в области $\mathfrak{S} \subseteq \mathbb{R}^n$ если по крайней мере для одной функции g_i

$$\mathcal{L}_{g_i} \mathcal{L}_f^{r_j-1} h_j \neq 0, \quad (2)$$

где $\mathcal{L}_f \lambda = \frac{\partial \lambda(x)}{\partial x} f(x) = \sum_{i=1}^n \frac{\partial \lambda(x)}{\partial x_i} f_i(x)$ – производная Ли функции λ по полю f .

Это означает, что по крайней мере один вход u_k влияет на выход y_j после r_j интегрирований.

Число $r = \sum_{j=1}^m r_j$ называется *относительной степенью системы*. Если $r = n$ и матрица

$$A(x) = \begin{pmatrix} \mathcal{L}_{g_1} \mathcal{L}_f^{r_1-1} h_1(x) & \cdots & \mathcal{L}_{g_m} \mathcal{L}_f^{r_1-1} h_1(x) \\ \vdots & \cdots & \vdots \\ \mathcal{L}_{g_1} \mathcal{L}_f^{r_m-1} h_m(x) & \cdots & \mathcal{L}_{g_m} \mathcal{L}_f^{r_m-1} h_m(x) \end{pmatrix} \quad (3)$$

полноранговая, то исходная динамическая система в области \mathfrak{S} эквивалентна системе:

$$y_j^{(r_j)} = \mathcal{L}_f^{r_j} h_j + \sum_{i=1}^m \mathcal{L}_{g_i} \mathcal{L}_f^{r_j-1} h_j u_i = B(x) + A(x)u. \quad (4)$$

Нелинейная обратная связь

$$u = A(x)^{-1}[v - B(x)] \quad (5)$$

переводит в области \mathfrak{S} исходную нелинейную динамическую систему в линейную:

$$y^{(r_j)} = v_j. \quad (6)$$

Управление нелинейной системой состоит из двух петель обратной связи, одна из которых реализует линеаризующую трансформацию, а вторая осуществляет управление системой любым известным методом из линейной теории управления.

Известными недостатками, ограничивающими область применения линеаризации по обратной связи на практике, являются условия постоянства относительной степени системы r и полноранговость матрицы $A(x)$ во всем фазовом пространстве \mathfrak{S} .

Численное продолжение по параметру.

Если рассматривать нелинейный объект управления без его динамики, то задача управления выродается в решение нелинейного уравнения

$$\varphi(\xi) = 0, \quad (7)$$

где $\varphi: \mathbb{R}^m \rightarrow \mathbb{R}^m$ – некоторая гладкая функция.

Пусть $\Omega \subset \mathbb{R}^m$ – открытое множество и $C(\bar{\Omega})$ – множество непрерывных отображений из его замыкания $\bar{\Omega}$ в \mathbb{R}^m . Две функции $F_0, F_1 \in C(\bar{\Omega})$ гомотопны (гомотопически эквивалентны) если существует непрерывное отображение

$$H: \bar{\Omega} \times [0, 1] \rightarrow \mathbb{R}^m, \quad (8)$$

такое, что $H(\xi, 0) = F_0(\xi)$, $H(\xi, 1) = F_1(\xi)$ для всех $\bar{\Omega}$. По теореме об инвариантности степени Брауэра [6] при гомотопической эквивалентности уравнение $H(\xi, \lambda) = 0$ имеет решения (ξ, λ) для любых $\lambda \in [0, 1]$.

Численное продолжение по параметру рассматривает разрешимость уравнения для случая $F_1 = \varphi$ и F_0 – некоторой функции с известными корнями. Сущность алгоритмов численного продолжения по параметру состоит в поиске кривой, начинающейся в точке $(\xi_0, 0)$, где $F_0(\xi_0) = 0$, и заканчивающейся в точке $(\xi^*, 1)$, где ξ^* – искомое решение.

2. Основной результат

В этом разделе мы изложим основной результат, являющийся теоретическим базисом предложенного подхода, начиная от простых фактов и заканчивая описанием гибридного алгоритма управления, сочетающего линеаризацию по обратной связи с методом продолжения по параметру.

В [10] указано, что задача управления состоянием аффинной системы связана с решением $\xi^*(t): \mathbb{R}_+ \rightarrow \mathbb{R}^n$ нестационарного нелинейного уравнения $\varphi(\xi, t) = 0$, в котором $\frac{\partial \varphi}{\partial \xi} = g(z)$,

$\frac{\partial \varphi}{\partial t} = f(z)$: управление $u(t) = \frac{d\xi^*(t)}{dt}$ такое, что

$\varphi(\xi^*(0), 0) = x(0)$, переводит систему из состояния $x(0)$ в $x(T) = 0$ при $T \rightarrow \infty$. Ниже мы приведем другой метод, отличающийся от описанного в [10] тем, что не генерирует разрывных управляющих траекторий.

Метод продолжения по параметру для нестационарных нелинейных уравнений. Рассмотрим решение нестационарного уравнения

$\varphi(\xi, t) = 0$. Составим параметризованное одно- временно по времени и по параметру λ гомотопическое отображение:

$$H(\xi, \lambda, t) = (1 - \lambda)(\xi - \xi_0) + \lambda\varphi(\xi, t), \quad (9)$$

где ξ_0 – начальное приближение к решению.

Сформулируем и дадим краткие доказательства ряда утверждений, на которых базируется метод продолжения по параметру для решения систем нестационарных уравнений.

Предположение 1. Если $\xi(t)$ и $\lambda(t)$ – решение уравнения, то $rank DH_{\xi, \lambda}(\xi(t), \lambda(t), t) = m$.

Это предположение является стандартным допущением при использовании методов продолжения по параметру, согласно которому вдоль траектории $(\xi(t), \lambda(t))$ могут существовать только предельные точки, где $D_\lambda H \notin im D_\xi H$. Точки бифуркации, в которых $rank DH_{\xi, \lambda}(\xi, \lambda, t) < m$ и $D_\lambda H \in im D_\xi H$ не рассматриваются, хотя это принципиально может быть сделано с привлечением соответствующих техник численной реализации методов продолжения по параметру [6].

Уравнение $H(\xi, \lambda, t) = 0$ для $\lambda \in [0, 1]$ определяет неявно заданную функцию $\xi(t)$, параметризованную от $\lambda(t)$ и удовлетворяющую уравнению, полученному дифференцированием по времени:

$$\dot{H}(\xi(t), \lambda(t), t) = \frac{\partial H}{\partial \xi} \dot{\xi}(t) + \frac{\partial H}{\partial \lambda} \dot{\lambda}(t) + \frac{\partial H}{\partial t} = 0. \quad (10)$$

Обозначив

$$A = \begin{pmatrix} \frac{\partial H}{\partial \xi} & \frac{\partial H}{\partial \lambda} \end{pmatrix}, B = -\frac{\partial H}{\partial t}, \tau = \begin{pmatrix} \dot{\xi} \\ \dot{\lambda} \end{pmatrix}, \quad (11)$$

можно представить как линейное матричное уравнение относительно $\dot{\xi}$ и $\dot{\lambda}$:

$$A\tau = B. \quad (12)$$

Лемма 1. Уравнение (12) всегда имеет решение.

Поскольку $A = DH_{\xi, \lambda}(\xi, \lambda, t)$, то для того, чтобы недоопределенное уравнение имело решение, необходимо и достаточно, чтобы $rank DH_{\xi, \lambda}(\xi, \lambda, t) = m$, что основывается на предположении 1.

Лемма 2. Все решения могут быть представлены в виде $\tilde{\tau} = \alpha \cdot \tau + \bar{\tau}$, где $\bar{\tau} = A^+ B$, $\tau \in ker A$, $\alpha \in \mathbb{R}$.

Достаточно очевидно, поскольку $dim ker A = 1$ и все нуль-пространство A может быть параметризовано одной переменной $\alpha \in \mathbb{R}$, а пространство решений W неоднородного уравнения вида

определяется как $W = \{A^+ B\} \oplus ker A$.

Докажем утверждение о липшицевости отображения $\Psi: \mathcal{D} \rightarrow \mathbb{R}^{m+1}$, необходимое в дальнейшем.

Теорема 1. Пусть $H: \mathcal{D} \rightarrow \mathbb{R}^m - C^1$ -непрерывная функция на некотором открытом множестве $\mathcal{D} \subset \mathbb{R}^{m+2}$, и матрица якобиана A полноранговая $rank A = m$ для всех $(\xi, \lambda, t) \in \mathcal{D}$. Тогда для каждого $(\xi, \lambda, t) \in \mathcal{D}$ существует уникальный вектор $\tilde{\tau} \in \mathbb{R}^{m+1}$, такой, что

$$\tilde{\tau} = \alpha \cdot \tau + \bar{\tau}, \bar{\tau} = A^+ B, \quad (13)$$

$$A \cdot \tau = 0, \|\tau\|_2 = 1, \det \begin{pmatrix} A \\ \tau^T \end{pmatrix} > 0, \alpha = \text{const} \in \mathbb{R}_+,$$

и отображение

$$\tilde{\Psi}: \mathcal{D} \rightarrow \mathbb{R}^{m+1}, \tilde{\Psi}: (\xi, \lambda, t) \mapsto \tilde{\tau}, \quad (14)$$

локально липшицево на \mathcal{D} .

Доказательство. Вектор $\tilde{\tau} = \alpha \cdot \tau + \bar{\tau}$ определен как сумма двух компонентов, про один из которых τ известно (уравнение 2.1.9 в [6]), что он является липшицевой функцией на \mathcal{D} . Отсюда необходимо доказать липшицевость функции $\bar{\tau} = A^+ B = (DH_{\xi, \lambda})^+ DH_t$. Единственность $\bar{\tau}$ следует из единственности псевдоинверсии Мура-Пенроуза.

Можно полагать, что DH – липшицево на \mathcal{D} с константой γ , что влечет за собой существование и ограниченность вторых производных H . Поскольку $rank A = m$, то операция взятия псевдоинверсии A^+ непрерывна и дифференцируема [11]. Отсюда произведение $A^+ B$ липшицево, поскольку его компоненты липшицевы.

Функция $\tilde{\Psi}$ задает автономное дифференциальное уравнение

$$\frac{d}{dt} \begin{pmatrix} \xi \\ \lambda \end{pmatrix} = \tilde{\Psi}(\xi, \lambda, t), \xi(0) = \xi_0, \lambda(0) = 0, \quad (15)$$

которое имеет единственное решение $\xi(t), \lambda(t)$, согласно теореме существования и единственности решения задачи Коши.

Теорема 2. Множество $H^{-1}(\{0\})$ односвязно.

Доказательство. По условию теоремы 2.1 из [12], если для отображения $F: \mathbb{R}^n \rightarrow \mathbb{R}^k$, $k \leq n$ выполняется

$$\sup \{ \| (DF(x) D^T F(x))^{-1} \| \mid x \in \mathbb{R}^n \} < \infty,$$

то $F^{-1}(0)$ – связное подмножество размерности $(n - k)$ в \mathbb{R}^n .

В соответствии с предположением 1, $rank DH = m$, следовательно $rank D^T H = m$, $rank DH \cdot D^T H = m$, инверсия матрицы $(DH \cdot D^T H)^{-1}$

определена и ее норма ограничена.

Теорема 3. Необходимым условием существования интегральной кривой $(\xi(t), \lambda(t))$ уравнения (15), соединяющей точки $(\xi_0, 0)$ и $(\xi^*, 1)$, является $\alpha > 0$.

Доказательство. На основании леммы 2 и теоремы 1 очевидно, что задача Коши вида (15) имеет единственное решение $(\xi(t), \lambda(t))$, удовлетворяющее уравнению (9) для каждого $t: H(\xi(t), \lambda(t), t) = 0$.

Для определения знака α рассмотрим поведение кривой $\gamma(t) = (\xi(t), \lambda(t))$ вблизи $t = 0$. Поскольку

$$A = \left(\lambda \frac{\partial \varphi}{\partial \xi} + (1-\lambda)E \varphi - \xi \right), B = -\lambda \frac{\partial \varphi}{\partial t}, \quad (16)$$

то вблизи $t = 0$ уравнение (9) ведет себя как стационарное с $B = 0$. Покажем, что $\alpha \neq 0$. Если $\alpha = 0$, то в момент времени $t = 0$ из леммы 2 следует, что $\dot{\xi}(0) = 0, \dot{\lambda}(0) = 0$, отсюда $\dot{\lambda}(t) = 0$ для любого t и точка $\lambda = 1$ недостижима. Поскольку вблизи $t = 0$ уравнение стационарно и решение определяется с помощью известного метода продолжения по параметру [6], то ориентация вектора τ согласно условию $\begin{pmatrix} A \\ \tau^T \end{pmatrix} > 0$ не должна изменяться. Отсюда $\alpha > 0$.

Сформулируем и дадим краткое доказательство предположению относительно поведения кривой решения $\gamma(t) = (\xi(t), \lambda(t))$.

Теорема 4. Существует такое число $\alpha_0 \in \mathbb{R}$, что интегральная кривая $\gamma(t) = (\xi(t), \lambda(t))$ уравнения (15) при $\alpha > \alpha_0$ имеет конечную длину между точками $(\xi_0, 0)$ и $(\xi^*, 1)$.

Доказательство. Рассмотрим структуру правой части (15). Поскольку константа α может быть выбрана произвольно и сколь угодно большой, то слагаемым $\bar{\tau}$ можно пренебречь и записать:

$$\tilde{\tau} = \alpha \cdot \tau + \bar{\tau} \xrightarrow{\alpha \rightarrow \infty} \alpha \cdot \tau. \quad (17)$$

Известно, что функция $\bar{\tau}$ липшицева на \mathcal{D} (теорема 1), из этого автоматически следует ограниченность $\bar{\tau}$. Тогда всегда можно выбрать конечное α таким образом, чтобы $\tilde{\tau} = \alpha \cdot \tau + \varepsilon, \varepsilon \ll \alpha \cdot \tau$.

Поскольку α – конечное число, то $\alpha \cdot \tau$ как правая часть (15) удовлетворяет известным результатам о конечности траектории решения (лемме 2.1.13 и теореме 2.1.14 из [6]). Отсюда при соответствующем выборе $\alpha > \alpha_0$ кривая $(\xi(t), \lambda(t))$ не имеет предельных точек и диффеоморфна прямой – имеет конечную длину между $\lambda_0 = 0$ и $\lambda_1 = 1$.

Продолжение по параметру для управления нелинейными аффинными системами.

Сопоставим с объектом управления систему с линейной динамикой, в которой m входов u, n состояний z, m выходов λ с такими же относительными степенями r_i , как и у (1):

$$\dot{z} = Az + Bu, \eta = Cz, \frac{d^{(r_i)}}{dt^{(r_i)}} \eta = u_i. \quad (18)$$

Запишем уравнение гомотопического отображения, связывающего динамику выходов систем (1) и (18):

$$H = (1-\lambda)\eta + \lambda y = 0. \quad (19)$$

По определению относительной степени выхода, каждый компонент H_i должен быть продифференцирован r_i раз по t до тех пор, пока не станет явной функцией от какого-либо входа u . Получаем после дифференцирования:

$$\begin{aligned} H_i^{(r_i)} = & -\sum_{k=1}^{r_i-1} C_{r_i}^k \eta_i^{(r_i-k)} \lambda^{(k)} + (1-\lambda)u_i + \\ & + (y_i - \eta_i) \lambda^{(r_i)} + \sum_{k=1}^{r_i-1} C_{r_i}^k y_i^{(r_i-k)} \lambda^{(k)} + \\ & + \lambda \left(\mathcal{L}_f^{r_i} h_i + \sum_{k=1}^m \mathcal{L}_{g_k} \mathcal{L}_f^{r_i-1} h_i u_k \right) = 0, \end{aligned} \quad (20)$$

что дает:

$$\begin{aligned} H^{(r_i)} = & \mathcal{A}_{i,1}(x, z, \Lambda)u + \\ & + \mathcal{A}_{i,2}(x, z, \Lambda)\lambda^{(r_i)} + \mathcal{B}_i(x, z, \Lambda), \end{aligned} \quad (21)$$

где $\Lambda = (\lambda, \dot{\lambda}, \ddot{\lambda}, \dots, \lambda^{(r_i-1)})$, C_n^k – биномиальные коэффициенты.

Рассматривая все компоненты H_i после дифференцирования в соответствии с относительными степенями выходов r_i можно записать алгебраическое условие, задающее непрерывную деформацию системы (18) в (1):

$$\begin{aligned} \bar{H} = & \mathcal{A}_1(x, z, \bar{\Lambda})u + \\ & + \mathcal{A}_2(x, z, \bar{\Lambda})\lambda^{(r_{\max})} + \mathcal{B}(x, z, \bar{\Lambda}) = 0, \end{aligned} \quad (22)$$

где $r_{\max} = \max\{r_i\}$, $\bar{\Lambda} = (\lambda, \dot{\lambda}, \ddot{\lambda}, \dots, \lambda^{(r_{\max}-1)})$,

По аналогии с методом продолжения по параметру для нестационарных нелинейных уравнений, связанная траектория $(u(t), \lambda(t))$, удовлетворяющая для $t \in [0, T)$ уравнению, определяется из следующей системы дифференциально-алгебраических уравнений:

$$\begin{pmatrix} u \\ \lambda^{(r_{\max})} \end{pmatrix} = \alpha \cdot \tau + \bar{\tau}, \quad (23)$$

$$\bar{\tau} = \mathcal{A}^+ \mathcal{B},$$

$$\mathcal{A} \cdot \tau = 0, \|\tau\|_2 = 1, \det \begin{pmatrix} \mathcal{A} \\ \tau^T \end{pmatrix} > 0,$$

где $\alpha > 0, \alpha \in \mathbb{R}$ – некоторая скалярная константа; \mathcal{A}^+ – инверсия Мура–Пенроуза матрицы \mathcal{A} .

Уравнение (22) задает обратную связь по состоянию, а (23) – динамику регулятора. Также стоит заметить, что, согласно (20), выражение для \mathcal{B} и \mathcal{A}_2 зависит в явной форме от y , что реализует обратную связь по выходу.

Переключательная стратегия для регуляризации при линейаризации по обратной связи. Все объекты управления на практике подвержены вариации параметров. Предложенный в предыдущем подразделе вариант управления определенно чувствителен к параметрическим неопределенностям в объекте управления. С другой стороны, при линейаризации по обратной связи вариации параметров объекта управления могут быть скомпенсированы за счет регулятора для линейаризованной системы [5, 9]. Рассмотрим гибридный метод, сочетающий в себе возможность применения внешнего контура регулирования и устойчивый к изменению относительной степени системы.

Система вида (20) с выходом H может быть линейаризована по обратной связи. Зафиксируем $\lambda^{(r_{\max})} = \text{const}$, $\lambda^{(r_{\max})} \in \{-1, 1\}$, откуда получаем из (20)

$$\bar{H} = F(x, z, \bar{\Lambda}) + G(x, z, \bar{\Lambda})u$$

$$F(x, z, \bar{\Lambda}) = \mathcal{A}_2(x, z, \bar{\Lambda})\lambda^{(r_{\max})} + \mathcal{B}(x, z, \bar{\Lambda}) \quad (24)$$

$$G(x, z, \bar{\Lambda}) = \mathcal{A}_1(x, z, \bar{\Lambda}).$$

Можно записать нелинейное преобразование координат $v \mapsto u$, переводящее нелинейную систему (24) в линейную $H^{(r)} = v$

$$u = [\mathcal{A}_1(x, z, \bar{\Lambda})]^{-1} (v - \mathcal{A}_2(x, z, \bar{\Lambda})\lambda^{(r_{\max})} - \mathcal{B}(x, z, \bar{\Lambda})). \quad (25)$$

Стратегия переключений состоит в том, что в областях $\bar{\mathcal{E}}$, где $\det(\lambda \mathcal{A}(x) + (1-\lambda)E) \rightarrow 0$ и линейаризация по обратной связи невозможна, необходимо осуществлять управление по методу продолжения по параметру согласно (23).

Следует заметить, что при $v = 0$ уравнение (25) является специальным случаем стратегии продолжения по параметру вдали от предельных точек. Если $\hat{\lambda} \neq 0$, то уравнение (25), полученное как решение (23) со следующим масштабированием переменных:

$$\alpha = \frac{\text{sign}(\hat{\lambda}^{(r_{\max})}) - \bar{\lambda}^{(r_{\max})}}{\hat{\lambda}^{(r_{\max})}}, \quad (26)$$

$$\begin{pmatrix} \hat{u} \\ \hat{\lambda}^{(r_{\max})} \end{pmatrix} = \tau, \begin{pmatrix} \bar{u} \\ \bar{\lambda}^{(r_{\max})} \end{pmatrix} = \bar{\tau}$$

удовлетворяет уравнению (23) при $\alpha = 1$, $r_{\max} = 1$ ($\dot{\lambda} = \text{const}$), с учетом последующего масштабирования $u := u/|\hat{\lambda}|$, $\dot{\lambda} := \dot{\lambda}/|\hat{\lambda}| = \text{sign}(\dot{\lambda})$.

Поскольку с вырождением матрицы \mathcal{A}_1 при применении линейаризации по обратной связи всегда наблюдается разрывность по крайней мере одного управляющего воздействия в $u(t)$, то практическим способом переключения между стратегиями управления является детектирование насыщения по входам $\max |u_i| > u_{\max}$. Входные сигналы всегда ограничены $|u| \leq u_{\max}$ в реальных приложениях. Это приводит к следующему гибриднему алгоритму линейаризации по обратной связи: мы начинаем с положительного знака $s = 1$ производной параметра $\lambda^{(r_{\max})} = s$ и линейаризации по обратной связи (25). В случае насыщения входа осуществляется переключение к процедуре продолжения по параметру. После возвращения всех входных сигналов в рабочие границы знак $\lambda^{(r_{\max})}$ изменяется на противоположный $s := -s$ и снова применяется линейаризация по обратной связи. Формальная процедура для расчета управляющих воздействий может быть представлена следующим образом:

1. $u := [\mathcal{A}_1(x, z, \bar{\Lambda})]^{-1} (v - \mathcal{A}_2(x, z, \bar{\Lambda})s - \mathcal{B}(x, z, \bar{\Lambda}))$

2. Если $\max |u_i| > u_{\max}$, то вычислить

$$\begin{pmatrix} \hat{u} \\ \hat{\lambda}^{(r_{\max})} \end{pmatrix} = \tau, \mathcal{A} \cdot \tau = 0, \|\tau\|_2 = 1, \det \begin{pmatrix} \mathcal{A} \\ \tau^T \end{pmatrix} > 0$$

$$k_\lambda := (\text{sign}(\hat{\lambda}^{(r_{\max})}) - \bar{\lambda}^{(r_{\max})}) / \hat{\lambda}^{(r_{\max})}$$

$$k_u := (\text{sign}(\max |u_i|)u_{\max} - \max |u_i|) / \max |u_i|$$

- 2.1. Если $k_u > k_\lambda$, то $\hat{u} := \hat{u} \cdot k_\lambda$, $s := \text{sign}(\hat{\lambda}^{(r_{\max})})$, $\hat{\lambda}^{(r_{\max})} := s$,

- 2.2. Иначе, если $k_u > 1$, то $\hat{u} := \hat{u} \cdot k_u$, $\hat{\lambda}^{(r_{\max})} := \hat{\lambda}^{(r_{\max})} \cdot k_u$,

- 2.3 $u = \hat{u} + \bar{u}$, $\lambda^{(r_{\max})} = \hat{\lambda}^{(r_{\max})} + \bar{\lambda}^{(r_{\max})}$.

Поскольку совокупность областей $\bar{\mathcal{E}}$ образует компактное пространство, то использование описанного алгоритма в целом дает линейаризованную систему. Ситуация, когда $x \in \bar{\mathcal{E}}$ может рассматриваться как действие возмущения на выход H .

3. Некоторые приложения

Абстрактная система с двумя входами и выходами. Рассмотрим абстрактный пример

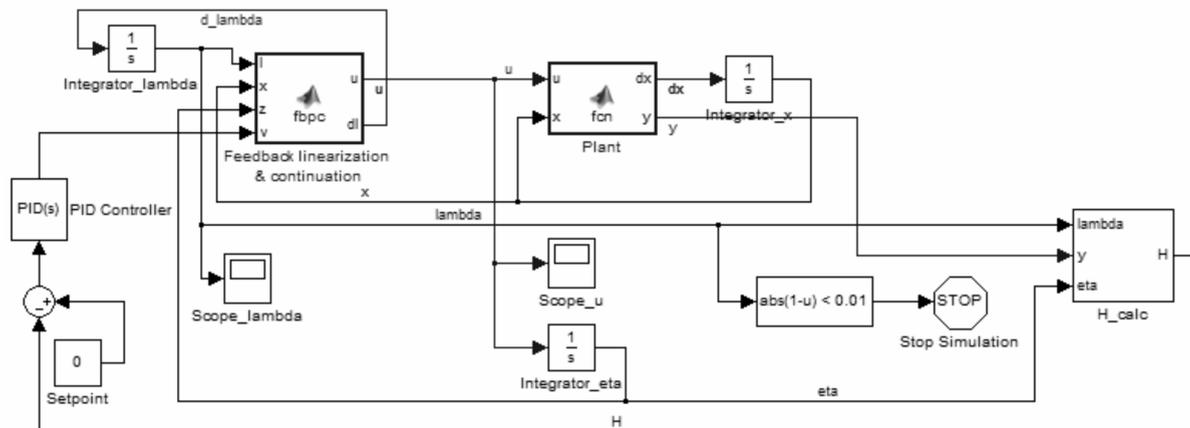


Рис. 1. Модели объекта управления и регулятора в Simulink

МИМО системы, меняющей свою относительную степень в пространстве состояний

$$\begin{aligned} \dot{x}_1 &= u_1 + x_2^3 \\ \dot{x}_2 &= u_2 + x_1^3 \\ y_1 &= x_1^3 - x_1 + 1 \\ y_2 &= x_2^4 \cos(2x_2) \end{aligned} \quad (27)$$

с начальным состоянием $x(0) = (1, 1)^T$. Необходимо решить задачу обнуления выхода $y \rightarrow 0$.

Дифференцируя выход, получаем

$$\begin{aligned} \dot{y}_1 &= (3x_1^2 - 1)(u_1 + x_2^3) = g_{11}u_1 + f_1; \\ \dot{y}_2 &= (4x_2^3 \cos(2x_2) - 2x_2^4 \sin(2x_2)) \times \\ &\times (u_2 + x_1^3) = g_{22}u_2 + f_2. \end{aligned} \quad (28)$$

Очевидно, что система в области $x \in [0, 1]$ не может быть полностью линеаризована по обратной связи, поскольку существуют такие x^* , где $g_{11}(x^*) = 0$ или $g_{22}(x^*) = 0$.

Сопоставим с (27) линейную систему вида

$$\dot{\eta}_1 = u_1, \dot{\eta}_2 = u_2 \quad (29)$$

с начальными условиями $\eta(0) = (0, 0)^T$.

По соотношению (20) получаем для уравнения $\dot{H} = 0$

$$\begin{aligned} \mathcal{A}_1 &= \begin{pmatrix} g_{11} & 0 \\ 0 & g_{22} \end{pmatrix} \cdot u + (1 - \lambda)E, \\ \mathcal{A}_2 &= y - \eta, \mathcal{B} = - \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}. \end{aligned} \quad (30)$$

Модель в Simulink для управления системой изображена на рис. 1. Результат моделирования представлен на рис. 2.

Управление трехфазным асинхронным электродвигателем. Асинхронный электродвигатель является классическим примером нелинейной системы, которая не может быть линеаризована по обратной связи [2]. Модель двигателя

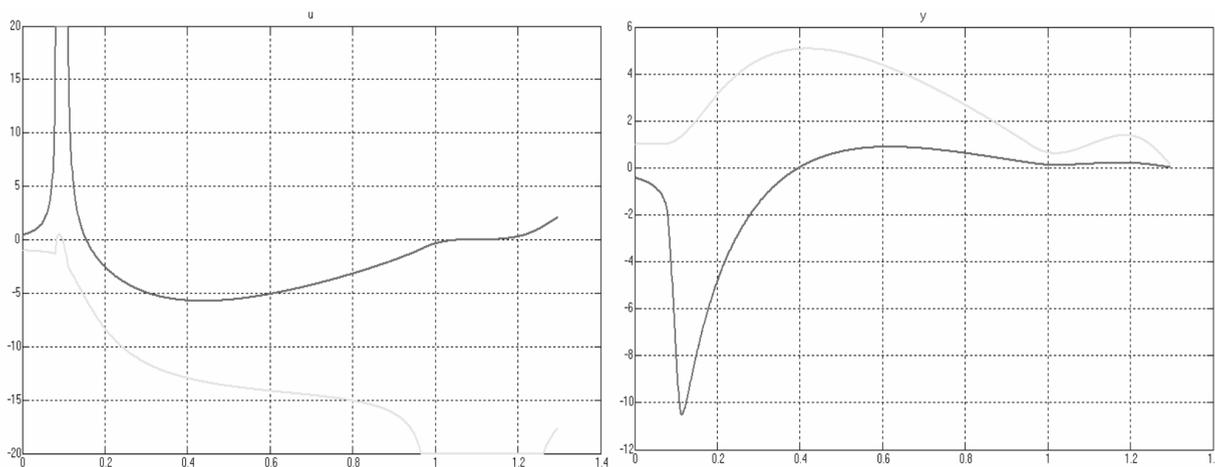


Рис. 2. Управляющие воздействия и реакция выхода объекта управления

в пространстве состояний задается следующей системой уравнений:

$$\begin{aligned} \dot{\omega} &= p^2 \frac{M_{sr} \Phi_r i_{sq}}{J L_r} - \frac{p T_m}{J} \\ \dot{\Phi}_r &= -\tau_r^{-1} \Phi_r + \tau_r^{-1} M_{sr} i_{sd} \\ \dot{i}_{sd} &= \beta \tau_r^{-1} \Phi_r - \tau_1^{-1} i_{sd} + \omega_s i_{sq} + \frac{u_{sd}}{L_1} \\ \dot{i}_{sq} &= -\beta \omega \Phi_r - \tau_1^{-1} i_{sq} - \omega_s i_{sd} + \frac{u_{sq}}{L_1}, \end{aligned} \quad (31)$$

в которой используется следующая параметризация

$$\begin{aligned} \tau_r &= \frac{L_r}{R_r}, \mu = p^2 \frac{M_{sr}}{J L_r}, \beta = \frac{M_{sr}}{L_r L_1} \\ L_1 &= L_s - \frac{M_{sr}^2}{L_r}, \\ R_1 &= R_s + R_r \left(\frac{M_{sr}}{L_r} \right)^2, \tau_1 = \frac{L_1}{R_1}, \end{aligned} \quad (32)$$

где R_s, R_r – соответственно сопротивления статора и ротора; L_s, L_r – индуктивности статора и ротора; M_{sr} – взаимная индуктивность. Пространство состояний формируют токи статора (i_{sd}, i_{sq}), магнитный поток ротора Φ_r и синхронная скорость вращения ротора ω . Входными переменными являются напряжения статора u_{sd} и u_{sq} .

Электромагнитный момент, развиваемый на валу, можно записать как произведение магнитного потока ротора на ток статора:

$$T_{em} = p \frac{M_{sr}}{L_r} (i_{sq} \Phi_{rd} - i_{sd} \Phi_{rq}), \quad (33)$$

где p – число пар полюсов статора.

Синхронную скорость ω_s , учитывающую скорость скольжения, можно записать следующим образом:

$$\omega_s = \omega + \frac{M_{sr} i_{sq}}{\tau_r \Phi_r}. \quad (34)$$

Цель управления – установка механической скорости $y_1 = \omega / p$ и квадрата магнитного потока ротора $y_2 = \Phi_r^2$.

Первое дифференцирование выходов дает

$$\begin{aligned} \dot{y}_1 &= \dot{\omega} / p = \mu \Phi_r i_{sq} / p - T_m / J; \\ \dot{y}_2 &= -2\tau_r^{-1} \Phi_r^2 + 2M_{sr} \tau_r^{-1} \Phi_r i_{sd}. \end{aligned} \quad (35)$$

После повторного дифференцирования входы u_1 и u_2 появляются в явном виде:

$$\begin{aligned} \begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \end{pmatrix} &= \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} + \begin{pmatrix} 0 & a_{12} \\ a_{21} & 0 \end{pmatrix} \\ b_1 &= -\mu \Phi_r (i_{sq} \tau_1^{-1} + i_{sd} \tau_r^{-1} + \omega_s i_{sd}) + \\ &+ \mu \tau_r^{-1} M_{sr} i_{sd} i_{sq} - \mu \beta \omega \Phi_r^2 \\ b_2 &= \frac{2}{\tau_r^2} (2 + \beta M_{sr}) \Phi_r^2 - \left(3 \frac{M_{sr}}{\tau_r} + \frac{M_{sr}}{\tau_1} \right) \frac{2i_{sd} \Phi_r}{\tau_r} + \end{aligned} \quad (36)$$

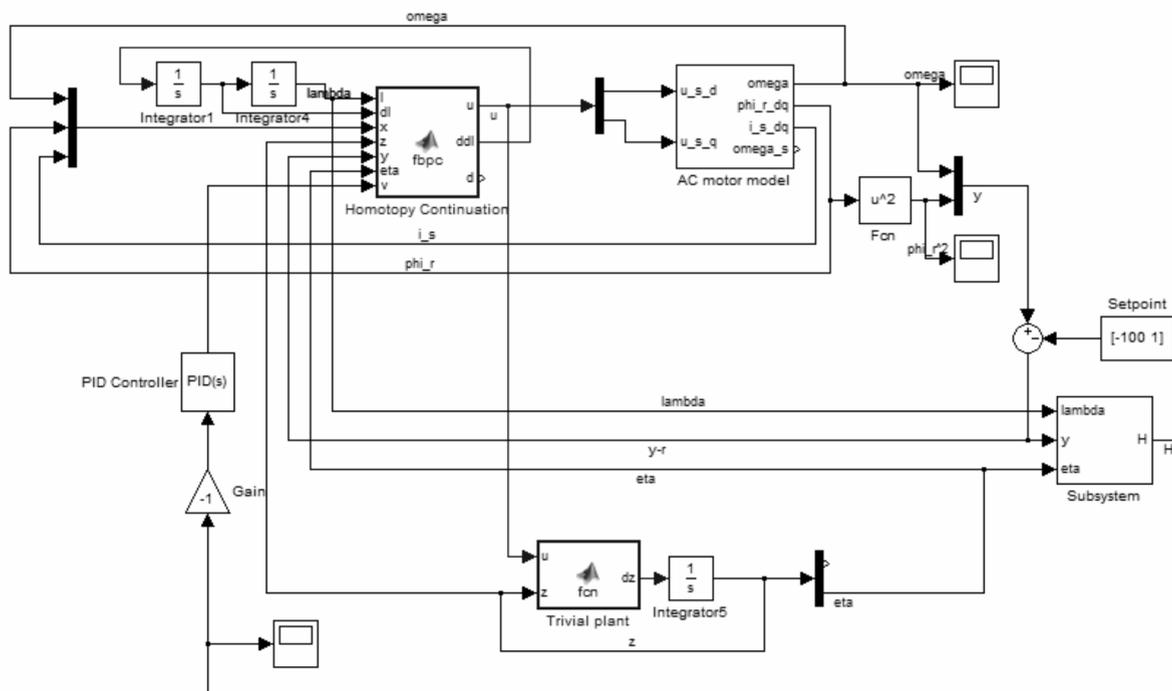


Рис. 3. Модели объекта управления и регулятора в Simulink

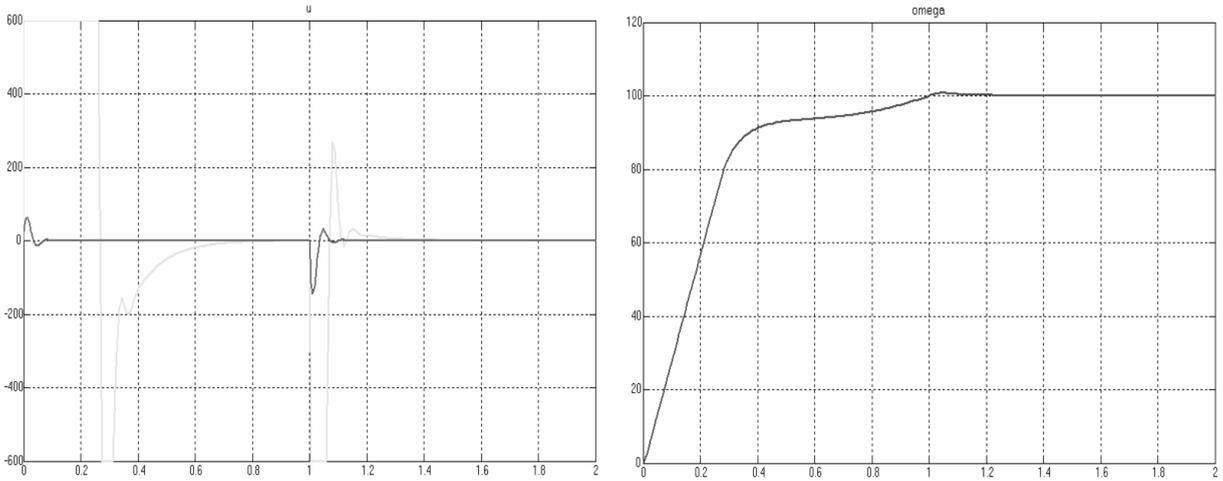


Рис. 4. Управляющие воздействия и механическая скорость на валу двигателя

$$\begin{aligned}
 &+2M_{sr}\omega_s \frac{i_{sq}\Phi_r}{\tau_r} + 2M_{sr}^2 \frac{i_{sd}^2}{\tau_r^2} \\
 &a_{12} = \mu L_1 \Phi_r \\
 &a_{21} = \frac{2M_{sr}\Phi_r}{\tau_r L_1}.
 \end{aligned}$$

Сопоставим с (31) линейную систему вида

$$\dot{z}_1 = u_1, \dot{z}_2 = u_2, \dot{z}_3 = z_2, \dot{z}_4 = z_1, \eta_1 = z_3, \eta_2 = z_4 \quad (37)$$

с начальным состоянием $z = (0, 0, 0, 0)^T$.

Используя (20) и (21), можно записать следующие уравнения для смешанной динамики (31) и (37):

$$\begin{aligned}
 \ddot{H} &= \mathcal{A}_1(x, z, \lambda)u + \\
 &+ \mathcal{A}_2(x, z, \lambda)\ddot{\lambda} + \mathcal{B}(x, z, \lambda, \dot{\lambda}) = 0 \\
 \mathcal{A}_1 &= \begin{pmatrix} 0 & \lambda a_{12} + 1 - \lambda \\ \lambda a_{21} + 1 - \lambda & 0 \end{pmatrix} \\
 \mathcal{A}_2 &= \lambda \begin{pmatrix} y_1 - \eta_1 \\ y_2 - \eta_2 \end{pmatrix} \\
 \mathcal{B} &= -\lambda \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} - 2\dot{\lambda} \begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \end{pmatrix} + 2\dot{\lambda} \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \end{pmatrix}.
 \end{aligned} \quad (38)$$

Уравнение (38) задает алгебраическое условие непрерывной деформации системы (37) в (31). Оно используется в областях пространства состояний, где линейризация по обратной связи невозможна, т. е. где $\Phi_r \neq 0$. Управляющие сигналы вычисляются как решение уравнения (23). В остальных случаях используется линейризация по обратной связи с внешним контуром управления для линейризованной системы $\dot{H} = v$:

$$u = \mathcal{A}_1^{-1}(v + \mathcal{B}_{\lambda := \text{sign}(\dot{\lambda})}). \quad (39)$$

Переключение между (38) и (39) осуществляется на основании регистрации превышения предельных значений питающего напряжения $u_{\max} = 300$, как это описано выше, в другой главе.

Численный эксперимент проведен с целью моделирования пуска асинхронного двигателя и стабилизации скорости и магнитного потока. В области $\Phi_r \approx 0$ асинхронный двигатель не может быть линейризован по обратной связи и обычно используется разомкнутое управление [2]. Предложенная параметризация позволяет использовать линейризацию по обратной связи и во время старта двигателя. Модель объекта управления и регулятора в среде Simulink показана на рис. 3. Результаты моделирования приведены на рис. 4.

В статье рассмотрено решение задачи установившегося значения на выходе аффинной по управлению нелинейной системы. Представленный подход сочетает линейризацию по обратной связи продолжением по параметру вокруг особых точек нелинейной обратной связи. В рамках разработки алгоритмов управления метод продолжения по параметру обобщен на случай нестационарных нелинейных систем алгебраических уравнений. Применение предложенного метода управления рассмотрено на примере многоканальной абстрактной системы и трехфазного асинхронного электродвигателя.

Предложенный метод управления позволяет управлять аффинными нелинейными системами с особыми точками в фазовом пространстве, ко-

торые не могут быть линеаризованы по обратной связи. В результате открываются новые возможности по применению способов проведения тра-

ектории состояния системы через сингулярные точки, разработанные в рамках методов продолжения по параметру.

СПИСОК ЛИТЕРАТУРЫ

1. **Chwa, D.** Tracking Control of Differential-Drive Wheeled Mobile Robots Using a Backstepping-Like Feedback Linearization [Text] / D. Chwa // In Proc. of IEEE Transactions on Systems, Man, and Cybernetics. –2010. –Part A. –P. 1285–1295.
2. **Fekih, A.** On nonlinear control of induction motors: comparison of two approaches [Text] / A. Fekih, F.N. Chowdhury // In proc. of American Control Conf. –2004. –Vol. 2.
3. **Wang, Jihong.** Tracking control of nonlinear pneumatic actuator systems using static state feedback linearization of the input–output map [Text] / Jihong Wang, Kotta Ulle, Ke Jia // Proc. Estonian Acad. Sci. Phys. Math. –2007. –Vol. 56. –№ 1. –P. 47–66.
4. **Мирошник, И.В.** Теория автоматического управления. Нелинейные и оптимальные системы [Текст] / И.В. Мирошник. –СПб.: Питер, 2006. –272 с.
5. **Alberto Isidori.** Nonlinear Control Systems [Text] / Isidori Alberto. –Springer, 1995. –564 p.
6. **Eugene, L.** Allgower and Kurt Georg. Introduction to Numerical Continuation Methods [Text] / L. Eugene. –2003. –388 p.
7. **Жулин, С.С.** Численный метод и программный комплекс для поиска экстремали в задачах оптимального управления на основе процедуры продолжения по параметру: Дис. ... канд. физ.-мат. наук [Текст] / С.С. Жулин. –М., 2009. –149 с.
8. **Аввакумов, С.Н.** Некоторые алгоритмы оптимального управления. Управление, устойчивость и обратные задачи динамики [Текст] / С.Н. Аввакумов, Ю.Н. Киселев // Сб. науч. тр. ИММ УрО РАН. –2006. –12. –№ 2. –С. 3–17.
9. **Wang, Qian.** State Probabilistic Control of Nonlinear Uncertain Systems. Article [Text] / Qian Wang, Robert F. Stengel // Probabilistic and Randomized Methods for Design under Uncertainty. –2006.
10. **Borisevich, A.** Some aspects of numerical continuation methods in control of nonlinear affine systems [Text] / A. Borisevich, M. Krupskaya // Proc. Int. Symp. Applied Natural Sciences 2011. –Trnava, 2011. –P. 111–115.
11. **Golub, G.H.** The Differentiation of Pseudo-Inverses and Nonlinear Least Squares Problems Whose Variables Separate [Text] / G.H. Golub, V. Pereyra // SIAM J. on Numerical Analysis. –Apr. 1973. –Vol. 10. –№ 2. –P. 413–432.
12. **Immo Diener.** On the global convergence of path-following methods to determine all solutions to a system of nonlinear equations [Text] / Immo Diener // J. Mathematical Programming: Series A and B. –Nov. 1, 1987. –Vol. 39. –Iss 2.

ИМИТАЦИОННАЯ МОДЕЛЬ СИСТЕМЫ ПЕРЕДАЧИ СПЕКТРАЛЬНО-ЭФФЕКТИВНЫХ СИГНАЛОВ С OFDM

В сетях беспроводного широкополосного доступа используются сигналы с ортогональным частотным мультиплексированием (сигналы с OFDM) с двоичной фазовой, многопозиционной фазовой, амплитудно-фазовой и др. видами манипуляции амплитуды и фазы колебания. При использовании таких методов манипуляции на каждой поднесущей частоте применяется прямоугольная форма вещественной огибающей сигналов. Одним из недостатков сигналов с OFDM является высокий уровень внеполосных излучений по краям занимаемой полосы частот (малая скорость спада уровня внеполосных излучений). Это оказывается серьезным препятствием на пути увеличения скорости передачи информации за счет увеличения числа каналов передачи в отведенной для передачи информации полосе частот телекоммуникационных систем.

Основной метод снижения уровня внеполосных излучений – использование на каждой поднесущей частоте оптимальных спектрально-эффективных сигналов, формы которых получены в результате решения оптимизационной задачи, при наличии ограничений на уровень внеполосных излучений. В статье рассматривается построение имитационной модели системы передачи спектрально-эффективных сигналов с OFDM по каналу связи с аддитивным белым гауссовским шумом (АБГШ).

Описание имитационной модели

Имитационная модель включает в себя источник цифровой двоичной последовательности символов, модулятор, формирователь спектрально-эффективных сигналов с OFDM, непрерывный канал передачи, демодулятор спектрально-эффективных сигналов с OFDM,

а также блоки для определения вероятности ошибки и анализа спектральных характеристик сигналов.

Применяемый метод формирования спектрально-эффективных сигналов с OFDM основан на линейном преобразовании вектора входных символов [1], что позволяет использовать обратное быстрое преобразование Фурье (ОБПФ).

Алгоритм работы имитационной модели заключается в следующем.

1. Задается количество N используемых поднесущих частот, объем выборки, равный $2*N$, отношение энергии сигнала к односторонней спектральной плотности мощности шума (отношение сигнал/шум), вектор коэффициентов в разложении огибающей сигналов в ряд Фурье.

2. Формируется псевдослучайная выборка последовательности символов указанного объема.

3. Производится преобразование информационных символов в символы канального алфавита, последовательный поток данных преобразуется в параллельный (по числу используемых поднесущих).

4. На основании заданных коэффициентов в разложении огибающей в ряд Фурье формируется матрица M линейного преобразования для формирования спектрально-эффективных огибающих. Осуществляется соответствующее линейное преобразование входных данных.

5. К полученным символам применяется обратное быстрое преобразование Фурье.

6. Сигнал до передачи в канал подается на блок контроля и анализа спектральных характеристик.

7. В канале к последовательности сигналов добавляется шумовое воздействие:

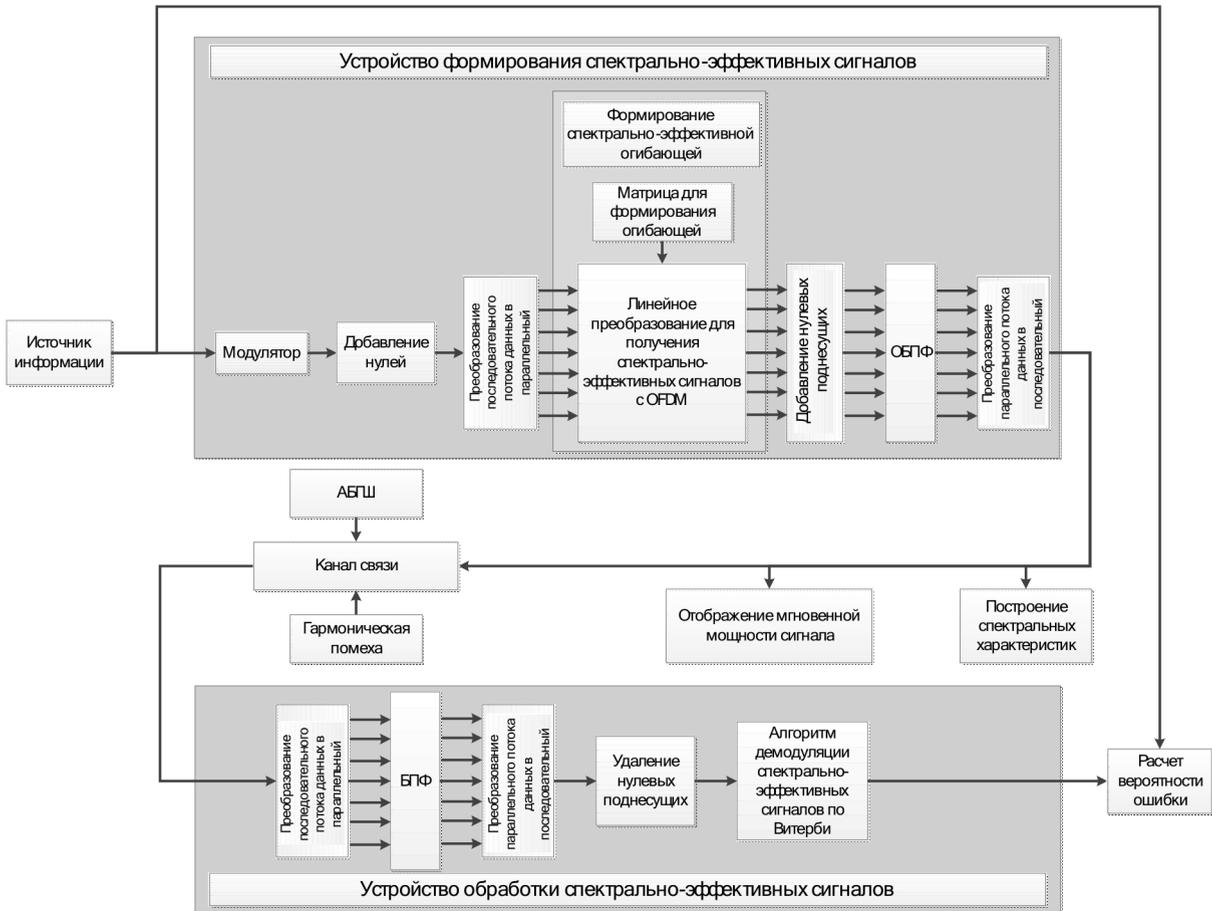


Рис. 1. Блок-схемы устройств формирования и обработки оптимальных канальных сигналов

АБГШ с учетом заданного отношения сигнал/шум;

гармоническое колебание с учетом заданного отношения сигнал/помеха.

8. В демодуляторе производится выделение используемых поднесущих и обратные операции преобразования данных, быстрое преобразование Фурье, демодуляция данных одним из двух методов: методом последовательного анализа или демодуляции по решетке.

9. Полученная последовательность символов сравнивается с исходной последовательностью и определяется средняя вероятность ошибочного приема.

Оптимальные спектрально-эффективные сигналы с OFDM получены путем решения оптимизационной задачи [1], в основе которой процедура минимизации функционала, определяющего

спектральные характеристики энергетического спектра. В частности, задача минимизации уровня внеполосных излучений сводится к минимизации функционала при наличии граничных условий и различного рода ограничений, например, на энергию и длительность сигнала, коэффициент корреляции, величину пик-фактора колебаний и является задачей вариационного исчисления.

Численное решение вариационной задачи для четной функции огибающей $a(t)$ спектрально-эффективных сигналов на каждой поднесущей частоте сводится к получению коэффициентов разложения функции $a(t)$ в ряд Фурье на интервале $[-T/2; T/2]$:

$$a(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos\left(\frac{2\pi}{T} kt\right),$$

где коэффициенты разложения имеют значения

a_0	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}
1,7823	0,626	-0,1239	0,053	-0,0294	0,0187	-0,013	0,0095	-0,0073	0,0057	-0,0046

Скорость спада уровня внеполосных излучений энергетического спектра случайной последовательности спектрально-эффективных сигналов, при условии использования фазовой манипуляции на 180 градусов, составляет величину $1/f^6$.

На рис. 1 приведена общая структурная схема имитационной модели, разделенная условно на две части: устройство формирования спектрально-эффективных сигналов и устройство обработки спектрально-эффективных сигналов.

Устройство формирования спектрально-эффективных сигналов выполняет процедуры построения канальных сигналов для передачи по каналам связи информационных последовательностей. Устройство формирования сигналов позволяет создавать на поднесущих частотах различные формы комплексной огибающей сигналов: от прямоугольной формы до спектрально-эффективной. Выбор форм происходит с помощью матрицы значений огибающей сигналов (подблок «Матрица для формирования огибающей»).

Блок «Устройство формирования спектрально-эффективных сигналов» состоит из следующих подблоков:

«Матрица для формирования огибающей»;

«Модулятор»;

«Добавление нулей»;

«Преобразование последовательного потока данных в параллельный»;

«Линейное преобразование для получения спектрально-эффективных сигналов с OFDM»;

«Добавление нулевых поднесущих»;

«ОБПФ»;

«Преобразование параллельного потока данных в последовательный».

Подблок «Модулятор»

На вход данного подблока поступает псевдослучайная последовательность единиц и нулей с вероятностью появления символов, равной 0,5. Выполняется фазовая модуляция (ФМ-4). Блок обладает следующими параметрами:

смещение фазы – $\pi/4$;

порядок – код Грея.

Подблок «Добавление нулей»

В подблоке происходит выполнение операции приведения формата сигнала к формату, требуемому для перемножения данных на числа, записанные в матрице формирования огибающей сигналов.

Подблок «Преобразование последовательного потока данных в параллельный»

В подблоке осуществляется процедура преобразования входного последовательного потока данных в параллельный поток.

Подблок «Матрица для формирования огибающей»

На основе информации о количестве поднесущих, коэффициентах разложения спектрально-эффективной огибающей сигналов и требуемом количестве спектрально-эффективных огибающих в матрице формируются числа, необходимые для формирования огибающей сигналов. При использовании 64 информационных поднесущих и 10 коэффициентов разложения спектрально-эффективной огибающей размерность матрицы для формирования огибающей составляет 84.

Подблок «Линейное преобразование для получения спектрально-эффективных сигналов с OFDM»

В подблоке выполняется линейное преобразование чисел, необходимое для получения на выходе сигнала, содержащего заданное количество спектрально-эффективных и прямоугольных огибающих. Линейное преобразование заключается в матричном перемножении входного вектора с выхода подблока добавления нулей и сформированной матрицы для формирования огибающей. Количество спектрально-эффективных огибающих и форма комплексной огибающей сигналов определяется матрицей значений огибающей сигналов. На выходе блока получаются числа, соответствующие спектрально-эффективным сигналам.

Подблок «Добавление нулевых поднесущих»

На выходе данного блока формируется сигнал с информационными и двумя блоками защитных поднесущих по бокам от информационных. Данное преобразование необходимо для того, чтобы размерность данных на входе подблока ОБПФ была кратна второй степени.

Подблок «Обратное быстрое преобразование Фурье»

Осуществляет обратное преобразование Фурье по алгоритму быстрого преобразования Фурье:

размерность преобразования Фурье – 256 точек; основание преобразования – 2.

Сформированный описанным выше образом сигнал передается в блок симуляции канала передачи.

Подблок «Преобразование параллельного потока данных в последовательный»

Осуществляется процедура преобразования входного параллельного потока данных в последовательный поток.

«Построение спектральных характеристик»

Осуществляются процедуры построения спектральных характеристик рассматриваемых сигналов.

«Отображение мгновенной мощности сигнала»

Данный блок предназначен для отображения мгновенной мощности сигнала.

«Канал передачи»

Осуществляет процедуры моделирования канала передачи в случае воздействия аддитивного белого гауссовского шума (АБГШ) и гармонической помехи.

Устройство обработки спектрально-эффективных сигналов выполняет процедуры приема и демодуляции оптимальных канальных сигналов. Устройство обработки сигналов позволяет осуществлять прием на поднесущих частотах различных форм комплексной огибающей сигналов.

Блок *«Устройство обработки спектрально-эффективных сигналов»* состоит из следующих подблоков:

«Преобразование последовательного потока данных в параллельный»;

«Быстрое преобразование Фурье»;

«Преобразование параллельного потока данных в последовательный»;

«Удаление нулевых поднесущих»;

«Алгоритм демодуляции спектрально-эффективных сигналов по Витерби».

Подблок *«Преобразование последовательного потока данных в параллельный»*

Осуществляет процедуру преобразования входного последовательного потока данных в параллельный.

Подблок «Быстрое преобразование Фурье»

Осуществляет расчет прямого преобразования Фурье над входным сигналом посредством реализации алгоритма быстрого прямого преобразования Фурье:

размерность преобразования Фурье – 256 точек;
основание преобразования – 2.

Подблок «Преобразование параллельного потока данных в последовательный»

Осуществляет процедуру преобразования входного параллельного потока данных в последовательный.

Подблок «Удаление нулевых поднесущих»

В блоке происходит удаление защитных поднесущих частот, добавленных при формировании сигнала.

Подблок «Алгоритм демодуляции спектрально-эффективных сигналов по Витерби»

В данном блоке осуществляется демодуляция спектрально-эффективных сигналов по алгоритму Витерби. При использовании сигналов с прямоугольными огибающими производится классическая демодуляция встроенным в данный блок демодулятором. Выбор формы используемой огибающей осуществляется с помощью матрицы для формирования огибающей. Встроенный демодулятор после выполнения алгоритма Витерби обладает следующими параметрами:

смещение фазы – $\pi/4$;

порядок – код Грея;

тип входных данных – бит;

тип решений – жесткие решения.

Полученный сигнал поступает в блок *«Расчета вероятности ошибки»*.

«Расчет вероятности ошибки»

Данный блок предназначен для расчета вероятностных характеристик ошибочного приема передаваемой информации.

Временные и спектральные характеристики сигналов

Во временной области сигнал с OFDM представляет собой суперпозицию большого количества отрезков гармонических колебаний различной частоты. С увеличением количества спектрально-эффективных сигналов форма огибающей суммарного сигнала приближается к форме огибающей одиночного сигнала. Пик-фактор такого сигнала может принимать относительно высокие значения.

Пик-фактор Π сигналов с OFDM представляет собой отношение наибольшей (пиковой) мощности к средней мощности сигнала $s(t)$:

$$\Pi = \frac{\max_{t \in [0; T]} \{|s(t)|^2\}}{P_{cp}}, P_{cp} = \frac{1}{T} \cdot \int_0^T |s(t)|^2 dt.$$

Большое внимание к изучению возможностей снижения пик-фактора колебаний связано в первую очередь с тем, что именно этот параметр сигналов с OFDM существенно ограничивает область их применения, особенно в портативных приемно-передающих устройствах с малым потреблением

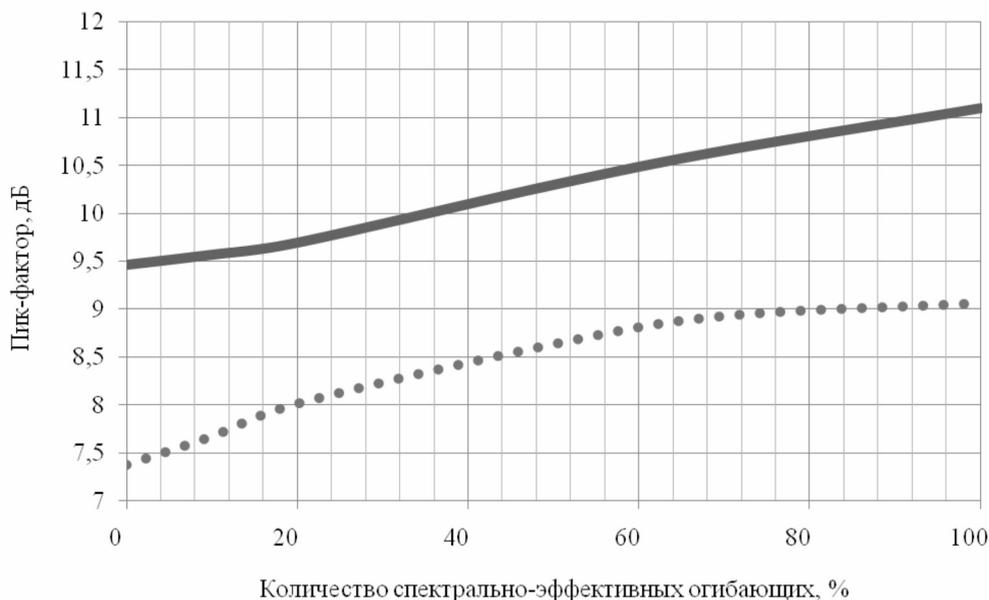


Рис. 2. Пик-фактора сигналов с OFDM при различном количестве спектрально-эффективных сигналов (—) без переноса; (.....) с переносом

мощности. Высокое значение пик-фактора колебаний приводит к амплитудному ограничению сигналов с OFDM в выходных цепях передатчика и появлению в сигнале межканальных помех и, как следствие, к снижению помехоустойчивости приема информации и увеличению значений уровня внеполосных излучений.

Результаты исследования пик-фактора сигналов с OFDM при различном количестве спектрально-эффективных огибающих показаны на рис. 2.

Увеличение количества спектрально-эффективных огибающих от 0 % до 100 % приводит к росту пик-фактора более чем на 2,5 дБ как при рассмотрении случая переноса на несущую частоту, так и в случае отсутствия переноса на несущую частоту. Данный эффект объясняется

тем фактом, что при переходе к спектрально-эффективным сигналам форма результирующей огибающей суммарного сигнала приближается к огибающей одиночного сигнала. Пик-фактор таких сигналов превышает пик-фактор классических сигналов с OFDM с прямоугольными огибающими.

Спектральные характеристики (значение занимаемой полосы частот в зависимости от значения $G(f)/G(0)$) оптимальных сигналов с минимальным уровнем внеполосных излучений с различным количеством спектрально-эффективных сигналов на поднесущих частотах при условии фиксированной средней мощности представлены в табл. 1.

Полученные результаты экспериментальных исследований модели справедливы для случая

Таблица 1

Значение занимаемой полосы частот в зависимости от значения $G(f)/G(0)$ для случая фиксированной средней мощности огибающих

$G(f)/G(0)$, дБ	Доля спектрально-эффективных сигналов, %			
	0	20	50	100
-20	0,630	0,506	0,506	0,506
-30	1,910	0,690	0,540	0,526
-40	—	—	—	0,584
-50	—	—	—	0,680

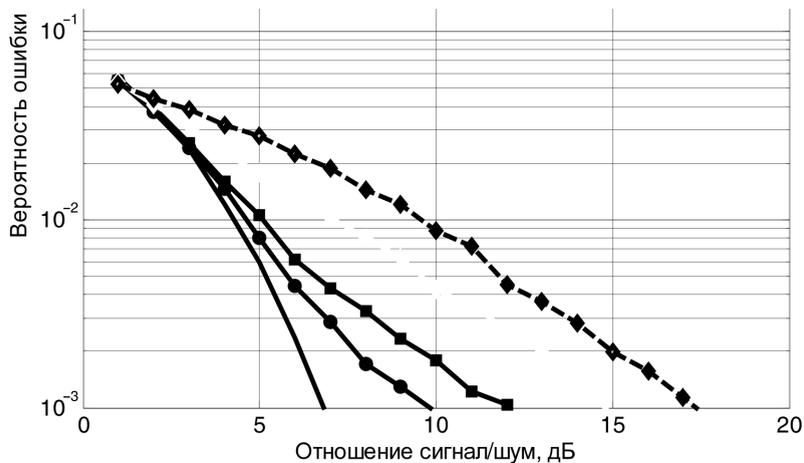


Рис. 3. Помехоустойчивость приема оптимальных сигналов с OFDM (ФМ-4) для случая фиксированной средней мощности при применении классического алгоритма приема (—) 0,0; (—●—) 0,1; (—■—) 0,2; (—◆—) 0,5; (---◆---) 1,0

фиксированной средней мощности передаваемых сигналов и при большой длине последовательности символов.

Помехоустойчивость приема оптимальных сигналов

В качестве помехи использовался аддитивный нормальный случайный процесс со спектральной плотностью средней мощности, равной N_0 . Результаты экспериментальных исследований приведены на рис. 3.

В качестве алгоритма демодуляции сигналов использовался алгоритм когерентного оптимального поэлементного приема для сигналов с прямоугольной огибающей. Такой алгоритм применяется в существующих системах с OFDM – WiMAX. Увеличение доли оптимальных сигналов с минимальным уровнем внеполосных излучений (0; 0,1; 0,2; 0,5; 1,0 на рис. 3) приводит к увеличению вероятности ошибки и энергетическому проигрышу по отношению сигнал/шум при фиксированной вероятности ошибки (табл. 2),

что обусловлено происходящим при этом увеличением помех из-за неортогональности между сигналами, расположенными на соседних поднесущих частотах. Худшей помехоустойчивостью обладают сигналы с OFDM со 100 % спектрально-эффективных сигналами на поднесущих частотах.

Из анализа зависимостей на рис. 3 видно, что при классическом алгоритме приема переход от сигналов с прямоугольными огибающими к сигналам со 100 % спектрально-эффективных огибающих приводит к энергетическому проигрышу в 5,3 дБ для вероятности ошибки 10^{-2} и 10,8 дБ для вероятности ошибки 10^{-3} .

Для повышения достоверности приема оптимальных сигналов с минимальным уровнем внеполосных излучений применяется алгоритм Витерби [3]. Основой этого алгоритма является использование при принятии решения информации о принятых символах, переданных на соседних поднесущих частотах. Принятие решения о переданном комплексном символе осуществляет-

Таблица 2

Значения отношения сигнал/шум (дБ) при фиксированной вероятности ошибки для различной доли оптимальных сигналов при фиксированной средней мощности и классическом алгоритме приема

Вероятность ошибки	Доля спектрально-эффективных огибающих		
	0	0,5	1,0
10^{-2}	4,3	7,1	9,6
10^{-3}	6,8	14,8	17,4

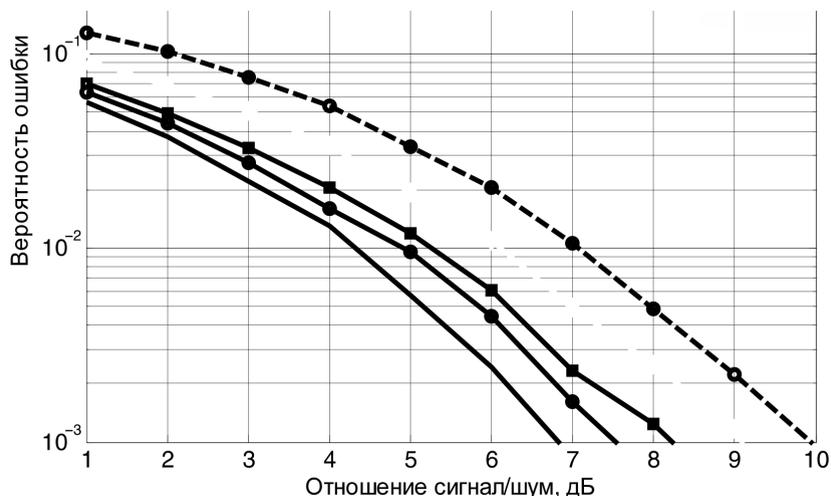


Рис. 4. Помехоустойчивость приема оптимальных сигналов с OFDM (ФМ-4) для случая фиксированной средней мощности при применении алгоритма приема по Витерби (—) 0,0; (—●—) 0,1; (—■—) 0,2; (—◊—) 0,5; (---◊---) 1,0

ся после анализа всех поднесущих в сигнале. Это приводит к тому, что в процессе декодирования в памяти декодера сохраняется набор возможных комплексных символов для каждой поднесущей, а окончательное принятие решения о символах осуществляется после учета всех связей между всеми поднесущими.

Результаты измерения вероятности ошибок для оптимальных сигналов с OFDM (вид модуляции на каждой поднесущей ФМ-4) для случая фиксированной средней мощности при применении алгоритма приема по Витерби показаны на рис. 4 и приведены в табл. 3.

Видно, что энергетический проигрыш при переходе от сигналов с прямоугольными огибающими к сигналам со 100 % спектрально-эффективных огибающих (2,8 дБ для вероятности ошибки 10^{-2} и 3,1 дБ для вероятности ошибки 10^{-3}) меньше по сравнению со случаем применения классического алгоритма приема (10,8 дБ для вероятности ошибки 10^{-3}). То есть

энергетический выигрыш за счет применения алгоритма приема по Витерби составляет более 7 дБ для вероятности ошибки 10^{-3} .

Переход к применению оптимальных сигналов с OFDM с минимальным уровнем внеполосных излучений позволяет существенно сократить занимаемую полосу частот. При этом, как видно из результатов имитационного моделирования, полоса занимаемых частот по уровню -50 дБ по отношению к спектральной маске классических сигналов с OFDM уменьшается более чем на 40 % или в 1,47 раза.

Энергетические потери при переходе от классических сигналов с OFDM к оптимальным составляют не более 2,8 дБ для вероятности ошибки 10^{-2} и 3,1 дБ для вероятности ошибки 10^{-3} .

Работа выполнена при финансовой поддержке ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007–2013 гг.», госконтракт № 07.514.12.4008.

Таблица 3

Значения отношения сигнал/шум (дБ) при фиксированной вероятности ошибки для различной доли оптимальных сигналов при фиксированной средней мощности и алгоритме приема по Витерби

Вероятность ошибки	Доля спектрально-эффективных огибающих		
	0	0,5	1,0
10^{-2}	4,3	6,1	7,1
10^{-3}	6,8	9,1	9,9

СПИСОК ЛИТЕРАТУРЫ

1. **Макаров, С.Б.** Метод формирования спектрально-эффективных OFDM-сигналов на основе неортогональных базисных функций [Текст] / С.Б. Макаров, А.В. Рашич // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2009. –№ 2 (76). –С. 94–98.

2. **Рашич, А.В.** Снижение пик-фактора сигналов с ортогональным частотным уплотнением [Текст] /

А.В. Рашич, С.Б. Макаров // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2008. –№ 2(55). – С. 79–84.

3. **Макаров, С.Б.** Формирование и прием спектрально-эффективных сигналов с OFDM [Текст] / С.Б. Макаров, А.В. Рашич // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2011. –№ 6-2 (138). –С. 19–26.

УДК 004.056

С.С. Бондарчук, А.Л. Додохов, А.Г. Сабанов

ТЕХНОЛОГИЯ ЗАЩИТЫ ПЕРСОНАЛЬНЫХ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ СУБД ORACLE

В связи с вступлением в силу Федерального закона от 25 июля 2011 г. № 261-ФЗ «О внесении изменений в Федеральный закон № 152-ФЗ О персональных данных» вопросу разработки практических методов защиты персональных данных (ПДн) уделяется повышенное внимание.

Как известно, самые большие базы данных, содержащие персональные данные, используют промышленные масштабируемые СУБД. Одной из самых распространенных в России является СУБД Oracle. Как показано в работе [1], эта СУБД имеет в настоящее время самые развитые встроенные сервисы безопасности по сравнению с другими СУБД, однако встроенные средства шифрования не удовлетворяют требованиям российского законодательства, и для защиты ПДн необходима разработка наложенных средств в соответствии с ГОСТ – 28147-89, глубоко интегрированных с безопасными сервисами, заложенными вендором. Как известно, разработка таких решений является сложной задачей, поскольку к ним одновременно предъявляются повышенные и противоречащие друг другу требования как по производительности, так и по безопасности – для задачи шифрования бах данных давно известна зависимость: чем лучше зашифрованы данные, тем больше падает производительность вычислений с данными.

В данной статье рассматривается метод по-

строения решения по защите персональных данных на платформе Oracle в соответствии с требованиями российского законодательства.

Постановка задачи. Одно из самых уязвимых мест при передаче, обработке и хранении персональных данных – это непосредственно базы данных (БД), где содержатся ПДн. Для БД известны следующие основные функции защиты информации:

- защита доступа – доступ к данным пользователь получает только при успешном прохождении им процедур идентификации и аутентификации;
- разграничение доступа – каждый пользователь, включая администратора, имеет доступ только к необходимой ему согласно занимаемой должности информации;
- шифрование данных – шифровать необходимо как передаваемые в сети данные для защиты от перехвата, так и данные, записываемые на носитель, для защиты от кражи носителя и несанкционированного просмотра/изменения нештатными средствами системы управления БД (СУБД);
- аудит доступа к данным – действия с критичными данными должны протоколироваться. Доступ к журналу не должны иметь пользователи, на которых он ведется.

Задачу данной статьи можно сформулировать следующим образом: необходима разработка

средства криптографической защиты конфиденциальной информации (не составляющей государственную тайну), при ее хранении в логической структуре реляционных таблиц на серверах баз данных. В качестве операционной среды для работы серверных компонентов рассматривается окружение сервера баз данных Oracle версий 9i, 10g, 11g. В качестве операционной среды для работы клиентских компонентов, как правило, используется операционная система Windows XP и выше.

Главное требование к разрабатываемому решению – возможность его использования в существующих прикладных задачах без их существенной переделки.

Кроме этого, основными бизнес-задачами обеспечения безопасности при построении решения являются:

- разделение доступа и персонификация действий пользователей (это требование вытекает из требований закона [2] и рекомендаций [3]);
- защита от администраторов СУБД и сетевого администратора, объясняемая тем, что, как правило, все крупные утечки из баз данных производятся злоумышленниками, получившими права администратора;
- прозрачность решения для существующих и разрабатываемых приложений;
- возможность предоставления защищенного удаленного доступа.

Основные рассматриваемые роли:

- администратор безопасности;
- администратор;
- привилегированный пользователь;
- пользователь;
- гость.

Архитектура предлагаемого решения.

В основе предлагаемого решения лежит классическая клиент-серверная архитектура. Основные криптографические преобразования производятся на стороне сервера.

В процессе разработки метода предложенное в итоге решение получило название «средство криптографической защиты информации (СКЗИ) «Крипто БД»». Схема реализации решения представлена на рис. 1.

СКЗИ «Крипто БД» является наложенным средством защиты, реализованным с использованием только документированных возможностей, предоставляемых СУБД Oracle. Криптографическое ядро СКЗИ реализовано на языке PL/SQL и Java, также штатно поддерживаемым сервером БД.

Основные компоненты СКЗИ «Крипто БД»:

- библиотеки, реализующие алгоритмы шифрования;

- библиотеки работы с ключами шифрования;
- сервис вычисления ключей шифрования;

- библиотеки работы с репозиторием;

- библиотеки и утилиты для управления ключами шифрования;

- библиотеки и утилиты для зашифрования и расшифрования.

Архитектура СКЗИ «Крипто БД» реализует хранение ключевой информации таким образом, что среда функционирования криптосредства (СФК) не имеет к этой информации непосредственного доступа. Ни один из компонентов СКЗИ «Крипто БД» не передает другому компоненту ни ключи шифрования в явном виде, ни адреса этих объектов. Вместо этого используют

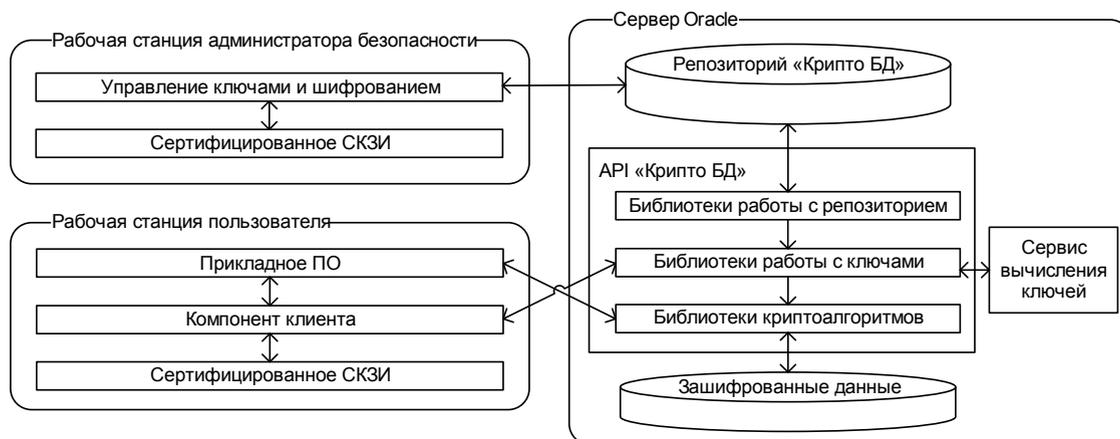


Рис. 1. Схема реализации СКЗИ «Крипто БД»

ся идентификаторы соответствующих ключей, не содержащие их адреса.

Применение СКЗИ «Крипто БД». СКЗИ «Крипто БД» предназначено для использования в существующем и во вновь разрабатываемом прикладном программном обеспечении.

Взаимодействие приложения и серверного компонента «Крипто БД». Общая схема работы приложения, использующего зашифрованные с помощью СКЗИ «Крипто БД» данные, приведена на рис. 2.

После успешной аутентификации и авторизации пользователя приложения на сервере БД создается процесс в памяти (сеанс пользователя БД), который обеспечивает взаимодействие приложение клиента – сервер БД. В этот момент компонент клиента «Крипто БД» (рис. 1) выполняет следующие действия (нумерация в соответствии с рис. 2).

1. Получение зашифрованных ключей шифрования данных (КШД):

определяется факт подключения смарт-карт или USB-ключей (далее по тексту – смарт-карты) к АРМ пользователя;

читается список сертификатов на подключенных смарт-картах;

извлекаются зашифрованные КШД из репозитория «Крипто БД» (см. рис. 1) с использованием в качестве идентификатора пользователя его сертификата;

каждый КШД передается вместе с открытым ключом сервера БД и случайным числом (вырабатывается сервисом вычисления ключей (см. рис. 1), которые также зашифрованы открытым ключом соответствующего сертификата пользователя.

2. Расшифрование КШД:

полученные зашифрованные значения КШД, открытый ключ сервера и случайное число расшифровываются компонентом клиента с использованием стандартного интерфейса, реализованного соответствующим поставщиком криптографии, установленном на АРМ пользователя (см. рис. 1). На данном этапе производится дополнительная авторизация пользователя (ввод PIN-кода смарт-карты) для получения доступа к закрытому ключу.

3. Зашифрование КШД на открытом ключе сервера:

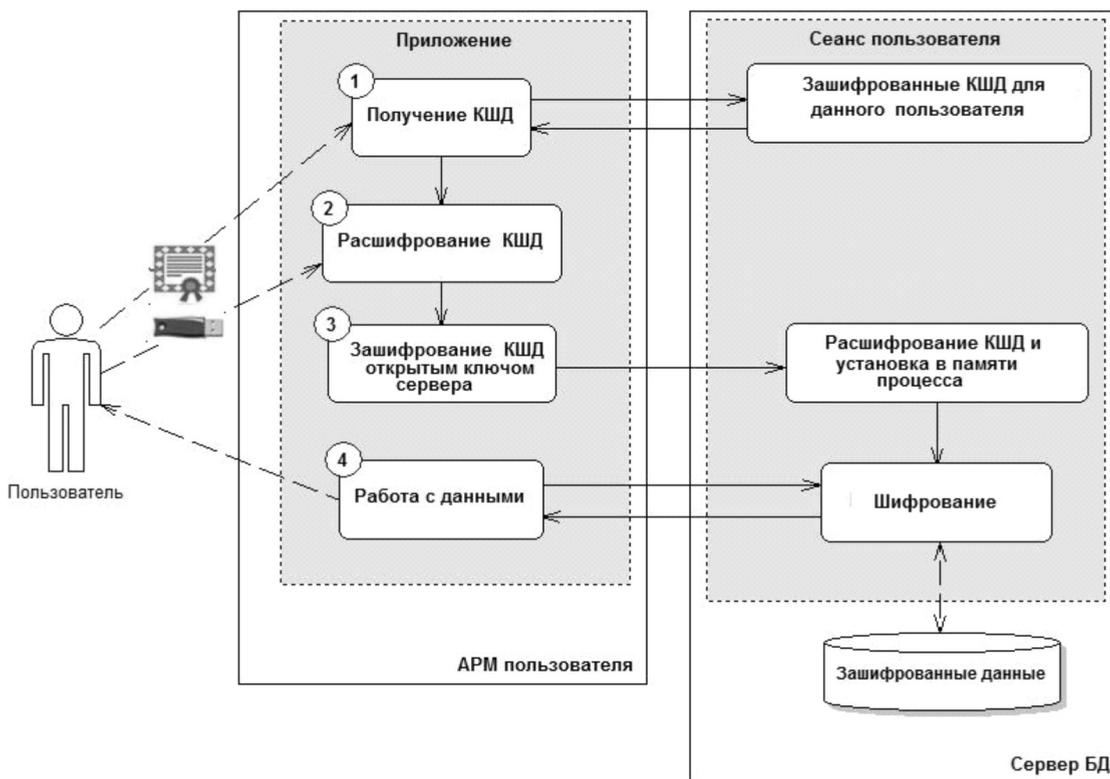


Рис. 2. Схема работы приложения с зашифрованными данными средствами СКЗИ «Крипто БД»

непосредственно после расшифрования КШД, открытого ключа сервера, случайного числа, КШД зашифровывается на открытом ключе сервера БД. Полученное значение и расшифрованное случайное число передается сервису вычисления ключей (см. рис. 1);

сервис вычисления ключей идентифицирует сеанс обмена ключами, используя полученное значение случайного числа. Далее, расшифровывается закрытым ключом сервиса КШД и кэшируется в сеансе пользователя.

4. Работа с зашифрованными данными:

чтение и запись зашифрованных данных производится через слой промежуточных представлений, реализующих вызовы API «Крипто БД». Промежуточные представления автоматически создаются «Крипто БД» в момент первичного зашифрования таблиц.

Основные характеристики СКЗИ «Крипто БД». Размеры ключей.

Размеры ключей для защиты ключей шифрования:

закрытый ключ – 256 бит;

открытый ключ – 512 бит (ГОСТ Р 34.10-2001).

Размеры ключей шифрования – 256 бит.

Реализованные алгоритмы криптографического преобразования. Реализован алгоритм криптографического преобразования, соответствующий ГОСТ 28147-89, в режимах:

режим простой замены (ECB);

режим гаммирования (OFM);

режим гаммирования с обратной связью (CFB);

режим сцепления блоков (CBC);

режим простой замены с диверсификацией ключа шифрования (ECB-UKM);

режим гаммирования с диверсификацией ключа шифрования (OFB-UKM);

режим гаммирования с обратной связью с диверсификацией ключа шифрования (CFB-UKM);

режим сцепления блоков с диверсификацией ключа шифрования (CBC-UKM);

режим простой замены с выработкой имитовставки (ECB-MAC);

режим гаммирования с выработкой имитовставки (Counter mode-MAC);

режим гаммирования с обратной связью и выработкой имитовставки (CFB-MAC);

режим сцепления блоков с выработкой имитовставки (CBC-MAC);

режим простой замены с диверсификацией

ключа шифрования и выработкой имитовставки (ECB-MAC-UKM);

режим гаммирования с диверсификацией ключа шифрования и выработкой имитовставки (OFB-MAC-UKM);

режим гаммирования с обратной связью, с диверсификацией ключа шифрования и выработкой имитовставки (CFB-MAC-UKM);

режим сцепления блоков с диверсификацией ключа шифрования и выработкой имитовставки (CBC-MAC-UKM).

Совместимость предлагаемого решения с программным обеспечением других разработчиков. СКЗИ «Крипто БД» совместимо с ПО, в котором реализованы российские криптографические алгоритмы, поддержка сертификатов открытых ключей X.509 и защищенный обмен сообщениями (CMS) в соответствии с документами RFC 4357, 4490 и 4491.

Архитектура предлагаемого решения, применение в нем только промышленных решений, открытых стандартов и рекомендаций, а также выполненные пилотные проекты в ряде государственных структур позволяют предполагать, что в более 80 % случаев интеграция данного решения с ПО сторонних разработчиков будет проходить в «прозрачном» режиме, т. е. не потребует доработок программного обеспечения при внедрении в существующие информационные системы.

Встроенные в СУБД Oracle алгоритмы криптографии, применяемой для защиты данных, ключей шифрования, контроля целостности не соответствуют требованиям законодательства РФ. Внедрение внешних криптоалгоритмов разработчиками Oracle не предусмотрено. Для применения СУБД Oracle для защиты персональных данных разработан метод наложенных средств шифрования, использующий алгоритмы ГОСТ 28147-89. Данный метод может применяться для защиты конфиденциальных данных, в частности, персональных данных. Предложенное решение в виде СКЗИ «Крипто БД» успешно прошло сертификацию по требованиям ФСБ России к СКЗИ класса КС1 (исполнение 1) и КС2 (исполнение 2). Согласно полученному сертификату соответствия СФ/124-1569 данное СКЗИ может использоваться для криптографической защиты информации, не составляющей государственную тайну, хранящейся в таблицах баз данных под управлением СУБД Oracle.

СПИСОК ЛИТЕРАТУРЫ

1. Додохов А.Л. Исследование применения СУБД Oracle для защиты персональных данных [Текст] / А.Л. Додохов, А.Г. Сабанов // Доклады Томского гос. ун-та систем управления и радиоэлектроники. –2011. –№ 2 (24). –С. 267–270.

2. Федеральный закон от 27 июля 2006 г. № 152-ФЗ «О персональных данных» (в ред. от 25 июля 2011 г.).

3. Нормативно-методический документ ФСБ России «Методические рекомендации по обеспечению с помощью криптосредств безопасности персональных данных при их обработке в информационных системах персональных данных с использованием средств автоматизации» от 21 февраля 2008 г.

УДК 519.725.2

П.К. Семенов

ПРАКТИЧЕСКИЕ АСПЕКТЫ АЛГОРИТМА ТАЛА–ВАРДИ СПИСОЧНОГО ДЕКОДИРОВАНИЯ ПОЛЯРНЫХ КОДОВ

В работе [1] предложены конструкция полярных кодов на основе ядра Арикана, асимптотически достигающая пропускной способности двоичного симметричного канала без памяти, и алгоритмы кодирования и декодирования. Описанный в [2] алгоритм Тала–Варди списочного декодирования для полярных кодов существенно уменьшает вероятность ошибки декодирования и, как было экспериментально установлено, приближает ее к границе декодирования по максимуму правдоподобия.

Алгоритм Тала–Варди – многостадийный, является расширением стандартного декодера полярных кодов, построенных на основе ядра Арикана. Хотя последний использует вычисления в логарифмической области, все расчеты в декодере Тала–Варди необходимо проводить напрямую с вероятностями, что приводит к быстрому убыванию вычисляемых характеристик к нулю и нехватке точности стандартных чисел с плавающей запятой для динамического диапазона. В связи с этим в [2] предложено нормирование оценок вероятностей на каждой стадии, хотя соответствующие случайные события не составляют полную группу. В результате это может привести к следующим затруднениям:

1) появляется зависимость по данным при обработке разных элементов списка в алгоритме Тала–Варди, хотя теоретически они могут быть обработаны независимо, параллельно;

2) становится невозможным получение хотя

бы истинного порядка значения оцениваемой вероятности, что может затруднить использование декодера Тала–Варди в каскадных конструкциях.

Обе эти проблемы разрешаются переходом в логарифмическую область, однако хорошо известный для полярных кодов способ вычислений с большой точностью, основанный на логарифмических отношениях правдоподобия (ЛОПП), не применим в силу специфики алгоритма Тала–Варди. Решение этой задачи представляет существенный интерес в силу востребованности эффективной аппаратной реализации (см. [3, 4]).

Данная статья посвящена переводу вычислений в алгоритме Тала–Варди в логарифмическую область. Также моделирование вычислений с фиксированной точностью показало, что за счет увеличения длины списка можно полностью перейти к целочисленным вычислениям, без выделения битов на квантование дробной части.

Полярные коды на основе ядра Арикана

Назовем ядром Арикана матрицу $G = \begin{pmatrix} 1 & 0 \\ & 1 \end{pmatrix}$, а через $G^{\otimes s}$ обозначим s -ю кронекеровскую степень – матрицу размера $2^s \times 2^s$. Пусть B_{2^s} – это $2^s \times 2^s$ -матрица такой перестановки для вектора длины 2^s , что номер новой позиции i -го элемента получается как обратная запись представленного в виде s -битной последовательности числа i . Например, $2 \equiv (1 \ 0 \ 0) \rightarrow (0 \ 0 \ 1) \equiv 1$, $3 \equiv (0 \ 1 \ 1) \rightarrow (1 \ 1 \ 0) \equiv 6$. Обозначим ска-

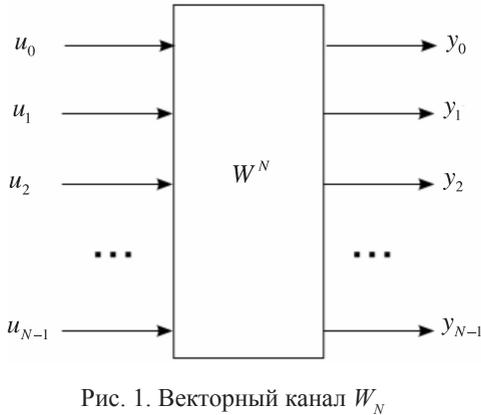


Рис. 1. Векторный канал W_N

лярный канал со входом u и выходом y как $W(y|u)$, отождествляя его с функцией плотности условной вероятности выходного символа. В случае передачи последовательности данных $u_0^{N-1} = (u_0 \ u_1 \ \dots \ u_{N-1})$ размера N через канал $W(y|u)$ будем рассматривать эквивалентный векторный канал $W^N(y_0^{N-1} | u_0^{N-1}) = \prod_{i=0}^{N-1} W(y_i | u_i)$ с выходом $y_0^{N-1} = (y_0 \ y_1 \ \dots \ y_{N-1})$ (рис. 1). В статье [1] предложен метод поляризации двоичного симметричного канала $W^N(y_0^{N-1} | u_0^{N-1})$ без памяти с двоичным входом, где $N = 2^s$, при передаче через него преобразованного блока данных $u_0^{N-1} B_N G^{2^s}$. Метод заключается в представлении $W^N(y_0^{N-1} | u_0^{N-1})$ как совокупности $\bar{W} \equiv \{W_N^{(i)}(y_0^{N-1}, u_0^{i-1} | u_i), 0 \leq i < N\}$ двоичных каналов (рис. 2), обладающей следующими свойствами.

1. Эквивалентность W и \bar{W} : суммарная пропускная способность $\sum_{i=0}^{N-1} C(W_N^{(i)})$ равна $C(W)$ при условии, что для каждого $W_N^{(i)}(y_0^{N-1}, u_0^{i-1} | u_i)$ известны точные значения информационных символов u_0^{i-1} (на рис. 2 это условие обозначено прерывистой линией).

2. Асимптотическая поляризация каналов из \bar{W} : при $N \rightarrow \infty$ относительная доля $W_N^{(i)}$ в \bar{W} таких, что $C(W_N^{(i)}) \rightarrow 1$, стремится к $C(W)$, а пропускная способность остальных $N(1 - C(W))$ каналов стремится к нулю.

Поляризационный эффект достигается применением невырожденного преобразования $B_N G^{2^s}$ к данным. Таким образом, поляризующее преобразование канала $W^{2N}(y_0^{2N-1} | u_0^{2N-1})$ основано на поляризации векторных каналов меньшей размерности $W^i(y_0^{i-1} | u_0^{i-1})$, $i = N, \frac{N}{2}, \frac{N}{4}, \dots, 1$.

В [1] показано, что эта декомпозиция для

плотностей $W_N^{(i)}(y_0^{N-1}, u_0^{i-1} | u_i)$ вероятностей выражается как

$$W_{2N}^{(2i)}(y_0^{2N-1}, u_0^{2i-1} | u_{2i}) = \frac{1}{2} \sum_{u_{2i+1}} W_N^{(i)}(y_0^{N-1}, u_{0,e}^{2i-1} \oplus u_{0,o}^{2i-1} | u_{2i} \oplus u_{2i+1}) \times \quad (1)$$

$$\times W_N^{(i)}(y_N^{2N-1}, u_{0,o}^{2i-1} | u_{2i+1});$$

$$W_{2N}^{(2i+1)}(y_0^{2N-1}, u_0^{2i} | u_{2i+1}) = \frac{1}{2} W_N^{(i)}(y_0^{N-1}, u_{0,e}^{2i-1} \oplus u_{0,o}^{2i-1} | u_{2i} \oplus u_{2i+1}) \times \quad (2)$$

$$\times W_N^{(i)}(y_N^{2N-1}, u_{0,o}^{2i-1} | u_{2i+1});$$

$$W_0^{(0)}(y | u) = W(y | u), \quad (3)$$

где $u_{0,e}^{2i-1}$ и $u_{0,o}^{2i-1}$ обозначают компоненты вектора u_0^{2i-1} только с четными и нечетными номерами соответственно. Вероятности передаваемых битов в каналах $W_N^{(i)}(y_0^{N-1}, u_0^{i-1} | u_i)$ оцениваются через рекуррентные формулы (1), (2) с условиями выхода из рекурсии (3).

При $N \rightarrow \infty$ канал $W_N^{(i)}$ будет либо абсолютно без шума, либо абсолютно ненадежным. Поэтому информационные символы u_i , передаваемые по каналам плохого качества, можно считать всегда фиксированными («замороженными»). И если сопоставить i -ю строку матрицы G^{2^s} каналу $W_N^{(i)}(y_0^{N-1}, u_0^{i-1} | u_i)$, то r наиболее надежным каналам из \bar{W} соответствует линейный $(2^s, r)$ -код длины 2^s и размерности r , называемый *полярным кодом*. Таким образом, чтобы задать полярный код порядка s , достаточно указать множество A индексов неиспользуемых строк матрицы G^{2^s} .

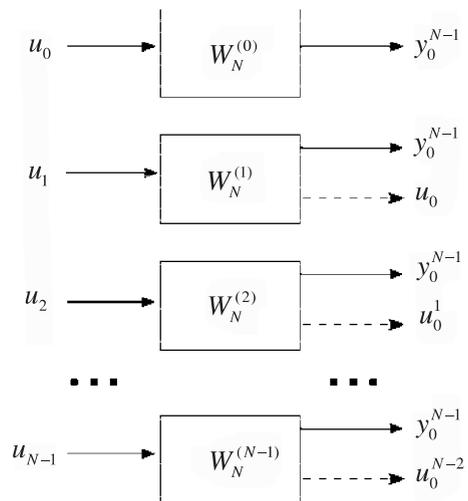


Рис. 2. Подканалы $W_N^{(i)}$ (-----) достоверная передача через канал

Списочный декодер Тала–Варди

Идея традиционного декодера [1] для полярных кодов заключается в последовательной оценке значений передаваемых битов по подканалам $W_N^{(i)}(y_0^{N-1}, u_0^{i-1} | u_i)$. При этом для $(i+1)$ -го канала в качестве достоверных битов u_0^i используются оценки входов подканалов всех предшествующих номеров. Таким образом, если на какой-то стадии происходит ошибка, то и оценка всех оставшихся битов будет ошибочной. Алгоритм Тала–Варди позволяет исправить эту ситуацию. На рис. 3 представлено идеальное дерево всех различных вариантов значений передаваемых битов u_i для полярного $(4,3)$ -кода с множеством «замороженных» битов $A = \{0\}$. Количество ветвей в таком дереве равно 2^r , где r – размерность кода. Пусть l – желаемый размер списка наиболее правдоподобных ветвей этого дерева. Правило выбора l наиболее надежных ветвей при оценке бита u_i соответствует выбору таких ветвей, отвечающих наиболее вероятным оценкам u_i – рис. 4.

Вычисления в логарифмической области

Для реализации правила выбора на i -й ста-

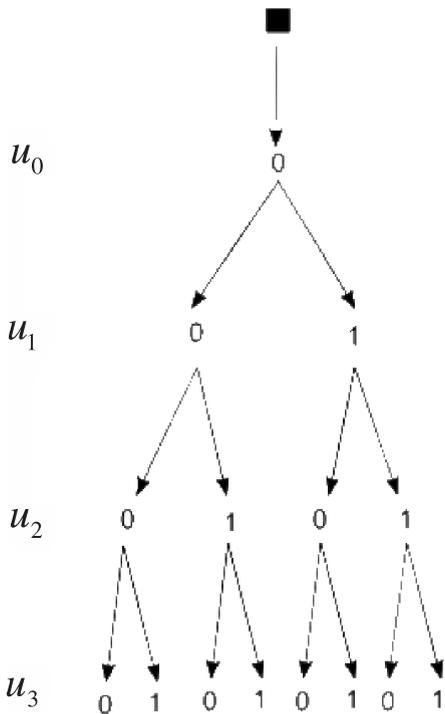


Рис. 3. Идеальное дерево вариантов для полярного $(4,3)$ -кода

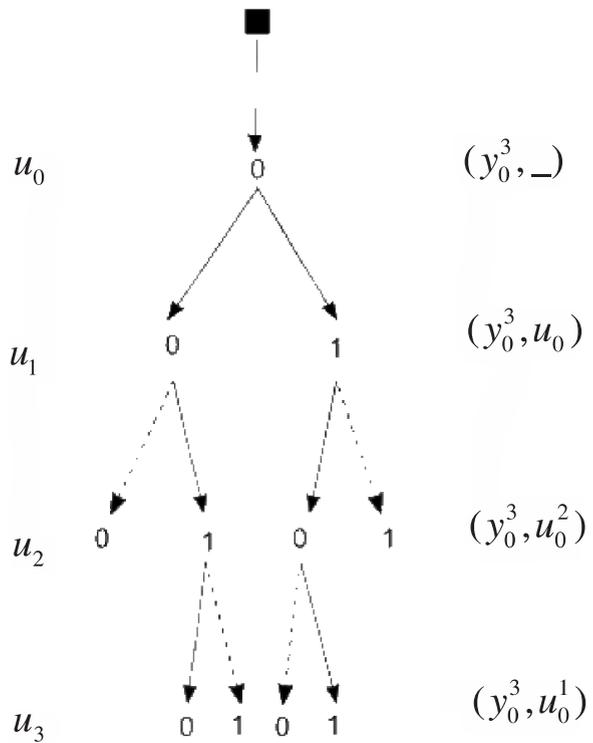


Рис. 4. Выбор только двух ветвей (непрерывная линия) на каждой стадии для полярного кода длины 4 и размерности 3

дии декодировании необходимо иметь значения вероятностей обоих случайных событий: «передавался 0» и «передавалась 1». Таким образом, переход к хорошо известной для полярных кодов области логарифмических отношений правдоподобия (ЛОПП) невозможен. В связи с этим для реализации формулы (1) в логарифмической области требуется решение следующей задачи: имея $\log(a_t) = \log W_N^{(i)}(y_0^{N-1}, u_{0,e}^{2i-1} \oplus u_{0,o}^{2i-1} | t)$ и $\log(b_t) = \log W_N^{(i)}(y_0^{N-1}, u_{0,e}^{2i-1} | t)$ для $t = 0, 1$, вычислить $\log W_{2N}^{(2i)}(y_0^{2N-1}, u_0^{2i-1} | u_{2i}) = \log \frac{1}{2}(a_0 b_0 + a_1 b_1) = \log \frac{1}{2}(e^{\log(a_0) + \log(b_0)} + e^{\log(a_1) + \log(b_1)})$. Не ограничивая общности, будем считать, что $\log(a_0) + \log(b_0) > \log(a_1) + \log(b_1)$. Тогда искомое выражение можно представить как $\log \frac{1}{2} + \log(a_0) + \log(b_0) + \log(1 + e^{\log(a_1) + \log(b_1) - \log(a_0) - \log(b_0)})$. В случае, когда $\log(a_1) + \log(b_1) - \log(a_0) - \log(b_0)$ мало (например, меньше $q = -70$), то $\log(1 + e^{\log(a_1) + \log(b_1) - \log(a_0) - \log(b_0)}) \approx 0$. В итоге аппроксимация для формулы (1) представляется в следующем виде:

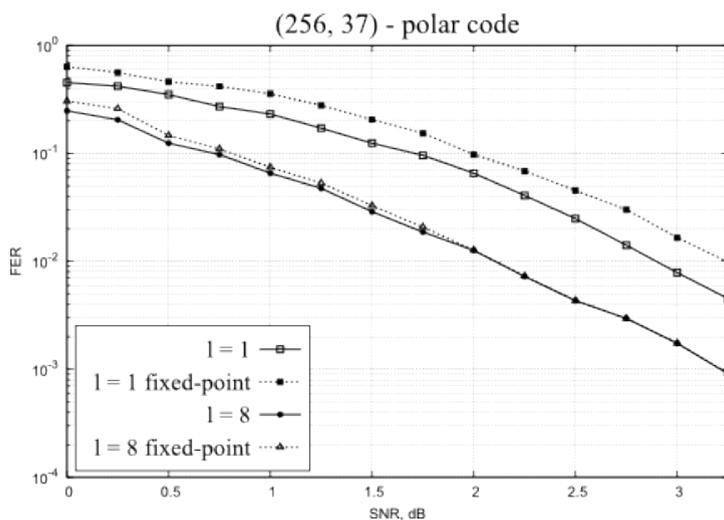
$$\log W_{2N}^{(2i)}(y_0^{2N-1}, u_0^{2i-1} | u_{2i}) \approx \begin{cases} \log \frac{1}{2} + \log(a_0) + \log(b_0), c < -q \\ \log \frac{1}{2} + \log(a_0) + \log(b_0) + \log(1 + e^c), -q \leq c \leq q, \\ \log \frac{1}{2} + \log(a_1) + \log(b_1), c > q \end{cases}$$

где $c = \log(a_i) + \log(b_i) - \log(a_0) - \log(b_0)$.

Реализация формулы (2) проводится аналогично. Как было показано в [3], для традиционного декодера полярных кодов, работающего с ЛОПП, достаточно обеспечить 4–6 битов на дробную часть при использовании чисел с фиксированной запятой. На рис. 5 а, б представлены

графики вероятности ошибки декодирования в зависимости от отношения сигнал/шум (в дБ) для полярных (256,37)- и (256,163)-кодов. Видно, что чем длиннее список, тем менее важна точность вычислений: так, для длины списка $l = 8$ вероятность ошибки декодирования для вычислений с плавающей запятой и для целочисленных вычис-

а)



б)

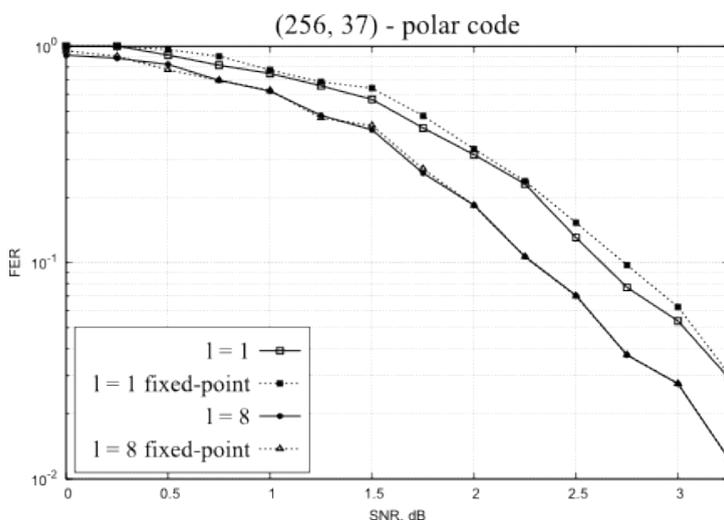


Рис. 5. Вероятность ошибки декодирования:

а – для (256,37)-кода для чисел с плавающей запятой и фиксированной (0 бит на дробную часть);
 б – для (256,163)-кода для чисел с плавающей запятой и фиксированной (0 бит на дробную часть)

лений (числа с фиксированной запятой и нулем бит на дробную часть) имеют несущественные отличия.

Представленный способ модификации алгоритма Тала–Варди позволяет использовать списочный декодер в каскадных конструкциях и

предоставляет возможность эффективной аппаратной реализации на платформах с целочисленной арифметикой.

Работа выполнена в рамках ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса РФ на 2007–2013 гг.» при финансовой поддержке Министерства образования и науки РФ, госконтракт 07.514.11.4101.

СПИСОК ЛИТЕРАТУРЫ

1. **Arikan, E.** Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels [Text] / E. Arikan // IEEE Trans. on Inf. Th. –2009. –№ 7 (55). –P. 3051–3073.

2. **Tal, I.** List decoding of polar codes [Text] / I. Tal, A. Vardy // In proc. of the IEEE Int. Symp. on Inf. Th. (ISIT). – 2011. – P. 1–5.

3. **Leroux, C.** Hardware Implementation of Successive

Cancellation Decoders for Polar Codes [Электронный ресурс] / C. Leroux, A. Raymond, G. Sarkis [et al.] // Submitted to J. Signal Process. Syst. –2011. –Режим доступа: <http://arxiv.org/pdf/1111.4362.pdf>

Leroux, C. Hardware Architectures for Successive Cancellation Decoding of Polar Codes [Text] / C. Leroux, I. Tal, A. Vardy [et al.] // In proc. of the IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP). –2011. –P. 1665–1668.

УДК 621.396.677.8

В.П. Акимов, С.Б. Глыбовский, С.С. Щесняк

КОРРЕКЦИЯ ФАЗОВЫХ ОШИБОК ПАРАБОЛИЧЕСКОГО ЗЕРКАЛА С ПОМОЩЬЮ ВТОРИЧНОГО СЕТЧАТОГО РЕФЛЕКТОРА

Хорошо известно [1], что при наличии неровностей различного характера на поверхности параболического рефлектора зеркальной антенны возникают фазовые ошибки, существенно ухудшающие ее направленные свойства с ростом частоты. В данной статье рассматривается метод повышения рабочей частоты антенны с рефлектором большого размера, заключающийся в добавлении к зеркальной системе дополнительного корректирующего рефлектора (КР), выполненного из ламинированной микросетки с металлическими нитями и расположенного непосредственно над первичным рефлектором. Структура КР и его расположение пояснены на рис. 1.

КР представляет собой слой равной толщиной δ с диэлектрической проницаемостью ϵ_r , внутри которого симметрично расположена сетка из тонких металлических нитей с проводимостью σ_i и магнитной проницаемостью μ_i . Рассматриваемый метод имеет следующие преимущества: установка вторичного сетчатого КР поверх первичного неровного металлического рефлектора может обеспечить меньшие искажения формы сетчатой отражающей поверхности, а значит – добиться улучшения качества профиля; в силу своей легкости КР не оказывает заметных весовых нагрузок на опоры первичного рефлектора; ламинирование нитей сетки позволяет избежать коррозии отражающей поверхности.

Цель данной статьи – построение численно-аналитической модели параболической антенны, оборудованной КР со структурой, показанной на рис. 1, а также расчет основных ее характеристик с учетом неровностей первичного рефлектора.

Для детального анализа зеркальных антенн большого размера широко используется токовый метод, основанный на приближении физи-

ческой оптики. Метод применительно к цельно-металлическим зеркалам подразумевает расчет приближенного распределения поверхностных «квазиоптических» токов, наведенных полем падающей волны, что справедливо, когда радиус кривизны зеркала много больше длины волны. При этом токи в каждой точке рефлектора такие же, какие наводились бы на бесконечном плоском рефлекторе, лежащем в касательной плоскости. Аналогичный подход для рассматриваемой антенны с параболическим КР требует рассмотрения отражательных свойств соответствующей плоской бесконечной структуры, что было сделано нами ранее [2]. Получены аналитические выражения для коэффициентов отражения E - и H -поляризованных компонент поля падающей плоской ЭМВ длиной λ_0 в вакууме (R^{EE} и R^{HH} соответственно), а также для коэффициента кросс-поляризации $R^{EH} = -R^{HE}$. Выражения выведены применительно к двум вариантам конфигурации нитей сетки: типа 1 – квадратные ячейки и идеальные электрические контакты ни-

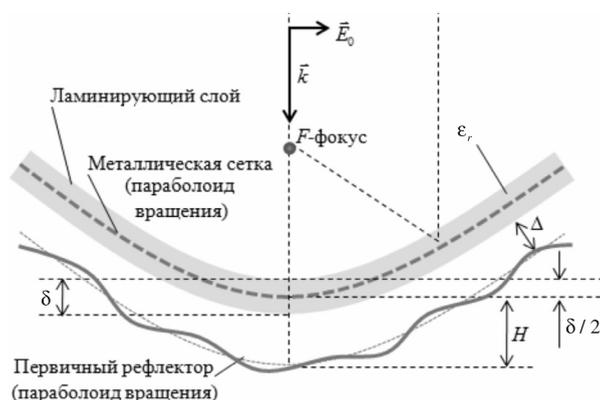


Рис. 1. Корректирующий сетчатый рефлектор, установленный поверх первичного сплошного металлического рефлектора

тей в узлах, и типа 2 – параллельные нити. При этом считалось, что выполняются условия тонкопроволочного приближения: $r_0 \ll \alpha$; $\alpha \ll \lambda$; $\alpha \ll \delta/2$, где $\lambda = \lambda_0 / \sqrt{\epsilon_r}$ – длина волны в диэлектрике, r_0 – радиус нити, α – расстояние между соседними нитями, что позволило использовать усредненные граничные условия Конторовича для сетки в однородном пространстве [3]. Полученные в [2] выражения позволяют анализировать отражательные свойства плоского бесконечного КР над плоским металлическим рефлектором при произвольных параметрах ламинирующего слоя и параметрах сетки, удовлетворяющих указанным выше приближениям. Показано, что для сетки типа 1 зависимости фазы коэффициента отражения от величины Δ воздушного зазора между первичным рефлектором и корректирующим представляют собой кривые с преобладанием участков медленного изменения фазы. Этот эффект приводит к слабой зависимости фазы отраженной волны от точки отражения на поверхности КР при наличии неровностей первичного рефлектора. Однако было также показано присутствие узких областей резонанса между рефлекторами, где фаза может меняться быстро. Следует также упомянуть о том, что в [2] сделан вывод о нецелесообразности применения в составе КР сетки типа 2, обладающей анизотропными свойствами из-за искажения фазовых характеристик под влиянием кросс-поляризации. Поэтому далее подразумевается конфигурация нитей, соответствующая сетке типа 1.

Рассмотрим задачу о параболическом КР, установленном поверх первичного параболического рефлектора (см. рис. 1). Неровности первичного рефлектора пока не учитываются. Радиус кривизны в любой точке на поверхности КР считается много большим длины волны. Кроме того, как было показано в работе, в условиях, соответствующих рефлекторам большого размера сантиметрового и миллиметрового диапазонов, радиус кривизны существенно превышает расстояние распространения поверхностной волны вдоль профиля КР. Поэтому можно рассматривать локальное отражение падающей волны от рассматриваемой структуры, используя полученные ранее формулы по аналогии с токовым методом. Особенность данного метода заключается в том, что на поверхности КР присутствуют, вообще говоря, касательные составляющие как вектора \vec{H} , так и вектора \vec{E} , и в качестве вторичных источни-

ков поля по принципу Гюйгенса–Френеля необходимо учитывать не только электрические, но и магнитные токи. В данной ситуации касательную составляющую полного электрического поля на поверхности КР можно записать в виде:

$$\vec{E}_\tau = \{\vec{t}^H [E_0^H (1 + R^{HH}) + E_0^E R^{EH}] + \vec{\tau}^E [E_0^E (1 - R^{EE}) - E_0^H R^{HE}]\} \exp(-i\vec{k}\vec{r}),$$

где $\vec{\tau}^E$, \vec{t}^H – касательные составляющие ортов, задающих локальное направление электрического поля E - и H -поляризации; E_0^E , E_0^H – E - и H -компоненты вектора-амплитуды падающей волны; \vec{k} – ее волновой вектор; \vec{r} – радиус-вектор точки отражения на поверхности КР.

Аналогичным образом находится и вектор \vec{H}_τ . Затем поле в некоторой точке наблюдения может быть рассчитано по следующей форме [4]:

$$\vec{E}^S \vec{a} = \int_S \{[\vec{n} \times \vec{H}_\tau] \vec{E}^D + [\vec{n} \times \vec{E}_\tau] \vec{H}^D\} dS, \quad (1)$$

где \vec{n} – внешняя нормаль к поверхности рефлектора; \vec{E}^D , \vec{H}^D – поля точечного электрического диполя с моментом \vec{a} , расположенного в точке наблюдения, вычисляемые по известным в литературе выражениям [1].

Относительное снижение КНД антенны по сравнению со случаем сплошного ровного металлического рефлектора определяется исходя из изменения уровня основной компоненты поля $\frac{|\vec{E}^S \cdot \vec{a}|}{|\vec{a}|}$ в фокусе, рассчитываемого по формуле (1).

Для вычисления интеграла удобно использовать сферическую систему координат, показанную на рис. 2 а. Если КР располагается над первичным рефлектором таким образом, что толщина всей структуры постоянна, внешняя поверхность КР уже не является параболической, что необходимо учитывать. При этом наблюдается существенное снижение КНД антенны из-за расфокусировки зеркала, что говорит о необходимости юстировки профиля КР с целью обеспечения параболической формы сетчатой поверхности (S_0 на рис. 2 а). Подобная юстировка подразумевается в дальнейшем, однако внешняя поверхность S ламинирующего слоя КР, по которой производится интегрирование в (1), по-прежнему отличается от параболоида вращения.

Каждой точке на поверхности первичного рефлектора (R, ϑ, φ) соответствует точка $(R_0, \vartheta_0, \varphi_0)$ на сетчатой параболической поверхности таким образом, что данные две точки соединяются от-

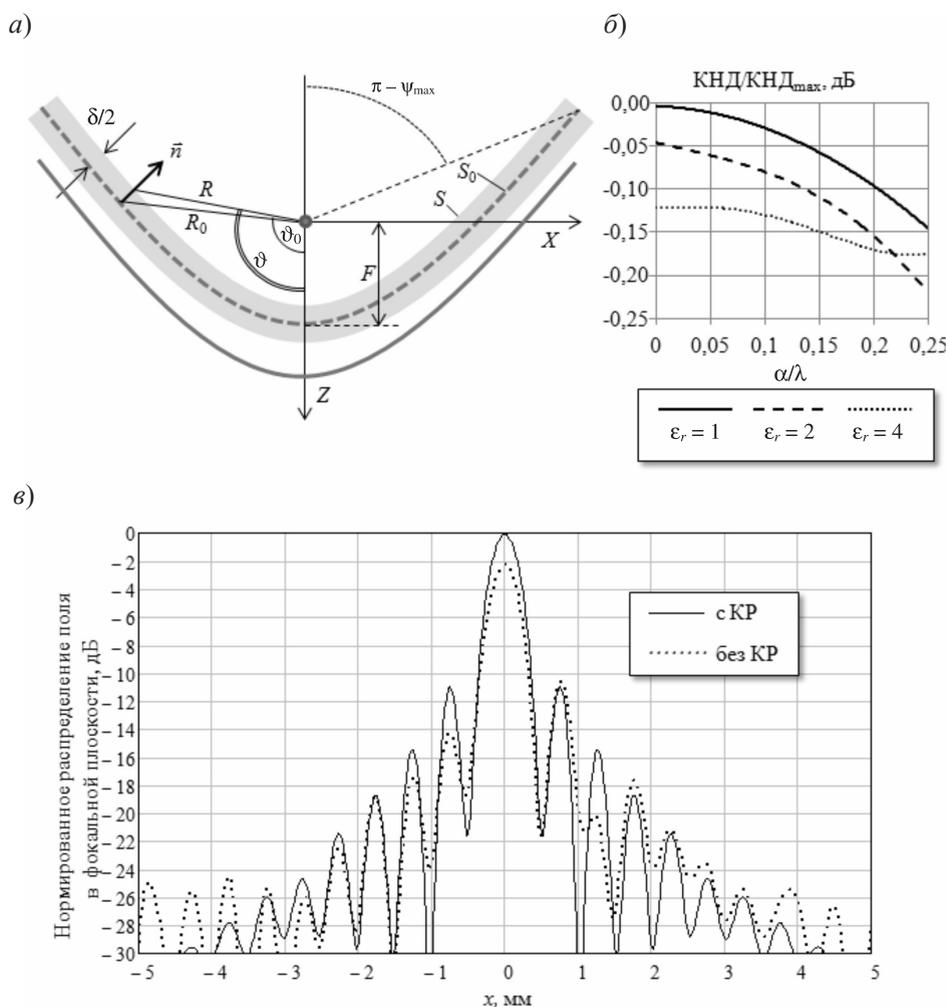


Рис. 2. Используемая сферическая система координат (а) и влияние вносимой со стороны КР фазовой ошибки на КНД антенны (б). Нормированное распределение поля основной компоненты в фокальной плоскости (в пределах квадрата размером 10 мм) с КР и без него (в)

резком нормали, общей для обеих поверхностей. Переход в (1) от координат ϑ, φ к координатам на параболоиде ϑ_0, φ_0 значительно упрощает выражения при разложении полей по ортам E - и H -поляризации. Связь координат определяется следующими выражениями:

$$\vartheta = \vartheta_0 + \arccos \left[\frac{R_0 - \delta \cos(\vartheta_0 / 2) / 2}{\sqrt{R_0^2 + (\delta / 2)^2 - \delta R_0 \cos(\vartheta_0 / 2)}} \right];$$

$$\varphi = \varphi_0, \quad (2)$$

где функция $R_0(\vartheta_0) = 2F / (1 + \cos(\vartheta_0))$ описывает параболическую поверхность сетки S_0 с фокусным расстоянием F . Внешняя поверхность S КР задается формулой $R^2 = R_0^2 + (\delta / 2)^2 - \delta R_0 \cos(\vartheta_0 / 2)$.

Величина воздушного зазора Δ между пер-

вичным рефлектором и корректирующим в выражениях зависит от текущей координаты на поверхности как $\Delta \approx H \cos(\vartheta_0 / 2) - \delta / 2$, где H – расстояние между центром первичного рефлектора и центром параболоида сетки.

По формуле (1) с учетом связи (2) проведены расчеты КНД параболической антенны, оборудованной КР в отсутствие каких-либо неровностей по методу интегрирования Гаусса пятого порядка. Выбранные параметры КР следующие: $\lambda_0 = 1$ мм, $H = 2$ мм, $\delta = 0,5$ мм, $r_0 / \alpha = 0,1$. Параметры первичного рефлектора соответствуют размерам главного зеркала радиотелескопа типа РТ-70 (диаметр 70 м, фокусное расстояние 21 м). На рис. 2 б показано, как вносимая со стороны КР фазовая ошибка влияет на КНД антенны в зависимости от плотности сетки, применяемой в со-

ставе КР при различных значениях ε_r . Фазовая ошибка связана со слабой зависимостью R^{EE} и R^{HH} от Δ , а также от наличия внешней половинны ламинирующего слоя. Это приводит к тому, что использовать КР целесообразно только при наличии неровностей первичного рефлектора с достаточно высокой амплитудой. Однако точная оценка корректирующей способности КР требует математического моделирования случайных поверхностных возмущений.

Выбранный метод моделирования случайных неровностей заключается в получении отдельных реализаций, описывающих искажения профиля и переходе к отдельным реализациям электродинамических параметров системы. Затем полученные данные усредняются по ансамблю реализаций. Данный подход выбран как дающий более полную информацию о механизме работы КР и приводящий к меньшей трудоемкости вычислений. Следует заметить, что для анализа в приближении токового метода должно выполняться условие $L \gg h_0$, где h_0 – максимальное отклонение неровной по-

$$U(x, y) = \begin{cases} \exp \left[w - w \cdot \left(1 - \left(\frac{x}{dx \cdot r} \right)^2 - \left(\frac{y}{dy \cdot r} \right)^2 \right)^{-1} \right], & \left(\frac{x}{dx} \right)^2 + \left(\frac{y}{dy} \right)^2 < r^2 \\ 0, & \left(\frac{x}{dx} \right)^2 + \left(\frac{y}{dy} \right)^2 \geq r^2. \end{cases} \quad (3)$$

Параметры, входящие в (3), выбраны обеспечивающими наилучшие аппроксимирующие свойства: $dx = dy = L$ – характерная величина «шага» неровностей; $w = 0,89$; $r = 1,5$. Радиус корреляции при этом равен $3L/2$. Функция неровностей строится во вспомогательных координатах следующим образом:

$$h'(x', y') = \sum_{i=0}^{2N} \sum_{j=0}^{2N} C_{i,j} U(x' - x'_i, y' - y'_j),$$

где $x'_i = (i / N - 1)R_0(\psi_{\max}) \sin(\psi_{\max})$; $y'_j = (j / N - 1)R_0(\psi_{\max}) \sin(\psi_{\max})$ ($i, j \in 0, 1, \dots, 2N$); $N = [R_0(\psi_{\max}) \sin(\psi_{\max}) / L]$; $C_{i,j}$ – набор случайных коэффициентов с равномерной плотностью распределения на отрезке $[0; 1]$; ψ_{\max} – половинный угол раствора параболоида.

Для построения координатной сетки на поверхности параболоида с постоянной площадью ячейки выведено следующее преобразование координат:

$$\vartheta'(\vartheta_0) = A' \{ 0,25 \sin(\vartheta_0 / 2) [\sec^4(\vartheta_0 / 2) + 1,5 \sec^2(\vartheta_0 / 2)] + 0,375 \ln[\operatorname{tg}(0,25\pi + 0,5\vartheta_0)] \},$$

верхности от невозмущенной, L – характерная протяженность неровностей вдоль поверхности. Наличие неровностей будет проявляться в изменении формы сетчатой поверхности КР, а также в изменении расстояния $\Delta(\vartheta_0, \varphi_0)$ между исходным и корректирующим рефлекторами. Если функция $h(\vartheta_0, \varphi_0)$ описывает неровности первичного рефлектора, а КР при этом имеет идеальную форму, то $\Delta(\vartheta_0, \varphi_0) = H \cos(\vartheta_0 / 2) - \delta / 2 - h(\vartheta_0, \varphi_0)$. В случае собственных неровностей КР необходимо учитывать также меняющееся расстояние между точками интегрирования и наблюдения $R'(\vartheta_0, \varphi_0) = R(\vartheta_0) - h(\vartheta_0, \varphi_0) \sec[\vartheta(\vartheta_0) - 0,5\vartheta_0]$, причем R' входит только в фазовую часть подынтегрального выражения (1) с точностью до величины h_0 / F . Функция неровностей нормируется на заданную максимальную амплитуду h_0 таким образом, что $h(\vartheta_0, \varphi_0) = h_0 \tilde{h}(\vartheta_0, \varphi_0)$, причем \tilde{h} строится на специально созданной сетке поверхностных координат с постоянной площадью ячейки. Для построения \tilde{h} применяется специальная финитная бесконечнодифференцируемая функция вида [5]:

где $A' = \psi_{\max} \{ 0,25 \sin(\psi_{\max} / 2) [\sec^4(\psi_{\max} / 2) + 1,5 \sec^2(\psi_{\max} / 2) + 0,375 \ln[\operatorname{tg}(0,25\pi + 0,5\psi_{\max})]] \}^{-1}$.

В результате можно прийти к следующему описанию поверхностных неровностей в выбранной сферической системе координат:

$$\tilde{h}(\vartheta_0, \varphi_0) = (\max |h'|)^{-1} \cdot h' [R_0(\vartheta'(\vartheta_0)) \sin(\vartheta'(\vartheta_0)) \cos \varphi_0, R_0(\vartheta'(\vartheta_0)) \sin(\vartheta'(\vartheta_0)) \sin \varphi_0].$$

Для неровностей подобного вида отношения СКО к амплитуде равно 0,28. Далее приводятся результаты расчета характеристик зеркальной антенны с КР, учитывающего неровности первичного рефлектора посредством изложенных выше выражений. На рис. 2 в показаны типичные распределения основной компоненты поля в фокальной плоскости в отсутствие и при наличии идеального КР ($\alpha / \lambda = 1 / 10$, $r_0 / \alpha = 1 / 10$) для неровностей первичного рефлектора с параметрами $h_0 = 0,2\lambda$, $N = 7$. Значения остальных величин соответствуют выбранным ранее. Сравнение проводится для фиксированной реализации $h(\vartheta_0, \varphi_0)$. Нормировка графиков на рис. 2 в произведена на

максимально достижимый уровень поля в фокусе (в случае идеального одиночного металлического параболоида). Собственные неровности КР отсутствуют. В отсутствие КР поверхностные неровности приводят к снижению КНД антенны на 39,3 % (-2,1 дБ), в то время как при наличии юстированного КР КНД падает всего на 2,1 % (-0,1 дБ), что хорошо видно по уровню поля в максимуме на рис. 2 в. Целесообразно ввести в рассмотрение количественную характеристику ξ степени коррекции, равную отношению потери КНД за счет неровностей в отсутствие КР к аналогичной величине при наличии КР:

$$\xi = \frac{\Delta \text{КНД}_{\text{БЕЗ КР}}}{\Delta \text{КНД}_{\text{С КР}}}$$

Результаты расчета показали, что при наличии неровностей одиночного металлического рефлектора с увеличением амплитуды h_0 в пределах

от 0 до λ_0 наблюдается относительное снижение уровня поля в фокусе 98–99 %, что связано с растущей фазовой ошибкой по всей площади зеркала. То есть антенна полностью теряет направленные свойства. При наличии КР ситуация иная: относительное снижение уровня поля в фокусе составляет не более 1 % (для периода сетки от 0,05 до 0,2 длины волны в диэлектрике). Таким образом, при отсутствии собственных неровностей КР обладает способностью коррекции фазовых ошибок, вызванных неровностями первичного (корректируемого) рефлектора. Если поверхность сетки КР идеально параболическая, то единственными причинами фазовых ошибок служат резонансные области фазовых характеристик R^{EE} и R^{HH} [2], а также верхняя половина ламинирующего слоя. Однако установлено, что области резонансов, возникающие на поверхности параболического КР, имеют «квазиодномер-

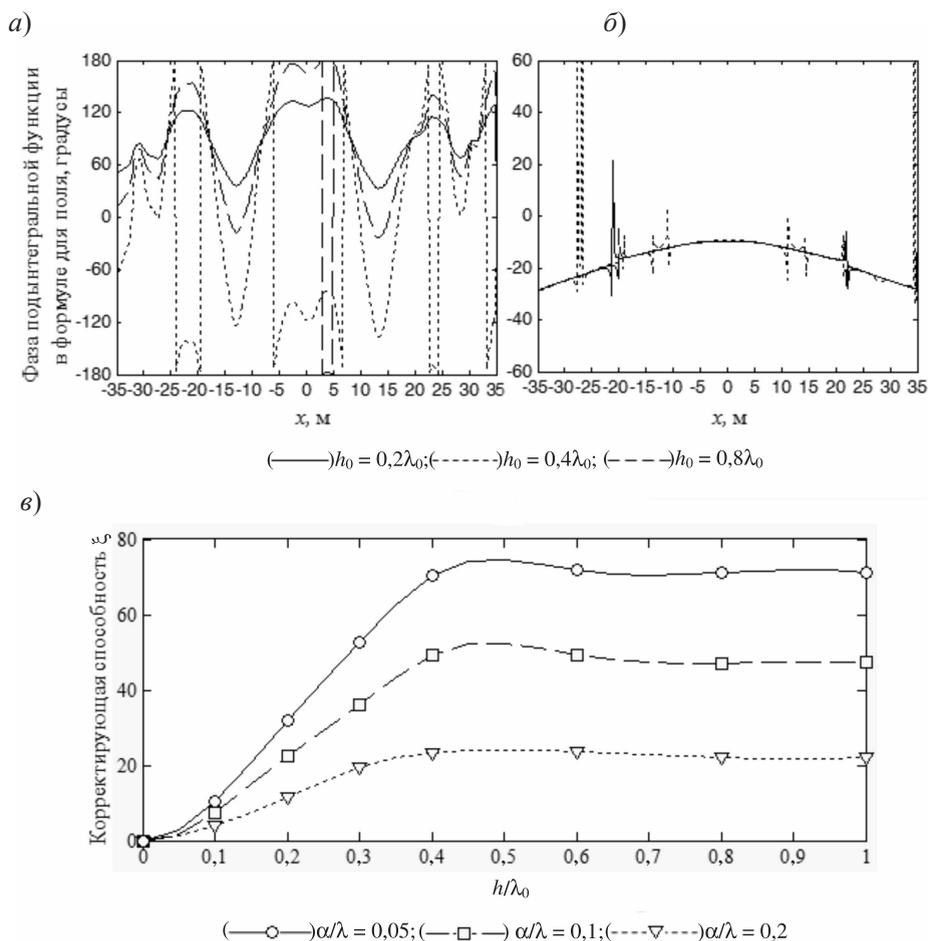


Рис. 3. Зависимость фазы подынтегрального выражения для расчета поля от x -координаты точки на поверхности в отсутствие КР (а) и при наличии КР (б); зависимость корректирующей способности от амплитуды неровностей при различных значениях периода металлической сетки (в)

ный» характер, т. е. их площадь остается малой даже при больших амплитудах неровностей первичного рефлектора. Сказанное иллюстрируется на рис. 3 а, б, где приведены зависимости фазы подынтегральной функции в выражении (1) от координаты x точки интегрирования на поверхности для трех значений амплитуды h_0 неровностей первичного рефлектора.

Кривые в отсутствие КР приведены на рис. 3 а. При этом наблюдается изменение фазы вдоль зеркала в пределах от 40 до 320 градусов в зависимости от h_0 . На рис. 3 б представлены кривые для тех же реализаций $h(\vartheta_0, \varphi_0)$, однако при наличии КР с выбранными ранее параметрами. Видно, что присутствует медленно меняющаяся фазовая ошибка, вызванная ламинирующим слоем, не превышающая 20 градусов. Также присутствуют резонансные участки быстро меняющейся фазы, которые занимают лишь малую часть графиков даже при $h_0 = 0,8\lambda_0$. Расчеты показали, что с ростом h_0 происходит лишь реконфигурация резонансных областей, в то время как площадь, занимаемая ими на поверхности рефлектора, растет крайне медленно. В итоге указанный эффект приводит к значительно более слабому снижению КНД с ростом h_0 , нежели в отсутствие КР.

В работе была проведена серия расчетов корректирующей способности ξ для различных параметров КР с усреднением по множеству реализаций $h(\vartheta_0, \varphi_0)$ в каждом случае. Результаты представлены на рис. 3 в для трех значений периода металлической сетки α : 0,05 λ , 0,1 λ и 0,2 λ . Из рисунка видно, что корректирующая способность растет с увеличением амплитуды неровностей первичного рефлектора. При этом применение КР целесообразно при значениях h_0 , превосходящих $\lambda_0 / 40 \div \lambda_0 / 20$ в зависимости от плотности сет-

ки, когда наблюдается выигрыш в КНД по сравнению со случаем отсутствия коррекции. Видно также, что наилучшие корректирующие свойства проявляются для неровностей с амплитудой более $\lambda_0 / 2$. Для указанных значений кривые на рис. 3 в выходят на насыщение, связанное с полным разрушением формы фокального пятна в отсутствие КР (спад уровня поля в фокусе прекращается). Также для больших h_0 можно заметить слабое снижение ξ , что объясняется увеличением площади резонансных областей, упомянутых выше.

Таким образом, применение КР позволяет снизить потери КНД за счет неровностей первичного рефлектора в $20 \div 75$ раз (в зависимости от параметров сетки) в случае, когда собственными неровностями КР можно пренебречь. При этом показана возможность работы КР при неровностях первичного рефлектора с амплитудой более половины длины волны, что делает способ коррекции фазовых ошибок интересным для применения в существующих зеркальных системах антенн большого размера. Разработанный численно-аналитический метод, основанный на приближении физической оптики совместно с применением усредненных граничных условий, позволяет рассчитывать все электродинамические свойства зеркальной антенны с КР, являясь универсальным по отношению к моделируемым неровностям. В частных случаях метод сводится к анализу таких систем, как одиночный сетчатый параболический, а также параболическое зеркало с диэлектрическим покрытием. На рассматриваемый способ коррекции фазовых ошибок нами получен патент на полезную модель [6].

СПИСОК ЛИТЕРАТУРЫ

1. Марков, Г.Т. Антенны; Учебник для студентов радиотехнических специальностей [Текст] / Г.Т. Марков, Д.М. Сазонов. –М.: Энергия, 1975. –С. 441–443.
2. Акимов, В.П. Отражающая поверхность на основе ламинированной сетки из металлических нитей для коррекции неровностей рефлекторов зеркальных антенн [Текст] / В.П. Акимов, С.Б. Глыбовский, В.К. Дубрович // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2011. –№ 3 (126). –С. 52–61.
3. Акимов, В.П. Коэффициенты отражения электромагнитных волн от сложных сетчатых структур, параллельных границе раздела сред [Текст] / В.П. Акимов, М.И. Астрахан, Г.И. Поликарпов // Вопро-

сы электромагнитной совместимости и расчета антенн и радиолоний (технические и научно-методические материалы). –ВАС, 1991. –284 с.

4. Каценеленбаум, Б.З. Высокочастотная электродинамика [Текст] / Б.З. Каценеленбаум. –М.: Наука, 1966. –240 с.

5. Щесняк, С.С. Расчет оптимальных размеров и параметров зеркальной системы и предварительная расчетная оценка допустимых погрешностей изготовления ее элементов: Отчет по этапу 2 НИР: 1/2005 [Текст] / Рук. С.С. Щесняк; исполн. А.В. Никитин. –СПб.: ООО «Научный центр прикладной электродинамики», 2005.

6. Патент RU 113079 U1 МПК H01Q15/16.

РАЗРАБОТКА МЕТОДИКИ СИНТЕЗА РАСПРЕДЕЛЕННОГО П-РЕГУЛЯТОРА ПО ПОКАЗАТЕЛЮ КОЛЕБАТЕЛЬНОСТИ

Известны следующие направления в решении проблемы синтеза регуляторов для распределенных систем [3]:

1) аналитическое конструирование оптимальных регуляторов, рассмотренное в работах Т.К. Сиразетдинова, Г.Л. Дегтярева и др.;

2) частотный метод синтеза, описанный в работах В.А. Бессекерского, Е.П. Попова, В.В. Солодовникова, И.М. Першина и др.;

3) параметрический синтез регуляторов, при котором задается структура распределенного регулятора, а параметры его подбираются в процессе экспериментальных исследований. Основные вопросы параметрического синтеза регуляторов на основе структурной теории изучены в трудах А.Г. Бутковского, В.Л. Рожанского и И.М. Бегимова.

Остановимся более подробно на втором направлении, с помощью которого разработаем методику расчета настроек распределенного регулятора по заданному показателю колебательности с применением расширенных частотных характеристик объекта управления [1].

Постановка задачи

Для системы управления распределенным объектом (задана его математическая модель, либо имеется возможность проводить экспериментальные исследования с реальным объектом) необходимо синтезировать регулятор, реализующий пропорциональный закон управления. При этом на запасы устойчивости разомкнутой системы наложены следующие ограничения:

запасы по модулю $\Delta L(G) \geq \Delta L_3$;

запасы по фазе $\Delta \varphi(G) \geq \Delta \varphi_3$;

показатель колебательности $M = M_{\text{зад}}$, где $\Delta L_3, \Delta \varphi_3, M_{\text{зад}}$ – заданные величины.

Процедура синтеза

Процедура синтеза регулятора состоит из следующих этапов.

1. Для выбранных значений $\eta \in [1, m_1]$, $\gamma \in [1, m_2]$ (где m_1 и m_2 – заданные числа) строим частотные характеристики объекта. Положим, что число характеристик равно \tilde{m} . Для каждой пространственной частотной характеристики определим значение обобщенной координаты $G_i (i = \overline{1, \tilde{m}})$.

2. Используя комплексный передаточный коэффициент объекта управления $W((j-m)\omega, \eta, \gamma)$, определяем частоты ω_η среза модуля разомкнутой системы.

3. По выбранному значению показателя колебательности определяем угол μ согласно выражению $\mu = \arcsin\left(\frac{1}{M_{\text{зад}}}\right)$. На комплексной плоскости строим луч ОЕ под углом μ к отрицательной вещественной полуоси.

4. Вычерчиваем M_η -окружности, имеющие центр на отрицательной вещественной полуоси, касающиеся расширенных амплитудно-фазовых характеристик (РАФЧХ) объекта и луча ОЕ. Определяем радиус r_η построенных M_η -окружностей.

5. С помощью выражения $r_\eta = \frac{M_{\text{зад}}}{(M_{\text{зад}}^2 - 1) \cdot K_\eta(G)}$ определяем коэффициенты усиления регулятора $K_\eta(G)$.

6. Определяем параметры весового коэффициента регулятора n_η .

7. С помощью РАФЧХ объекта произведем расчеты значения коэффициента E_1 , обеспечивающего заданный показатель колебательности.

8. Определяем значение коэффициента E_1 .

Пример.

Рассмотрим процедуру синтеза, реализующую



Рис. 1. Структурная схема системы управления

го пропорциональный закон управления, на примере системы управления температурным полем объекта. Структурная схема системы управления изображена на рис. 1.

Объект управления представлен на рис. 2.

Математическая модель объекта управления имеет следующий вид:

$$\frac{\partial Q(x, y, z, t)}{\partial t} = a \left(\frac{\partial^2 Q(x, y, z, t)}{\partial x^2} + \frac{\partial^2 Q(x, y, z, t)}{\partial y^2} + \frac{\partial^2 Q(x, y, z, t)}{\partial z^2} \right), \quad (1)$$

$$0 < x < L_x, 0 < y < L_y, 0 < z < L_z.$$

Граничные и начальные условия задаются следующими соотношениями:

$$Q(x, 0, z, \tau) = Q(x, L_y, z, \tau) = 0, \quad (2)$$

$$\frac{\partial Q(0, y, z, \tau)}{\partial x} = \frac{\partial Q(L_x, y, z, \tau)}{\partial x} = 0,$$

$$\lambda \frac{\partial Q(x, y, L_z, \tau)}{\partial z} = U(x, y, \tau), \quad (3)$$

$$\frac{\partial Q(x, y, 0, \tau)}{\partial z} = 0, \quad (4)$$

$$Q(x, y, z, 0) = 0. \quad (5)$$

Постановка задачи синтеза. Произведем синтез распределенного регулятора для системы управления объектом, при этом будем полагать, что запас устойчивости по фазе $\Delta\varphi \geq \pi/6$, показатель колебательности $M = 2$.

Передаточная функция синтезируемого регулятора согласно [2] имеет вид:

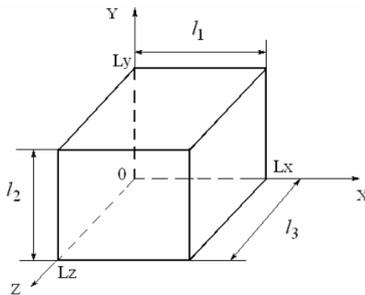


Рис. 2. Объект управления

$$W(x, y, p) = E_1 \left[\frac{n_1 - 1}{n_1} - \frac{1}{n_1} \nabla^2 \right], \quad (6)$$

где E_1 – заданное число (общий коэффициент усиления); n_1 – весовой коэффициент ($n_1 \geq 1$); ∇^2 – лапласиан.

Процедура синтеза. Математическая модель объекта описывается системой дифференциальных уравнений в частных производных. Передаточные функции таких объектов управления могут быть представлены в виде совокупности передаточных функций по каждой пространственной моде [4]. Известно, что для тепловых объектов управления передаточные функции по каждой моде входного воздействия могут быть аппроксимированы передаточными функциями вида:

$$W_\eta(p) = \frac{K_\eta}{T_\eta p + 1} e^{-\tau_\eta p}, \quad (\eta = 1, 2 \dots). \quad (7)$$

Значения параметров K_η , T_η , τ_η определяются по результатам численного моделирования.

Для этого на вход системы подадим входное воздействие вида:

$$\alpha(x, y, p) = C_{\eta, \gamma}(s) \sin(\Psi_\eta \cdot x) \sin(\bar{\Psi}_\gamma \cdot y), \quad (8)$$

где $C_{\eta, \gamma} = e^{j\omega\tau}$; $\Psi_\eta = \frac{\pi\eta}{x_l}$; $\bar{\Psi}_\gamma = \frac{\pi\gamma}{r_l}$.

При определении указанных параметров для $\eta = 1, 3$ учитывались динамические свойства процесса распространения теплового поля внутри объекта.

По результатам моделирования определим значения параметров K_η , T_η , τ_η .

Вычисленные значения равны:

$$K_1 = 0,7; \quad T_1 = 70; \quad \tau_1 = 6;$$

$$K_3 = 0,5; \quad T_3 = 67; \quad \tau_3 = 6.$$

Для частотного анализа объекта положим в (6) $p = (j - m)\omega$ (где ω – круговая частота) и определим модуль M_η и фазу φ_η :

$$M_\eta(\omega) = \frac{K_\eta}{((T_\eta \omega)^2 + 1)^{1/2}} \quad \eta = \overline{1, 2} \dots \quad (9)$$

$$\varphi_\eta(\omega) = -\omega\tau_\eta - \arctg(T_\eta \omega). \quad (10)$$

По результатам вычислений строим расширенные частотные характеристики объекта управления (рис. 3) [2].

Коэффициент усиления пространственно-усилительного звена по каждой составляющей ряда входного воздействия [2] имеет вид:

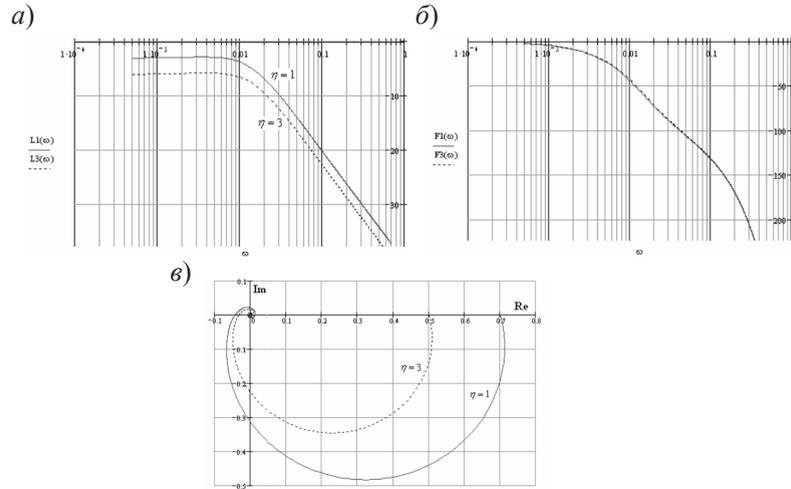


Рис. 3. Расширенные частотные характеристики объекта:
а – РЛАЧХ; б – РФЧХ; в – РАФЧХ

$$K_{\eta,\gamma,\xi} = E_1 \left[\frac{n_1 - 1}{n_1} + \frac{1}{n_1} (\psi_\eta^2 + \tilde{\varphi}_\gamma^2) \right], \quad (11)$$

($\eta, \gamma = \overline{1, \infty}$; $\xi = \overline{1, 4}$).

Представим (11) в следующей форме:

$$K_{\eta,\gamma,\xi} = E_1 \left[\frac{n_1 - 1}{n_1} + \frac{1}{n_1} \tilde{G} \right], \quad (12)$$

где $\tilde{G} = \psi_\eta^2 + \tilde{\varphi}_\gamma^2$ – дискретная функция ($\eta, \gamma = \overline{1, \infty}$). Значения функции \tilde{G} зависят не только от η, γ , но и от x_L, y_L .

Из (12) следует, что коэффициент усиления не зависит от ξ . Таким образом, передаточная функция пространственно-усилительного звена может

быть представлена бесконечной совокупностью коэффициентов усиления (12). Работать с бесконечным набором функций (12) не всегда удобно. Перейдем от набора функций (12) к функциональной зависимости $K(G)$. Для этого заменим \tilde{G} непрерывной функцией G с областью определения $[0, \infty]$. В этом случае при изменении G от 0 до ∞ охватятся все дискретные значения \tilde{G} .

Выражение (12) с учетом изложенного выше может быть записано в виде:

$$K(G) = E_1 \left[\frac{n_1 - 1}{n_1} + \frac{1}{n_1} G \right], \quad 0 \leq G \leq \infty. \quad (13)$$

Произведем расчеты значения коэффициента E_1 , обеспечивающего заданный показатель

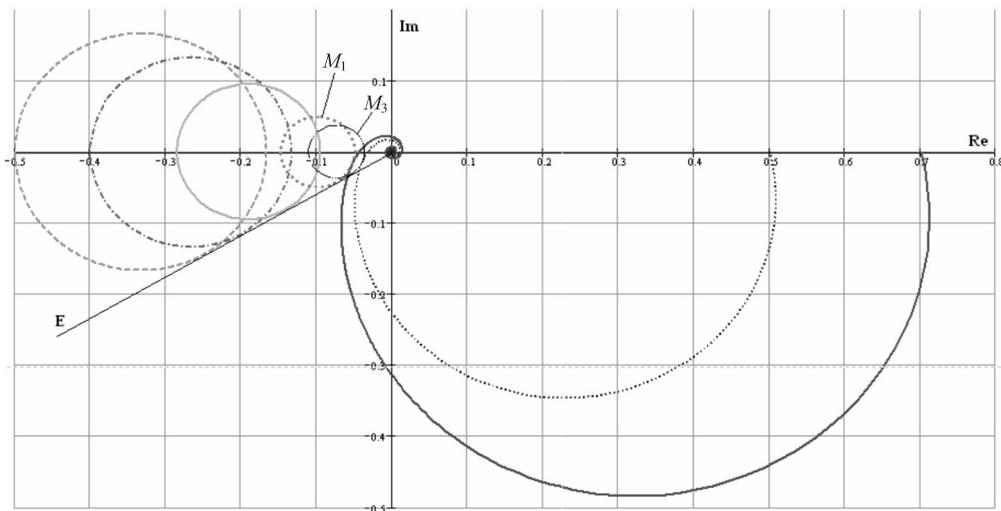


Рис. 4. М-окружности и РАФЧХ

колебательности. Для этого построим РАФЧХ объекта. По выбранному значению показателя колебательности определим угол μ согласно выражению $\mu = \arcsin\left(\frac{1}{M_{\text{зад}}}\right)$. На комплексной плоскости построим луч ОЕ под углом μ к отрицательной вещественной полуоси.

Вычертим M_{η} -окружности, имеющие центр на отрицательной вещественной полуоси, касающиеся РАФЧХ объекта и луча ОЕ. Определим радиус r_{η} построенных M_{η} -окружностей.

С помощью выражения (14) определим значение коэффициента $K_{\eta}(G)$:

$$r_{\eta} = \frac{M_{\text{зад}}}{(M_{\text{зад}}^2 - 1) \cdot K_{\eta}(G)} \quad (14)$$

$$K_1(G) = 13,5; K_3(G) = 18.$$

Определим параметры n_1, E_1 :

$$K_1(G) = E_1 \left[\frac{n_1 - 1}{n_1} + \frac{G_1}{n_1} \right], \quad (15)$$

$$K_3(G) = E_1 \left[\frac{n_1 - 1}{n_1} + \frac{G_3}{n_1} \right]. \quad (16)$$

Поделив (15) на (16), придем к следующему результату:

$$n_1 = \frac{\Delta M - 1 - \Delta M \cdot G_1 + G_3}{\Delta M - 1},$$

$$\text{где } G_1 = \Psi_1 + \overline{\Psi_1} = \frac{\pi \cdot 1}{Lx} + \frac{\pi \cdot 1}{Ly}; \quad G_3 = \Psi_3 + \overline{\Psi_3} = \frac{\pi \cdot 3}{Lx} + \frac{\pi \cdot 3}{Ly}; \quad \Delta M = \frac{K_3(G)}{K_1(G)} = 1,333.$$

Произведем вычисления и получим n_1 :

$$n_1 = 61,85.$$

Подставляя значения $n_1, K_3(G)G_3$ в (16), получим E_1 .

Произведя расчеты по изложенной выше методике, получим следующие результаты:

$$n_1 = 61,85; E_1 = 12.$$

Запишем передаточную функцию распределенного регулятора:

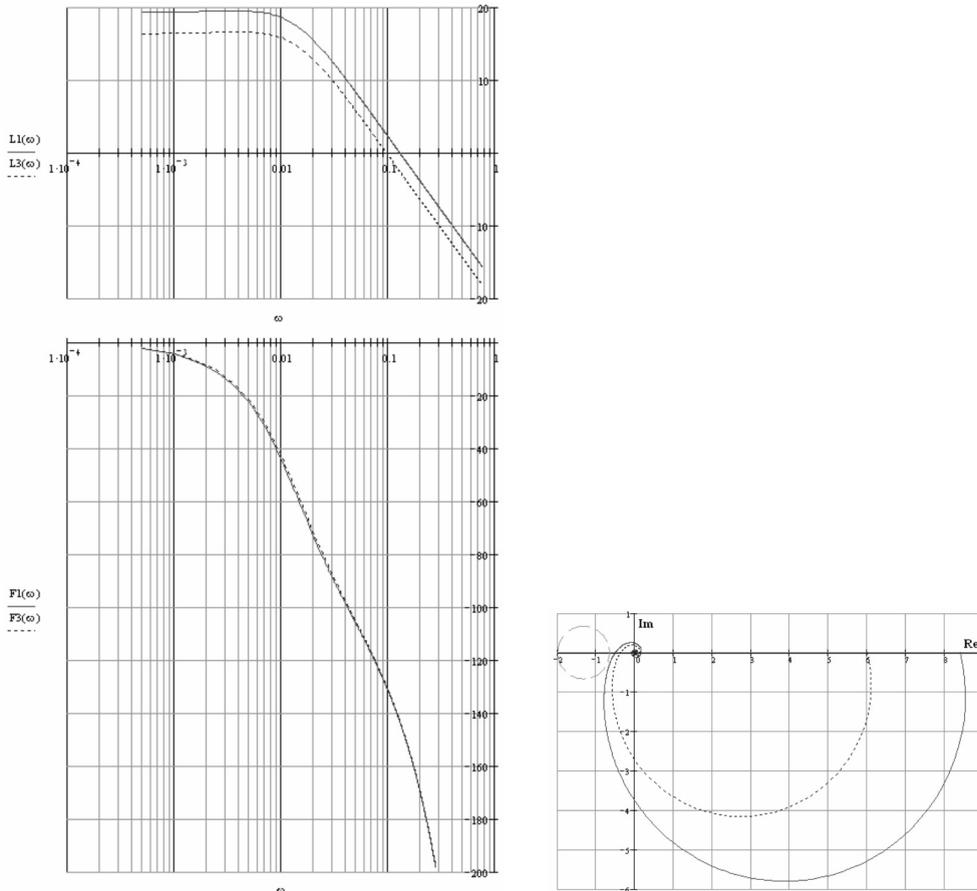


Рис. 5. Расширенные частотные характеристики системы

$$W(x, y, p) = 12 \cdot \left[\frac{61,85 - 1}{61,85} - \frac{1}{61,85} \nabla^2 \right], \quad (17)$$

$$W(x, y, p) = 12 \cdot [0,985 - 0,0161 \cdot \nabla^2].$$

Используя полученную передаточную функцию, построим расширенные частотные характеристики разомкнутой системы управления.

Как видно из графиков (рис. 5), замкнутая система будет устойчива, а запасы устойчивости по

модулю и по фазе у разомкнутой системы не менее заданных.

Разработанная методика позволяет рассчитывать настройки распределенного регулятора, реализующего пропорциональный закон управления, по заданному значению показателю колебательности, а следовательно, и с требуемой степенью затухания переходного процесса.

СПИСОК ЛИТЕРАТУРЫ

1. Дудников, Е.Г. Основы автоматического регулирования тепловых процессов: Учеб. пособие для вузов [Текст] / Е.Г. Дудников. – М., –Л.: Госэнергоиздат, 1956. –264 с

2. Ляшенко, А.Л. Частотный анализ объектов с распределенными параметрами с помощью расширенных частотных характеристик [Текст] / А.Л. Ляшенко // Матер. VI науч. конф. Управление и информационные

технологии. –СПб.: ОАО «Концерн «ЦНИИ Электроприбор», 2010. –С. 65–70.

3. Першин, И.М. Анализ и синтез систем с распределенными параметрами [Текст] / И.М. Першин. –Пятигорск: РИА на КМВ, 2007. –244 с.

4. Рапопорт, Э.Я. Структурное моделирование объектов и систем управления с распределенными параметрами [Текст] / Э.Я. Рапопорт. –М.: Высш. школа, 2003. –299 с.

УДК 681.3 (075.8)

Н.В. Ростов

МНОГОКРИТЕРИАЛЬНОЕ НОРМИРОВАНИЕ ЭТАЛОННЫХ МОДЕЛЕЙ В ЗАДАЧАХ СИНТЕЗА СИСТЕМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

Синтез линейных САУ с регуляторами различных типов обычно проводится с использованием явных или неявных эталонных моделей (ЭМ). Такими моделями могут быть желаемая передаточная функция (ПФ) или ЛАХ разомкнутой системы, желаемое размещение полюсов и нулей ПФ замкнутой системы или интегральный квадратичный критерий с заданными матрицами весовых коэффициентов. Параметрическая оптимизация линейных и нелинейных САУ итерационными методами также часто осуществляется по ЭМ, задающей желаемые показатели качества проектируемой САУ.

Однако из-за субъективности выбора ЭМ динамические возможности синтезированной системы могут оказываться не полностью использованными, либо задаваемая эталонной моделью динамика может быть недостижимой для реальной системы из-за энергетических ограничений, влияния нелинейностей или недостаточных запа-

сов робастности при действии внешних возмущений и отклонениях параметров объекта управления (ОУ) от расчетных значений. В связи с этим важное значение имеет оптимизация выбора ЭМ, представляемой в нормированном виде.

В теории автоматического управления разработаны методы нормирования ЭМ по критериям, оценивающим быстродействие, степень устойчивости, колебательность и другие показатели САУ [1, 2]. На практике требуется выбирать ЭМ одновременно по совокупности показателей, следовательно, необходимо многокритериальное (МК) нормирование. При этом векторные критерии, по которым осуществляется нормирование, и структура ЭМ должны соответствовать классу (целевому назначению) САУ и структурным особенностям объекта управления и регулятора.

Ниже предлагается метод МК-нормирования, позволяющий определять Парето-оптимальные значения параметров операторных и векторно-

матричных ЭМ, которым соответствуют наилучшие показатели ЭМ по заданной совокупности критериев.

Известные методы нормирования

Рассмотрим теоретические методы нормирования ЭМ замкнутых систем в форме Вышнеградского с оператором $q = \omega_o s$, где ω_o – параметр, масштабирующий время.

При нормировании ПФ в классе низкочастотных фильтров Баттерворса $W_{ЭМ}(q) = 1 / Q_n(q)$ их переходные характеристики имеют перерегулирование не более 20 % и относительное время переходного процесса $\tau_n = t_n \omega_o = 10 \div 20$. Характеристические полиномы с рекомендуемыми нормированными коэффициентами имеют следующий вид:

$$Q_2(q) = q^2 + 1,4142q + 1;$$

$$Q_3(q) = (q+1)(q^2 + q + 1);$$

$$Q_4(q) = (q^2 + 0,7654q + 1)(q^2 + 1,8478q + 1);$$

$$Q_5(q) = (q+1)(q^2 + 0,6180q + 1)(q^2 + 1,6180q + 1).$$

Значения коэффициентов для полиномов более высокой степени до $n = 10$ приведены в [1].

При нормировании ЭМ замкнутых систем, представляемых в виде

$$W_{ЭМ}(q) = \frac{A_3 q^2 + A_2 q + A_1}{q^n + A_n q^{n-1} + \dots + A_2 q + A_1},$$

параметры A_i связаны с коэффициентами характеристического полинома

$$Q_n(s) = s^n + a_n s^{n-1} + \dots + a_2 s + a_1$$

выражениями

$$A_1 = a_1 / \omega_o^n = 1;$$

$$A_2 = a_2 / \omega_o^{n-1}, \dots, A_n = a_n / \omega_o.$$

Для ПФ без нулей ($A_2 = 0$ и $A_3 = 0$ в числителе) задание параметров знаменателя A_i в виде коэффициентов бинома Ньютона обеспечивает равные (кратные) отрицательные вещественные полюса и переходную характеристику ЭМ без перерегулирования. Для получения минимального времени переходного процесса значения A_i должны обеспечивать комплексные полюса. Для ПФ с одним нулем ($A_3 = 0$ в числителе) рекомендуется задавать параметры знаменателя такими, чтобы полюса имели одинаковые отрицательные вещественные части и мнимые части, образующие арифметическую прогрессию. Для ПФ

с двумя нулями – геометрическую прогрессию. При этом влияние нулей ослабляется, и переходные процессы в ЭМ имеют перерегулирование не более 20 %.

Таблицы значений параметров для таких ПФ с порядком $n = 2 \div 10$ приведены в [2]. Масштаб по времени для нормируемых процессов определяется выбираемым параметром $\omega_o = \sqrt[n]{a_1}$.

По нормированным ПФ замкнутых систем (без нулей, с одним нулем и с двумя нулями) могут определяться эталонные ПФ разомкнутых систем с астатизмом первого, второго и третьего порядка соответственно:

$$W_{ЭМ}^{раз}(q) = \frac{W_{ЭМ}(q)}{1 - W_{ЭМ}(q)} = \frac{1}{q(q^{n-1} + A_n q^{n-2} + \dots + A_2)};$$

$$W_{ЭМ}^{раз}(q) = \frac{A_2 q + 1}{q^2(q^{n-2} + A_n q^{n-3} + \dots + A_3)};$$

$$W_{ЭМ}^{раз}(q) = \frac{A_3 q^2 + A_2 q + 1}{q^3(q^{n-3} + A_n q^{n-4} + \dots + A_4)}.$$

Таблицы значений параметров A_i для таких ПФ с порядком $n = 2 \div 10$ приведены в [2].

Однако значения нормированных параметров ЭМ, определяемые данными методами по условиям желаемого размещения полюсов, по совокупности других важных для САУ показателей могут быть неоптимальными, имеющими лишь рекомендательный характер.

Канонические формы представления ЭМ

Параметры нормированной и используемой при синтезе ПФ вида

$$W_{ЭМ}(q) = \frac{B_n q^{n-1} + B_{n-1} q^{n-2} + \dots + B_2 q + B_1}{q^n + A_n q^{n-1} + \dots + A_2 q + A_1}; \quad (1)$$

$$W_{ЭМ}(s) = \frac{b_n s^{n-1} + b_{n-1} s^{n-2} + \dots + b_2 s + b_1}{s^n + a_n s^{n-1} + \dots + a_2 s + a_1} \quad (2)$$

связаны следующими выражениями:

$$A_1 = a_1 / \omega_o^n = 1;$$

$$A_2 = a_2 / \omega_o^{n-1}, \dots, A_n = a_n / \omega_o;$$

$$B_1 = b_1 / \omega_o^n, B_2 = b_2 / \omega_o^{n-1}, \dots, B_n = a_n / \omega_o.$$

Более общими, чем операторная модель (1), являются векторно-матричные ЭМ в формах управляемости и наблюдаемости:

$$\dot{X}_c = A_c X_c + B_c G; y = C_c X_c; \quad (3)$$

$$\dot{X}_o = A_o X_o + B_o G; y = C_o X_o, \quad (4)$$

где X_c и X_o – векторы переменных состояния в данных канонических базах; y – скалярный выход ЭМ; G – вектор внешних воздействий; A_c, B_c, A_o, B_o – матрицы; C_c, C_o – вектор-строки нормируемых параметров канонических моделей.

В отличие от операторной модели с одним входом, вектор G в векторно-матричных ЭМ (3) и (4) включает в себя также воздействия по переменным состояниям, дающие возможность оценивать показатели инвариантности – ошибки при ступенчатых возмущениях для получения более полной информации о динамических возможностях ЭМ.

Нормированные ЭМ можно составлять с обеспечением требуемого порядка астатизма разомкнутой системы, необходимыми условиями которого являются: $B_1 = A_1 = 1$ – для астатизма первого порядка; $B_1 = A_1, B_2 = A_2$ – для астатизма второго порядка; $B_1 = A_1, B_2 = A_2, B_3 = A_3$ – для астатизма третьего порядка. В зависимости от учитываемых параметров B_i нормированные ЭМ могут быть без нулей, если только $B_1 \neq 0$; с одним нулем, если $B_1 \neq 0, B_2 \neq 0$; с двумя нулями, если $B_1 \neq 0, B_2 \neq 0, B_3 \neq 0$ и т. д.

Таким образом, количество нормируемых параметров в различных вариантах ЭМ n -го порядка не превышает $m = 2(n - 1)$.

Общая постановка задачи МК-нормирования

Рассматривается задача минимизации вектора частных критериев

$$F(\theta) = [f_1(\theta), \dots, f_L(\theta)]^T; f_j(\theta) \rightarrow \min_{\theta \in D^m}$$

в области значений нормируемых параметров $\theta = (A_2, \dots, A_n, B_2, \dots, B_n)^T$, заданной прямыми и критериальными ограничениями

$$D^m : \{f_j(\theta) \leq f_j^o\}, j = \overline{1, L}.$$

При этом векторный критерий может быть совокупностью интегральных квадратичных оценок

$$F_1(\theta) = \left[\frac{1}{N} \sum_{n=0}^N e_{\text{ЭМ}}^2(t_n), \frac{1}{N} \sum_{n=0}^N y_{\text{ЭМ}}^2(t_n) \right], \quad (5)$$

вычисляемых по переходной характеристике ЭМ произвольного вида, где $e_{\text{ЭМ}}(t_n) = g(t_n) - y_{\text{ЭМ}}(t_n)$ и $y_{\text{ЭМ}}(t_n)$ – ошибка и производная выхода ЭМ. Первый частный критерий косвенно оценивает быстроедействие ЭМ, а второй – ее колебательность.

Частными критериями могут быть прямые показатели качества – относительное время пере-

ходного процесса и перерегулирование:

$$F_2(\theta) = [\tau_{\text{п}}, \sigma]^T, \quad (6)$$

но они не могут быть определены в случаях сильно колебательной или расходящейся переходной характеристики ЭМ.

Векторный критерий может включать показатели инвариантности ЭМ – динамическую и статическую ошибки:

$$F_3(\theta) = [|e_{\text{дин}}^f|, |e_{\text{стат}}^f|]^T, \quad (7)$$

оцениваемые при действии ступенчатого внешнего возмущения, и точностные показатели ЭМ – добротности по скорости и ускорению:

$$F_4(\theta) = [K_v, K_a]^T, \quad (8)$$

вычисляемые по временной диаграмме при гармоническом входном воздействии.

Комбинации частных критериев (5)–(8) в векторном критерии должны соответствовать классам САУ, для которых нормируются ЭМ. Кроме того, к ЭМ может предъявляться требование положительности вещественной части ее АФЧХ $\text{Re}\{W_{\text{ЭМ}}(j\omega)\} > 0$ для обеспечения монотонности переходных процессов или требование соответствия показателей ЭМ техническим критериям, таким, как модульный и симметричный оптимальности, используемым при настройке контуров тока и скорости в электроприводных системах [3].

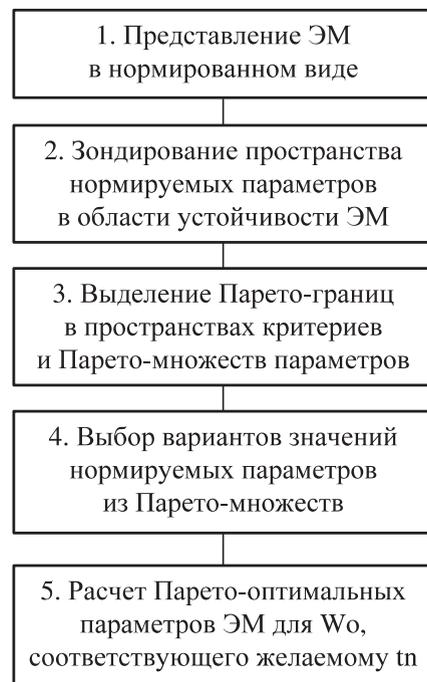


Рис. 1. Сценарий МК-нормирования

Определение. Значения вектора нормированных параметров $\theta^* \in R^m$ принадлежат m -мерной области эффективных (Парето-оптимальных) значений $P^m \in D^m \subset R^m$, где D^m – область устойчивости ЭМ, если их отображения $F(\theta^*)$ в пространстве частных критериев сформированного векторного критерия лежат в ε -окрестности L -мерной левой нижней границы отображения $F(D^m)$, которая является Парето-границей $F(P^m)$.

В случаях малости или невыпуклости получаемой Парето-границы, а также при задании

высоких приоритетов для отдельных частных критериев наряду с Парето-оптимальными могут выбираться и малоэффективные значения нормированных параметров, оптимальные по Слей-теру [6].

Поиск Парето-оптимальных параметров ЭМ

На рис. 1 представлена схема алгоритма-сценария проведения МК-нормирования.

На первом этапе выбирается вариант структуры ЭМ с требуемым для синтезируемой САУ

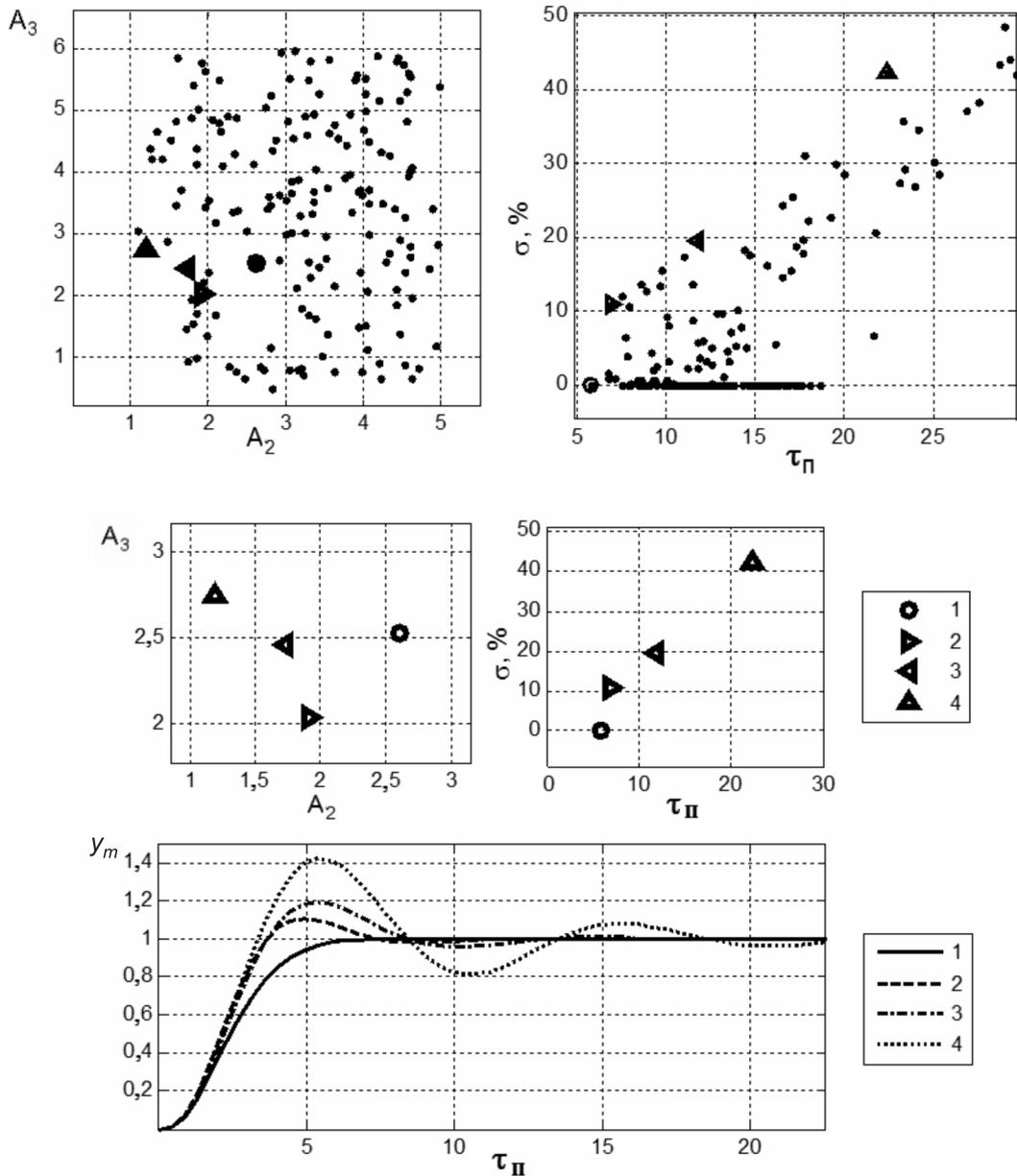


Рис. 2. Результаты нормирования параметров ЭМ без нулей



порядком астатизма и числом нулей. На втором этапе проводится прямое зондирование заданной ограниченной области пространства нормируемых параметров на регулярной или случайной m -мерной сетке значений параметров или целенаправленный поиск Парето-решений с использованием генетического алгоритма (ГА). На третьем в пространстве частных критериев выделяется окрестность Парето-границы и соответствующее ей множество близких к Парето-оптимальным значений нормируемых параметров. На четвертом из Парето-множества выбираются альтернативные варианты значений нормированных параметров, для которых на пятом этапе осуществляется расчет параметров ненормированной ЭМ с использованием параметра ω_0 , соответствующего желаемому времени переходного процесса в системе $t_n = \tau_n / \omega_0$.

При использовании ГА рекомендуется проводить поиск по *принципу двух векторных критериев*, предполагающему введение в рассмотрение:

- основного критерия $F_{\text{осн}}(\theta)$ невысокой раз-

мерности, формируемого из противоречивых частных критериев квадратичного вида, вычисляемых процедурой ГА при генерировании множества решений и локализации Парето-границы, соответствующей этому критерию;

- дополнительного критерия $F_{\text{доп}}(\theta)$, составляемого из непротиворечивых и неквадратичных частных критериев, также оцениваемых при зондировании, но используемых на четвертом этапе при отборе альтернативных вариантов Парето-решений, определяемых по основному критерию.

Выбор окончательных вариантов нормированных параметров производится по обоим критериям с учетом требований по синтезу САУ и предпочтений проектировщика (лица, принимающего решение).

Нормирование параметров ЭМ третьего порядка

Пример 1. Осуществим нормирование параметров A_2 и A_3 эталонной модели без нулей по векторному критерию (6).

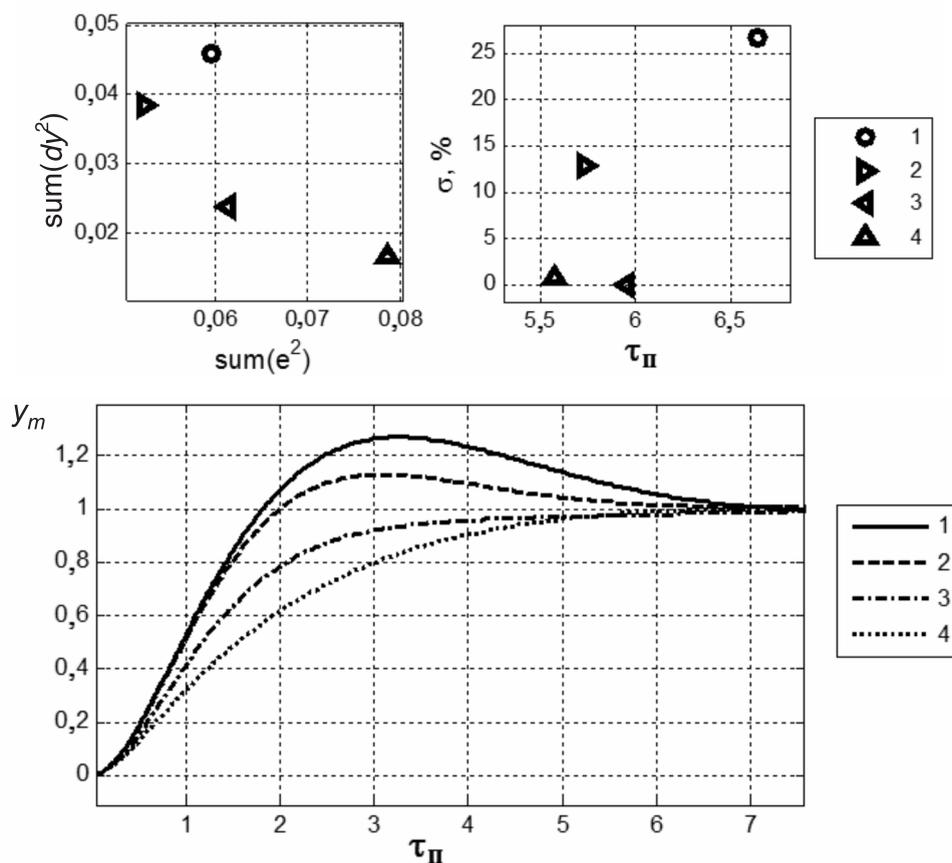


Рис. 3. Результаты нормирования параметров ЭМ с одним нулем

На рис. 2 представлены результаты прямого зондирования на случайной сетке параметров и приведены переходные характеристики для четырех выбранных вариантов их значений, из которых только первые два варианта являются Парето-оптимальными.

Пример 2. Проведем нормирование параметров A_2 , A_3 и B_2 эталонной модели с одним нулем, используя при зондировании по основному векторному критерию (5) процедуру ГА.

На рис. 3 приведены переходные характеристики для выбранных четырех вариантов значений нормированных параметров:

- 1) $A_2 = 2,3662$; $A_3 = 2,6192$; $B_2 = 2,1345$;
- 2) $A_2 = 2,7402$; $A_3 = 2,4690$; $B_2 = 2,0514$;
- 3) $A_2 = 3,3167$; $A_3 = 2,8835$; $B_2 = 1,8161$;
- 4) $A_2 = 3,7845$; $A_3 = 4,6922$; $B_2 = 1,9282$.

Последние три варианта являются Парето-оптимальными и по основному, и по дополнительному критерию (6). Однако при нормировании с учетом критериев (7) и (8) предпочтительнее первый вариант ЭМ с более высоким перерегулированием.

Предложенный метод позволяет нормировать параметры операторных и канонических векторно-матричных ЭМ замкнутых САУ высокого порядка с различной степенью астатизма. Его применение способствует повышению эффективности принятия решений в задачах алгебраического синтеза САУ с последовательными и модальными регуляторами по методикам, изложенным в [3].

Разработанный сценарий МК-нормирования по разным векторным критериям с применением процедур зондирования на сетках параметров или генетического алгоритма обеспечивает относительно невысокие затраты машинного времени при расчете таблиц значений нормированных параметров ЭМ замкнутых САУ.

По построенным предложенным методом эталонным моделям с Парето-оптимальными нормированными параметрами могут эффективнее решаться задачи синтеза систем стабилизации и следящих систем со статическими и астатическими ОУ и осуществляться их параметрическая оптимизация с учетом нелинейностей итерационными методами [4, 5].

СПИСОК ЛИТЕРАТУРЫ

1. Синтез регуляторов и теория оптимизации систем автоматического управления [Текст] / Под ред. Н.Д. Егупова / Методы классической и современной теории автоматического управления: Учеб. в 3 т.; Т. 2. –М.: Изд-во МГТУ им. Н.Э. Баумана, 2000.
2. **Ким, Д.П.** Линейные системы [Текст] / Д.П. Ким // Теория автоматического управления. Т. 1. –М.: Физматлит, 2007. –312 с.
3. **Козлов, В.Н.** Теория автоматического управления. Компьютерные технологии: Учеб. пособие [Текст] / В.Н. Козлов, Н.В. Ростов. –СПб.: Изд-во Политехнического ун-та, 2008. –332 с.
4. **Ростов, Н.В.** Параметрическая оптимизация цифровых модальных регуляторов [Текст] / Н.В. Ростов // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2010. –№ 3 –С. 39–45.
5. **Ростов, Н.В.** Синтез и компьютерная оптимизация цифровых последовательных регуляторов высокого порядка [Текст] / Н.В. Ростов // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2010. –№ 4. –С. 53–58.
6. **Черноруцкий, И.Г.** Методы оптимизации. Компьютерные технологии [Текст] / И.Г. Черноруцкий. –СПб.: БХВ-Петербург, 2011. –384 с.



УДК 002.53:004.89:005.6

Н.И. Лычагин, В.П. Вершинина, В.П. Дмитриев

УПРАВЛЕНИЕ ЗНАНИЯМИ В СИСТЕМЕ МЕНЕДЖМЕНТА КАЧЕСТВА ПРЕДПРИЯТИЯ – РАЗРАБОТЧИКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Понятие «управление знаниями» появилось в середине 90-х гг. прошлого века в крупных корпорациях, для которых проблемы обработки информации приобрели особую остроту и стали критическими. При этом стало очевидным, что основное узкое место – работа со знаниями (сохранение, поиск, тиражирование). Почти все предприятия обладают большими объемами формализованной и неформализованной информации, которая нередко рассредоточена. Проблема заключается в том, чтобы организовать доступ к ней, придав форму, удобную для использования.

К концу XX в. формируется понятие менеджмента качества, а затем и всеобщего управления качеством (TQM) как интегральной концепции менеджмента организации. В работах российских и зарубежных авторов прослеживается идея о необходимости систематической, целенаправленной и научно обоснованной деятельности по работе со знаниями и неразрывной связи этих действий с менеджментом качества [8–10].

В целях поддержки и улучшения организационных процессов создания, сохранения, извлечения, перемещения и применения знаний современные предприятия разрабатывают автоматизированные системы управления знаниями (СУЗ).

В данной статье предлагается вариант внедрения процесса управления знаниями в базовые процессы системы менеджмента качества предприятия. В качестве предприятия рассматривается организация – разработчик программного обеспечения (ПО) для телекоммуникационной отрасли путем создания СУЗ, основным ресурсом знаний которой является документация на ПО.

Система менеджмента качества предприятия – разработчика ПО

Скоординированная деятельность по общему управлению организации осуществляется

путем создания и внедрения систем менеджмента. Управление организацией как единой системой взаимосвязанных и взаимодействующих процессов систем менеджмента позволяет обоснованно выделять и перераспределять внутренние ресурсы, устанавливать и поддерживать надежные отношения с потребителями и поставщиками.

Ядром общей системы менеджмента организации является система менеджмента качества (СМК). Под СМК понимается совокупность взаимосвязанных и взаимодействующих процессов и необходимых ресурсов организации для достижения целей в области качества [2].

СМК современной организации строится на базовых принципах международных стандартов ИСО серии 9000. Основу СМК типового предприятия – разработчика ПО составляет иерархия процессов, охватывающих все жизненно важные виды деятельности предприятия и направленных на обеспечение удовлетворенности потребителей продукции (рис. 1). Под продукцией понимаются программные средства (ПС).

Первый уровень иерархии составляют общие процессы СМК – это процессы высшего уровня иерархии – организационные.

Второй уровень состоит из базовых (ключевых) процессов СМК – это процессы, реализующие модель жизненного цикла продукции (процессы, вошедшие в жизненный цикл производимой и поставляемой продукции и оказывающие влияние на ее качество), а также процессы постоянного улучшения продукции, процессов и СМК в целом.

Третий уровень содержит процессы поддержки СМК, обеспечивающие функционирование базовых процессов и решение конкретных задач функционирования СМК.

Технология поддержки жизненного цикла ПС на предприятии – разработчике ПО для телекоммуникационной отрасли, применяемая при работе с потребителями, базируется на основе

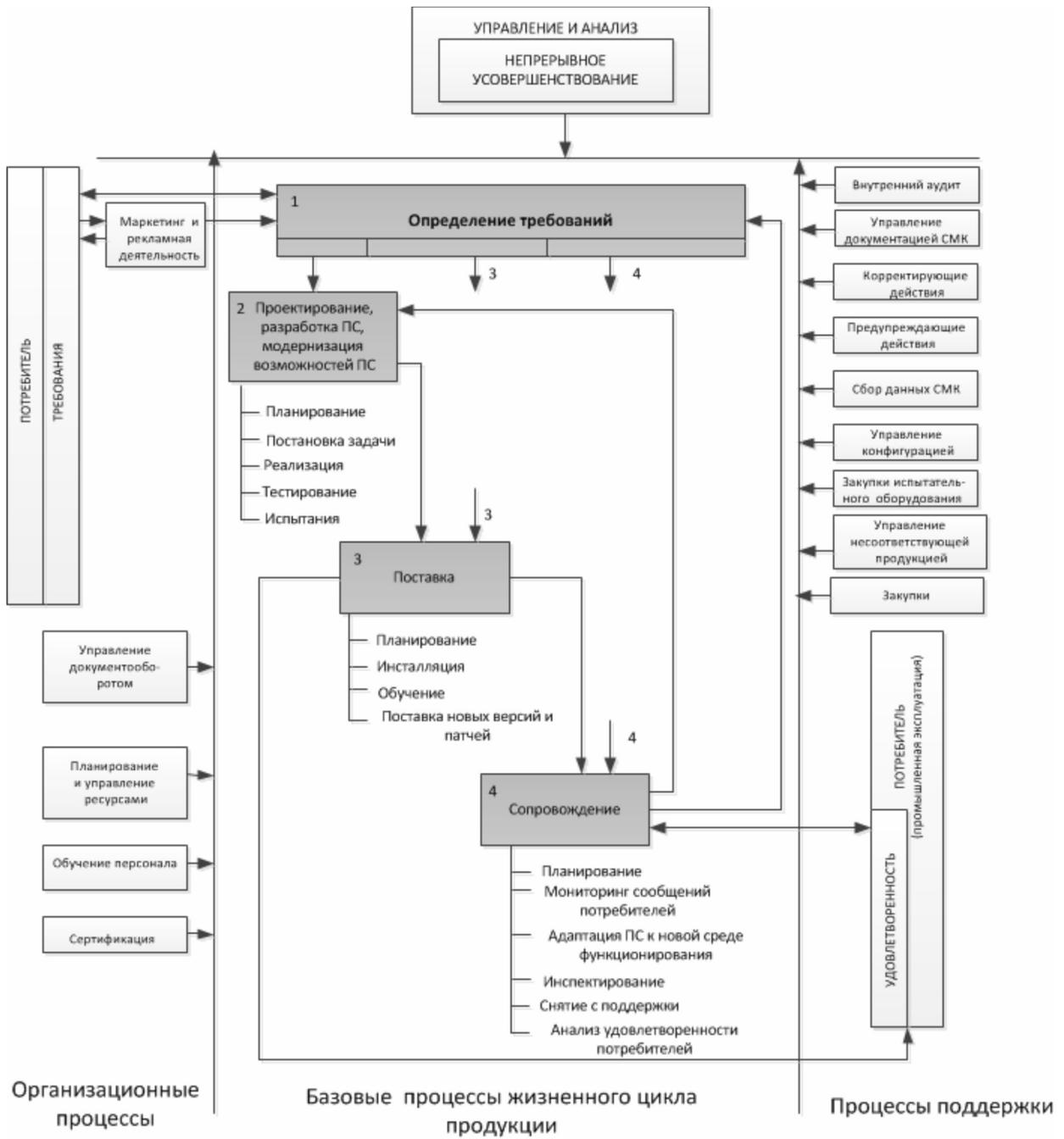


Рис. 1. Схема процессов СМК предприятия – разработчика ПО

ГОСТ Р ИСО/МЭК 12207 и ГОСТ 34.601 [1, 3]. Под жизненным циклом продукции понимается совокупность процессов, связанных с последовательным изменением состояния продукции от формирования исходных требований к нему до окончания его эксплуатации. Жизненный цикл состоит из стадий – логически завершенных ча-

стей. Стадии характеризуются определенным состоянием продукции, видом предусмотренных работ и их результатом.

Особое место среди процессов СМК занимают базовые процессы, связанные с проектированием, разработкой и сопровождением ПС, т. к. они являются наиболее трудоемкими и качественное их



выполнение во многом определяет способность организации выполнить требования, предъявляемые потребителем к поставляемой продукции.

Базовые процессы жизненного цикла ПС включают в себя четыре процесса:

- определение требований;
- проектирование и разработку компонент ПО и документации;
- поставку (инсталляцию);
- сопровождение.

Базовые процессы могут разделяться на набор подпроцессов (работ), которые будут выполняться в жизненном цикле разрабатываемой и поставляемой продукции.

Необходимость реализации конкретных работ, задач и условия выполнения процесса определяются владельцами процессов. Детализация процессов производится в стандартах, методологических инструкциях и руководящих документах по конкретным направлениям деятельности предприятия.

Основным ресурсом знаний является документация на ПО. Эти знания используются во всех основных процессах: при разработке проектной документации, разработке программных продуктов и документации, тестировании, сопровождении, внедрении и т. д.

Документация на ПО является органичной составной частью любого программного продукта (ПП). Программы могут стать ПП только в совокупности с комплексом документов, полностью соответствующих их содержанию и достаточных для его освоения, применения и изменения. Ошибки в документации не менее опасны, чем ошибки в структуре, интерфейсах, текстах программ. Поэтому к разработке, полноте, корректности и качеству документации необходимо такое же тщательное отношение, как к разработке и изменениям текстов программ и данных [7].

Документация на ПО позволяет полностью использовать всю функциональность, заложенную на этапах проектирования и разработки. Основные знания для работы с программами и их использования несет в себе техническая (эксплуатационная) документация.

Система управления знаниями

Концепция разрабатываемой СУЗ предпри-

ятия – разработчика ПО для телекоммуникационной отрасли состоит в комбинации двух типов СУЗ, основанных на прагматических и содержательных аспектах представления знаний, их извлечения и обработки [4, 6].

Ключевым компонентом СУЗ является подсистема поиска, обеспечивающая оперативный отбор и доставку релевантной информации по запросам. Поиск информации производится не в хранилище документов на ПО, а в онтологии документации на ПО. Используется комбинация синтаксического и семантического поиска, т. е. поиск выполняется в экземплярах онтологии с учетом их семантических свойств и связей. Результатом такого поиска являются знания, которые используются эксплуатирующим систему специалистом (сотрудником предприятия или заказчика).

В рамках данной концепции вся документация на ПО модифицируется с точки зрения компонентной структуры, выявляются минимальные информационные единицы учета – модули документации (МД). МД – дискретный фрагмент текста документа в электронном формате, который может содержать как изображения, так и текст с форматированием и гиперссылками. Вместо разработки и поддержания в актуальном состоянии цельных документов документаторы создают или изменяют МД.

Основными компонентами СУЗ являются:

- онтология инфокоммуникационной системы – используется для описания информационных ресурсов с учетом семантики обрабатываемой информации;
- поисковая подсистема – выполняет поиск информации по запросам, навигацию, рубрицирование и визуализацию экземпляров онтологии инфокоммуникационной системы;
- подсистема компоновки документов – выполняет формирование требуемых цельных документов в определенных форматах;
- подсистема разграничения доступа – обеспечивает разграничение доступа к информационным ресурсам;
- подсистема целостности онтологии – обеспечивает контроль целостности онтологии.

Продукт Microsoft SharePoint 2010 представляет собой технологическую платформу СУЗ.

Архитектура СУЗ и основные участники

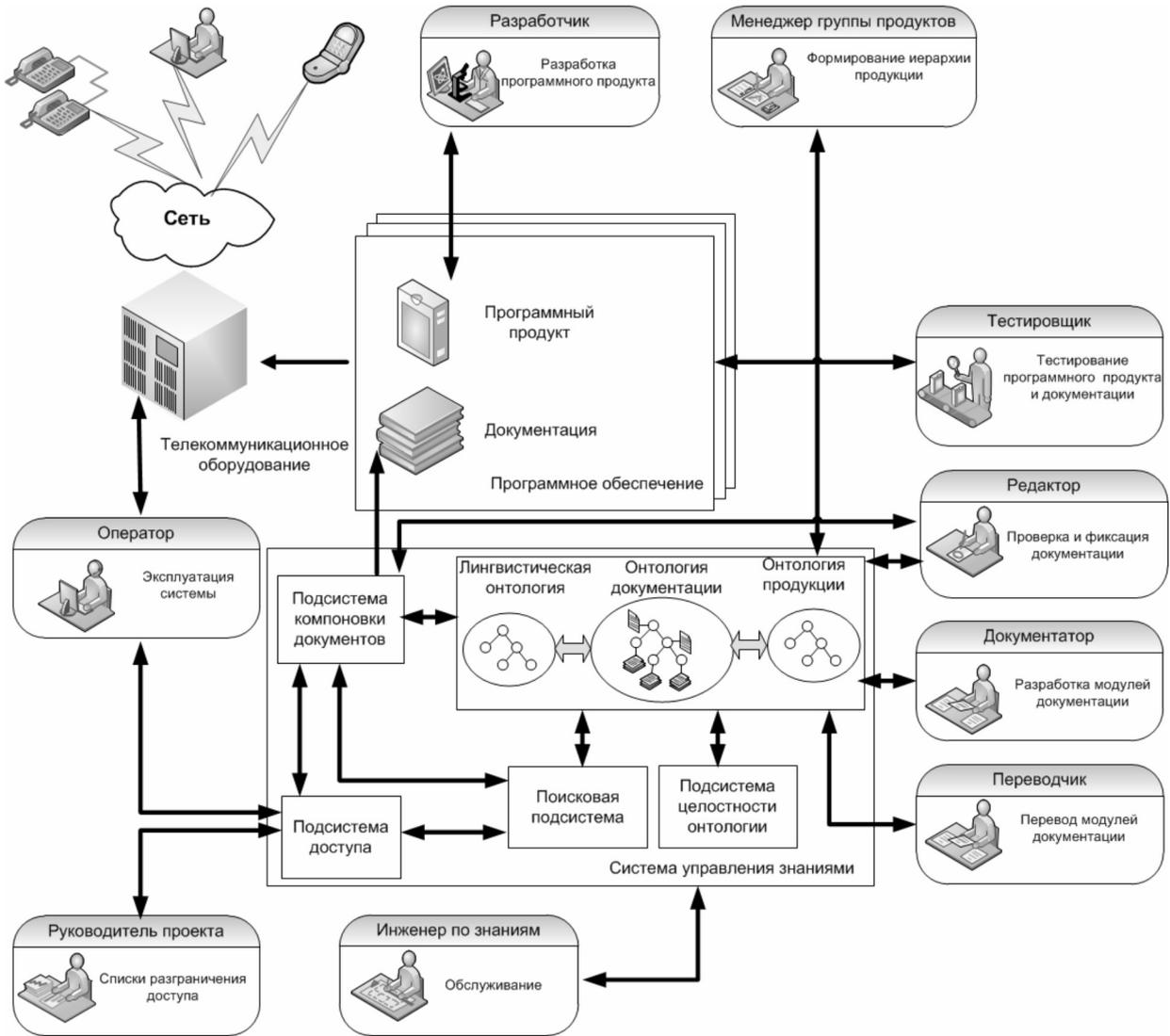


Рис. 2. Архитектура СУЗ и основные участники процесса управления знаниями

процесса управления знаниями представлены на рис. 2.

Онтология инфокоммуникационной системы разделена на функциональные составляющие: онтологию документации на программное обеспечение, лингвистическую онтологию, онтологию продукции [5].

В онтологию документации на ПО выделены концепты, относящиеся к документации на ПО. Для каждого концепта выделяется подмножество ключевых атрибутов, служащих для однозначной идентификации экземпляров. Онтология документации на ПО представляет собой библиотеку документов SharePoint. Для каждого концепта

(вида МД) онтологии документации на ПО создан тип контента (Content Type) со своим набором столбцов (атрибутов).

Понятия онтологии, предназначенные для поддержки решения задач информационного поиска, должны быть связаны со значениями терминов предметной области. Такого рода онтологию принято называть *лингвистической* или *словарной онтологией*. В основе лингвистической онтологии предметной области лежит карта бизнес-процессов модели eTOM (enhanced Telecom Operations Map), разработанной международной некоммерческой организацией TeleManagement Forum (TM Forum). Концепт лингви-



стической онтологии инфокоммуникационной системы рассматривается как пара – сигнатура (термин – функциональная возможность) из иерархии бизнес-процессов и множество ключевых слов, синонимов и сокращений к сигнатуре. Лингвистическая онтология реализована на основе набора терминов управляемых метаданных SharePoint.

Создание экземпляров лингвистической онтологии и формирование отношений между ними выполняется инженером по знаниям по запросу сотрудника группы документации. Наименование экземпляра функциональной возможности и ее место в иерархии определяется на этапе эскизного или технического проектирования, фиксируется в проектной документации.

В онтологию продукции выделены концепты, относящиеся к программным продуктам (например, подсистема, продукт, группа продуктов и т. д.). Онтология продукции реализована на основе списков SharePoint.

Создание экземпляров онтологии продукции и формирование отношений между ними выполняется менеджерами групп продуктов. Наименование экземпляра продукции и ее место в иерархии определяется на этапе эскизного или технического проектирования, фиксируется в проектной документации.

Компоненты программного обеспечения реализованы при помощи языков JavaScript, XML, XSLT, XPath, библиотек jQuery и AJAX, Web-служб Microsoft SharePoint 2010.

Рассмотренный подход используется при разработке СУЗ компании «Петер-Сервис» (Санкт-Петербург), которая является ведущим разработчиком решений для телекоммуникаци-

онной отрасли, специализируясь на разработке, внедрении и обслуживании OSS/BSS систем для крупных операторов связи. Документация на ПО содержит около 90 % всей информации о разрабатываемых в компании продуктах.

На сегодняшний день в онтологии документации на ПО выделено около 100 концептов и 7 типов отношений, создано около 22 000 экземпляров. Объем документации на ПО, переведенный на компонентную структуру, составляет примерно 40 %. Лингвистическая онтология включает один концепт – «Функциональная возможность», иерархия строится на основе отношения типа «класс-подкласс», создано около 600 экземпляров. В онтологии продукции выделено 6 концептов, таксономия строится на основе отношения типа «часть-целое» (включения), создано около 5000 экземпляров.

Внедрение СУЗ в базовые процессы СМК предприятия – разработчика ПО обеспечивает:

повышение качества документации на ПО (гипертекст, метайнформация, динамика);

повышение эффективности процесса управления знаниями, находящимися в документации на ПО;

повышение качества процесса поиска информации о поставляемой продукции примерно на 40–50 % за счет того, что в механизме поиска учитываются значения семантических свойств и связей МД;

активизирование обмена знаниями между сотрудниками;

уменьшение трудозатрат на разработку документации на ПО на примере справочников настроечных параметров на программные продукты примерно в 3–4 раза, т. к. такие документы автоматически генерируются на основе свойств и связей МД.

СПИСОК ЛИТЕРАТУРЫ

1. ГОСТ 34.601-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы стадии создания.
2. ГОСТ Р ИСО 9000-2008 Система менеджмента качества. Основные положения и словарь.
3. ГОСТ Р ИСО/МЭК 12207-99 Информационная технология. Процессы жизненного цикла программных средств.
4. **Дмитриев, П.И.** Моделирование знаний инфокоммуникационной системы на основе онтологического подхода [Текст] / П.И. Дмитриев, Л.П. Вершинина

// Тр. XIII Междунар. науч.-практич. конф. молодых ученых, студентов и аспирантов «Анализ и прогнозирование систем управления». –СПб.: ПГУПС, 2012. –С. 377–383.

5. **Дмитриев, П.И.** Онтологическая модель системы управления знаниями инфокоммуникационной системы [Текст] / П.И. Дмитриев // Телекоммуникации. –2011. –№ 12. –С. 8–13.

6. **Дмитриев, П.И.** Основные возможности Microsoft SharePoint 2010 для построения корпоративной системы управления знаниями [Текст] / П.И. Дми-

триев // Молодой ученый. –2011. –№4. –Т.1. –С. 86–88.

7. **Липаев, В.В.** Документирование сложных программных средств [Текст] / В.В. Липаев. –М.: СИНТЕГ, 2005. –216 с.

8. **Мирошников, В.В.** Система управления знаниями в области качества [Текст] / В.В. Мирошников, Д.И. Булатицкий // Информационные технологии. –2006. –№7. –С. 16–22.

9. **Попов, Э.В.** Управление знаниями [Текст] / Э.В. Попов [и др.] // Вестник РФФИ. –2004. –№ 4. –С. 5–29.

10. **Бабкин, А.В.** Модель национальной инновационной системы на основе экономики знаний [Текст] / А.В. Бабкин, Т.Ю. Хватова // Экономика и управление. –2010. –№ 12 (62). –С. 170–176.



Математическое моделирование: методы, алгоритмы, технологии

УДК 303.732.4

Ю.А. Голуб

МЕТОДЫ И МОДЕЛИ ИССЛЕДОВАНИЯ И РАЗРАБОТКИ АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ ИЗБИРАТЕЛЬНОГО РАСПРОСТРАНЕНИЯ ИНФОРМАЦИИ И ИНИЦИАТИВНОГО ОБСЛУЖИВАНИЯ

Наиболее распространенным классом информационных систем (ИС), предназначенных для обеспечения деятельности и управления организацией, являются в настоящий момент информационные системы, функционирующие в режиме регламентного информационного обслуживания. Однако на практике широкое распространение имеют задачи, требующие обеспечения без запроса со стороны пользователей, или задачи обеспечения информацией по произвольным запросам. Исследования в области информационного обеспечения подобных задач начались в 70-е гг. XX в. в связи с обеспечением руководителей и научных сотрудников научно-технической информацией. В результате были введены такие режимы информационного обеспечения, как избирательное распределение информации (ИРИ), дифференцированное обслуживание руководителей (ДОР), режим ретроспективного информационного обслуживания по произвольным запросам (ИОПЗ). Позднее был предложен термин «режим инициативного информационного обеспечения (ИИО)», который в настоящее время может быть применен к информационному обеспечению слабоструктурированных управленческих задач, обладающих сложной внутренней логикой и требующих большого количества разнообразной информации, поступающей из различных источников. Поэтому поставлена задача исследования и разработки методов и моделей создания автоматизированных информационных систем, обеспечивающих избирательное распространение информации, ретроспективный поиск по произвольным запросам и другие виды инициативного обслуживания пользователей.

На основе определения системы, реализующего системно-целевой подход [1], предложена многоуровневая модель структуры информационной системы (рис. 1).

Разработаны методы оценки взаимосвязей между стратами многоуровневой модели, базирующиеся на идее решающих матриц Г.С. Поспелова [2] и информационного подхода к анализу систем А.А. Денисова [6].

Четырехстратную структуру, представленную на этапе предварительных исследований, можно сократить до трехстратной, пропустив функциональную страту, которая практически отсутствует в существующих в настоящее время информационных системах вуза.

Для организации сложной экспертизы применим вначале метод решающих матриц (МРМ), предложенный Г.С. Поспеловым как средство стратифицированного расчленения проблемы с большой неопределенностью на подпроблемы и пошагового получения оценок.

С помощью метода решающих матриц можно оценить относительные информационные потребности, затем – компоненты информационной страты, обеспечивающие их, а затем – компоненты нижней страты, т. е. технические и программные средства, используемые для создания информационной страты [3].

Тогда в модели могут быть реализованы следующие страты (рис. 2, сверху вниз): информационные потребности с относительными оценками $\alpha_1, \alpha_2, \dots, \alpha_j, \dots, \alpha_{na}$; информационные массивы $\beta_1, \beta_2, \dots, \beta_p, \dots, \beta_{np}$; составляющие (компоненты) информационной супермагистрали $\gamma_1, \gamma_2, \dots, \gamma_k, \dots, \gamma_{ny}$.



Рис. 1. Представление информационной инфраструктуры

Каждая строка решающей матрицы характеризует относительную значимость компонентов нижележащего уровня для реализации вышестоящего. Вначале оценивается относительная значимость α_j компонентов верхнего уровня – информационных потребностей. Оценка проводится методом нормирования. Проверяется выполнение условия нормирования:

$$\sum_{j=1}^{n\alpha} \alpha_j = 1. \quad (1)$$

Затем оценивается a_{ij} – относительная значимость компонентами информационного слоя (баз данных, массивов научно-технической информации и т. п.) для реализации информационных потребностей пользователей.

Проверяется выполнение условия нормирования для каждой j :

$$\sum_i a_i = 1. \quad (2)$$

На основе α_j и a_{ij} вычисляется β_i – относительная значимость компонент информационной страты:

$$\beta_i = \sum_j a_{ij} \alpha_j. \quad (3)$$

Проверяется выполнение условия нормирования:

$$\sum_{i=1}^{n\beta} \beta_i = 1. \quad (4)$$

Далее оценивается b_{ki} – относительная значимость компонентов информационной суперма-

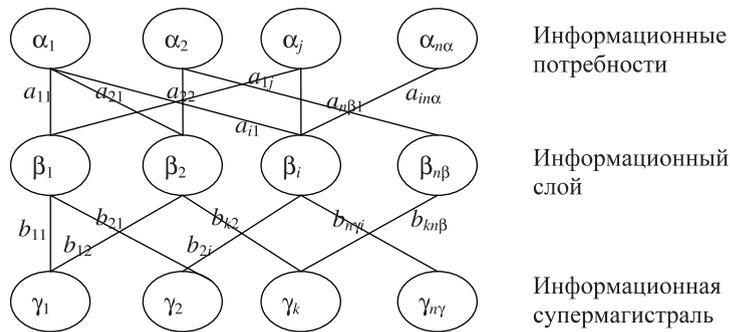


Рис. 2. Графическое представление модели решающих матриц

гистрала (т. е. информационных и программных средств) для реализации информационной страты.

Проверяется выполнение условия нормирования для каждой i :

$$\sum_k b_k = 1. \quad (5)$$

Вычисляется γ_k – относительные веса компонент информационной супермагистрала для реализации информационной страты:

$$\gamma_k = \sum_i b_{ki} \beta_i. \quad (6)$$

Проверяется выполнение условия нормирования $\sum_{k=1}^{n\gamma} \gamma_k = 1$.

При оценке технических и программных средств возможно формирование двух отдельных матриц и оценка значимости их компонентов для реализации информационной страты.

Более объективный анализ можно получить с помощью информационного подхода к разработке модели организации сложных экспертиз, базирующегося на использовании методов структуризации и информационных оценок степени соответствия исследуемых компонентов, что реализуется на основе многоуровневой оценки влияния компонентов каждой нижележащей страты на реализацию компонентов вышестоящей [4, 5].

В основу построения модели положена идея модели решающих матриц, а оценки между уровнями уточняются на основе информационного подхода А.А. Денисова, позволяющего приводить разнородные критерии (количественные и качественные) к единым информационным единицам, что помогает сопоставлять эти критерии и получать оценки потребности в различных видах услуг, используемых для проведения сравнительного анализа.

В качестве информационной меры, характеризующей значимость компонентов нижележащего уровня на вышестоящий условимся принять предложенную А.А. Денисовым меру потенциала H , которая позволяет учесть одновременно два критерия p и q :

$$H_i = -q_i \log(1 - p_i'), \quad (7)$$

где p_i' – степень влияния нижестоящего уровня на вышестоящий (изменяется в пределах от 0,7 до 0,99); q_i – вероятность выбора и реализации компонентов нижестоящего уровня с точки зрения лиц, принимающих решение, при их выборе на вышестоящем уровне.

В модели, приведенной на рис. 3, вводятся оценки p' и q между стратами. Оценка p' производится единичными экспертами, для которых определяются сферы компетентности.

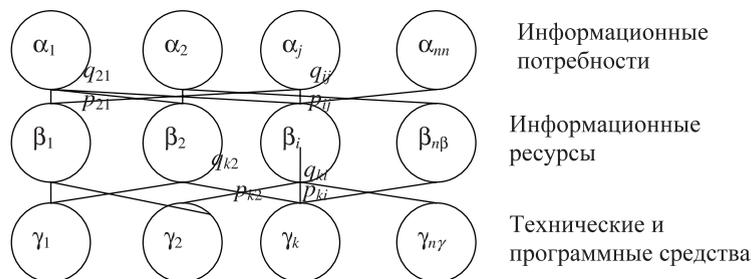


Рис. 3. Графическое представление модели, базирующейся на информационном подходе

Компоненты верхней страты могут быть оценены либо методом нормирования, либо для верхней страты информационная мера степени влияния информационных потребностей может быть оценена как степень влияния p' потребностей пользователей на реализацию целей вуза, т. е. без учета q :

$$H_{\alpha_j} = -\log(1 - p_{\alpha_j}). \quad (8)$$

Для получения значения α_j необходимо выполнить процедуру нормирования:

$$H_{\Sigma_{\alpha}} = \sum H_{\alpha_j}. \quad (9)$$

На рис. 2 приняты следующие обозначения:

$\alpha_j = \frac{H_{\alpha_j}}{H_{\Sigma_{\alpha}}}$ – относительная значимость компонентов верхнего уровня – информационных потребностей; H_{β} – информационная мера степени влияния информационного слоя (информационных баз) на реализацию информационных потребностей

$$H_{\beta_i} = -q_{ij} \log(1 - p_{ij}); \quad (10)$$

p_{β} – степень влияния компонент информационной страты (информационных баз) на реализацию информационных потребностей; q_{ij} – вероятность выбора и реализации компонент информационной страты с точки зрения лиц, принимающих решение при их выборе на вышестоящем уровне; H_{γ} – информационная мера степени влияния компонент информационной супермагистрали (технических средств) на реализацию информационных баз

$$H_{\gamma_{ki}} = -q_{ki} \log(1 - p_{ki}); \quad (11)$$

p_{γ} – степень влияния компонент информационной супермагистрали (технических средств) на реализацию информационных баз; q_{ki} – относительная значимость компонент информационной супермагистрали (технических средств) на реализацию информационных баз; β_i – относительная значимость компонент информационной страты

$$\beta_i = \sum a_{ij} \cdot \alpha_j; \quad (12)$$

$b_{ki} = \frac{H_{\gamma_{ki}}}{H_{\Sigma_{\gamma_{ki}}}}$ – относительная значимость компонент супермагистрали для реализации компонент информационной страты; γ_k – относительные веса компонент информационной супермагистрали

$$\gamma_k = \sum b_{ki} \cdot \beta_i. \quad (13)$$

Рассматриваемую модель можно представить

в обобщенном формализованном виде:

$$\begin{aligned} \beta_i &= \sum_j a_{ij} \alpha_j = \sum_j \left(\frac{H_{\beta_{ij}}}{H_{\Sigma_{\beta_{ij}}}} \cdot \frac{H_{\alpha_j}}{H_{\Sigma_{\alpha}}} \right) \\ \gamma_k &= \sum_i b_{ki} \beta_i = \sum_i \frac{H_{\gamma_{ki}}}{H_{\Sigma_{\gamma_{ki}}}} \left[\sum_j \left(\frac{H_{\beta_{ij}}}{H_{\Sigma_{\beta_{ij}}}} \cdot \frac{H_{\alpha_j}}{H_{\Sigma_{\alpha}}} \right) \right] \\ \gamma_k &= \sum_i \left[\frac{q_{ki} \log(1 - p_{ki}')}{\sum_i q_{ki} \log(1 - p_{ki}')} \times \right. \\ &\quad \left. \times \sum_j \left(\frac{q_{ij} \log(1 - p_{ij}')}{\sum_i q_{ij} \log(1 - p_{ij}')} \cdot \frac{\log(1 - p_{\alpha_j}')}{\sum_j \log(1 - p_{\alpha_j}')} \right) \right]. \end{aligned} \quad (14)$$

На основе проведенных исследований и оценки элементов многоуровневой структуры информационной системы вуза поставлена многокритериальная оптимизационная задача выбора компонент многоуровневой структуры информационной системы для обеспечения информационных потребностей определенных групп потребителей. Предложен подход к решению многокритериальной оптимизационной задачи, базирующийся на поэтапном решении однокритериальных оптимизационных задач по принятым критериям и обобщении результатов решения этих задач на основе их графического сопоставления и выбора на этой основе компонент страт многоуровневой модели, необходимых для удовлетворения принятым критериям.

Проведены экспериментальные исследования предложенных моделей на примере разработки многоуровневой структуры информационной системы вуза и многомерной конфигурации информационной системы районной Администрации.

В статье рассмотрена задача исследования и разработки методов и моделей создания автоматизированных информационных систем, обеспечивающих избирательное распространение информации, ретроспективный поиск по произвольным запросам и другие виды инициативного обслуживания пользователей.

На основе определения системы, реализующем системно-целевой подход, предложена многоуровневая модель структуры информационной системы.

Разработаны методы оценки взаимосвязей между стратами многоуровневой модели, базиру-

ющиеся на идее решающих матриц Г.С. Поспелова и информационного подхода к анализу систем А.А. Денисова.

Поставлена многокритериальная оптимизационная задача выбора компонентов многоуров-

невой структуры информационной системы для обеспечения информационных потребностей определенных групп потребителей.

Проведены экспериментальные исследования предложенных моделей.

СПИСОК ЛИТЕРАТУРЫ

1. **Волкова, В.Н.** Теория систем и системный анализ: Учебник [Текст] / В.Н. Волкова, А.А. Денисов. –М.: Изд-во «Юрайт», 2012. –579 с.

2. **Волкова, В.Н.** Методы организации сложных экспертиз: Учеб. пособие [Текст] / В.Н. Волкова, А.А. Денисов. –СПб.: Изд-во Политехн. ун-та, 2010. –128 с.

3. **Волкова, В.Н.** Автоматизированные информационные системы в высшей школе: история и перспективы [Текст] / В.Н. Волкова, Ю.А. Голуб. –СПб.: Изд-во Политехн. ун-та, 2011. –112 с.

4. **Волкова, В.Н.** Концепция многоуровневой ин-

формационной системы и ее реализация на примере вуза [Текст] / В.Н. Волкова, Ю.А. Голуб // Экономика, статистика и информатика. Вестник УМО. –2009. –№ 1-2. –С. 54–62.

5. **Волкова, В.Н.** Информационная система: к вопросу определения понятия [Текст] / В.Н. Волкова, Ю.А. Голуб // Прикладная информатика. –М.: Market. –2009. –№ 5 (23). –С. 112–120.

6. **Денисов, А.А.** Современные проблемы системного анализа: Учебник. [Текст] / А.А. Денисов. –3-е изд., перераб. и доп. –СПб.: Изд-во Политехн. ун-та, 2008. –304 с.

УДК 364

В.Н. Деркаченко, А.Ф. Зубков, Н.В. Ковалерова, М.А. Бармин

МАТЕМАТИЧЕСКИЕ МОДЕЛИ ПРОГНОЗИРОВАНИЯ ДИНАМИКИ РАЗВИТИЯ РЫНКА ЖИЛОЙ НЕДВИЖИМОСТИ

Недвижимость – важнейшая составная часть национального богатства, рынок недвижимости – необходимая составляющая любой национальной экономики. Он является предпосылкой существования всех других рынков: рынка труда, капитала, товаров и услуг.

Рынок жилой недвижимости оказывает большое воздействие на все стороны жизни и деятельности людей, выполняя ряд общих и специальных функций. Он находится под влиянием факторов, определяющих социально-экономическое развитие как страны в целом, так и отдельных регионов.

На практике для описания тенденции развития рынка недвижимости широко используются модели кривых роста, представляющие собой различные функции времени. При таком подходе изменение исследуемого показателя связывают лишь с течением времени и считается, что влияние других факторов несущественно или косвенно сказывается через фактор времени.

Разработка математических моделей прогнозирования и прогнозная оценка показателей – важная и актуальная задача.

В статье выполнен сравнительный анализ динамики цен на рынке жилой недвижимости в регионах Приволжского ФО и Пензенской области, разработаны математические модели и сделан прогноз ввода в действие жилых домов на 2012 г., а также получена зависимость стоимости квартир на первичном и вторичном рынках в г. Пенза. Статистический анализ и построение моделей проводились на основе информации Российского статистического ежегодника [1].

В качестве показателей динамики использовались абсолютный прирост, темп роста и темп прироста [2].

Поквартальная динамика изменения цен на первичном рынке жилой недвижимости Пензенской области за период с 2002 по 2009 г. и темпы прироста средней цены 1 м² показаны на рис. 1.

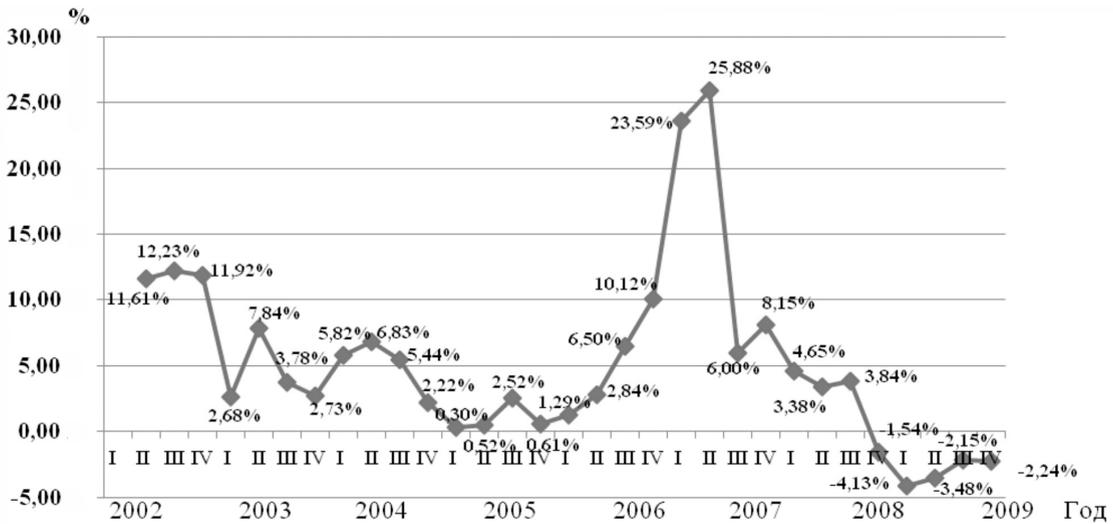


Рис. 1. Темп прироста (в % к предыдущему кварталу) средней цены 1 м² на рынке первичного жилья в 2002–2009 гг.

Сравнительная оценка динамики цен за квадратный метр на рынке первичной недвижимости Пензенской области с динамикой цен в регионах Приволжского ФО за 2002–2009 гг. показана на рис. 2. На рисунке представлены регионы, в которых на протяжении всего рассматриваемого периода наиболее часто фиксировались самые высокие (Нижегородская область) и самые низкие (Ульяновская и Саратовская области) цены на рынке первичного жилья.

Сравнительная оценка динамики цен за квадратный метр на рынке вторичной недвижимости Пензенской области с динамикой средней цены на вторичном рынке 1 м² по Приволжскому ФО также позволяет сделать вывод о том, что данный показатель по Пензенской области ниже в среднем на 7 % показателя по Приволжскому ФО за рассматриваемый период времени.

На рис. 3 приведена динамика цены квадратного метра на вторичном рынке недвижимости

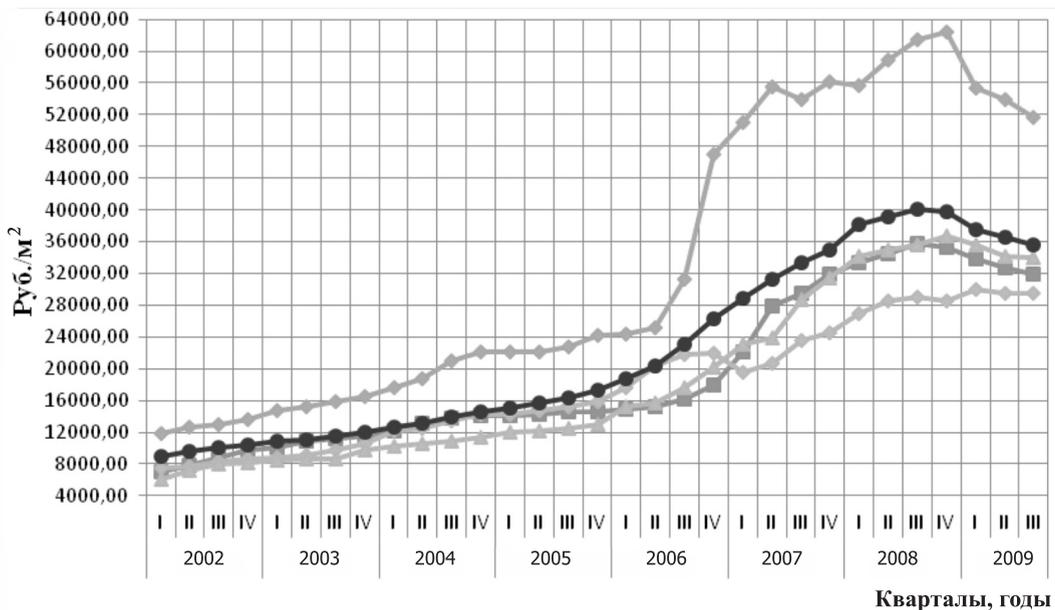


Рис. 2. Сравнительная динамика цен на первичном рынке недвижимости в регионах Приволжского ФО
(◆) Нижегородская область; (■) Пензенская область; (◇) Саратовская область;
(▲) Ульяновская область; (●) Средняя цена 1 м² по ПФО

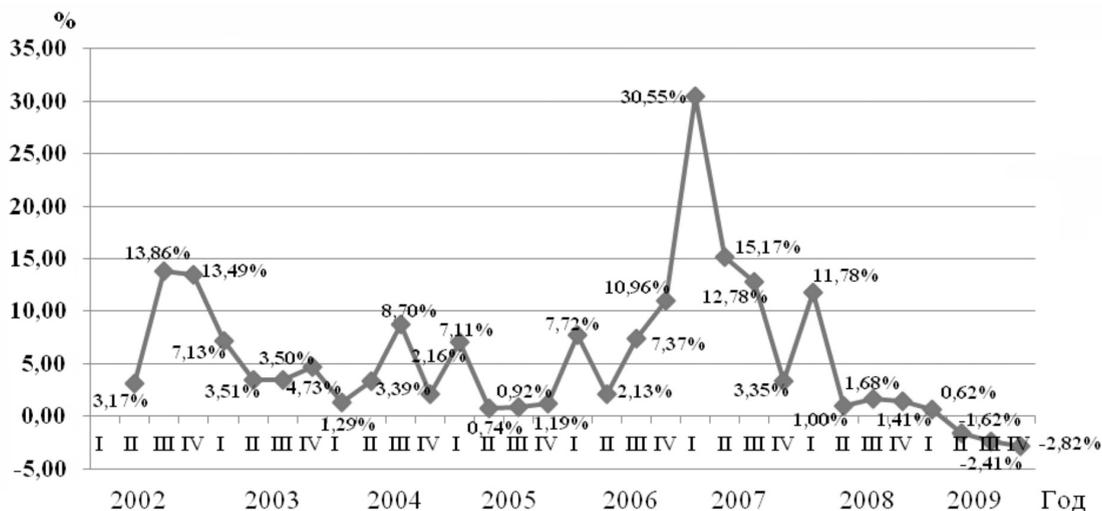


Рис. 3. Темп прироста (в % к предыдущему кварталу) средней цены 1 м² на рынке вторичного жилья в 2002–2009 гг.

Пензенской области в 2002–2009 гг.

Проведенный статистический анализ рынка жилой недвижимости Пензенской области за 2002–2009 гг. позволил сделать следующие выводы.

Положительный абсолютный прирост до IV квартала 2008 г. говорит о тенденции увеличения цены квадратного метра на рынке первичного жилья (исходя из динамики цен, в среднем на 1108,81 руб.), отрицательный абсолютный прирост, начиная с IV квартала 2008 г., говорит о снижении цены (в среднем на 920,99 руб.). Однако положительное значение показателя среднего абсолютного прироста за весь рассматриваемый период свидетельствует о том, что имела место тенденция ежеквартального увеличения цены квадратного метра на рынке новостроек в среднем на 782,26 руб. Значение среднего темпа прироста показывает, что в каждом квартале по сравнению с предыдущим цена квадратного метра увеличилась в среднем на 4,95 %.

В Пензенской области на всем рассматриваемом промежутке времени средняя цена за квадратный метр на рынке первичного жилья ниже аналогичного значения показателя в целом по Приволжскому ФО.

До II квартала 2009 г. на вторичном рынке жилья имела место тенденция увеличения цены за квадратный метр в среднем на 1098,17 руб. (об этом свидетельствуют положительные значения абсолютных приростов). В течение всего рассматриваемого периода времени цена за квадратный

метр на вторичном рынке жилья возрастала в среднем на 911 руб. ежеквартально или на 5,43 % (по отношению к предыдущему периоду). Рекордный рост цен на вторичном рынке жилой недвижимости отмечен в начале 2007 г., когда цена за квадратный метр увеличилась по сравнению с IV кварталом 2006 г. на 30,55 % или на 5587,48 руб. в денежном эквиваленте.

Цена квадратного метра на вторичном рынке жилой недвижимости в Пензенской области в 2002–2009 гг. была ниже, чем в большинстве субъектов Приволжского ФО.

На основе регрессионного анализа и программных средств [3] построены следующие модели:

линейная – $y = 14687,33 + 3485,14 \cdot t$, ($R^2 = 0,59$);

логарифмическая – $y = 14828,14 + 10995,66 \times \ln(t)$, ($R^2 = 0,74$);

обратная – $y = 36465,63 - 23462,1 \cdot \frac{1}{t}$, ($R^2 = 0,76$);

показательная – $y = 13838,2 \cdot 1,164^t$, ($R^2 = 0,63$);

модель роста – $y = e^{9,6+0,154t}$, ($R^2 = 0,63$);

экспоненциальная – $y = 13838,629 \cdot e^{0,154t}$, ($R^2 = 0,63$);

степенная – $y = 13838,6 \cdot t^{0,487}$, ($R^2 = 0,79$).

Для прогнозирования выбрана степенная модель, имеющая наибольшее значение индекса детерминации. Прогноз по модели на 2012 г. составил 38097,5 руб. за 1 м² на рынке первичного жилья.

Предпочтительной моделью для вторичного

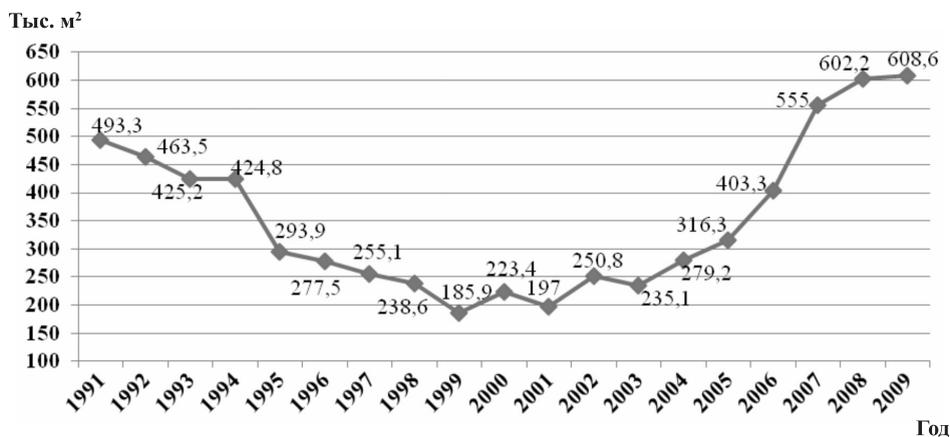


Рис. 4. Динамика ввода в действие жилых домов в Пензенской области

рынка жилья также является степенная модель:

$$y = 13418,9 \cdot t^{0,559}, (R^2 = 0,84).$$

Прогнозная оценка средней стоимости квадратного метра жилья на вторичном рынке Пензенской области в 2012 г. составила 42908,5 руб.

Для прогнозирования ввода жилья в Пензенской области разработана модель с переменной структурой:

$$y = 539,072 - 39,863 \cdot t - 907,167 \cdot F + 90,566 \cdot Ft,$$

где F – фиктивная переменная ($F = 0$ до 1999 г.; $F = 1$ с 2000 г., см. рис. 4).

Применение модели с переменной структурой оправдано, поскольку моделируемый временной ряд имеет нестабильную тенденцию. Прогнозное значение данного показателя на 2010 г., полученное по модели, составляет 645,97 тыс. м². При этом реальное значение показателя [1] – 625,0 тыс. м², и ошибка аппроксимации составила 3,3 %. На 2011 г. рассчитанное значение показателя составило 696,64 тыс. м², а прогножное значение ввода в действие жилых домов в Пензенской области в 2012 г. – 747,37 тыс. м².

Для оценки стоимости квартир построены зависимости цены y , руб. одно-, двух- и трехкомнатных квартир на первичном и вторичном рынках жилья г. Пензы от следующих факторов:

x_1 – общая площадь квартиры, м²;

x_2 – площадь кухни, м²;

$d_1 = 1$, если жилье расположено в центре города; 0, если – нет;

$d_2 = 1$, если дом кирпичный; 0, если – нет;

$d_3 = 1$, если не первый и не последний этажи; 0, если – нет.

По реальным статистическим данным получено шесть уравнений линейной многофакторной регрессии. Методом пошагового исключения незначимых факторов получили следующие уравнения со значимыми факторами:

- стоимость однокомнатной квартиры на первичном рынке

$$y = 56125,41 + 35877,1 \cdot x_1 (R^2 = 0,52; F = 20,2);$$

- стоимость двухкомнатной квартиры на первичном рынке

$$y = 1013664 + 17772,3 \cdot x_1 (R^2 = 0,43; F = 13,65);$$

- стоимость трехкомнатной квартиры на первичном рынке

$$y = 882765,2 + 20677,1 \cdot x_1 + 459404,1 \cdot d_1 (R^2 = 0,68; F = 18,62);$$

- стоимость однокомнатной квартиры на вторичном рынке

$$y = 592110,9 + 27454,7 \cdot x_1 + 224852,1 \cdot d_1 (R^2 = 0,86; F = 18,8);$$

- стоимость двухкомнатной квартиры на вторичном рынке

$$y = 571334,2 + 28392,9 \cdot x_1 + 227320,8 \cdot d_1 (R^2 = 0,75; F = 25,6);$$

- стоимость трехкомнатной квартиры на вторичном рынке

$$y = 660352,1 + 26249,8 \cdot x_1 + 521884,9 \cdot d_1 (R^2 = 0,78; F = 31,1).$$

Все полученные уравнения регрессии статистически значимы и могут использоваться для прогноза стоимости квартиры в зависимости от общей площади (x_1) и района (d_1). Другие рассмотренные факторы не оказывают существенного влияния на стоимость квартир в г. Пензе.

Район расположения жилья оказывает меньшее влияние на стоимость квартиры в случае,

если жилье первичное. Здесь вероятно влияние степени готовности квартиры для заселения и наличие необходимых для проживания бытовых условий.

Таким образом, в статье рассмотрена поквартальная динамика изменения цен на первичном и вторичном рынках жилой недвижимости Пензенской области за период с 2002 по 2009 г. и дана сравнительная оценка динамики цен на рынке недвижимости в регионах Приволжского ФО.

Построены математические модели зависимости цены за квадратный метр жилья в зависимости от времени, модель с переменной структурой ввода в эксплуатацию жилья в Пензенской области, получены прогнозные оценки на 2012 г. Методом пошагового регрессионного анализа выявлены факторы, оказывающие существенное влияние на стоимость квартир на первичном и вторичном рынках недвижимости. Результаты прогноза могут использоваться для принятия управленческих решений на региональном уровне.

СПИСОК ЛИТЕРАТУРЫ

1. Российский статистический ежегодник 2011: Стат. сб. [Текст] / Росстат. –М., 2011.
2. Статистика: Учебник [Текст] / Под ред. В.С. Мхитаряна. –М.: Экономистъ, 2006. –671с.
3. Деркаченко, В.Н. Методы социально-экономического прогнозирования: Учебник [Текст] / В.Н. Деркаченко, А.Ф. Зубков. –Пенза: Изд-во ПГТА, 2008. –С.156–184.

УДК 62-533.4

А.В. Кривошейкин, А.П. Молчанов, М.А. Чесноков

ДИНАМИЧЕСКАЯ ФИЛЬТРАЦИЯ ЭЛЕМЕНТОВ РЕЧЕВЫХ СИГНАЛОВ И СИГНАЛОВ С НЕПЕРИОДИЧЕСКИМ ЛИНЕЙЧАТЫМ СПЕКТРОМ

Постановка задачи

В системах радио- и гидролокации используются сигналы с периодическим спектром. Для оптимального приема таких сигналов применяются гребенчатые фильтры, выделяющие сигнал из аддитивной смеси с шумами [1]. При этом период спектра считается заранее известным и неизменным в процессе приема.

Речевой сигнал характеризуется присутствием в нем основного тона, т. е. сигнала с периодическим спектром, о наличии которого на приемной стороне заранее неизвестно, как неизвестен и его период. Поэтому приемник речевого сигнала должен решать две задачи: поиск сигнала основного тона и при его обнаружении реализацию оптимального приема сигнала.

В [2] предложен алгоритм выделения сигнала основного тона, определения его периода и оптимального приема на основе непрерывно подстраиваемого гребенчатого фильтра в условиях непрерывно изменяющегося периода основного тона. Реализация этого алгоритма при точной

настройке гребенчатого фильтра и при наличии всех гармоник – обертонов основного тона значительно увеличивает отношение сигнал/шум на выходе приемника.

Однако неизбежно существующая погрешность настройки, а также отсутствие некоторых обертонов в реальном сигнале и наличие шумов приводят при реализации алгоритма к подчеркиванию спектральных составляющих, которых нет в обрабатываемом исходном сигнале. Это существенно изменяет тембр голоса при воспроизведении звука и уменьшает его разборчивость.

В статье предложена и исследована система фильтрации аддитивной смеси шума и речевого сигнала с периодическим и непериодическим спектром.

Алгоритм работы системы фильтрации

Как известно из [3], слуховой аппарат здорового человека способен воспринимать речевую информацию при отрицательных уровнях отношения сигнал/шум, например мощность шума превосходит мощность речевого сигнала на

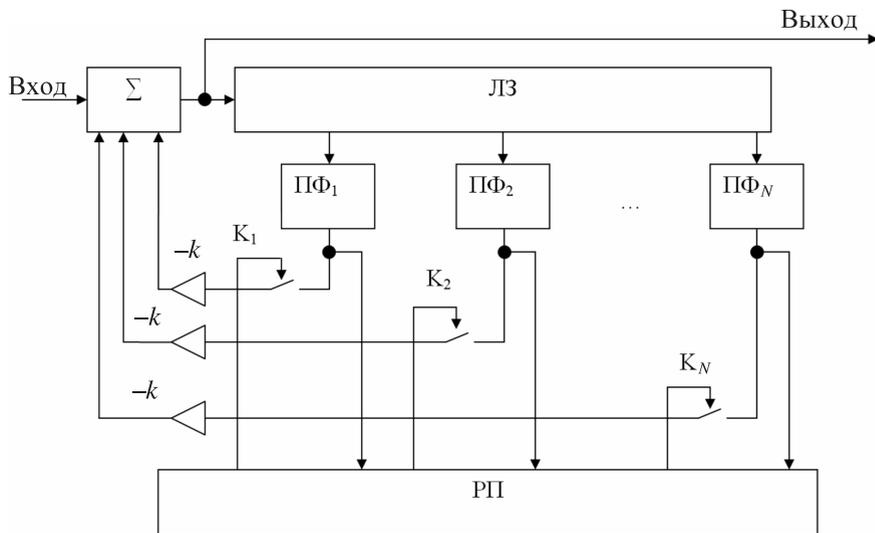


Рис. 1. Структурная схема системы фильтрации

10–12 дБ. Поэтому рассматривается принцип построения системы фильтрации, основанный на сочетании метода построения гребенчатого фильтра и модели Фланагана [4], описывающей процесс прохождения звукового сигнала в слуховом аппарате человека и использующей понятие формантной области частот [5]. В этой области находится некоторое количество гармоник основного тона, производимого голосовыми связками и усиленного вследствие резонанса в полости рта. Таким образом, область звуковых частот может быть разделена на формантные области, в которых достигаются максимумы спектральной характеристики звукового сигнала. На рис. 1 представлена структурная схема системы фильтрации.

Входной сигнал через сумматор поступает на линию задержки (ЛЗ), к отводам которой подключены каналные полосовые фильтры. Число отводов N равно числу формантных областей. Сигналы обратной связи каждого из каналов через ключи и

усилители поступают на входы каналов обратной связи сумматора. Длительность задержки на каждом отводе равна половине периода составляющей сигнала, частота которого совпадает с центральной частотой полосового фильтра. Выполнение этого условия, а также отрицательность значений коэффициентов передачи усилителей обеспечивает наличие положительной обратной связи (ОС), лежащее в основе работы гребенчатых фильтров.

Ключи управляются сигналами, вырабатываемыми блоком решающего правила (РП), задачей которого является определение номеров каналов, подлежащих включению в цепь положительной обратной связи, а также определение значений коэффициентов передачи усилителей в выбранных каналах. Из блока ПФ на каналные входы блока РП поступает N сигналов. В каждом из каналов определяется мощность сигнала $W(n)$, попавшего в полосу пропускания ПФ канала, где n – номер канала.

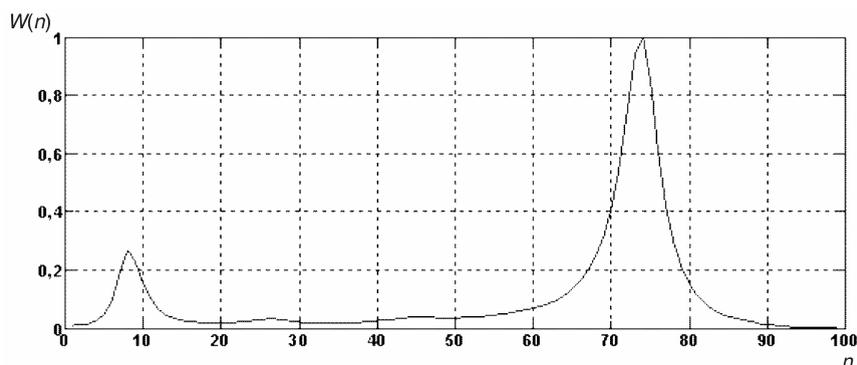


Рис. 2. Нормированное энергетическое распределение $W(n)$ между каналами для гласной «а»

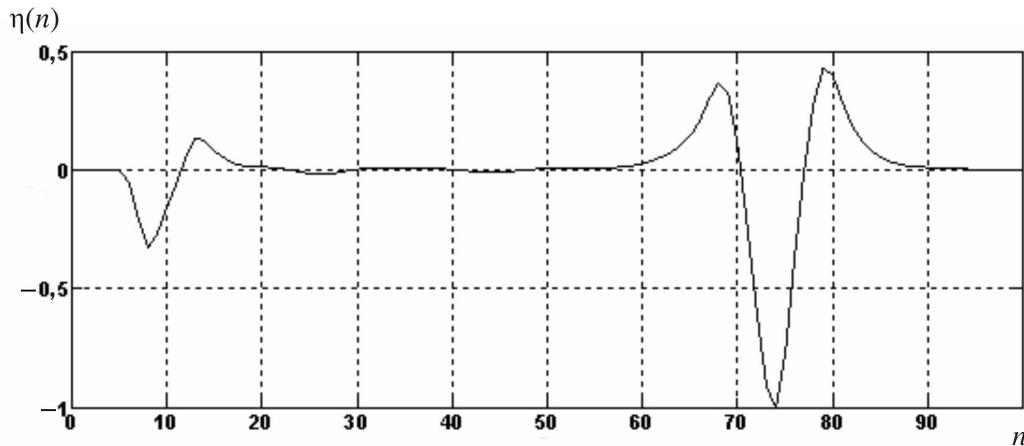


Рис. 3. Нормированное распределение $\eta(n)$ между каналами для гласной «а»

На рис. 2 в качестве примера представлена нормированная функция $W(n)$ распределения мощности сигнала по каналам для гласного звука «а», полученная в результате моделирования в среде MATLAB. В цепь ОС включаются только те каналы, которые соответствуют максимумам дискретной функции $W(n)$.

Рассмотрим алгоритм работы блока РП. Предполагая, что функция $W(n)$ может быть аппроксимирована функцией Гаусса, воспользуемся тем обстоятельством, что в точке максимума функции Гаусса ее вторая производная имеет минимум. Поэтому для поиска максимума дискретной функции $W(n)$ используется дискретная функция $\eta(n)$, являющаяся второй производной, вычисленной через вторые конечные разности функции $W(n)$:

$$\eta(n) = \frac{W(n^+) + W(n^-) - 2W(n)}{2 \cdot \Delta^2},$$

где n – дискретное значение аргумента, в котором вычисляется функция $\eta(n)$; Δ – шаг дискретизации; n^- , n^+ – соответственно предыдущее и последующее значения аргумента, отстоящие от значения аргумента n на один шаг дискретизации.

На рис. 3 представлено нормированное распределение $\eta(n)$ для звука гласной «а», полученное в результате моделирования в среде MATLAB.

Поиск аргумента n^* , доставляющего минимум функции $\eta(n)$, произведем простым перебором. В окрестности искомого значения n^* в процессе перебора аргумента n функция $\eta(n)$ переходит от положительного значения к отрицательному и далее после перехода точки n^* функция $\eta(n)$ изменяется от отрицательного значения к положи-

тельному. Таким образом, существуют точки n^- и n^+ , в которых функция $\eta(n)$ принимает близкие к нулю значения.

Не все номера каналов, т. е. значения частот, на которых расположены локальные максимумы в спектре звукового сигнала, соответствуют центральным частотам формантных областей. Они могут возникнуть за счет выделения шума голосовым аппаратом, имеющим максимумы на амплитудно-частотной характеристике. Спектральные максимумы, появившиеся в результате присутствия шума в исходном сигнале, будем считать ложными. Для исключения ложных максимумов используется разность между значениями n_k^+ и n_k^- и вводится понятие относительной ширины формантной области:

$$D = \frac{n_k}{n_k^+ - n_k^-}.$$

Значение параметра D должно быть сопоставлено с некоторым пороговым значением. Данное пороговое значение было получено экспериментально для фрагмента речи, гласного звука «а». Из ряда опытов была замечена корреляция между положением естественных, не ложных, максимумов распределения $W(n)$ и длины частотной полосы, занимаемой отрицательными значениями распределения $\eta(n)$ в окрестностях ее минимума. Пороговое значение зависит от возможной зашумленности сигнала, поэтому решение о его значении принимается из следующего условия: при определенном входном ОСШ должны быть выявлены алгоритмом первые две формантные области гласного звука «а», без появления ложных, что также определяет диапазон рабочих зна-

Таблица значений ОСШ выходного сигнала

Тип сигнала	Естественный речевой	Синтезированный
ОСШ вх. сигнала, дБ	ОСШ вых. сигнала, дБ	ОСШ вых. сигнала, дБ
-24	-21,7	-20,5
-18	-15,5	-13,2
-12	-5,7	-5,4
-6	1,4	1,7
0	7,1	7,5
6	13,6	14,1

чений ОСШ входного сигнала. Если рассчитанное значение D меньше порогового, то блок РП рассматривает максимум в спектре звукового сигнала как ложный, и соответствующий ему канал под номером n_k исключается из цепи обратной связи. Алгоритм переходит к оценке следующего максимума. Для тех максимумов, которые удовлетворяют требованиям к формантной области, блок РП замыкает ключи и включает соответствующие каналы в цепь ОС. Предложенный алгоритм заложен в основу математической модели системы фильтрации и реализован в среде MATLAB. Проведено компьютерное моделирование с использованием аудиофайлов формата wave.

Моделирование производилась для двух типов сигналов:

синтезированного входного сигнала в виде суммы двух синусоид с некрратными частотами и отстоящих друг от друга не менее чем на 50 Гц;

естественного, представленного вокализированными фрагментами человеческого голоса.

В обоих случаях использовалась модель с частотой дискретизации 32 кГц, в качестве по-

межи использовался белый шум. Производилась оценка относительного увеличения отношения сигнал/шум (ОСШ). Результаты моделирования приведены в таблице.

Как видно из таблицы, использование предложенной системы фильтрации при ОСШ на входе системы от -12 дБ и выше приводит к увеличению ОСШ приблизительно на 6 дБ на выходе системы.

Предложенные в статье структура устройства и алгоритм обработки речевого сигнала используются при выделении речевого сигнала из смеси с шумами. Учет особенностей обработки сигнала слуховым аппаратом человека позволяет увеличить отношение сигнал/шум даже в условиях, когда мощность сигнала значительно меньше мощности шума.

Результаты проведенного моделирования могут использоваться при построении вокодеров, в системах глубоководной связи водолазов при работе на грунте, а также при построении слуховых аппаратов.

СПИСОК ЛИТЕРАТУРЫ

1. **Финкельштейн, М.И.** Гребенчатые фильтры [Текст] / М.И. Финкельштейн. –М.: Сов. радио, 1969. –320 с.

2. **Бабкина, Л.Н.** Способ адаптивной фильтрации речевых сигналов в слуховых аппаратах [Текст] / Л.Н. Бабкина, А.П. Молчанов // Патент RU-2047946 от 10.11.95.

3. **Молчанов, А.П.** Электрические модели улитки

органа слуха [Текст] / А.П. Молчанов, Л.Н. Бабкина. –Л.: Наука, 1978. –181 с.

4. **Фланаган, Л.** Анализ, синтез и восприятие речи [Текст] / Л. Фланаган; пер. с англ. А.А. Пирогова. –М.: Связь, 1968. –397 с.

5. **Бондарко, Л.В.** Основы общей фонетики [Текст] / Л.В. Бондарко, Л.А. Вербицкая, М. Гордина. –4-е изд. –СПб.: Академия, 2004. –160 с.

УДК 681.5

С.Л. Морева, О.И. Золотов, А.Л. Ляшенко

РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ МОДЕЛИ ТЕМПЕРАТУРНЫХ ПОЛЕЙ АКТИВНОЙ ЗОНЫ РЕАКТОРА РБМК-1000

Высокий уровень надежности современных атомных станций – доказанный наукой и временем факт. Тем не менее, обеспечение безопасности является приоритетом ядерной промышленности, а ее совершенствованию уделяется самое пристальное внимание. Работа с высокими технологиями и огромная ответственность персонала атомных станций (в первую очередь перед населением) требуют, чтобы учитывались даже маловероятные риски возникновения аварии. Для этого проводится постоянный мониторинг состояния реакторной установки. Диагностирование физического состояния энергоблока позволяет выявить возможные нарушения функций безопасности, вызванные отказами оборудования и систем, оценить приоритетность восстановления этих функций и определить оптимальную последовательность действий оператора, направленных на обеспечение безопасности и перевод энергоблока в нормальный режим работы.

Описание объекта и постановка задачи

В качестве примера рассмотрим реактор АЭС РБМК-1000. Данный реактор является канальным реактором с графитовым замедлителем и водяным кипящим теплоносителем. В качестве топлива используется диоксид урана UO_2 с добавкой оксида эрбия Er_2O_3 . Активная зона представляет собой цилиндрическую кладку диаметром 11,8 и высотой 7 м, состоящую из вертикальных графитовых колонн [1].

Графитовая кладка цилиндрической формы состоит из вертикальных графитовых блоков сечением 250*250 мм. Блоки по вертикальной оси имеют сквозные отверстия диаметром 114 мм, предназначенные для размещения технологических каналов и каналов для контроля и управления. Часть технологического канала, размещенная в активной зоне, представляет собой циркониевую трубу диаметром 88 мм с толщиной стенки 4 мм. Для обеспечения теплового контакта с блоками кладки на трубу надеты графитовые кольца. Каждое чередующееся кольцо имеет не-

посредственный контакт по боковой поверхности либо с трубой, либо с внутренней поверхностью графитового блока, а также одно с другим по торцам. Это расположение определено из условия недопустимости заклинивания канала в кладке вследствие ее радиационно-термической усадки, а также термической и радиационной ползучести циркониевой трубы канала. В технологические каналы устанавливаются тепловыделяющие сборки (ТВС). Каждая ТВС содержит 18 твэлов (топливных стержней), дистанционированных специальными стальными решетками, и опирается на стальную концевую решетку. Твэлы диаметром 13,6 мм расположены по двум окружностям. На первом радиусе расположено 6 твэлов, на втором – 12.

Геометрическое расположение графитовой кладки, образующей вместе с топливными каналами активную зону, представлено на рис. 1.

Технологические каналы предназначены для размещения ТВС, дополнительных поглотителей и организации потока теплоносителя. Теплоноситель (вода) поступает в топливные каналы снизу, при температуре 270° С. Участок канала, на котором вода нагревается до температуры насыщения, имеет высоту около 2,5 м от низа активной зоны. На остальной ее части имеет место процесс развитого кипения. Расход теплоносителя по каждому топливному каналу может регулироваться независимо индивидуальным запорно-регулирующим клапаном.

Приблизительно 95 % энергии, выделяющейся в результате реакции деления ядер урана, передается теплоносителю, омывающему ТВС. Около 5 % мощности реактора выделяется в графите за счет замедления нейтронов и поглощения гамма-излучения. Эта доля тепла также передается теплоносителю через стенку технологического канала. Для снижения термического сопротивления, предотвращения окисления графита и улучшения теплопередачи от графита к технологическим и специальным каналам полость кладки заполнена циркулирующей газовой смесью гелия и азота, ко-

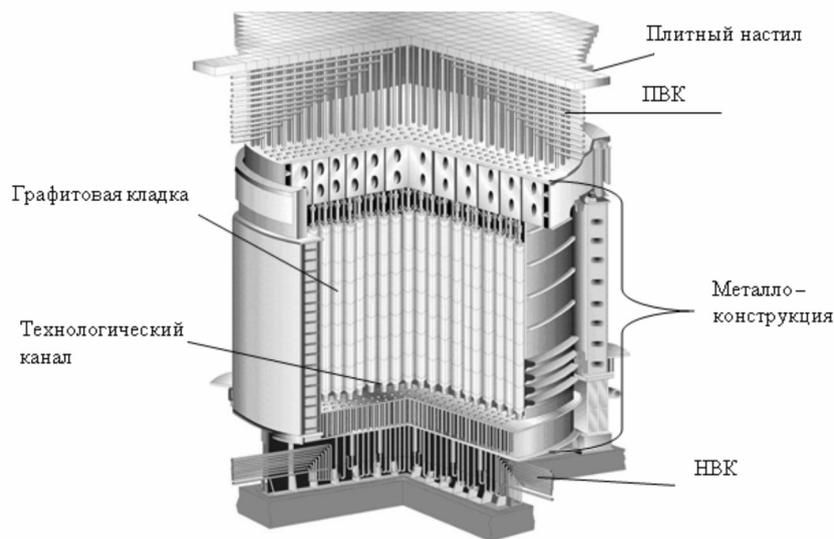


Рис. 1. Активная зона реактора

торая используется одновременно и для контроля целостности каналов путем измерения влажности и температуры газа [1].

Мониторинг параметров реакторной установки связан с необходимостью анализа непрерывных физических процессов, протекающих в активной зоне. С этой целью разработан ряд программных продуктов, рассмотренных в [2, 7]. Важная роль при исследовании ядерных установок отводится расчету тепловыделяющей системы, математической моделью которой является уравнение теплопроводности, выведенное на основе первого закона термодинамики [3].

Составим математическую модель температурных полей для графитовой кладки.

Разработка математической модели

При моделировании рассматривались тепловые процессы, протекающие в графитовых кладках, набранных из графитовых блоков. На рис. 2 показан фрагмент кладки в количестве девяти блоков. В технологическом канале располагается тепловыделяющая сборка, омываемая теплоносителем. Высота графитовой кладки 8330 мм. Поверхности ТВС будем рассматривать как функции входа, а контрольные точки во внутренних углах графитовых кладок – как функции выхода. На рисунке представлены рассматриваемые ячейки, ориентированные в горизонтальной плоскости осей пространственных координат X - Z , ось Y в данном случае является осью ординат.

Ниже представлен фрагмент математической модели, составленный для ячейки 5 (см. рис. 2).

Запишем дифференциальное уравнение теплопроводности применительно к графитовой кладке:

$$\frac{\partial T_{s1}(x, y, z, t)}{\partial t} = a_{\Gamma} \left[\frac{\partial^2 T_{s1}(x, y, z, t)}{\partial x^2} + \frac{\partial^2 T_{s1}(x, y, z, t)}{\partial y^2} + \frac{\partial^2 T_{s1}(x, y, z, t)}{\partial z^2} \right] + \frac{a_{\Gamma} W_{\Gamma}}{\lambda_{\Gamma} V_{\Gamma}}, \quad (1)$$

где a_{Γ} – коэффициент температуропроводности графита; x, y, z – пространственные координаты; W_{Γ} – мощность, выделяемая при торможении нейтронов; λ_{Γ} – коэффициент теплопроводности графита; V_{Γ} – объем графитовой кладки; $X2 < x < X3, Y1 < y < Y2, Z2 < z < Z3$.

При анализе тепловых процессов в технологических каналах, уравнение теплопроводности удобнее записывать в сферических координатах.

Запишем дифференциальное уравнение теплопроводности применительно к циркониевой трубе:

$$\frac{\partial T_{s2}(y, \varphi, R, t)}{\partial t} = a_{\text{ц}} \left[\frac{\partial^2 T_{s2}(y, \varphi, R, t)}{\partial R^2} + \frac{1}{R} \frac{\partial T_{s2}(y, \varphi, R, t)}{\partial R} + \frac{\partial^2 T_{s2}(y, \varphi, R, t)}{\partial y^2} \right], \quad (2)$$

где $a_{\text{ц}}$ – коэффициент температуропроводности циркония; $Y1 < y < Y2, R2 < R < R3$.

Запишем дифференциальное уравнение те-

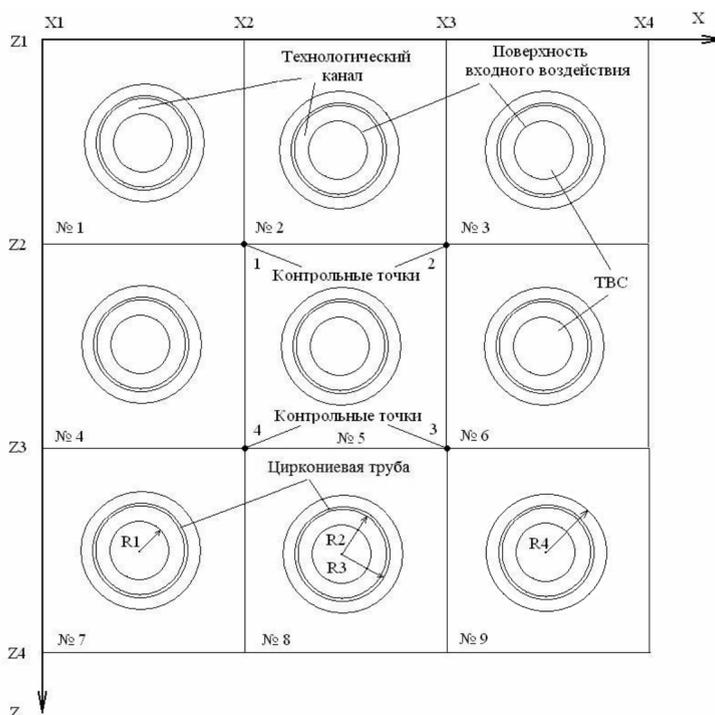


Рис. 2. Схема расположения рассматриваемых ячеек технологических каналов

плопроводности применительно к теплоносителю в канале [5]:

$$\frac{\partial T_{53}(y, \varphi, R, t)}{\partial t} = a_{\Gamma} \left[\frac{\partial^2 T_{53}(y, \varphi, R, t)}{\partial R^2} + \frac{1}{R} \frac{\partial T_{53}(y, \varphi, R, t)}{\partial R} + \frac{\partial^2 T_{53}(y, \varphi, R, t)}{\partial y^2} \right] - \vartheta(R, t) \cdot \frac{\partial T_{53}(y, \varphi, R, t)}{\partial \varphi}, \quad (3)$$

где ϑ – скорость движения теплоносителя в канале; a_{Γ} – коэффициент температуропроводности теплоносителя; $Y1 < y < Y2$, $R1 < R < R2$.

Функции входа U заданы в виде выражения:

$$U = T_{\text{ТВС}}(y, \varphi, R, t), \quad (4)$$

где $T_{\text{ТВС}}$ – температура на поверхности ТВС; $Y1 < y < Y2$, $\Phi1 < \varphi < \Phi2$, $R = R1$.

Численные значения температуры на поверхности ТВС будем искать из значений энергораспределения, полученных с помощью информационно-вычислительной системы (ИВС) «Скала-микро».

Аналогично составлены дифференциальные уравнения теплопроводности для остальных ячеек.

В разработанной математической модели присутствуют уравнения, записанные в прямоугольной и цилиндрической системе координат. Подробно вопрос стыковки данных уравнений рассмотрен в [4]. Для их стыковки выделены зоны цилиндрической формы с радиусом $R4$ в графитовых блоках, они совпадают с границами сквозных отверстий в графитовых блоках (см. рис. 2). Для этих зон составлено дополнительно дифференциальное уравнение теплопроводности.

Запишем дифференциальное уравнение теплопроводности применительно к выбранной ячейке графитовой кладки:

$$\frac{\partial T_{11}(y, \varphi, R, t)}{\partial t} = a_{\Gamma} \left[\frac{\partial^2 T_{11}(y, \varphi, R, t)}{\partial R^2} + \frac{1}{R} \frac{\partial T_{11}(y, \varphi, R, t)}{\partial R} + \frac{1}{R^2} \frac{\partial^2 T_{11}(y, \varphi, R, t)}{\partial \varphi^2} + \frac{\partial^2 T_{11}(y, \varphi, R, t)}{\partial y^2} \right] + \frac{a_{\Gamma} W_{\Gamma}}{\lambda_{\Gamma} V_{\Gamma}}; \quad (5)$$

$$Y1 < y < Y2, \Phi1 < \varphi < \Phi2, R3 < R < R4.$$

Данное дифференциальное уравнение отражает общий характер процесса. Для получения полной математической модели объекта, соответствующей конкретной задаче, необходимо за-

дание условий однозначности.

Граничные условия для графитовой кладки зададим соотношениями:

$$T_{11}(y, \varphi, R3, t) = T_{52}(y, \varphi, R3, t); \quad (6)$$

$$\lambda_{\Gamma} \frac{\partial T_{11}(y, \varphi, R3, t)}{\partial R} = \lambda_{\Pi} \frac{\partial T_{52}(y, \varphi, R3, t)}{\partial R}; \quad (7)$$

$$T_{51}(x, y, Z2, t) = T_{21}(x, y, Z2, t); \quad (8)$$

$$\lambda_{\Gamma} \frac{\partial T_{51}(x, y, Z2, t)}{\partial Z} = \lambda_{\Gamma} \frac{\partial T_{21}(x, y, Z2, t)}{\partial Z}; \quad (9)$$

$$T_{51}(X3, y, z, t) = T_{61}(X3, y, z, t); \quad (10)$$

$$\lambda_{\Gamma} \frac{\partial T_{51}(X3, y, z, t)}{\partial X} = \lambda_{\Gamma} \frac{\partial T_{61}(X3, y, z, t)}{\partial X}; \quad (11)$$

$$T_{51}(x, y, Z3, t) = T_{81}(x, y, Z3, t); \quad (12)$$

$$\lambda_{\Gamma} \frac{\partial T_{51}(x, y, Z3, t)}{\partial Z} = \lambda_{\Gamma} \frac{\partial T_{81}(x, y, Z3, t)}{\partial Z}; \quad (13)$$

$$T_{51}(X2, y, z, t) = T_{41}(X2, y, z, t); \quad (14)$$

$$\lambda_{\Gamma} \frac{\partial T_{51}(X2, y, z, t)}{\partial X} = \lambda_{\Gamma} \frac{\partial T_{41}(X2, y, z, t)}{\partial X}; \quad (15)$$

где $T_{21}, T_{61}, T_{81}, T_{41}$ – температура в графите ячеек 2, 6, 8 и 4 соответственно; λ_{Γ} – коэффициент теплопроводности графита; λ_{Π} – коэффициент теплопроводности циркония.

Граничные условия для циркониевой трубы зададим соотношениями:

$$T_{52}(y, \varphi, R3, t) = T_{51}(y, \varphi, R3, t); \quad (16)$$

$$\lambda_{\Pi} \frac{\partial T_{52}(y, \varphi, R3, t)}{\partial R} = \lambda_{\Gamma} \frac{\partial T_{51}(y, \varphi, R3, t)}{\partial R}. \quad (17)$$

$$T_{52}(y, \varphi, R2, t) = T_{53}(y, \varphi, R2, t); \quad (18)$$

$$\lambda_{\Pi} \frac{\partial T_{52}(y, \varphi, R2, t)}{\partial R} = \lambda_{\Gamma} \frac{\partial T_{53}(y, \varphi, R2, t)}{\partial R}. \quad (19)$$

Граничные условия для теплоносителя зададим соотношениями:

$$T_{53}(y, \varphi, R2, t) = T_{52}(y, \varphi, R2, t); \quad (20)$$

$$\lambda_{\Gamma} \frac{\partial T_{53}(y, \varphi, R2, t)}{\partial R} = \lambda_{\Pi} \frac{\partial T_{52}(y, \varphi, R2, t)}{\partial R}. \quad (21)$$

Данная математическая модель получилась сложной. Решить полученную систему дифференциальных уравнений в частных производных аналитически (выделить передаточную функцию) не представляется возможным. Данные дифференциальные уравнения решались методом «сеток» [6].

Для численного анализа рассматриваемого объекта управления были составлены дискретная

модель уравнений и вычислительный алгоритм. Упрощенная блок-схема вычислительного алгоритма решения дифференциальных уравнений представлена на рис. 3.

В процессе составления дискретных моделей были решены задачи «стыковки» граничных условий, обеспечения устойчивости вычислительной схемы и выбраны шаги дискретизации по пространственным переменным.

Для проверки адекватности математической

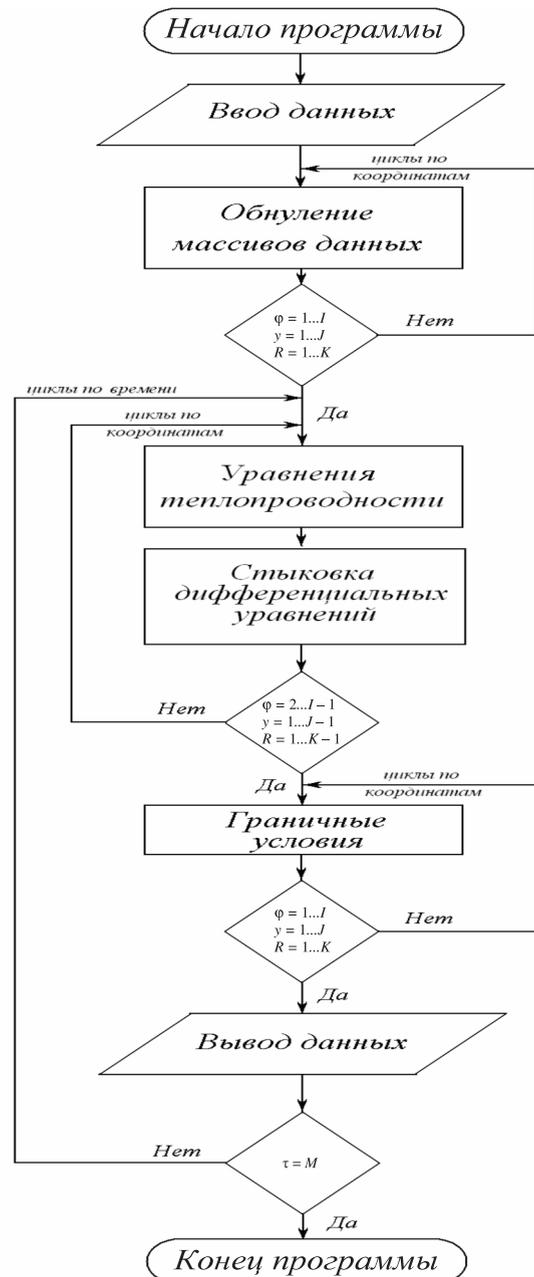


Рис. 3. Блок-схема вычислительного алгоритма

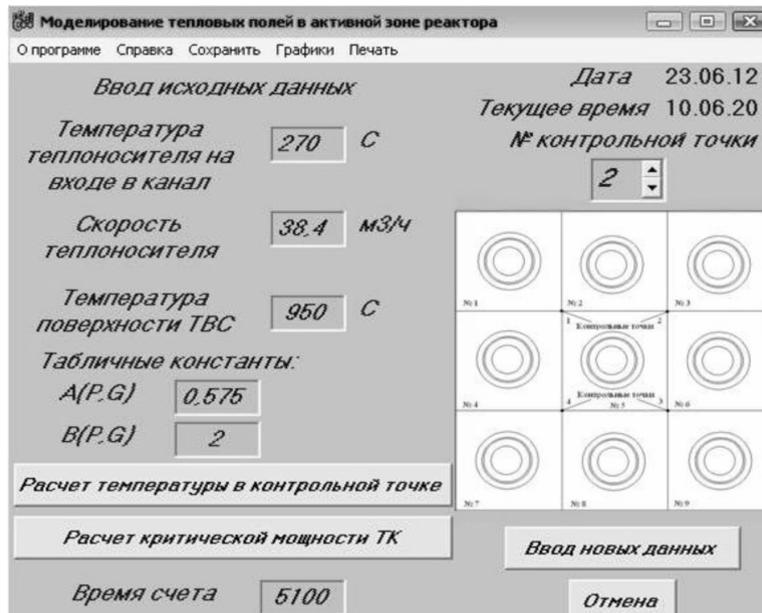


Рис. 4. Окно ввода данных

модели и компьютерного моделирования специально разработано программное обеспечение. В качестве системы программирования выбран язык C++ и среда для разработки Borland C++ Builder. Выбор языка программирования C++ обусловлен тем, что он применяется на практике во многих отраслях промышленности, в т. ч. и на АЭС.

На рис. 4 представлен общий интерфейс программы.

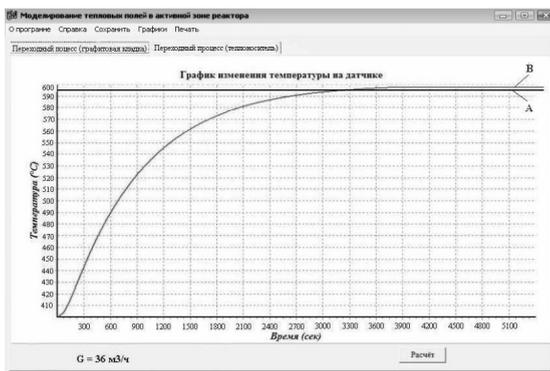
С помощью разработанного ПО рассчитаны значения температур в контрольных точках (рис. 5), соответствующих местам расположения термодатчиков, установленных в графитовой кладке на реакторе.

На рис. 5 графики *A* соответствуют данным, полученным с ИВС «Скала-микро», графики *B* соответствуют данным, полученным расчетным путем.

Результаты моделирования проверены с помощью информационно-вычислительной системы «Скала-микро», обеспечивающей контроль над работой реактора.

Значения, полученные путем моделирования, совпали с показаниями датчиков, установленных в графитовой кладке на реакторе. Полученные результаты свидетельствуют об адекватности разработанной математической модели.

a)



б)

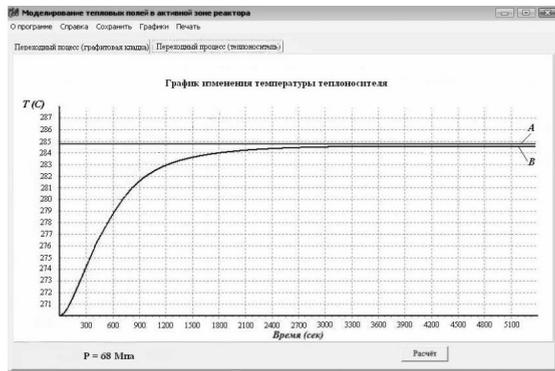


Рис. 5. Графики переходных процессов:

a – график переходного процесса в контрольной точке 2;

б – график переходного процесса в теплоносителе на выходе из канала

Результаты вычислительных экспериментов позволяют сделать вывод о том, что предложенное ПО точно моделирует тепловые процессы, протекающие в выделенном участке реактора, и может быть расширено с целью моделирования и мониторинга тепловых процессов во всем реакторе.

Разработанная методика моделирования позволит проводить диагностирование физического состояния энергоблока, выявлять возможные нарушения в его функционировании и прогнозировать возможные варианты работы при различных, в т. ч. и нештатных ситуациях.

СПИСОК ЛИТЕРАТУРЫ

1. **Абрамов, М.А.** Канальный ядерный энергетический реактор РБМК [Текст] / М.А. Абрамов, В.И. Авдеев, Е.О. Адамов; Под общ. ред. Ю.М. Черкашова. –М.: ГУП НИКИЭТ, 2006. –632с.

2. **Киселев, А.Е.** Моделирование аварий со снижением подачи теплоносителя для канала реактора РБМК-1000 [Текст] / А.Е. Киселев, А.В. Лемеха, В.Ф. Стрижов // Вопросы атомной науки и техники. Сер. Физика ядерных реакторов. –1998. –Вып. 1. –С. 48–65. –М.: Российский научный центр «Курчатовский институт».

3. **Лыков, А.В.** Теория теплопроводности [Текст] / А.В. Лыков. –М.: Высш. школа, 1967. –599 с.

4. **Морева, С.Л.** Математическое моделирование системы с распределенными параметрами на примере активной зоны реактора РБМК-1000 [Текст] / С.Л. Мо-

рева, А.Л. Ляшенко // Матер. IV Междунар. науч. конф. Системный синтез и прикладная синергетика. –Таганрог.: Изд-во ТТИ ЮФУ, 2011. –С. 218–225.

5. **Морева, С.Л.** Разработка программного комплекса для моделирования тепловых процессов в активной зоне реакторных установок [Текст] / С.Л. Морева, А.Л. Ляшенко // Народное хозяйство. Вопросы инновационного развития. –М.: МИИ Наука, 2012. –С. 243–247.

6. **Самарский, А.А.** Численные методы математической физики [Текст] / А.А. Самарский, А.В. Гулин. –М.: Научный мир, 2000. –316 с.

7. **Allison, C.M.** SCDAP/RELAP5/MOD2 Code Manual [Text] / C.M. Allison, E.C. Johnson [et al.] // EG&G Idaho, INC. Sept. 1989.

УДК 004.8

В.В. Дармахеёв

ЭФФЕКТИВНОСТЬ ПРИМЕНЕНИЯ РЯДА ГЕНЕТИЧЕСКИХ ОПЕРАТОРОВ СЕЛЕКЦИИ

Эволюционные вычисления для генерации моделей автоматов и преобразователей используются давно и успешно [6]. Однако при разработке генетических алгоритмов возникают проблемы. К одной из них относится трудность достижения сходимости алгоритма. Эта проблема тесно связана с выбором генетических операторов, включая операторы селекции. Настоящая статья посвящена вопросу выбора операторов селекции.

Генетический алгоритм разработан для решения прикладной задачи, заключающейся в генерации моделей преобразователей продукционных правил, представленных на естественном языке, в формулы логики предикатов первого порядка.

Данный генетический алгоритм (генератор) основан на следующих положениях [5]:

- автомат (особь) представляется в виде хромосомы;
- хромосома состоит из динамического набора молекул ДНК и имеет графовое представление, при этом одна молекула ДНК представляет собой описание одного состояния автомата;
- молекула ДНК состоит из фиксированного количества генов (минимальная неделимая составная часть молекулы ДНК);
- первая популяция $P(0)$ генерируется случайным образом, при этом значение каждого гена выбирается из соответствующей области допустимых значений.

Во время проведения вычислительных экспериментов была обнаружена проблема выбора особей, к которым впоследствии будут при-

меняться операторы скрещивания и мутации. Задача выбора особей для родительского пула, на основе которого будет формироваться следующее поколение, нетривиальна. В литературе встречается несколько различных подходов к формированию родительского пула [2–4]. Один из таких подходов [2] предполагает участие в скрещивании и мутации всех имеющихся в популяции особей. То есть с каждым шагом эволюции популяция увеличивается экспоненциально, что соответственно также экспоненциально увеличивает вычислительные затраты на каждом шаге эволюции. С точки зрения эффективности использования вычислительных ресурсов оптимальным является подход, при котором размер популяции на каждом шаге эволюции остается неизменным, т. е. при формировании следующего поколения (особи, к которым применяются операторы скрещивания и мутации) допускается только часть особей, отобранных из популяции с помощью операторов селекции [4].

Для решения этой задачи разработаны и программно реализованы следующие операторы селекции: пропорциональной; селекции колесом рулетки; турнирной.

Оператор пропорциональной селекции заключается в выборе из всей популяции только лучших особей, т. е. особей, обладающих наибольшей фитнес-функцией.

Как показали вычислительные эксперименты, применение пропорциональной селекции приводит к резкому уменьшению роста фитнес-функции на некотором этапе эволюции. Анализ полученных результатов привел к выводу, что это происходит из-за наполнения популяции особями, имеющими похожую структуру. Дело в том, что при такой селекции в родительский пул попадают только лучшие особи. Так как особи, полученные в результате скрещивания и мутации более приспособленных родителей, чаще всего имеют большую фитнес-функцию, чем потомки менее приспособленных особей, родительский пул на каждом шаге эволюции занимают все больше особей, имеющих похожую структуру (особей с общими предками). Пока популяция разнообразна, фитнес-функция растет очень быстро благодаря отбору только лучших особей. Уменьшение разнообразия популяции с каждым шагом эволюции, в конечном счете, приводит к тому, что потенциал роста быстро исчерпывается. Это связано с тем, что операции скрещивания между похожими

особями, как правило, незначительно изменяют значение фитнес-функции.

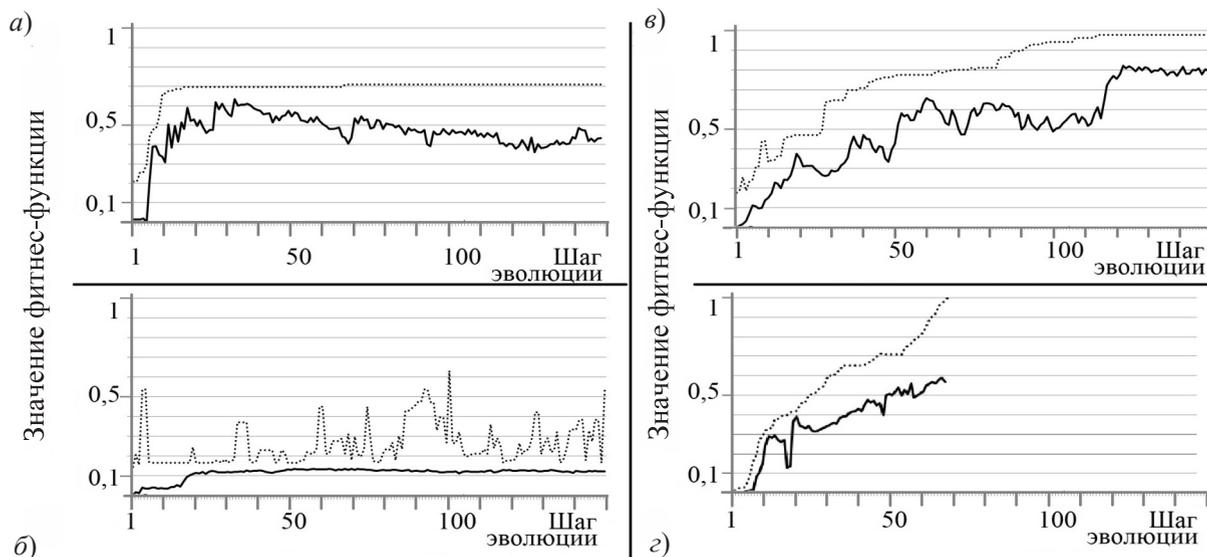
Селекция колесом рулетки заключается в произвольном выборе из популяции особей для родительского пула.

При такой селекции на каждом новом шаге эволюции популяция обладает достаточным разнообразием особей для эффективного проведения операции скрещивания. Об этом говорит тот факт, что значение фитнес-функции лучшей особи родительского пула почти всегда меньше фитнес-функции лучшей особи полученной популяции. Однако случайный выбор особей часто приводит к непопаданию в родительский пул особей, обладающих лучшей фитнес-функцией, поэтому изменение значения фитнес-функции лучшей особи в процессе эволюции носит скачкообразный характер. Хотя этот метод селекции в целом ведет к увеличению роста среднего значения фитнес-функции популяции, прирост фитнес-функции на каждом шаге эволюции является незначительным и может быть нивелирован неудачным выбором родительского пула. Поэтому процесс эволюции может продолжаться очень долго и часто не приводит к положительным результатам.

Оператор турнирной селекции заключается в следующем. Из популяции случайным образом выбирается некоторое количество особей, из которого в родительский пул попадает только особь, имеющая наибольшую фитнес-функцию. Этот процесс повторяется до тех пор, пока родительский пул не будет заполнен. Данный метод селекции является комбинацией селекции колесом рулетки и пропорциональной селекции.

Результаты экспериментов с этим типом селекции показали, что, к сожалению, данный оператор селекции обладает как плюсами обоих методов, так и их недостатками. Популяция, полученная из родительского пула, заполненного с помощью этого оператора селекции, обладает достаточным разнообразием особей. При этом попадание наиболее приспособленных особей в родительский пул необязательно. Поэтому прирост лучшей фитнес-функции может иметь скачкообразный характер. При этом данный метод селекции наиболее затратный с точки зрения требуемых вычислительных ресурсов.

Так как ни один оператор селекции в отдельности по результатам экспериментов не показал стабильного получения исправных преобразова-



Результаты испытаний операторов селекции:

a – пропорциональная селекция; *б* – колесо рулетки; *в* – турнирная селекция; *г* – ряд операторов селекции в равных долях
(.....) лучшая фитнес-функция; (—) средняя фитнес-функция

телей, то, на наш взгляд, для выборки родительского пула необходимо провести исследования по совместному применению всех рассмотренных операторов селекции [1].

Действительно, каждый из рассмотренных операторов имеет свои преимущества и недостатки, поэтому, пробуя нивелировать недостатки и использовать достоинства каждого оператора, мы исследовали возможности комбинированного применения всех трех операторов. Первые вычислительные эксперименты с применением такого подхода показали его эффективность. Первоначально родительский пул был разбит на три равные части, в каждую часть селекция производилась одним оператором.

Результаты вычислительных экспериментов с применением различных операторов селекции представлены на рисунке. Такой подход впервые позволил добиться приемлемой сходимости алгоритма. Поэтому следующим этапом стало проведение вычислительных экспериментов по определению оптимального сочетания операторов селекции.

При этом основная проблема – точная оценка эффективности применения операторов селекции. В случае применения операторов селекции по одному об эффективности селекции можно сделать однозначный вывод только на основе анализа скорости изменения среднего значения

фитнес-функции популяции. Однако такой способ оценки не дает однозначных результатов эффективности применения ряда операторов селекции. Логично при расчете эффективности оценивать и разнообразие полученного родительского пула.

Таким образом, расчет эффективности применения ряда операторов селекции необходимо осуществлять по двум критериям: скорости изменения среднего значения фитнес-функции популяции и разнообразию полученного родительского пула.

Одной из основных характеристик разнообразия популяции является распределение значений фитнес-функции особей популяции. При этом максимально возможный прирост фитнес-функции лучшей особи достигается в том случае, если в родительский пул помещаются все имеющиеся особи. Исходя из этого, наиболее эффективной селекцией можно считать ту, которая показала наименьшее изменение распределения значений фитнес-функции. Таким образом, в качестве оценки эффективности селекции была выбрана ошибка репрезентативности [7] распределения фитнес-функции родительского пула относительно всей популяции. Данный подход широко применяется при оценке репрезентативности выборки респондентов социологических опросов. Он отличается простотой и достоверностью. Ошибку репрезентативности можно вычис-

лить, используя среднееквадратичное отклонение значений фитнес-функции особей родительского пула от всей популяции:

$$\delta = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2},$$

где x_i – значение фитнес-функции i особи из родительского пула; n – количество особей в родительском пуле; \bar{x} – среднее значение фитнес-функции популяции на данном шаге эволюции.

Тогда в качестве ошибки репрезентативности можно взять среднее значение на всей эволюции среднееквадратичного отклонения значений фитнес-функции особей родительского пула от всей популяции:

$$E = \frac{1}{n} \sum_{i=1}^n \delta_i,$$

где δ_i – среднееквадратичное отклонение значений фитнес-функции особей родительского пула от всей популяции на i шаге эволюции; n – количество шагов эволюции.

В результате вычислительных экспериментов с применением ряда операторов селекции в генетический алгоритм выбран следующий оператор селекции: 5 % особей родительского пула отбирается с помощью оператора пропорциональной селекции; 65 % – с помощью турнирной селекции; 30 % – с помощью селекции колесом рулетки. При выбранных пропорциях достигнуто лучшее сочетание прироста фитнес-функции и ошибки

репрезентативности. Некоторые результаты вычислительных экспериментов по выбору пропорций ряда операторов селекции представлены в таблице.

Вычислительные эксперименты показали, что ряд операторов селекции, показавший наибольшее разнообразие популяции, имеет более быстрый рост фитнес-функции. При этом генетический алгоритм обладает близкой к 100 % сходимостью.

В результате проведения вычислительных экспериментов по применению генетических операторов селекции сделан вывод об эффективности совместного применения различных операторов селекции. Действительно, ряд операторов селекции позволяет свести к минимуму недостатки каждого из рассмотренных операторов, при этом такая селекция обладает плюсами каждого из рассмотренных операторов. Благодаря совместному применению операторов селекции удалось достичь почти пятикратного увеличения скорости работы генетического алгоритма относительно применения операторов селекции по отдельности.

В заключение хочется сделать вывод об оценке эффективности операторов селекции. Безусловно, главным критерием эффективности операторов селекции является время, затраченное генетическим алгоритмом на поиск решения.

Результаты испытаний ряда операторов селекции

Операторы селекции, %	Средний прирост фитнес-функции лучшей особи	Средний прирост фитнес-функции всей популяции	Ошибка репрезентативности
Проп. селекции – 33,3 Тур. селекции – 33,3 Колесо рулетки – 33,3	0,016431083	0,009753	0,12281772
Проп. селекции – 80 Тур. селекции – 10 Колесо рулетки – 10	0,09292	0,026303	0,1157645
Проп. селекции – 10 Тур. селекции – 80 Колесо рулетки – 10	0,052783613	0,024688	0,10643268
Проп. селекции – 10 Тур. селекции – 10 Колесо рулетки – 80	0,032521922	0,013234	0,09829084
Проп. селекции – 5 Тур. селекции – 65 Колесо рулетки – 30	0,108406406	0,040051	0,08964752

Однако немаловажно разнообразие особей, представленных в родительском пуле, т. к. оно имеет непосредственное влияние на эффективность операторов скрещивания. Влияние этого критерия на скорость поиска оптимального решения зависит от размера популяции и мощности множества, в

котором происходит поиск решения. Чем большей становится область поиска, тем значительнее становится влияние этого критерия. Также, исходя из значения разнообразия родительского пула, можно сделать вывод о необходимом размере популяции и родительского пула.

СПИСОК ЛИТЕРАТУРЫ

1. **Chellapilla, K.** Combining Mutation Operators in Evolutionary Programming [Text] / K. Chellapilla // IEEE transactions on Evolutionary Computation. –1998. –Vol. 2. –№ 3. –P. 91–96.

2. **Fogel, L.** Artificial Intelligence through Simulated Evolution [Text] / L. Fogel, A. Owens, M. Walsh –NY: Wiley, 1966.

3. **Naidoo, A.** The Induction of Finite Transducers Using Genetic Programming [Электронный ресурс] / A. Naidoo, N. Pillay // Proc. of Euro GP. Springer, 2007. –Режим доступа: <http://saturn.cs.unp.ac.za/~nelishiap/papers/eurogp07.pdf>

4. **Емельянов, В.В.** Теория и практика эволюци-

онного моделирования [Текст] / В.В. Емельянов, В.В. Курейчик, В.М. Курейчик. –М.: Физматлит, 2003. –С. 432.

5. **Найханова, Л.В.** Технология создания методов автоматического построения онтологий с применением генетического и автоматного программирования [Текст] / Л.В. Найханова. –Улан-Удэ: Изд-во БНЦ СО РАН, 2008. –244 с.

6. **Поликарпова, Н.** Автоматное программирование [Текст] / Н. Поликарпова, А. Шальто. –СПб.: Питер, 2009. – 176с.

7. Рабочая книга социолога [Текст]. –М.: Наука, 1977. –С. 257–297.

УДК 519.6

А.Г. Ченцов, П.А. Ченцов

МАРШРУТИЗАЦИЯ ПЕРЕМЕЩЕНИЙ С ОГРАНИЧЕНИЯМИ И НЕСТАЦИОНАРНЫМИ ФУНКЦИЯМИ СТОИМОСТИ

Рассматривается задача о посещении конечной системы мегаполисов при условии выполнения (внутренних) работ в период каждого посещения и ограничениях в виде условий предшествования (условия типа «одно после другого»). Имеется особенность в задании функций стоимости внешних перемещений и внутренних работ: упомянутые функции зависят от номера в очереди. Данный номер можно интерпретировать как дискретное время, что позволяет трактовать названную особенность как своеобразную нестационарность. Эта нестационарность может быть, в частности, связана с приоритетами клиентов, заинтересованных в обслуживании мегаполисов. В этой конкретизации клиенты могут в своей совокупности рассматриваться как система с противоречивыми интересами участников. Исполнитель, осуществляющий серию работ, сопровождающихся внешними перемещениями, оптимизирует систему возникающих

затрат (издержек) в условиях ограничений и противоречий, порождаемых претензиями клиентов.

Возможны и другие конкретизации рассматриваемой ниже общей постановки. В частности, допускаются случаи, когда номера в очереди соответствуют фактическим моментам времени, а упомянутая выше нестационарность отвечает динамике меняющихся состояний среды, в которой реализуются перемещения. Допускается также случай, когда номера в очереди определяют всякий раз промежуток реального времени, в течение которого осуществляются работы в пределах того или иного мегаполиса при однородных способах оценивания затрат (уровнях цен). Здесь «момент» является укрупненным представлением промежутка, естественным для уровня макрозадачи, касающейся организации внешних перемещений и увязки с внутренними работами посредством выбора пунктов прибытия и отправления.

В статье представлен оптимальный алгоритм маршрутизации, построенный на основе экономической модификации динамического программирования, причем снижение вычислительных затрат осуществляется за счет рационального учета ограничений (условий предшествования). Алгоритм реализован на ПЭВМ с использованием элементов параллельной структуры.

Излагаемые ниже результаты анонсированы в [1].

Постановка задачи

Введем сначала некоторые общие обозначения, используя сокращения: ДР – допустимое решение; МДП – метод динамического программирования; ПМ – подмножество; УП – упорядоченная пара. Через $:=$ обозначаем равенство по определению.

Семейством называется множество, все элементы которого сами являются множествами. Если h – объект, то $\{h\}$ есть def одноэлементное множество, содержащее h . Для любых двух объектов x, y через (x, y) обозначаем УП [3] с первым элементом x и вторым элементом y . Если даны три объекта u, v и w , то $(u, v, w) := ((u, v), w)$ – триплет. В этой связи отметим соглашение [4, гл. 1, раздел 3]: для любых множеств A, B и C полагается, что $A \times B \times C := (A \times B) \times C$. Если z есть УП, то через $pr_1(z)$ и $pr_2(z)$ обозначаем первый и второй элементы z , однозначно определяемые условием $z = (pr_1(z), pr_2(z))$; если $z \in A \times B$, где A и B – множества, то $pr_1(z) \in A$ и $pr_2(z) \in B$. Если U, V и W – множества, $p \in U \times V$ и $q \in W$, то в силу упомянутого выше соглашения [4] $(p, q) \in U \times V \times W$. Для всякого множества S через $\text{Fin}(S)$ обозначаем семейство всех непустых конечных ПМ S . Каждому непустому конечному множеству K сопоставляем мощность $|K| \in \mathbf{N}$, где $\mathbf{N} := \{1; 2; \dots\}$; полагаем $|\emptyset| := 0$.

В дальнейшем \mathbf{R} – вещественная прямая, $[0, \infty[:= \{\xi \in \mathbf{R} \mid 0 \leq \xi\}$ (неотрицательная полуось), $\mathbf{N}_0 := \{0\} \cup \mathbf{N} = \{0; 1; 2; \dots\}$ и

$$\overline{k, l} := \{i \in \mathbf{N}_0 \mid (k \leq i) \& (i \leq l)\} \quad \forall k \in \mathbf{N}_0 \quad \forall l \in \mathbf{N}_0.$$

Тогда $\overline{1, n} = \{i \in \mathbf{N} \mid i \leq n\} \quad \forall n \in \mathbf{N}$. Если K – непустое конечное множество, то через $(\text{bi})[K]$ обозначаем множество всех биекций [5] «промежутка» $\overline{1, |K|}$ на K . Напомним, что перестановкой непустого множества A называется всякая биекция A на себя.

Всюду в дальнейшем фиксируем число $N \in \mathbf{N}$,

$2 \leq N$, непустое множество X , точку $x^0 \in X$, а также множества

$$M_1 \in \text{Fin}(X), \dots, M_N \in \text{Fin}(X),$$

именуемые мегаполисами. Предполагается, что $x^0 \notin \bigcup_{i=1}^N M_i$ и, кроме того, $\forall p \in \overline{1, N} \quad \forall q \in \overline{1, N}$

$$(p \neq q) \Rightarrow (M_p \cap M_q = \emptyset).$$

Для дальнейшего удобно ввести (конечное) множество

$$X := \{x^0\} \cup \left(\bigcup_{i=1}^N M_i\right) \in \text{Fin}(X).$$

Ниже рассматриваются перемещения в X . Полагаем, что $\mathbf{P} := (\text{bi})[\overline{1, N}]$; если $\alpha \in \mathbf{P}$, то через α^{-1} обозначаем перестановку, обратную к α : $\alpha^{-1} \in \mathbf{P}$ и

$$\alpha^{-1}(\alpha(s)) = \alpha(\alpha^{-1}(s)) = s \quad \forall s \in \overline{1, N}.$$

В терминах множества \mathbf{K} , $\mathbf{K} \subset \overline{1, N} \times \overline{1, N}$, определяются условия предшествования: перестановка $\alpha \in \mathbf{P}$ (т. е. маршрут) допустима по предшествованию, если $\forall z \in \mathbf{K} \quad \forall t_1 \in \overline{1, N} \quad \forall t_2 \in \overline{1, N}$

$$((\alpha(t_1) = pr_1(z)) \& (\alpha(t_2) = pr_2(z))) \Rightarrow (t_1 < t_2).$$

Итак, элементы \mathbf{K} – суть УП индексов, именуемые адресными; для каждой адресной пары первый элемент называется отправителем, а второй – получателем. Требуется, чтобы каждый отправитель (точнее мегаполис с индексом отправителя) посещался ранее соответствующего получателя. Тогда $\mathbf{A} := \{\alpha \in \mathbf{P} \mid \alpha^{-1}(pr_1(z)) < \alpha^{-1}(pr_2(z)) \quad \forall z \in \mathbf{K}\}$ есть множество всех маршрутов $\alpha \in \mathbf{P}$, допустимых по предшествованию. Полагаем в дальнейшем, что для всякого непустого множества $\mathbf{K}_0, \mathbf{K}_0 \subset \mathbf{K}$, непременно $\exists z_0 \in \mathbf{K}_0 : pr_1(z_0) \neq pr_2(z_0) \quad \forall z \in \mathbf{K}_0$.

Тогда, как показано в монографии [2, раздел 2.2], $\mathbf{A} \neq \emptyset$. Итак, в рассматриваемом случае допустимые маршруты существуют. Нам потребуются также трассы (траектории), согласованные с тем или иным маршрутом. Речь идет при этом о перемещениях вида

$$\begin{aligned} x^0 \rightarrow (x_{1,1} \in M_{\alpha(1)} \rightarrow x_{1,2} \in M_{\alpha(1)}) \rightarrow \dots \\ \dots \rightarrow (x_{N,1} \in M_{\alpha(N)} \rightarrow x_{N,2} \in M_{\alpha(N)}), \end{aligned} \quad (1)$$

где перемещения в круглых скобках отвечают внутренним работам; фрагменты внутренних работ связываются внешними перемещениями. Разумеется, в (1) требуется, чтобы $\alpha \in \mathbf{A}$. В терминах (1) определяются допустимые трассы со звеньями, определяемыми посредством УП $z_1 = (x_{1,1}, x_{1,2}), \dots, z_N = (x_{N,1}, x_{N,2})$.

Этот кортеж дополняем начальной УП

$z^0 := (x^0, x^0)$, полагая, что $z_0 = z^0$ (в дальнейшем обозначение z^0 понимается только в упомянутом смысле). Возвращаясь к (1), полагаем далее, что для всякого $\alpha \in \mathcal{A}$ Z_α есть def множество всех кортежей $(z_i)_{i \in \overline{0, N}} : \overline{0, N} \rightarrow \mathbf{X} \times \mathbf{X}$, для каждого из которых $z_0 = z^0$ и $z_j \in M_{\alpha(j)} \times M_{\alpha(j)} \quad \forall j \in \overline{1, N}$; разумеется, Z_α есть непустое конечное множество. Каждую УП $(\alpha, (z_i)_{i \in \overline{0, N}}), \alpha \in \mathcal{A}, (z_i)_{i \in \overline{0, N}} \in Z_\alpha$, будем называть допустимым решением (ДР) формулируемой ниже задачи маршрутизации.

Мы намеренно игнорируем сейчас описание перемещений при выполнении внутренних работ, которые могут иметь достаточно сложную структуру; в (1) определены, собственно говоря, скачки $x_{1,1} \rightarrow x_{1,2}, \dots, x_{N,1} \rightarrow x_{N,2}$.

Появление данных скачков связано как раз с работами, которые (на макроуровне) не детализируются, что позволяет, не проигрывая в качестве, построить работоспособный алгоритм с учетом естественной иерархии: внутренние работы относятся к задачам нижнего уровня, а перемещения (1) – к задаче верхнего уровня (макрозадаче). При этом используется аддитивный способ агрегирования затрат, а тогда при решении задач нижнего уровня можно ограничиться оптимальными режимами, при которых упомянутым выше скачкам можно сопоставить экстремумы стоимостей соответствующих внутренних работ. Последние могут быть несложными и тогда решение задач нижнего уровня очевидно и может быть сразу отражено в виде стоимостей скачков $x_{s,1} \rightarrow x_{s,2}$, где $s \in \overline{1, N}$. В других случаях «внутренние» задачи могут потребовать разработки специальных методов решения, но во всех случаях иерархический подход представляется полезным (см., например, [6, 7]). В настоящей статье основное внимание уделяется решению макрозадачи.

Введем три типа функций стоимости: стоимости внешних перемещений, внутренних работ (имеются в виду экстремумы задач нижнего уровня) и терминального состояния. Функции первого типа суть

$$\begin{aligned} c_1 : \mathbf{X} \times M_1 \times \overline{1, N} &\rightarrow [0, \infty[, \dots, \\ \dots, c_N : \mathbf{X} \times M_N \times \overline{1, N} &\rightarrow [0, \infty[. \end{aligned} \quad (2)$$

Если $s \in \overline{1, N}$, $x \in \mathbf{X}$, $y \in M_s$ и $t \in \overline{1, N}$, то $c_s(x, y, t)$ есть стоимость перемещения из x в y , завершаемого «в момент» t . Индекс s используется для указания того, что оценивается перемещение на мегаполис M_s ; зависимость от t существенна и

может, в частности, отражать желания клиента, заинтересованного в обслуживании M_s .

Для оценивания внутренних работ используются функции

$$\begin{aligned} c_1 : M_1 \times M_1 \times \overline{1, N} &\rightarrow [0, \infty[, \dots \\ \dots, c_N : M_N \times M_N \times \overline{1, N} &\rightarrow [0, \infty[. \end{aligned} \quad (3)$$

Если $s \in \overline{1, N}$, $x \in M_s$, $y \in M_s$ и $t \in \overline{1, N}$, то $c_s(x, y, t)$ оценивает внутренние работы в пределах M_s , осуществляемые «в момент» t при условии, что (см. (1)) x есть пункт прибытия на M_s , а y – пункт отправления; в наиболее логичном варианте $c_s(x, y, t)$ есть экстремум внутренних затрат при фиксации x и y («рабочий» вариант, реализуемый в (1), имеет при $j \in \overline{1, N}$ следующий вид: $s = \alpha(j)$, $x = x_{j,1}$, $y = x_{j,2}$, $t = j$). Для оценивания терминального состояния используем функцию

$$f : \bigcup_{i=1}^N M_i \rightarrow [0, \infty[.$$

Заметим, что функции $c_1, \dots, c_N, c_1, \dots, c_N$ являются по существу объемными матрицами, а f сводится к вектору. Если $\alpha \in \mathcal{A}$ и $(z_i)_{i \in \overline{0, N}} \in Z_\alpha$, то полагаем, что

$$\begin{aligned} \Pi(\alpha, (z_i)_{i \in \overline{0, N}}) &:= \sum_{t=1}^N c_{\alpha(t)}(\text{pr}_2(z_{t-1}), \text{pr}_1(z_t), t) + \\ &+ \sum_{t=1}^N c_{\alpha(t)}(z_t, t) + f(\text{pr}_2(z_N)). \end{aligned} \quad (4)$$

В правой части (4) выделены три существенных компоненты аддитивного критерия, связанные с оцениванием внешних перемещений, внутренних затрат и терминального состояния. В качестве основной рассматриваем задачу

$$\Pi(\alpha, (z_i)_{i \in \overline{0, N}}) \rightarrow \min, \alpha \in \mathcal{A}, (z_i)_{i \in \overline{0, N}} \in Z_\alpha. \quad (5)$$

Напомним, что множество ДР непусто и конечно; задаче (5) сопоставляется значение

$$V := \min_{\alpha \in \mathcal{A}} \min_{(z_i)_{i \in \overline{0, N}} \in Z_\alpha} \Pi(\alpha, (z_i)_{i \in \overline{0, N}}) \in [0, \infty[\quad (6)$$

и непустое множество оптимальных решений (имеются в виду ДР $(\alpha^0, (z_i^0)_{i \in \overline{0, N}})$, где $\alpha^0 \in \mathcal{A}$ и $(z_i^0)_{i \in \overline{0, N}} \in Z_{\alpha^0}$, для которых $\Pi(\alpha^0, (z_i^0)_{i \in \overline{0, N}}) = V$). Наша цель – нахождение V и какого-либо оптимального ДР.

В [1] изложена схема решения на основе широко понимаемого МДП; особо отметим теорему 1 работы [1] (доказательство теоремы получается естественной комбинацией конструкций [6, 8]), доставляющую уравнение Беллмана, а также эко-

номичную вычислительную процедуру (см. [1, раздел 3]), не предусматривающую построение всего массива значений функции Беллмана. В настоящей статье ограничимся алгоритмической стороной дела, связанной с упомянутой вычислительной процедурой. Итак, рассмотрим алгоритм на функциональном уровне.

Слои функции Беллмана. Функция Беллмана [1] определена на множестве $\text{Pos} := \mathbf{X} \times \mathbf{N}$, где \mathbf{N} – семейство всех ПМ $\overline{1, N}$. Элементы Pos (а это УП) именуем позициями. Мы, однако, сразу будем выделять конечный набор ПМ Pos , на которых рекуррентно определяются функции, являющиеся, как показано в [1], сужениями настоящей функции Беллмана.

Пусть $\mathbf{G} := \{K \in \mathbf{N}_* \mid \forall z \in \mathbf{K} \quad (\text{pr}_1(z) \in K) \Rightarrow (\text{pr}_2(z) \in K)\}$, где \mathbf{N}_* – семейство всех непустых ПМ $\overline{1, N}$, то есть $\mathbf{N}_* := \mathbf{N} \setminus \{\emptyset\}$;

$$\mathbf{G}_s := \{K \in \mathbf{G} \mid s = |K|\} \quad \forall s \in \overline{1, N}.$$

При $\mathbf{K}_1 := \{\text{pr}_1(z) : z \in \mathbf{K}\}$ имеем $\mathbf{G}_1 = \{\{t\} : t \in \overline{1, N} \setminus \mathbf{K}_1\}$; $\mathbf{G}_N = \{\overline{1, N}\}$ (одноэлементное семейство). Следуя [2], вводим оператор вычеркивания $\mathbf{I} : \mathbf{N}_* \rightarrow \mathbf{N}_*$, полагая при $K \in \mathbf{N}_*$, что $\mathbf{I}(K) := K \setminus \{\text{pr}_2(z) : z \in \Sigma[K]\}$, где $\Sigma[K] := \{z \in \mathbf{K} \mid (\text{pr}_1(z) \in K) \& (\text{pr}_2(z) \in K)\}$. Кроме того, из результатов [9, 10] следует, что $\mathbf{G}_{s-1} = \{K \setminus \{j\} : K \in \mathbf{G}_s, j \in \mathbf{I}(K)\} \quad \forall s \in \overline{2, N}$.

Определяем \mathbf{M} в виде объединения всех множеств $M_i, i \in \overline{1, N} \setminus \mathbf{K}_1$. Тогда $D_0 := \{(x, \emptyset) : x \in \mathbf{M}\}, D_N := \{(x^0, \overline{1, N})\}$ суть крайние слои пространства позиций. Если $s \in \overline{1, N-1}$ и $K \in \mathbf{G}_s$, то последовательно определяем множества $J_s(K) := \{j \in \overline{1, N} \setminus K \mid \{j\} \cup K \in \mathbf{G}_{s+1}\}$ и $\mathbf{M}_s[K] := \bigcup_{i \in J_s(K)} M_i$, после чего вводим клетку $\mathbf{D}_s[K] := \{(x, K) : x \in \mathbf{M}_s[K]\}$. Тогда полагаем, что $D_s := \bigcup_{K \in \mathbf{G}_s} \mathbf{D}_s[K] \quad \forall s \in \overline{1, N-1}$.

Итак, построены множества D_0, D_1, \dots, D_N ; каждое из них есть непустое ПМ множества Pos . Получили слои пространства позиций. Далее определяем функции

$$\begin{aligned} V_0 : D_0 &\rightarrow [0, \infty[, \\ V_1 : D_1 &\rightarrow [0, \infty[, \dots, V_N : D_N &\rightarrow [0, \infty[\end{aligned} \quad (7)$$

посредством рекуррентной процедуры. Итак, полагаем, что

$$V_0(x, \emptyset) := f(x) \quad \forall x \in \mathbf{M}, \quad (8)$$

получая (в виде (8)) функцию на D_0 . Если $s \in \overline{1, N}$, то преобразование V_{s-1} в V_s задается правилом

$$\begin{aligned} V_s(x, K) := \min_{j \in \mathbf{I}(K)} \min_{z \in M_j \times M_j} [c_j(x, \text{pr}_1(z), \\ N - |K| + 1) + c_j(z, N - |K| + 1) + \\ + V_{s-1}(\text{pr}_2(z), K \setminus \{j\})] \quad \forall (x, K) \in D_s; \end{aligned} \quad (9)$$

учитываем здесь, что (см. монографию [2, раздел 4.9]) при $(u, U) \in D_s, t \in \mathbf{I}(U)$ и $v \in M_t$ непременно $(v, U \setminus \{t\}) \in D_{s-1}$. Заметим, что [1] $V = V_N(x^0, \overline{1, N})$, тогда из (9) вытекает, что

$$\begin{aligned} V = \min_{j \in \mathbf{I}(\overline{1, N})} \min_{z \in M_j \times M_j} [c_j(x^0, \text{pr}_1(z), 1) + \\ + c_j(z, 1) + V_{N-1}(\text{pr}_2(z), \overline{1, N} \setminus \{j\})]. \end{aligned} \quad (10)$$

Построение оптимального решения. Полагаем далее $\mathbf{z}_0 := z^0$. Считаем, что все слои функции Беллмана построены на основе (8), (9). Приступаем к построению (оптимального) ДР. Для этого, используя (10), выбираем $\mathbf{k}_1 \in \mathbf{I}(\overline{1, N})$ и $\mathbf{z}_1 \in M_{\mathbf{k}_1} \times M_{\mathbf{k}_1}$, для которых

$$\begin{aligned} V = c_{\mathbf{k}_1}(x^0, \text{pr}_1(\mathbf{z}_1), 1) + c_{\mathbf{k}_1}(\mathbf{z}_1, 1) + \\ + V_{N-1}(\text{pr}_2(\mathbf{z}_1), \overline{1, N} \setminus \{\mathbf{k}_1\}). \end{aligned} \quad (11)$$

Осуществляем перемещение $(x^0, \overline{1, N}) \rightarrow (\text{pr}_2(\mathbf{z}_1), \overline{1, N} \setminus \{\mathbf{k}_1\})$, где

$$(\text{pr}_2(\mathbf{z}_1), \overline{1, N} \setminus \{\mathbf{k}_1\}) \in D_{N-1}. \quad (12)$$

Используем (9) в условиях, когда $s = N-1$, а (x, K) есть позиция (12): выбираем $\mathbf{k}_2 \in \mathbf{I}(\overline{1, N} \setminus \{\mathbf{k}_1\})$ и $\mathbf{z}_2 \in M_{\mathbf{k}_2} \times M_{\mathbf{k}_2}$, для которых

$$\begin{aligned} V_{N-1}(\text{pr}_2(\mathbf{z}_1), \overline{1, N} \setminus \{\mathbf{k}_1\}) = \\ = c_{\mathbf{k}_2}(\text{pr}_2(\mathbf{z}_1), \text{pr}_1(\mathbf{z}_2), 2) + c_{\mathbf{k}_2}(\mathbf{z}_2, 2) + \\ + V_{N-2}(\text{pr}_2(\mathbf{z}_2), \overline{1, N} \setminus \{\mathbf{k}_1, \mathbf{k}_2\}). \end{aligned} \quad (13)$$

Заметим, что из (11), (13) следует, в частности, что

$$\begin{aligned} V = c_{\mathbf{k}_1}(x^0, \text{pr}_1(\mathbf{z}_1), 1) + c_{\mathbf{k}_2}(\text{pr}_2(\mathbf{z}_1), \\ \text{pr}_1(\mathbf{z}_2), 2) + c_{\mathbf{k}_1}(\mathbf{z}_1, 1) + c_{\mathbf{k}_2}(\mathbf{z}_2, 2) + \\ + V_{N-2}(\text{pr}_2(\mathbf{z}_2), \overline{1, N} \setminus \{\mathbf{k}_1, \mathbf{k}_2\}) = \\ = \sum_{t=1}^2 c_{\mathbf{k}_t}(\text{pr}_2(\mathbf{z}_{t-1}), \text{pr}_1(\mathbf{z}_t), t) + \\ + \sum_{t=1}^2 c_{\mathbf{k}_t}(\mathbf{z}_t, t) + V_{N-2}(\text{pr}_2(\mathbf{z}_2), \overline{1, N} \setminus \{\mathbf{k}_1, \mathbf{k}_2\}), \end{aligned} \quad (14)$$

где $(\text{pr}_2(\mathbf{z}_2), \overline{1, N} \setminus \{\mathbf{k}_1, \mathbf{k}_2\}) = (\text{pr}_2(\mathbf{z}_2), \overline{1, N} \setminus \{\mathbf{k}_1, \mathbf{k}_2\}) \in D_{N-2}$. Если $N = 2$, то в силу (8) оптимальное решение уже построено. Если $N > 2$, то упомянутую выше процедуру выбора локальных решений на основе (9) следует продолжать вплоть до исчерпания списка $\overline{1, N}$.

Вычислительный эксперимент

Для вычислений использовалась программа, написанная на Microsoft Visual C++ 2005. Вычисления производились на ПЭВМ с процессором Intel i7-2630QM с 8 Гб оперативной памяти, работающей под управлением Windows 7.

В настоящем разделе исследуется модельный пример ОЗМ. Речь пойдет о плоской задаче маршрутизации: имеется в виду случай $X = \mathbf{R} \times \mathbf{R}$. На плоскости фиксируется база $x^0 = (0, 0)$. В качестве $\rho: X \times X \rightarrow [0, \infty[$ используем функцию евклидова расстояния: при $x_1 \in X$ и $x_2 \in X$ число $\rho(x_1, x_2)$ есть евклидово расстояние между плоскими векторами x_1 и x_2 .

Полагаем, что $N = 27$. Каждый мегаполис M_i , $i \in \overline{1, 27}$, задается равномерной сеткой на окружности радиуса 7,5. Полагаем, что $|M_i| = 50$, $i \in \overline{1, 27}$. Центры окружностей определены на плоскости случайным образом, но так, что получающиеся при этом мегаполисы не пересекаются и не содержат базу. Точки M_i можно рассматривать как входы-выходы в помещение, в пределах которого исполнителю следует достичь заданной точки и, выполнив там некоторое действие, покинуть помещение, вновь используя один из портов (возможно, тот же, через который был произведен вход). В пределах каждой окружности случайным образом выбирается упомянутая точка – так называемый «рубильник», с которым связано выполнение внутренних работ. Цель внутренних

работ – приведение в действие «рубильника», связанного с каждым мегаполисом. Фиксируем точки $a_i \in X$, $i \in \overline{1, 27}$, определяющие положения «рубильников».

Далее вводятся непосредственно функции стоимости. Полагаем, что при каждом $s \in \overline{1, 27}$ функция c_s определяется условием: при $x \in X$, $y \in M_s$ и $t \in \overline{1, 27}$

$$c_s(x, y, t) = \rho(x, y)(1, 1 + \sin(t)).$$

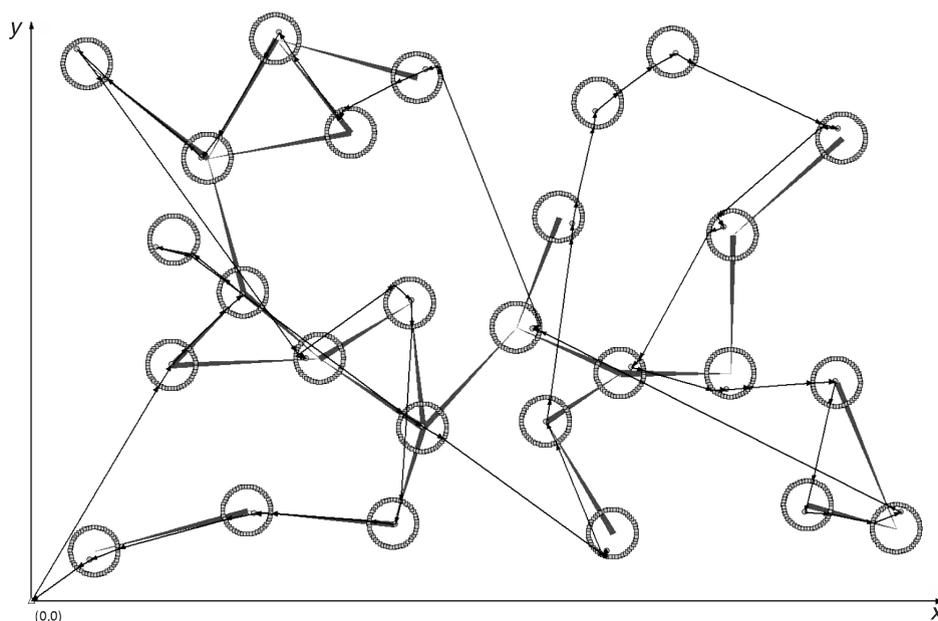
Стоимости внешних перемещений определены. Для оценивания внутренних работ используются функции c_i , $i \in \overline{1, 27}$, определяемые следующим образом: если $s \in \overline{1, 27}$, $x \in M_s$, $y \in M_s$ и $t \in \overline{1, 27}$, то

$$c_s(x, y, t) = \rho(x, a_s)(1, 1 + \cos(t)) + \rho(a_s, y)(1, 1 + \cos(t)).$$

Наконец, функцию f отождествляем с евклидовой нормой на плоскости X . Таким образом, определены функции стоимости $c_1, \dots, c_{27}, c_1, \dots, c_{27}, f$.

Что касается точных координат мегаполисов и «рубильников» – они не приводятся с целью экономии места (как и полученный в ходе решения маршрут).

Получены следующие результаты: $V = 1424,36$; время счета 3 ч 25 мин 32 с. Результат отображен на рисунке. Тонкими стрелками показан маршрут движения, толстыми – адресные пары.



Результаты работы алгоритма

В статье предлагается метод построения оптимального решения маршрутной задачи о посещении системы мегаполисов, осложненной ограничениями и необходимостью выполнения тех или иных работ в пределах упомянутых мегаполисов. Решение базируется на идеях динамического программирования. При этом используется специальный метод, позволяющий ограничиться частичным построением массива значений функции Беллмана за счет использования условий предшествования, моделирующих ограничения, типичные для многих инженерных задач (применения развиваемых методов могут быть связаны, в частности, с проблемой снижения облучаемости персонала АЭС при выполнении комплекса работ в условиях повышенной радиации (см. [11]).

Рассматриваемая постановка восходит на идейном уровне к известной труднорешаемой задаче коммивояжера [12–14], для которой в [15, 16] построен вариант МДП; отметим также метод ветвей и границ (см. [17]). Существенные особенности рассматриваемой в настоящей работе задачи связаны с наличием ограничений, многовариантностью на этапах перемещений между мегаполисами и включением (в постановку) фрагментов внутренних работ. Построенный в статье оптимальный алгоритм реализован в виде стандартной программы на ПЭВМ с элементами параллельной структуры, позволяющей решать задачи умеренной размерности с большим числом ограничений в виде условий предшествования.

СПИСОК ЛИТЕРАТУРЫ

1. **Ченцов, А.Г.** Динамическое программирование в одной нестационарной задаче маршрутизации [Текст] / А.Г. Ченцов, П.А. Ченцов // Известия Ин-та математики и информатики УдГУ. –2012. –Вып. 1 (39). –С.151–154.
2. **Ченцов, А.Г.** Экстремальные задачи маршрутизации и распределения заданий: вопросы теории [Текст] / А.Г. Ченцов. –М. –Ижевск: НИЦ «Регулярная и хаотическая динамика»; Ижевский институт компьютерных исследований, 2008. –240 с.
3. **Куратовский, К.** Теория множеств [Текст] / К. Куратовский, А. Мостовский. –М.: Мир, 1970. –416 с.
4. **Дьедонне, Ж.** Основы современного анализа [Текст] / Ж. Дьедонне. –М.: Мир, 1964. –430 с.
5. **Кормен, Т.** Алгоритмы: построение и анализ [Текст] / Т. Кормен, Ч. Лейзерсон, Р. Ривест. –М.: МЦНМО, 1999. –960 с.
6. **Ченцов, А.А.** Экстремальная задача маршрутизации перемещений с ограничениями и внутренними потерями [Текст] / А.А. Ченцов, А.Г. Ченцов, П.А. Ченцов, // Известия вузов. Математика. –2010. –№ 6. –С. 64–81.
7. **Ченцов, А.А.** Обобщенная задача курьера с неоднородными внутренними работами [Текст] / А.А. Ченцов, А.Г. Ченцов // Алгоритмы и програм. средства параллел. вычислений. –Екатеринбург: Изд-во УрО РАН, 2011. –Вып. 11. –С. 80–109.
8. **Тонков, Л.В.** К вопросу оптимального выбора маршрута в условиях временного дисконтирования [Текст] / Л.В. Тонков, А.Г. Ченцов // Кибернетика и систем. анализ. –1999. –№ 1. –С. 95–106.
9. **Григорьев, А.М.** Динамическое программирование в обобщенной задаче курьера с внутренними работами: элементы параллельной структуры [Текст] / А.М. Григорьев, Е.Е. Иванко, А.Г. Ченцов // Моделирование и анализ информационных систем. –Ярославль, 2011. –Т. 18. –№ 3. –С.101–124.
10. **Ченцов, А.Г.** Одна параллельная процедура построения функции Беллмана в обобщенной задаче курьера с внутренними работами [Текст] / А.Г. Ченцов // Автоматика и телемеханика. –2012. –№ 3. –С. 134–149.
11. **Сесекин, А.Н.** Использование метода динамического программирования для оптимизации траектории перемещения работников в радиационно-опасных зонах с целью минимизации облучения [Текст] / А.Н. Сесекин, О.Л. Ташлыков, С.Е. Щеклеин, М.Ю. Куклин, А.Г. Ченцов, А.А. Кадников // Известия вузов. Ядерная энергетика. –2006. –№ 2. –С. 41–48.
12. **Меламед, И.И.** Задача коммивояжера. Вопросы теории [Текст] / И.И. Меламед, С.И. Сергеев, И.Х. Сигал // Автоматика и телемеханика. –1989. –№ 9. –С. 3–34.
13. **Меламед, И.И.** Задача коммивояжера. Точные алгоритмы [Текст] / И.И. Меламед, С.И. Сергеев, И.Х. Сигал // Автоматика и телемеханика. –1989. –№ 10. –С. 3–29.
14. **Меламед, И.И.** Задача коммивояжера. Приближенные алгоритмы [Текст] / И.И. Меламед, С.И. Сергеев, И.Х. Сигал // Автоматика и телемеханика. –1989. –№ 11. –С. 3–26.
15. **Беллман, Р.** Применение динамического программирования к задаче о коммивояжере [Текст] / Р. Беллман // Кибернетический сборник. –М.: Мир, 1964. –Т. 9. –С. 219–228.
16. **Хелд, М.** Применение динамического программирования к задачам упорядочения [Текст] / М. Хелд, Р.М. Карп // Кибернетический сборник. –М.: Мир, 1964. –Т. 9. –С. 202–218.
17. **Литл, Дж.** Алгоритм для решения задачи о коммивояжере [Текст] / Дж. Литл, К. Мурти, Д. Суини, К. Кэрел // Экономика и математические методы. –1965. –Т.1. –Вып. 1. –С. 94–107.

УДК 004.056.5

И.А. Зикратов, А.И. Спивак, А.В. Разумовский

МЕТОД РАЗГРАНИЧЕНИЯ ДОСТУПА К ОБЪЕКТАМ В СИСТЕМЕ КОНТРОЛЯ ВЕРСИЙ

Промышленная разработка и эксплуатация сложных ИТ-систем, основанных на использовании распределенных вычислений, организуется коллективами разработчиков и пользователей. Примером таких систем может служить разработка программного обеспечения 3D анимации, где объектами являются исходные тексты программ, отдельные графические сцены, объекты анимации, обладающие определенными атрибутами, которые могут изменяться в зависимости от сценария фильма. Для художников-аниматоров особую важность приобретает возможность работы с версиями одних и тех же сцен, что предполагает разграничение прав доступа к ним со стороны субъектов-разработчиков.

Наиболее распространенным механизмом распределения доступа к версиям бинарных файлов в системах контроля версий являются блокировки. Механизм блокировки позволяет одному из разработчиков взять в монопольное использование файл или группу файлов для внесения в них изменений. На то время, пока файл заблокирован, все права у остальных пользователей отбираются, он остается доступным всем остальным разработчикам только на чтение, и любая попытка внести в него изменения отвергается сервером. Недостатки использования блокировок следующие:

блокировки мешают продуктивной работе, поскольку вынуждают ожидать освобождения заблокированных файлов;

создают административные проблемы в случае, если разработчик забудет снять блокировку с занятых им файлов.

Для разрешения указанных проблем приходится применять административные меры, в т. ч. включать в систему технические средства для сброса неверных блокировок, но и при их наличии на приведение системы в порядок расходуется время.

Метод разграничения доступа на основе дискреционной модели

Для решения задачи обеспечения совместной работы группы субъектов над одними и теми же

объектами нами взята за основу известная из теории информационной безопасности модель дискреционного разграничения доступа – HRU [1, 2].

Отличием предлагаемого метода от известных, используемых, например, в системах контроля версий Subversion [3] и CVS [4], является введение в систему совокупности матриц доступа, описывающих разграничение субъектов к версиям объектов, что позволило реализовать механизм контроля версий при производстве мультимедийного контента в системе 3D анимации.

Суть метода состоит в представлении версий объектов не в виде древовидной структуры, количество узлов и ветвей в которой является случайными и труднопрогнозируемыми величинами, а в виде совокупности матриц, на пересечении строк и столбцов которых записаны права доступа именованных субъектов к именованным объектам.

Для системы 3D анимации в качестве столбцов матрицы представлено множество объектов, которыми являются файлы сцен и объекты анимации, а в качестве строк – субъекты-разработчики. В отличие от классической модели HRU (в которой существуют: O – множество объектов; S – множество субъектов; R – множество прав доступа субъектов к объектам) для системы контроля введено множество V – множество версий объектов O .

Интерпретация идеи состоит в том, что при появлении i -й версии объекта создается дополнительная i -я матрица доступа, отражающая права к этой версии объектов со стороны субъектов. В случае отсутствия версий для какого-либо объекта, ячейки, соответствующие его версиям, будут пустыми. Пространство состояний такой системы описывается как $S \times O \times R \times V$.

Матрица прав доступа $M[s, o]$, где столбцами являются объекты, а строками – субъекты, содержит права доступа субъектов к объектам. При создании новой версии объекта (V_i) создается новая матрица $W_{vi}[s, o]$, где содержатся права доступа субъектов к новой версии объекта. Применяется сквозная нумерация версий объектов для простоты получения информации о последнем

изменении объекта, при этом не имеет значения, какой именно субъект создал новую версию объекта. При выполнении операции $a()$ система переходит из состояния Q в состояние Q' .

Соответствующие изменения в формальном аппарате модели HRU заключаются в добавлении следующих базовых операций:

1. Добавление права субъекту по отношению к версии объекта **Enter r W[s,o]**.

Начальное состояние $q = (S, O, V, M, W)$: $s \in S, o \in O, r \in R, v \in V$.

Конечное состояние $q = (S', O', V', M', W')$: $S' = S, O' = O, V' = V, W'_v[s, o] = W_v[s, o] \sqcup \{r\}$, если $(x_s, x_o) \neq (s, o) \Rightarrow W'_v[x_s, x_o] = W_v[x_s, x_o], M'[s, o] = M[s, o]$.

2. Удаление права у субъекта по отношению к версии объекта **Delete r W[s,o]**.

Начальное состояние $q = (S, O, V, M, W)$: $s \in S, o \in O, r \in R, v \in V$.

Конечное состояние $q = (S', O', V', M', W')$: $S' = S, O' = O, V' = V, W'_v[s, o] = W_v[s, o] \setminus \{r\}$, если $(x_s, x_o) \neq (s, o) \Rightarrow W'_v[x_s, x_o] = W_v[x_s, x_o], M'[s, o] = M[s, o]$.

3. Создание версии объекта **Create v'**.

Начальное состояние $q = (S, O, V, M, W)$: $v' \notin V$.

Конечное состояние $q = (S', O', V', M', W')$: $S' = S, V' = V \sqcup \{v'\}$, для $\forall (x_s, x_o) \in S \times O \Rightarrow M'[s, o] = M[s, o], W'_v[x_s, x_o] = \emptyset$.

4. Удаление версии объекта **Delete v'**.

Начальное состояние $q = (S, O, V, M, W)$: $v' \in V$.

Конечное состояние $q = (S', O', V', M', W')$: $S' = S, O' = O, V' = V \setminus \{v'\}$ для $\forall (x_s, x_o) \in S \times O \Rightarrow M'[s, o] = M[s, o], W'_v[x_s, x_o] = W_v[x_s, x_o]$.

Таким образом, наличие апробированного формального описания дискреционной системы контроля версий файлов позволяет устранить недостатки, присущие традиционным механизмам блокировки файлов, обеспечить ясную физическую интерпретацию и контролируемое масштабирование процесса создания версий файлов-сцен.

Реализация метода

Описанный в статье метод дискреционного разграничения доступа реализован в программном модуле системы хранения и контроля версий больших бинарных файлов (СКВ) инновационной системы 3D анимации (ИСА).

Архитектура СКВ имеет распределенный характер. В СКВ выделены три компонента: ядро СКВ, агенты СКВ и клиент СКВ. Благодаря тому, что данные (файлы) передаются только между

агентом и клиентом, такая архитектура позволяет снизить нагрузку на ядро СКВ. Интерфейсы компонентов для общения между собой и программными модулями, использующими СКВ для получения файлов и их версий, унифицированы и представляют собой REST интерфейс [5].

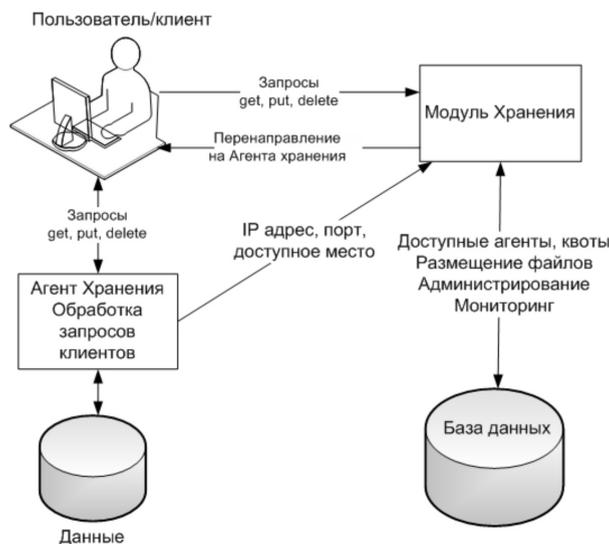
Все компоненты реализованы на языке Java. Это позволяет развернуть компоненты на базе любой операционной системы, имеющей JVM.

Концептуальная схема взаимодействия компонентов СКВ представлена на рисунке. Составными частями служат агент СКВ, ядро СКВ, база данных и клиент СКВ, в качестве которого могут выступать другие модули ИСА, а также сторонние программные продукты.

Ядро СКВ обеспечивает диспетчеризацию потоков данных внутри системы хранения и контроля версий больших бинарных файлов. Выполняя управление работой остальных компонентов, ядро СКВ не является узким местом всего модуля благодаря работе в режиме предоставления только управляющей информации, без необходимости обработки потока данных, который проходит между отправителями и получателями файлов (клиенты и агенты). Для обеспечения распределенного хранения файлов ядро СКВ взаимодействует с агентами и поддерживает в актуальном состоянии базу данных местоположений файлов, а также другую служебную информацию (версии файлов, дату и время их создания, размер и т. д.). Функционал модели, описанной в статье, реализован в ядре и используется при каждом обращении клиента к ядру.

Агент СКВ представляет собой программное обеспечение, выполняющееся на узле, предоставляющем свои ресурсы в виде дискового пространства. Функция по хранению файлов является основной для агентов СКВ, поэтому программный код агента позволяет управлять хранением файлов при помощи операций создания, получения и удаления файлов.

База данных предназначена для хранения метаданных ядра СКВ. Реализация базы данных выполнена с учетом требований высокой скорости работы и возможности простого горизонтального масштабирования. Хранение структуры в единой БД помогает снизить накладные расходы при изменении структуры в распределенной системе хранения и контроля версий файлов и ускоряет работу, по сравнению со стандартной схемой хранения структуры на каждом агенте.



Высокоуровневая структура СКВ

Клиент СКВ. Пользователь взаимодействует с ядром СКВ при первоначальном запросе и при необходимости с агентом СКВ. Взаимодействие между клиентом и компонентами СКВ осуществляется по REST интерфейсу.

Порядок взаимодействия компонент СКВ друг с другом

1. Пользователь (клиент СКВ) посылает запрос ядру СКВ при помощи сформированного сообщения в формате JSON [6].

2. При обработке запроса ядром СКВ выполняется обращение к базе данных с целью получения информации о выполняемой в рамках запроса операции и проверки полномочий на выполнение требуемой операции.

3. Пользователь получает результат выполнения действий на стороне ядра СКВ в виде сформированного запроса к заданному агенту СКВ.

4. Заключительный этап включает в себя обработку запроса на работу с файлом агентом СКВ и определение статуса выполнения операции.

Имеющиеся в составе СКВ функции по управлению доступом к версиям объектов реализуют описанный в статье метод контроля доступа. Это позволяет проводить дальнейшие исследования в направлении усовершенствования систем контроля доступа к объектам в системах контроля версий.

Разработанный метод разграничения доступа, в отличие от известных, позволяет использовать его системам контроля версии обеспечивать разграничение доступа на уровне отдельных версий без вмешательства в технологические процессы, осуществляемые группой разработчиков. Такой функционал является конкурентным преимуществом по сравнению с существующими системами контроля версий и дает возможность более гибкого, по сравнению с механизмом блокировки, управления доступом к объектам.

Работа выполнена в рамках ФЦП, госконтракт № 07.524.11.4009 «Разработка инновационной системы 3D анимации».

СПИСОК ЛИТЕРАТУРЫ

1. **Девянин, П.Н.** Модели безопасности компьютерных систем [Текст] / П.Н. Девянин. –М.: Изд. центр «Академия», 2005. –144 с.

2. **Цирлов, В.Л.** Основы информационной безопасности. Краткий курс [Текст] / В.Л. Цирлов. –М.: Феникс, 2008. –256 с.

3. Система управления версиями Subversion [Электронный ресурс] / Режим доступа: <http://subversion.apache.org/>

4. Система управления версиями CVS [Электрон-

ный ресурс] / Режим доступа: <http://cvs.nongnu.org/>

5. **Fielding, R.** Architectural Styles and the Design of Network-based Software Architectures [Электронный ресурс] / Roy Thomas Fielding. –Режим доступа: http://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf

6. RFC 4627. The application/json Media Type for JavaScript Object Notation (JSON) [Электронный ресурс] / Internet Engineering Task Force. –Режим доступа: <http://tools.ietf.org/html/rfc4627>



Конференция «Технологии Microsoft в теории и практике программирования»

Начиная с 2004 года факультет технической кибернетики Санкт-Петербургского государственного политехнического университета проводит конференцию студентов, аспирантов и молодых ученых Северо-Запада «Технологии Microsoft в теории и практике программирования». Цели конференции – подготовка участников к будущей работе в профессиональных программистских коллективах, создающих программный продукт высокого качества, поддержка изучения современных информационных технологий и инструментальных средств в соответствии с мировыми стандартами и действующими международными сертификационными требованиями, а также выявление талантливых молодых специалистов в области разработки и использования программных систем.

Особенность конференции заключается в том, что участник должен проявить свои знания и умения не столько в области программирования различных математических головоломок, сколько в области разработки и использования программных продуктов и систем в условиях, максимально приближенных к реальным процессам проектирования и разработки современных систем различной степени сложности.

Конференция 2012 года проводится в два этапа: на первом отбираются и ревьюируются работы участников, присылающих короткие статьи – заявки на участие. На их основе отбираются приглашенные докладчики на второй этап. Присланные статьи публикуются в сборнике материалов конференции. Второй этап проходит в виде докладов и демонстраций на секциях.

Все участники секционных презентаций награждаются дипломами первой, второй и третьей степени. Решение о премировании участников конкурса принимает конкурсная комиссия, в которую входят руководители секций и представители организаций спонсоров.

В 2012 году спонсором конференции является всемирно известная компания Microsoft.

В данном журнале публикуется семь статей победителей – участников конференции 2012 года. Публикуемые работы отличает инновационный характер, ориентация на реализацию полученных результатов, широкое применение технологий MS при создании программного обеспечения и документации.

Декан ФТК, профессор,
доктор технических наук

И.Г. Черноруцкий

УДК 004.4'2

А.В. Григорьева, В.А. Васильев, В.О. Сафонов

ПРИМЕНЕНИЕ ЭЛЕМЕНТОВ АСПЕКТНО-ОРИЕНТИРОВАННОГО ПОДХОДА ASPECT.NET ПРИ РАЗРАБОТКЕ ГЕОЛОКАЦИОННОЙ СИСТЕМЫ УПРАВЛЕНИЯ РЕСУРСАМИ ДЛЯ ОБЛАЧНОЙ ПЛАТФОРМЫ MICROSOFT WINDOWS AZURE

В связи с растущими темпами производства и развитием глобальных коммуникационных сетей задача учета ресурсов и управления ими становится все более важной. Возросла также необходимость в получении и обработке данных о состоянии и географическом положении ресурсов в реальном времени. В данной статье рассматривается применение элементов аспектно-ориентированного подхода ASPECT.NET при разработке открытой геолокационной системы управления ресурсами, которая позволит получать данные о местоположении ресурсов в реальном времени, а также оперативно реагировать на изменения с помощью сложной бизнес-логики правил и отчетов, тем самым эффективно решая задачу управления ресурсами.

Данный проект является исследовательским и находится на ранней стадии, которая характеризуется высокой изменчивостью пользовательских требований к системе [1].

Архитектурно разрабатываемая геолокационная система управления ресурсами представляет собой сложное объектно-ориентированное приложение, разворачиваемое с помощью облачной платформы Microsoft Windows Azure [2]. Важная отличительная особенность разрабатываемой системы – возможность гибкой настройки конфигурации на конкретного пользователя, что позволит создавать специальные правила для управления конкретными ресурсами и получать разнообразные отчеты в реальном времени. Для этого система логически разделена на два слабо связанных между собой модуля: модуль взаимодействия с ресурсами и устройствами, позволяющий прозрачно для системы добавлять и отключать различные устройства и получать данные от них, и модуль управления, ответственный за создание и редактирование новых ресурсов, создание правил, отчетов, настройку конфигурации,

управление устройствами и пользователями.

Модуль управления представляет собой трех-слойную структуру, состоящую из следующих компонентов:

- слой пользовательского интерфейса и визуализации, который содержит динамические веб-страницы и компоненты для пользовательского интерфейса, а также набор кросс-платформенных веб-служб для поддержки доступа к основным функциям системы из произвольного клиентского приложения;

- слой бизнес-логики, который содержит реализацию основных функций системы, таких, как управление пользователями системы (создание, редактирование пользователей и пользовательских групп, назначение администраторов); управление ресурсами (создание, редактирование ресурсов и групп ресурсов); управление правилами (создание правил и настройка их периодичности); управление отчетами (создание, редактирование и трансформация отчетов); мониторинг и отслеживание ресурсов;

- слой доступа к данным, который содержит в себе классы для реализации доступа к различным источникам данных (SQL-источники данных, NoSQL-источники данных, кэши) и данные о сущностях системы (ресурсы, устройства, пользователи, аккаунты и т. д.).

Модуль взаимодействия с ресурсами и устройствами представляет собой отдельный независимый компонент для работы с устройствами. Любой ресурс в системе представляет определенное устройство в реальном мире (например, это может быть датчик температуры, скорости, GPS-датчик, и т. д.), которое передает данные о своем состоянии в систему. Важная отличительная особенность системы – возможность работы с различными типами устройств без необходимости сложной настройки соедине-

ния. Для достижения этой цели реализована концепция «драйверов» для различных устройств. Архитектурно модуль состоит из трех важных частей:

- кластера серверов, «слушающих» различные порты и принимающих данные от устройств;
- драйверов, получающих данные от конкретных устройств по определенному протоколу и передающих их на сервер;
- подсистемы связывания и передачи данных от устройств в модуль управления для дальнейшей обработки, реализующей слабое связывание между двумя модулями посредством использования единых источников данных

Таким образом, схематично процесс добавления нового устройства в систему выглядит следующим образом: пользователь создает ресурс – логическое представление устройства, определяет для него группу, настраивает для него правила (например, отображать ресурс на карте или отсылать письмо при поступлении данных от устройства) и отчеты. Затем пользователь определяет адрес и порт сервера, который будет принимать сообщения от устройства, и выбирает нужный драйвер. После этого при поступлении данных от устройства система будет автоматически применять правила и настройки, тем самым выдавая информацию в реальном времени. Взаимодействие с устройствами происходит в основном посредством XML-сообщений, что упрощает процесс обмена данными и позволяет легко добавлять в систему новые устройства. При необходимости добавить устройство нового типа, можно просто реализовать драйвер, который будет трансформировать данные от устройства в XML-формат и отправлять их на сервер.

Объектно-ориентированная методология позволяет добиться хорошего уровня декомпозиции распределенной облачной системы. Но, в свою очередь, применение объектно-ориентированной методологии программирования привнесло в создание облачных приложений традиционные проблемы «сквозной функциональности» (cross-cutting concerns), когда один компонент отвечает за реализацию

```
public class AzureTrace: Aspect {
```

```
    //Строка для установки соединения с хранилищем Azure
    const string DiagnosticsConnectionString = "...";
```

нескольких функций [3]. Например, в веб-службе часто перемешиваются бизнес-логика, требования безопасности, кэширование, протоколирование и пр. Аспектно-ориентированное программирование (АОП) предлагает решать проблему «сквозной функциональности» вынесением каждой задачи в свой модуль – аспект. Нужное поведение целевой программы достигается за счет вставки действий аспекта в заданные точки на этапе компиляции или в процессе выполнения.

Компания Microsoft позиционирует Enterprise Library (EL) [4] как набор наиболее удачных архитектурных решений, применимых в большинстве программных систем. По сравнению с динамическим применением аспектов в EL, когда во время выполнения программы среда .NET создает прокси-объекты в заданных местах, Aspect.NET [5] вплетает действия на уровне MSIL-инструкций после этапа компиляции целевой сборки, что влечет повышение производительности целевого приложения. Достоинствами библиотеки EL является встроенная поддержка со стороны среды выполнения .NET, а также возможность настройки аспектов без компиляции через конфигурационные XML-файлы. Однако при работе с облачными вычислениями перекompиляция для внедрения аспектов не вызывает трудностей, т. к. развертывание целевой системы осуществляется лишь один раз. Более того, такая «пост-обработка» дает возможность выбирать конкретные места применения действий аспектов. Наконец, Aspect.NET не требует вносить изменения в исходный код целевого приложения, т. е. целевой проект будет полностью изолирован от аспектов.

В работе предложены решения в виде аспектов Aspect.NET для следующих задач в рамках предметной области:

1. Протоколирование. Этот аспект перехватывает в целевом коде все вызовы метода System.TraceInformation() и перенаправляет их данные в хранилище диагностической информации Azure, что позволяет просматривать и обрабатывать ее с помощью специальных инструментальных средств.

```
//Настраиваем и запускаем диагностику Azure внутри метода WebRole.OnStart()
[AspectAction(«%before %call *RoleEntryPoint.OnStart()»)]
static public void RegisterAzureDiagnostics() {
    try {
        CloudStorageAccount storage =
            CloudStorageAccount.Parse(DiagnosticsConnectionString);
        DiagnosticMonitorConfiguration diagnosticConfig =
            DiagnosticMonitor.GetDefaultInitialConfiguration();

        //...Настраиваем diagnosticConfig
        DiagnosticMonitor.Start(storage, diagnosticConfig);
    }
    catch (Exception e) {
        //...Сообщаем об исключении с указанием вызвавшего его целевого метода
    }
}
//Перед каждым вызовом TraceInformation() убеждаемся, что соотв. Azure-подписчик
//зарегистрирован
[AspectAction(«%before %call *System.Diagnostics.Trace.TraceInformation»)]
public static void AspectTrace() {
    //Если его еще нет в списке подписчиков, то добавляем в этот список...
    if (!ContainsWADListener()) {
        System.Diagnostics.Trace.Listeners.Add(
            new Microsoft.WindowsAzure.Diagnostics.DiagnosticMonitorTraceListener());
        System.Diagnostics.Trace.AutoFlush = true;
    }
}
}
```

2. Прозрачная интеграция сторонних инструментов. Для платформы .NET разработано множество библиотек, каркасов и инструментов, таких, как PostSharp [6], MS Code Contracts [7], MS EL и пр., предоставляющих полезные технические сервисы и службы. Однако их интеграция в целевое приложение приводит к изменениям в исходном коде, что нежелательно при быстром прототипировании, когда необходимо удалить или заменить

данную подсистему. Например, в целях эксперимента было решено использовать для протоколирования при нажатии кнопки на странице веб-роли ResourcesView метод Microsoft.Practices.EnterpriseLibrary.Logging.Logger(). С помощью Aspect.NET можно создать аспект, к которому подключаются все соответствующие сторонние сборки EL, а затем с помощью его действий вставить нужный код в целевое приложение:

```
[ReplaceBaseClass]//Подменяет целевой класс ResourcesView данным наследником
public class ELLogger : ResourcesView {
    protected void LogButton_Click(object sender, EventArgs e) {
        Microsoft.Practices.EnterpriseLibrary.Logging.Logger.Write(«Message from the Logging
                                                                    Application Block»);

        base.LogButton_Click(sender, e);
    }
}
```

3. Обработка исключений. Если при обращении к некоторому веб-сервису возникает исключительная ситуация, аспект посылает ему еще не-

сколько запросов по таймауту. В случае неудачи данные запрашиваются у запасного веб-сервиса. Вся служебная информация, в т. ч. стек, отсыла-

ется в службу протоколирования Azure. Рассмотрим пример взаимодействия клиентского приложения, которое для получения данных (например, списка доступных ресурсов) использует облачную веб-службу, а затем визуально отображает

их. Вызов данной службы происходит в обработчике щелчка мыши `buttonInvoke_Click` по кнопке формы на локальной машине, а в виде результата она возвращает текст. Сам принцип применим и для взаимодействия служб в облаке.

```
public class ExceptionCatcher : Aspect {
    //Задаем правила, куда вставлять это действие
    [AspectAction(«%instead %call *buttonInvoke_Click(...) && args(..)»)]
    public static void RepairAction(object sender, EventArgs e) {
        int count=0;
        Exception servEx=null;
        while(count++<3) {
            try {
                //Пытаемся вызвать сервис несколько раз
                Target.buttonInvoke_Click(sender,e);
            } catch {Exception e) {
                servEx=e;
            }
        }
        if (servEx!=null) {
            try {
                GetAvailableResources2();//Пытаемся вызвать запасной сервис
            } catch {Exception e) {/*...*/}
        }
        public static void GetAvailableResources2 () {
            AvailableResourcesClient2 client = new AvailableResourcesClient2();
            string result = client.GetData();
            MessageBox.Show(result);
        }
    }
}
```

4. Кэширование. Предложен аспект, действия которого подменяют целевой метод, загружающий данные из MS SQL Azure, и сохраняют их в распределенном кэше `Microsoft.ApplicationServer.Caching.DataCacheFactory`. В целевое приложение можно встроить управляющий кэшем веб-компонент, который также содержится в сборке этого аспекта.

5. Управление конфигурацией MS Azure. С помощью счетчиков производительности платформы аспект постоянно отслеживает текущую

загрузку и по необходимости подключает или отключает новые вычислительные узлы, тем самым экономя денежные затраты клиента.

Применение АОП и системы Aspect.NET при разработке облачной геолокационной системы управления ресурсами позволяет повысить легкость ее сопровождения, увеличить скорость разработки и снизить затраты за счет повторного использования универсальных аспектов.

СПИСОК ЛИТЕРАТУРЫ

1. Ларман, К. Применение UML и шаблонов проектирования [Текст] / К. Ларман. –М.: Изд-во «Вильямс». –47 с.

2. Сайт проекта MS Windows Azure [Электронный ресурс] / Режим доступа: <http://www.windowsazure.com>

3. Safonov, V.O. Using aspect-oriented programming for trustworthy software development [Text] / V.O. Safonov // Wiley Interscience. –John Wiley & Sons, 2008. –338 p.

4. Сайт проекта MS Enterprise Library // Режим доступа: <http://wag.codeplex.com>

5. Григорьев, Д.А. Реализация и практическое применение аспектно-ориентированной среды программирования для Microsoft .NET [Текст] / Д.А. Григорьев // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2009. –№ 3. –С. 225–232.

6. Сайт проекта PostSharp [Электронный ре-

сурс] / Режим доступа: <http://www.sharpcrafters.com>

7. Сайт проекта Code Contracts [Электронный ресурс] / Режим доступа: <http://msdn.microsoft.com/en-us/devlabs/dd491992>

8. Сайт проекта Aspect.NET [Электронный ресурс] / Режим доступа: <http://aspectdotnet.org>

УДК 681.3.068

Н.В. Воинов, Л.П. Котлярова, И.Е. Поддубный

МЕТОДИКА РАЗРАБОТКИ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ ДЛЯ ПЛАТФОРМЫ WINDOWS PHONE 7

Современный рынок мобильных устройств требует подготовки специалистов, способных быстро осваивать и создавать новые приложения под новые платформы. Среди них мобильная платформа Windows Phone 7 [1] в настоящее время считается наиболее перспективной. Именно поэтому она использована в новом курсе для обучения специалистов по направлению «Программная инженерия» на факультете технической кибернетики СПбГПУ.

В курсе разработана новая методика создания приложений (рис. 1), включающая следующее:

1) подготовку окружения, в т. ч. установку средств разработки приложений – Visual Studio 2010, Service Pack 1 и Windows Phone SDK [3];

2) создание проекта лабораторной или курсовой работы;

3) анализ интерфейса и макета внешнего вида приложения;

4) анализ архитектуры проекта, который предполагает проектирование схемы классов будущего приложения;

5) разработку кода разметки интерфейса на языке разметки xaml;

6) разработку программной логики на языках **C#, Java и Visual Basic) с использованием технологий .NET Framework 4, Silverlight и XNA;**

7) отладку на симуляторе и реальном устройстве;

8) дальнейшее совершенствование проекта.

Методика демонстрирует студентам основные принципы разработки приложений и учит использованию важных возможностей платформы.

Основная задача при разработке нового приложения заключается в создании пользовательского интерфейса на языке разметки xaml (eXtensible Application Markup Language) и логики на языках программирования. xaml – это расширяемый язык разметки приложений, разработанный компанией Microsoft. Он основан на XML, содержит интуитивно понятные онтологические структуры, прост в освоении. Тесное взаимодействие объектов xaml и программной логики позволяет без труда работать с элементами управления в коде приложения.

Разработанная методика включает три группы лабораторных работ и курсовую работу, в рамках которой студенты создают собственные приложения. Каждая из групп лабораторных работ объясняет и демонстрирует на практике следующие возможности:

1) варианты разметки (Grid, StackPanel, Canvas), основные элементы управления, контекст ввода, навигацию между страницами и передачу данных, элементы Pivot и Panorama, меню приложения;

2) задачи запуска и выбора, элемент управления WebBrowser, элемент управления Map и определение местоположения, работа с акселерометром, работа с HTTP, работа с изолированным



Рис. 1. Структура методики создания мобильных приложений

хранилищем и локальной базой, привязка данных, основы Linq;

3) восстановление приложений из состояний Dormant и Tombstoned, работа с фоновыми службами, фоновая загрузка файлов, воспроизведение музыки, работа с будильником, живые плитки (Live Tiles), некоторые элементы управления расширения Microsoft.Phone.Controls.Toolkit.

Каждая работа включает описание задачи и методику решения. Структура описания включает: 1) формулировку задания; 2) определение осваиваемых тем; 3) описание типового решения; 4) возможные способы совершенствования.

В первом пункте определяется, какое приложение требуется разработать и какими функциями оно должно обладать.

Второй пункт содержит список изучаемых тем, что позволяет студентам, предпочитающим самостоятельно во всем разбираться, узнать, какие конкретно вопросы им необходимо изучить.

Третий пункт содержит полное описание методики разработки приложения. Описание включает пошаговые инструкции, примеры кода и снимки экрана. Создание каждой страницы разбивается на три этапа: разработку кода разметки страницы; подключение необходимых библиотек; разработку кода программы.

Четвертый пункт описания лабораторной работы содержит список возможных совершенствований приложения. После выполнения задания студентам необходимо самостоятельно внести некоторые изменения в код или добавить дополнительную функциональность, что позволяет контролировать процесс изучения материала и определять уровень способностей и заинтересо-

ванности каждого студента.

Работы рассчитаны на студентов, имеющих опыт разработки Windows-приложений на языках C#, Visual Basic, Java. Предполагаемое время выполнения одной лабораторной работы составляет два академических часа.

Курсовые работы предполагают использование полученных навыков при разработке игрового приложения средствами технологий Silverlight и XNA. Каждая из них рассчитана на выполнение в течение 12 академических часов. Методика включает пять курсовых работ на использование обеих технологий, что, в частности, позволяет освоить работу с событийной машиной, графикой, звуком и обработкой жестов.

Наиболее интересные приложения могут быть размещены в магазине Marketplace [2].

Предлагаемая методика позволит студентам без особых трудностей научиться разрабатывать приложения для Windows Phone 7 разного уровня сложности. По методике для учебного курса разработаны 21 лабораторная и 10 курсовых работ.

Для иллюстрации применения методики приведено описание курсовой работы «Тетрис».

Задание. Используя технологию XNA, написать игру «Тетрис» для Windows Phone 7. Допускается использование технологии Silverlight + XNA, если основная логика игры будет написана на XNA Framework.

Требования. В работе необходимо обеспечить:

управление уровнями сложности (изменение скорости игры);

подсчет очков и сохранение лучшего результата;

реализацию меню игры (при использовании технологии Silverlight + XNA) и звуковых эффектов;

вывод на экран набранных очков и уровня сложности;

создание собственного тайла.

По технологии Silverlight + XNA необходимо создать новый проект XNA Game Studio 4.0 – Windows Phone Silverlight and XNA Application [4].

Поскольку курсовые работы подразумевают создание полноценных игровых приложений с большим числом строк кода, в статье далее приведен только порядок действий без детализации программной логики:

«Создадим абстрактный класс Figure, который будет содержать основные методы для будущих классов фигур».

Перейдем к работе с библиотеками XNA:

«В классе игры определим глобальные переменные типа Texture2D, SpriteFont и SoundEffect (для этого нужно использовать пространство имен Microsoft.Xna.Framework.Audio). Внутри метода OnNavigatedTo() будем загружать изображения, шрифты и звуковые эффекты из ресурсов. Там же будем получать параметр от страницы MainPage.xaml с номером уровня сложности (скорости игры)».

Далее обучающимся следует ознакомиться с одним из основных методов XNA, в котором происходят игровые действия – OnUpdate(). Далее детализируем обработку касаний экрана:

«В методе OnUpdate() происходит обновление объектов, пересчет положений и обработка нажатий. Внутри данного метода будем перехватывать нажатия. При одиночном нажатии будем перемещать фигуру просто на одну клетку. При задержанном – будем осуществлять быстрое перемещение. В функции StartGame() будем определять, какие типы жестов считывать сенсором экрана (для этого нужно использовать пространство имен Microsoft.Xna.Framework.Input.Touch)».

Далее познакомим обучающихся с еще одним основным методом XNA, в котором происходит вывод изображений на экран – OnDraw():

«Метод OnDraw() отвечает за перерисовку игрового пространства. Он вызывается каждый игровой такт. Внутри него с помощью уже определенной переменной spriteBatch будем рисовать фон, кнопки, игровое поле, кирпичики на поле, фигуры и строки. Всю прорисовку необ-

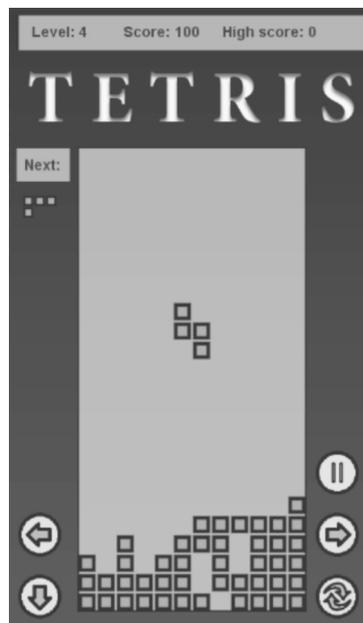


Рис. 2. Игровое приложение «Тетрис»

ходимо заключать в функциональные «скобки» spriteBatch.Begin() и spriteBatch.End()».

Данная часть курсовой обучает работе с методом OnDraw() и двухмерной графикой: рисованию изображений и текстовых строк на экране телефона с использованием библиотек XNA.

В результате описанных действий будет разработано игровое приложение, показанное на рис. 2.

В конце описаний курсовых работ, так же как и в конце описаний лабораторных работ, приводится список возможных совершенствований:

- совершенствование анимации;
- добавление фигур «пентамино» (или других).

Совершенствование анимации подразумевает самостоятельное изучение библиотек эффектов XNA.

Описанная методика протестирована на группе студентов разных курсов. В итоге получено 15 приложений. Они оценивались по степени сложности и завершенности по заранее определенным критериям с помощью экспертной оценки. Как показали результаты исследования (рис. 3), студенты старших курсов разрабатывали более завершенные и сложные приложения. Это связано с большим опытом разработки и с большей заинтересованностью в получении новых знаний.

В связи с тем, что методика рассчитана, в основном, на студентов старших курсов, участ-



Рис. 3. Результаты пилотирования методики

ники пилотного исследования были разделены на две группы. В первую вошли студенты первого и второго курсов, во вторую – третьего и четвертого. Дальнейшее исследование проводилось на второй группе: оценивалась степень освоения материала методики. В качестве критерия была выбрана оценка сложности приложений: чем сложнее приложение, тем лучше освоен материал.

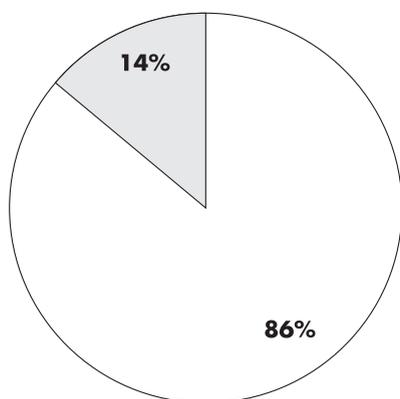


Рис. 4. Доля студентов старших курсов, полностью освоивших методику (□) полностью освоившие; (■) освоившие частично

Результаты показали (рис. 4), что более 85 % студентов старших курсов полностью освоили материал методики. В связи с этим можно сделать вывод, что методика обеспечивает обучение студентов разработке приложений для мобильных устройств.

Мобильные технологии активно развиваются. В конце 2012 г. ожидается выход новой версии платформы Windows Phone – Windows Phone 8. По предварительной информации от компании Microsoft, новая платформа не будет иметь кардинальных отличий от своего предшественника, т. е. в основе разработки, как и ранее, будут лежать технологии Silverlight и XNA [5]. В результате предложенная методика потребует минимальных изменений.

Планируется расширение курса, предполагающее добавление лабораторных работ по следующим темам: информационная безопасность; управление службами; взаимодействие с облачными сервисами; взаимодействие с социальными сетями.

Работа частично поддержана грантом РФФИ 11-07-90412-Укр_ф_а.

СПИСОК ЛИТЕРАТУРЫ

1. Microsoft Windows Phone 7 [Электронный ресурс] / Режим доступа: <http://www.microsoft.com/windowsphone>
2. Microsoft Windows Phone 7 Marketplace [Электронный ресурс] / Режим доступа: www.windowsphone.com/marketplace
3. Windows Phone 7 SDK [Электронный ресурс] / Режим доступа: <http://www.microsoft.com/en-us/download/details.aspx?id=27570>
4. Орлов, С. Технологии разработки программного обеспечения: Учебник. [Текст] / С. Орлов. –СПб.: Питер, 2002. –464 с.
5. Петзольд, Ч. Программируем Windows Phone 7 [Текст] / Ч. Петзольд. –Redmond, Washington: Microsoft Press, 2011. –695 с.

УДК 681.5

Л.Л. Малыгин, В.А. Царев, П.С. Черкас

АВТОМАТИЧЕСКОЕ УПРАВЛЕНИЕ ПАРАМЕТРАМИ СРЕДСТВ ФОРМИРОВАНИЯ ИЗОБРАЖЕНИЙ В СИСТЕМАХ РАСПОЗНАВАНИЯ ТЕКСТОВЫХ МЕТОК РЕАЛЬНОГО ВРЕМЕНИ

Система распознавания текстовых меток реального времени (СРТМРВ) представляет собой аппаратно-программный комплекс (АПК), работа которого заключается в распознавании в режиме реального времени текстовых меток (ТМ) на изображениях подвижных объектов. Эти изображения СРТМРВ получает с использованием средств формирования изображений (СФИ, включающих весь комплекс аппаратных средств (камеры, средства освещения, платы видеозахвата), участвующих в процессе формирования изображений. СРТМРВ также включает в себя алгоритмы обработки и анализа изображений, общее и специализированное программное обеспечение, ЭВМ.

В настоящее время с помощью СРТМРВ решается широкий круг задач, связанных с идентификацией подвижных объектов контроля по текстовым меткам на них. Ярким примером СРТМРВ являются системы автоматического распознавания автомобильных номеров реального времени (САРАНРВ).

Эффективность СРТМРВ зависит от нескольких факторов, в т. ч. основных:

- качества и корректности использования средств формирования изображений;
- эффективности алгоритмического и программного обеспечения;
- производительности ЭВМ.

Производительность современных ЭВМ и эффективность алгоритмов распознавания находятся на высоком уровне и непрерывно возрас-

тают. Основные проблемы применения СРТМРВ связаны с тем, что использование большинства современных СФИ не позволяет формировать высококачественные изображения текстовых меток в нестационарных сложных условиях использования. При этом если объект с ТМ на изображении размытый, нечеткий, недостаточно освещен или пересвечен (рис. 1), то это приводит к малой информативности такого изображения, что становится причиной значительного снижения эффективности всей системы, т. к. изображения являются входными данными для алгоритмов распознавания.

Многие СФИ (камеры, объективы) общего назначения имеют встроенные средства автоматической адаптации под изменяющиеся условия. Но они рассчитаны на достижение высокого качества всего изображения, а не ТМ на нем, что ограничивает их применение в СРТМРВ.

Таким образом, одним из методов улучшения эффективности СРТМРВ является повышение качества изображений с ТМ за счет автоматического управления параметрами специализированных СФИ.

Качество изображений

Под качеством изображения понимается степень соответствия его характеристик определенным объективным и субъективным требованиям.

Объективные требования в основном предъявляются алгоритмами распознавания ТМ и со-



Рис. 1. Низкокачественные изображения автомобильных номеров

держат ограничения на характеристики областей изображения, содержащих ТМ (размеры в пикселях по вертикали и горизонтали, яркость, контрастность, углы наклона и т. д.).

Субъективные требования формируются с учетом того, что изображения должны быть полезны и информативны для оператора системы.

В общем случае из всех требований к изображению можно выделить те, которые относятся к сюжету изображения, и те, которые относятся к параметрам всего изображения и его области с ТМ (например, яркость, контрастность и т. д.). Первые могут быть удовлетворены на этапе монтажа СФИ и их первоначальной настройки. Для выполнения требований из второй группы из-за использования СРТМРВ в нестационарных условиях на практике требуется периодически подстраивать параметры СФИ вручную. Из-за трудоемкости данной задачи эта подстройка во многих случаях не выполняется [5]. В этой связи актуально автоматическое управление параметрами СФИ с учетом требований к изображению.

С точки зрения повышения качества изображения с помощью управления параметрами СФИ целесообразно остановиться на рассмотрении требований из второй группы, относящихся к характеристикам изображения и его областей, содержащих образ ТМ.

Изображение можно представить как двумерный массив значений яркости $Im[x, y]$, при $x \in [1, width]$ и $y \in [1, height]$ [3]. Область изображения с образом ТМ задается с помощью координат прямоугольника $L = (x_1, y_1, x_2, y_2)$.

В общем случае некоторая i -я характеристика, используемая для оценки качества изображений, рассчитывается с помощью функции от изображения (либо его области) $R_i(Im) = a, a \in A_i$, где A_i – область значений функции R_i . Данные функции должны рассчитываться автоматически без участия оператора ЭВМ и обладать минимальными вычислительными затратами.

При этом все характеристики текущего изображения представляют собой вектор значений:

$$R(Im) = (R_1, \dots, R_n). \quad (1)$$

Требования к качеству изображения указывают оптимальные значения для каждой характеристики $R_0 = (R_{0_1}, \dots, R_{0_n})$, где $R_{0_i} \subseteq A_i$. То есть формируется некоторый эталон, который может включать как одно значение $R_{0_i} = a', a' \in A_i$, так и область значений (или несколько областей значений) $R_{0_i} = \{a' | a' \in A_i\}$ для каждой характеристики изображения.

Тогда качество изображения рассчитывается следующим образом:

$$Q(Im) = Q(R(Im), R_0) = (Q_1(R_1(Im), R_{0_1}), \dots, Q_n(R_n(Im), R_{0_n})). \quad (2)$$

В качестве функции $Q_i(R_i(Im), R_{0_i})$ может использоваться определенная метрика $|R_i(Im) - R_{0_i}|$ или более сложная методика расчета рассогласования величин. В некоторых случаях целесообразно использовать обобщенное качество изображения, описываемое в виде одного значения:

$$Q(Im) = k * Q_2^{P_1} * \dots * Q_n^{P_n} \text{ или} \quad (3)$$

$$Q(Im) = a_1 * Q_1 + \dots + a_n * Q_n,$$

где k – нормирующий коэффициент; наборы коэффициентов P_1, \dots, P_n и a_1, \dots, a_n определяют вес каждой характеристики.

Автоматическое управление параметрами СФИ

Для автоматического управления параметрами СФИ в СРТМРВ вводится новый модуль выработки управляющих решений (МВУР), представленный на рис. 2.

Добавление МВУР преобразует СРТМРВ в замкнутую систему автоматического управления (САУ) с отрицательной обратной связью. Сигналом обратной связи является качество текущего изображения. МВУР – автоматический регуля-

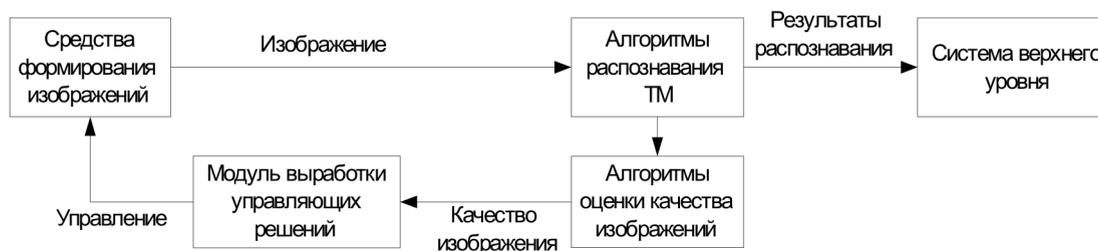


Рис. 2. Система автоматического управления параметрами СФИ в СРТМРВ

тор. Цель управления – максимизация качества изображений.

Для определения качества изображений используются программные средства, рассчитывающие значения определенных характеристик изображения. Это могут быть как основные алгоритмы распознавания ТМ, так и вспомогательные алгоритмы обработки и анализа изображений.

СФИ следует подбирать в соответствии с задачей, решаемой СРТМРВ. Среди всех параметров СФИ наибольшую ценность имеют настраиваемые в режиме реального времени параметры (далее HPR), которые необходимо выбирать с учетом требований к характеристикам изображений.

В общем случае МВУР может быть любым из автоматических регуляторов с подходящим законом управления. Однако многие СРТМРВ должны работать в сложных нестационарных условиях. Это приводит к тому, что априорно неизвестна зависимость характеристик изображения и его областей с ТМ от управляемых параметров СФИ. В таких случаях следует использовать адаптивный регулятор (АР), в котором часть требований к характеристикам изображений задается на этапе проектирования экспертами, а остальные требования могут быть рассчитаны в процессе работы системы. Закон управления в АР может быть задан с учетом основных известных зависимостей с неопределенными коэффициентами, которые рассчитываются в модуле адаптации и передаются в модуль управления [2].

Закон управления в АР должен позволять рассчитывать новые значения управляемых параметров HPR' с учетом их текущих значений HPR и качества изображения Q :

$$U(HPR, Q, M) = HPR', \quad (4)$$

где M – динамически формируемая модель, устанавливающая зависимость характеристик изображения от управляемых параметров СФИ с учетом текущих условий использования. В модели накапливаются данные об успешности управления при различных исходных значениях параметров СФИ и значениях характеристик изображения, которые в дальнейшем аппроксимируются и участвуют при выработке последующих оптимальных управляющих сигналов.

При определении эффективности СРТМРВ с автоматическим управлением необходимо уделять внимание всем подсистемам. Эффективность СФИ может быть определена по среднему

качеству формируемых изображений. Качество работы алгоритмов распознавания оценивается с помощью вероятностных критериев [1]. Для обеспечения работы системы в режиме реального времени необходимо оценивать временной интервал, в течение которого обрабатывается одно изображение в основных алгоритмах распознавания ТМ. Эффективность МВУР может быть оценена с помощью средств расчета показателей САУ и автоматических регуляторов. На быстродействие МВУР накладываются ограничения, связанные с тем, что объекты с ТМ могут двигаться с большой скоростью в поле зрения камеры:

$$K \ll T, \quad (5)$$

где K – время, необходимое на оценку качества изображения, выработку и применение новых значений управляемых параметров СФИ; T – время нахождения в поле зрения камеры объекта с ТМ, движущегося с максимально-допустимой скоростью.

Тестирование СРТМРВ с автоматическим управлением параметрами СФИ необходимо осуществлять либо с помощью специальных программных средств моделирования с обратной связью (виртуальные 3D полигоны), либо в реальных условиях использования.

Применение автоматического управления параметрами СФИ

На основе описанной выше методики разработан АПК «УМКА-Автомаршал», являющийся сетевой «умной» камерой [6] и предназначенный для применения в системах распознавания автомобильных номеров. Данный комплекс содержит в одном герметичном корпусе мини-ЭВМ, цифровую USB камеру с ИК фильтром и управляемой ИК подсветкой. Изображения с камеры поступают в специализированное ПО в мини-ЭВМ, которое отвечает за распознавание автомобильных номеров, оценку качества изображений, управление параметрами камеры и передачу изображений и результатов распознавания по локальной вычислительной сети [4].

АПК «УМКА-Автомаршал» протестирован в различных условиях использования (разные времена года, время суток и погодные условия). При этом формируемые им изображения обладают высоким качеством, что обеспечивает высокий уровень распознавания и подтверждает эффективность предложенной методики.



Перспективность описанных методов заключается в том, что оценка качества изображений позволяет формировать рекомендации оператору и администратору СРТМРВ. Например, при некорректно установленных и настроенных СФИ определенные характеристики изображений будут значительно хуже оптимальных значений. На основе этого си-

стема может информировать о необходимости подстройки определенных параметров вручную.

Предложенная методика автоматического управления параметрами СФИ в СРТМРВ позволяет значительно повысить эффективность подобных систем и упростить их установку, настройку и использование.

СПИСОК ЛИТЕРАТУРЫ

1. **Воскресенский, Е.М.** Метод оценки эффективности систем распознавания текстовых меток на сложном фоне с использованием дерева вероятностных характеристик [Текст] / Е.М. Воскресенский, В.А. Царев // Компьютерная оптика. –2008. –Т. 32. –№ 3. –С. 283–290.

2. **Дорф, Р.** Современные системы управления [Текст] / Р. Дорф, Р. Бишоп; Пер. с англ. Б.И. Копылова. –М.: Лаборатория Базовых Знаний, 2002. –832 с.

3. **Шапиро, Л.** Компьютерное зрение [Текст] / Л. Шапиро, Дж. Стокман; Пер. с англ. –М.: БИНОМ. Лаборатория знаний, 2006. –752 с.

4. **Vesnin, E.N.** The «Smart Camera» Adaptive Optoelectronic Complex [Text]/ E.N. Vesnin, A.E. Mikhailov, V.A. Tsarev, P.S. Cherkas // Pattern Recognition and Image Analysis. –Pleiades Publishing, Ltd. –2011. –Vol. 21. –№ 2. –P. 354–356.

5. ProSystem CCTV: первый и единственный журнал в России по системам видеонаблюдения: профессиональное издание для экспертов и специалистов по охранному телевидению и видеонаблюдению [Текст] –М.: Немецкая Фабрика Печати. –2010. –№ 42-43.

6. Smart Cameras [Text] / Ed. A.N. Belbachir.–XX. –2010. –404 p.

УДК 004.415

А.С. Иванов, В.П. Котляров, А.А. Летичевский

СТАТИЧЕСКИЙ АНАЛИЗ ТРЕБОВАНИЙ РЕАКТИВНЫХ СИСТЕМ

Требования к надежности телекоммуникационного оборудования высоки, что требует применения самых современных методов разработки и контроля качества. Описанные в данной статье подходы проверялись на приложениях телекоммуникационных систем, но они могут распространяться и на приложения многокомпонентных реактивных систем. Начальный этап разработки системы – формулировка требований. В документах, описывающих требования, часто содержится большое количество ошибок, среди которых встречаются ошибки неполноты и несоответствия начальным требованиям заказчика. Согласно исследованию Джонса [5] и Шалла [6] самыми действенными методами поиска ошибок в требованиях являются формальные инспекции кода, верификация и статический анализ моделей. Эффективность инспекций оценивается в 55 %, а эффективность остальных методов – в 65 %. Автоматизация инспекций – сложный и плохо формализуемый процесс,

в то время как автоматизация статического анализа и верификации обеспечены богатым набором методик и инструментов ведущих разработчиков программного обеспечения [7].

Статический анализ реактивных систем начинается после перевода системы с естественного языка на язык формальных спецификаций. В качестве языка спецификаций может использоваться нотация базовых протоколов [2], UCM [8], различные расширения UML [7], Spec# и AsmL [6]. По формальному описанию строится модель программы, позволяющая проводить статический анализ и верификацию. На основе модели нередко автоматически формируются тесты.

Реализация статического анализа системой VRS

Одной из эффективных систем статического анализа многокомпонентных систем является статический анализатор требований верифика-

тора VRS [3]. В данной системе для анализа используется модель, описанная в языке базовых протоколов [2]. Модель состоит из среды и погруженных в нее агентов, взаимодействующих между собой посредством чтения и изменения атрибутов. Агенты, работающие параллельно и асинхронно, могут быть представлены в виде автоматов. Состоянием в данном случае является множество всех состояний атрибутов. Переходы задаются базовыми протоколами. Базовый протокол представляет собой тройку Хоара [1] $\forall x(\alpha(r, x) \rightarrow \langle P(r, x) \rangle \beta(r, x))$ и описывает тот факт, что если в какой-то момент времени истинно предусловие α , то выполняется процесс P и состояние агента изменяется согласно постусловию β (в котором допустимо изменение состояний и других агентов) [2]. Пред- и постусловия являются формулами многосортной логики первого порядка, интерпретированной на некоторых областях данных. В постусловии атрибуты могут менять свои значения. Агент, переход которого задает базовый протокол, называется *ключевым* для этого протокола.

Статическая верификация подобных моделей опирается на анализ наборов базовых протоколов, фактически без построения полного пространства состояний и поиска в нем. В частности, применяя базовые протоколы к заданным условиям, можно проверять инвариантность этих условий. Статический анализ может показать детерминированность поведения системы и отсутствие тупиков [3].

Состояние s какого-либо агента a называется *недетерминированным*, если из него существует более одного перехода, т. е. применимо более одного протокола с ключевым агентом a .

Введем понятие *непротиворечивости* для систем базовых протоколов. Оно определяется отсутствием пересечений предусловий базовых протоколов для одного ключевого агента. Пусть $\alpha(a, x, r)$ и $\alpha'(a, x, r)$ – предусловия двух различных базовых протоколов b и b' с ключевыми агентами одного и того же типа, конкретизированные одним и тем же именем ключевого агента (в случае, если это имя является параметром), x и y – списки параметров, r – список атрибутов. Протоколы b и b' непротиворечивы, если формула

$$\neg(\exists x\alpha(a, x, r) \wedge \exists y\alpha'(a, x, r)) \quad (1)$$

тождественно истинна (отрицание невыполнимо). Система базовых протоколов называется *непротиворечивой*, если любая пара протоколов

с однопипными ключевыми агентами непротиворечива.

Если система базовых протоколов непротиворечива, то в каждом состоянии существует не более одного применимого базового протокола, а следовательно, все агенты такой системы имеют детерминированное поведение.

К любой исследуемой системе предъявляется требование отсутствия тупиков. Введем понятие *полноты* для систем базовых протоколов. Оно определяется следующим условием: дизъюнкция условий применимости всех базовых протоколов

$$\forall x_1\alpha_1(x_1, r_1) \vee \forall x_2\alpha_2(x_2, r_2) \vee \dots \quad (2)$$

должна быть тождественно истинной. Здесь $\alpha_i(x_i, r_i)$ – предусловие базового протокола b_i .

Если выполняется условие полноты и в любой момент времени для каждого типа агентов в системе найдется агент этого типа, то в каждом состоянии системы найдется хотя бы один применимый протокол. Таким образом, в полной системе отсутствуют тупики.

Требования полноты и непротиворечивости проверяются для всех исследуемых систем, но существует ряд свойств, обусловленных требованиями к конкретной системе, это могут быть, например, требования живучести, надежности. Такие свойства необходимо формулировать для каждой системы индивидуально в формализме конкретной модели.

Рассмотрим свойства, которые можно записать в виде формул базового языка, и поставим требование истинности этих формул на всех состояниях системы. В [3] они названы *условиями целостности*. Построим алгоритм проверки условия целостности для системы базовых протоколов. Пусть $Q(r)$ – формула базового языка, задающая условие целостности, $s_0(r)$ – формула базового языка, характеризующая начальное состояние системы, r – список атрибутов.

- В первую очередь необходимо проверить целостность начальных состояний системы: $s_0(r) \rightarrow Q(r)$. Эта формула должна быть общезначимой, в противном случае начальное состояние нарушает условие целостности, тогда нет смысла продолжать проверки и алгоритм должен завершить работу.

- Далее следует проверить инвариантность условия целостности. Для этого достаточно проверить для каждого протокола следующее условие: если протокол применим и условие целостности

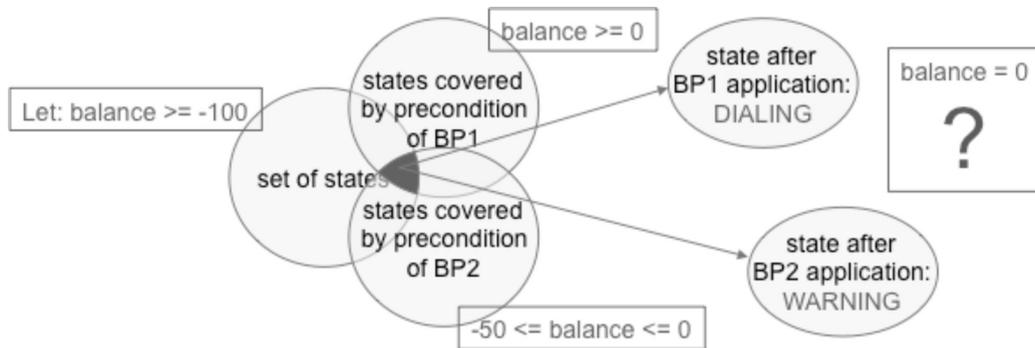


Рис. 1. Недетерминизм в требованиях

истинно, то после применения протокола оно также будет истинным. Формально это условие выражается в виде тождественной истинности формулы $\forall x(pt(\alpha(x,r) \wedge Q(r), \beta(x,r)) \rightarrow Q(r))$, где $\alpha(x,r)$ и $\beta(x,r)$ – пред- и постусловия рассматриваемого протокола, pt – предикатный трансформер, определенный как функция осуществления перехода из одного состояния системы в другое под действием заданного базового протокола.

Недостаток статического поиска недетерминизмов, нарушений условий целостности и тупиков заключается в том, что все найденные ошибки являются всего лишь подозрениями на ошибки и их актуальность требует дополнительного уточнения.

Состояния среды с ошибками можно получить исходя из проверки формул, которые выражают заданные свойства. Достижимость этих состояний можно опровергать статическими методами, сформулировав соответствующие условия целостности, или доказывать с помощью генерации трасс и поиска в пространстве состояний.

Пример поиска ошибок в системе. Рассмотрим простейшую систему, которая проверяет баланс абонента перед совершением звонка (рис. 1). Допустим, что минимальное значение счета может принимать значение -100 . Пусть при положительном балансе система переходит в состояние звонка, а при балансе от -50 до нуля сообщает об отсутствии средств.

Предусловие совершения звонка в терминах базового протокола

```
phone(p, dial);
balance >= 0;
```

Предусловие получения предупреждения об отсутствии средств на счету в терминах базовых протоколов

```
phone(p, dial);
```

```
-50 <= balance <= 0;
```

В приведенной спецификации специально допущено несколько ошибок. Применяя формулу (1), получаем случай, подозрительный на ошибку при $balance = 0$. В общем случае необходимо исследовать модель и либо исправить ее, либо пометить данную ошибку как ложное срабатывание.

Следующий шаг – проверка модели на полноту. Для этого к модели применяется формула (2). Как и в первом случае в спецификации специально допущена ошибка (рис. 2). Статический анализатор [3] определяет неполноту и строит формулу области, подозрительной на ошибку: $balance < 50$.

Рассмотрим более сложный пример недетерминизма в требованиях. Первый рассмотренный нами пример был тривиален и мог быть легко найден без применения статического анализа модели. Введем базовый протокол $bp1_1$ с предусловием $TST(m.active), ((a > 0) \& (b < 150))$ и постусловием $TST(m.idle), (a := 120)$. Введем базовый протокол $bp2_1$ с предусловием $TST(m.idle), (a > 0)$ и постусловием $TST(m.queued)$. Введем базовый протокол $bp2_2$ с предусловием $TST(m.idle), (a = b + 5)$ и постусловием $TST(m.queued)$. Применив формулу (1), в результате получаем вердикт и формулу, описывающую состояние, в котором возможно существует неопределенное поведение.

Из формулы становится очевидно, что в случае, если b равно 125, то поведение системы не определено.

Рассмотрим пример нарушения свойства безопасности в системе. Пусть описаны два базовых протокола: $bp1$ с предусловием $phone(p, dial); balance > 0;$ и постусловием $phone(p, dialing); balance := balance - 1;$ и базовый протокол $bp2$ с предусловием $phone(p, dial); -50 <= balance <= 0$ и

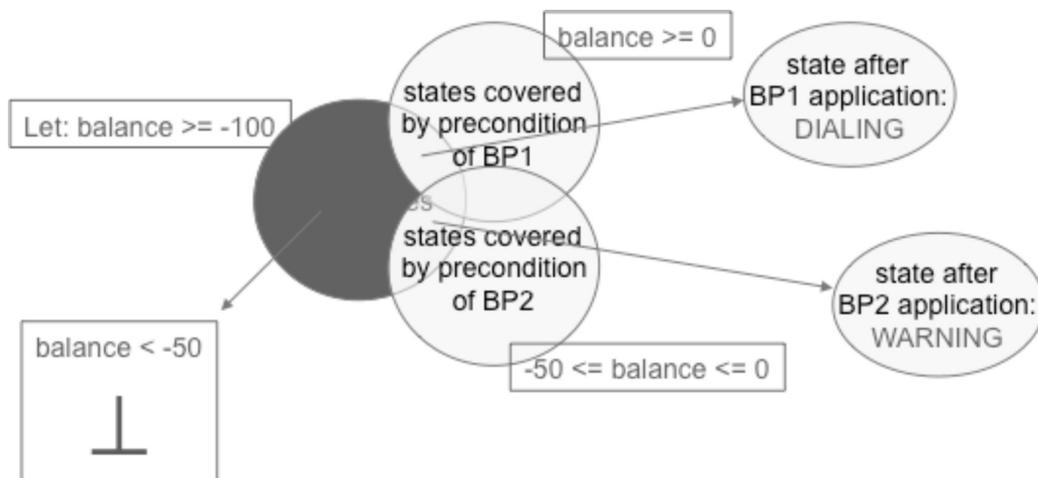


Рис. 2. Неполнота в требованиях

постусловием $\text{phone}(p, \text{warning})$. Пусть свойство безопасности задается формулой $\text{balance} > 0$.

В качестве результата статический анализатор вернет два вердикта:

BP1 breaks safety;

Precondition of BP2 breaks safety.

Для первого вердикта придется контр-пример: пусть $\text{balance} = 1$, применяется bp1 , $\text{balance} = 0$. Второй вердикт говорит о том, что если свойство выполняется, то протокол никогда не сможет быть применен, иначе свойство безопасности сформулировано неверно и должно быть скорректировано (рис. 3).

Автоматическое построение фильтров входных параметров. Существуют ситуации, когда внесение изменений в систему не представляется возможным, но требования для нее известны и могут быть верифицированы. Для таких систем можно провести анализ достижимости «плохих» состояний из начальных состояний системы. В случае если для каких-то входных значений система переходит в состояние тупика, недетерменизма или нарушения свойства безопасности, можно построить фильтр входных значений, который будет отбрасывать и протолировать такие воздействия на систему.

Для реализации данного подхода необходимо найти все «плохие» состояния и записать их в виде формулы безопасности. Если плохие состояния не достижимы из любого состояния системы, то можно проигнорировать такие «плохие состояния», иначе необходимо проверить, является ли состояние начальным. Если мы нашли начальное состояние из которого достижимо «плохое» со-

стояние, то статический анализатор создаст формулу, которая определит, какие входные условия необходимо отбросить. Данная формула может быть статически упрощена и скомпилирована в исполняемый модуль фильтрации, который устанавливается перед входом системы. В данный момент реализована система создания фильтров из полученных ограничений на язык Си.

Статический анализ с применением Microsoft Spec Explorer. Тема проверки моделей активно разрабатывается и компанией Microsoft в рамках Microsoft Research. Одним из инструментов, открытых для использования, является программа Spec Explorer [6]. Данный инструмент кодирует поведение системы в исполняемую форму (модель программы). Модель программы может быть менее специфицирована, чем имплементация, она должна лишь адекватно отражать состояния, нуждающиеся в проверке.

Инструмент Spec Explorer вводит три различных группы требований и делит их по вопросам: что система должна делать, что система может делать, что система не должна делать. Данные требования закладываются в систему путем описания на языке Spec# и AsmL.

Модель программы подается на model checker на явных состояниях, который позволяет искать (включая бесконечное число состояний) последовательности вызовов, в которых либо выполняются пред- и постусловия и инварианты, либо выполняется пользовательский тест. Система может найти только подозрительные на ошибки состояния, но при этом сразу же может преобразовать их в тесты и запустить на имплементации

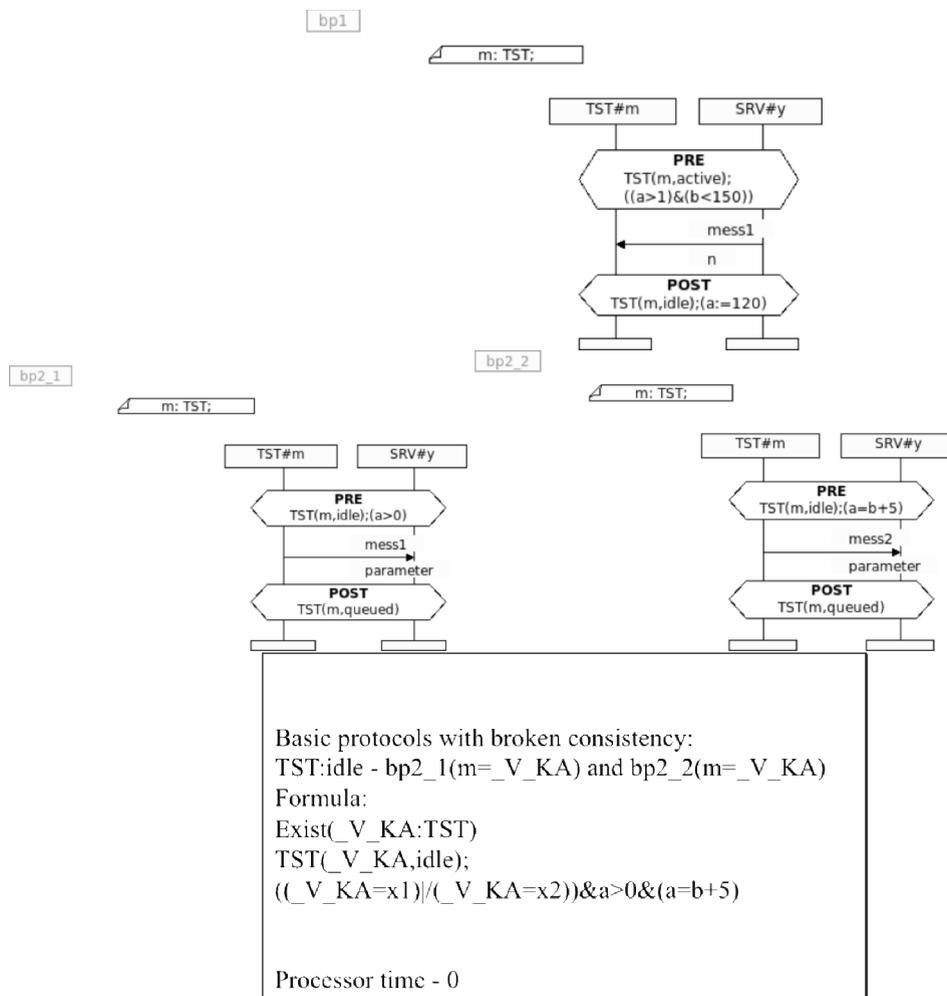


Рис. 3. Результат работы статического анализатора

программы. Ошибки, которые позволяет найти система Spec Explorer, могут относиться к одному из следующих типов: ошибка реализации (ошибка в коде программы); в модели; в спецификации; ошибка в дизайне (логическая непоследовательность в предполагаемом поведении системы).

Microsoft Spec Explorer позволяет переиспользовать результаты верификации для дальнейшего тестирования. Может одновременно проводить статический анализ модели, генерацию тестов по этой модели и анализ покрытия имплементации тестами. Однако он ориентирован только на платформу – Microsoft .NET. [6] и не может быть

применен к другим популярным языкам и платформам.

Технология статического анализа требований, описанная в данной статье, была применена в нескольких телекоммуникационных проектах и доказала свою эффективность в поиске ошибок на самых ранних стадиях разработки. Ее область применимости не ограничена телекоммуникационными проектами и может распространяться на любые реактивные системы, формализуемые с помощью описанной в статье нотации.

Работа поддержана грантом РФФИ 11-07-90412-Укр_ф_а

СПИСОК ЛИТЕРАТУРЫ

1. Hoare, C.A.R. Communicating Sequential Processes [Text] / C.A.R. Hoare. –Prentice Hall, 1985. –256 p.
2. Летичевский, А.А. Спецификация систем с помощью базовых протоколов [Текст] / А.А. Летичев-

ский, Ю.В. Капитонова, В.А. Волков, О.А. Летичевский, С.Н. Баранов, В.П. Котляров, Т. Вейгарг // Кибернетика и системный анализ. –2005. –№ 4.
3. Потененко, С.В. Статическая проверка требова-

ний и подходы к решению проблемы достижимости [Текст] / С.В. Потиеенко // Искусственный интеллект. –2009. –№ 1. –С. 192–197.

4. **Jones, C.** Software Defect-Removal Efficiency [Text] / C. Jones // IEEE Computer. –1996. –Р. 94–95.

5. **Shull.** What We Have Learned About Fighting Defects [Text] / Shull [et al.] // IEEE Computer. –2002. –Р. 78–81.

6. **Veanes, M.** Model-Based Testing of Object-Oriented Reactive Systems with Spec Explorer, in Formal Methods and Testing [Text] / M. Veanes, C. Campbell, W. Grieskamp [et al.]. –Springer Verlag, 2008. –Vol. 49. –Р. 39–76.

7. **Baranov, S.** The technology of Automation Verification and Testing in Industrial Projects [Text] / S. Baranov, V. Kotlyarov, A. Letichevsky, P. Drobintsev // Proc. of SPb IEEE Chapter, International Conf., May 18-21, 2005. –Р. 81–86.

8. **Никифоров, И.** Генерация формальной модели системы по требованиям, заданным в нотации USE CASE MAPS [Текст] / И. Никифоров, А. Петров, Ю. Юсупов // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2010. –№ 4 (103). –С. 191–195.

УДК 004.415

И.В. Никифоров, А.В. Петров, В.П. Котляров

СТАТИЧЕСКИЙ МЕТОД ОТЛАДКИ ТЕСТОВЫХ СЦЕНАРИЕВ, СГЕНЕРИРОВАННЫХ С ИСПОЛЬЗОВАНИЕМ ЭВРИСТИК

В современном подходе к разработке программного обеспечения активно используются модели приложений различной детальности. Они позволяют уже на ранних стадиях проектирования проводить статический анализ поведения модели, а также создавать тестовые сценарии, поэтому задача отладки моделей и автоматически генерируемых по модели тестовых сценариев на сегодняшний день является актуальной и важной для создания качественного программного продукта.

Пример современной технологии, в которой применяется модель для верификации и тестирования, – система автоматической генерации и

исполнения тестов VRS/ТАТ [1]. Для высокоуровневого описания модели в рассматриваемой технологической цепочке используется нотация Use Case Maps (UCM) [2, 3].

В настоящей работе стандартная нотация UCM, позволяющая в гибкой и удобной форме задавать потоки управления, расширена средствами задания потоков данных [4] с помощью механизма метаданных. Метаданные определяют условия изменения состояний приложения, реализуемые элементом UCM: описывают состояния предусловий и постусловий, взаимодействия с перемненными системы, а также действия и сигналы,

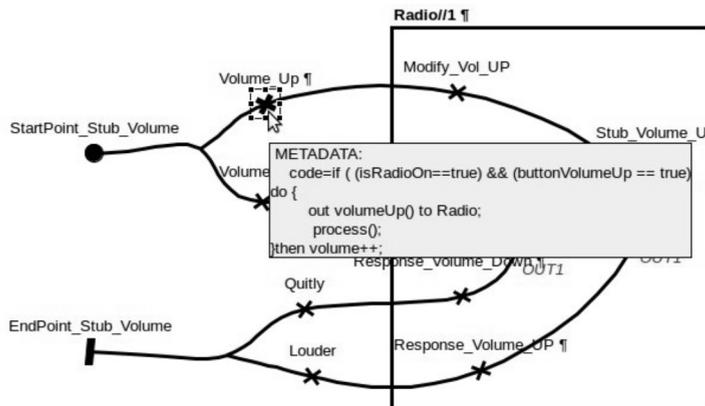


Рис. 1. Использование метаданных на UCM элементе

отправляемые и принимаемые. На рис. 1. приведен простой пример использования метаданных, описывающий процесс увеличения громкости радио, при нажатии на кнопку ‘Volume Up’.

В предусловие входят две переменные, которые проверяют состояние радио и нажатую кнопку. После чего происходит отправление сигнала увеличения громкости и изменение соответствующего параметра.

Использование одновременно потока управления и потока данных в модели делает нотацию UCM мощным языком моделирования промышленных проектов.

Для отладки модели в технологии VRS/TAT используется интерактивный режим, позволяющий в пошаговом режиме статического анализа отслеживать состояния модели приложения и диагностировать проблемные места поведения модели. На проектах средней и большой сложности подобный процесс отладки трудоемок, что приводит к существенным временным затратам в процессе разработки тестовых сценариев. Поэтому ставится задача создания методов быстрого и гибкого статического анализа для отладки автоматически генерируемых тестовых сценариев.

Автоматическая генерация тестовых сценариев

Структура процесса генерации символьных сценариев приведена на рис. 2. По исходным требованиям к системе вручную строится UCM модель. Методика формализации описана в работе [4]. Созданная UCM модель с помощью транслятора UCM2BP [5] преобразуется в формальную модель на языке базовых протоколов [6], являющемся входным языком для инструментов VRS/TAT.

Основная проблема процесса генерации тестовых сценариев по модели – взрыв состояний. Дерево поведения модели уже для проектов среднего размера может содержать огромное число поведений, поэтому полный обход всех поведений практически невозможен. Для решения этой проблемы используются эвристики (гиды) [7], которые содержат упорядоченные подмножества подцелей, направляющих поиск в соответствии с критериями покрытия актуальных поведений [8].

Затем формальная модель на языке базовых протоколов и эвристики поступают на вход трассового генератора VRS (см. рис. 2), осуществляю-

щего автоматический обход дерева и построение символьных тестовых сценариев.

Результат трассовой генерации – информация о покрытии, т. е. информация о том, по каким эвристикам тестовые сценарии сгенерированы успешно, а по каким – нет.

Задача автоматического получения символьных трасс по эвристикам не всегда решается без проблем. Причина заключается в особенности построения эвристик инструментом UCM2MSC, учитывающим зависимости от потока управления (явно отображенного на UCM диаграммах), но не учитывающим зависимости от данных, входящих в предусловия базовых протоколов и влияющих на поток управления. По этой причине некоторые эвристики не могут быть покрыты тестами, что приводит к анализу причин непокрытия, сосредоточенных либо в UCM модели, либо в автоматически сгенерированной эвристике.

Для целей отладки и выяснения причин расхождения эвристик с трассами используется анализатор последовательностей событий UCM диаграммы – UCM Events Analyzer (UCM EVA). Процесс анализа и отладки на рис. 2 выделен жирными линиями.

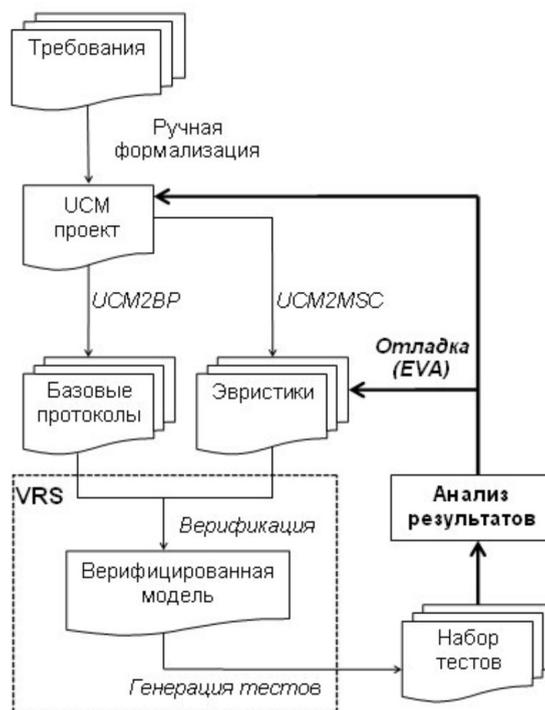


Рис. 2. Процесс генерации тестовых сценариев с использованием инструмента статического анализа – EVA

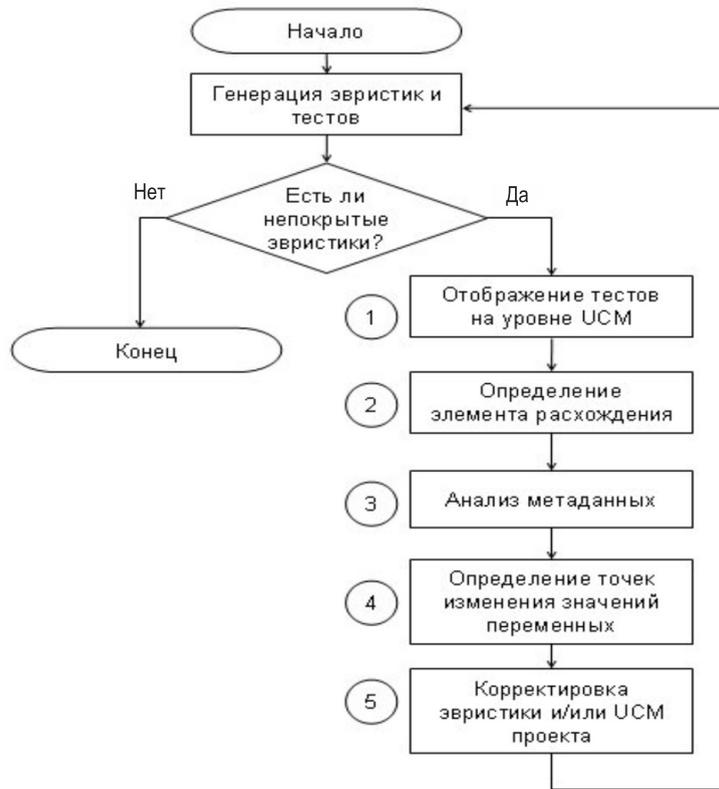


Рис. 3. Алгоритм поиска ошибок в эвристиках и их корректировка

Методика поиска ошибок

К наиболее частым причинам расхождения относятся: недостаточная детализация эвристик и ошибки в последовательности UCM элементов в эвристике из-за некорректного использования переменных (ошибки идентифицируются как deadlock на ветвях или ветвлениях).

В статье предложен метод поиска мест и причин ошибок, выявленных для случаев расхождения эвристики с трассой. На рис. 3 представлен итерационный алгоритм поиска и исправления ошибок.

1. Эвристики и полученные в VRS трассы представлены в виде MSC диаграмм с последовательностью имен базовых протоколов, поэтому первым шагом алгоритма является отображение базовых протоколов на UCM элементы.

2. Сравнивая эвристику и трассу в терминах UCM элементов, можно определить последний элемент трассы, удовлетворяющий порядку UCM элементов эвристики. Следующий за ним элемент назовем *элементом расхождения*.

3. По непокрытому в символьной трассе элементу эвристики – элементу расхождения можно рассмотреть соответствующие ему метаданные и

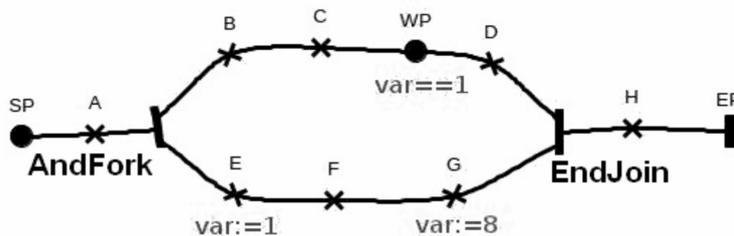


Рис. 4. UCM диаграмма с двумя параллельными участками и использованием разделяемых ресурсов

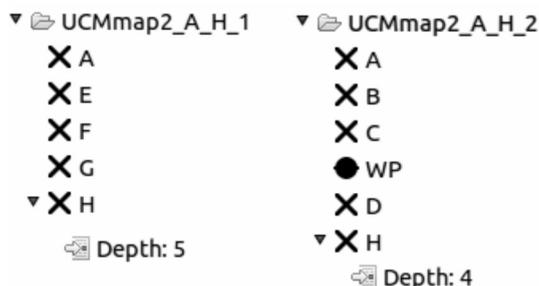


Рис. 5. Эвристики в терминах UCM, покрывающие ветви UCM диаграммы

зафиксированное в них предусловие.

4. Далее следует выделить переменные предусловия в отдельный список и проанализировать только те места на UCM диаграмме, в которых осуществляется изменение переменных этого списка. Анализ следует вести снизу вверх, начиная с событий, ближайших к элементу расхождения.

5. После обнаружения причины расхождения необходимо скорректировать эвристику или внести изменения в UCM модель.

Предложенную последовательность действий требуется повторять до тех пор, пока все эвристики не будут покрыты трассами.

Пример анализа UCM проекта

Наглядная иллюстрация поиска расхождения трассы с эвристикой – пример многопоточной системы с разделяемыми ресурсами. На рис. 4 приведен пример UCM диаграммы с двумя параллельными участками, заданными с помощью элементов параллельного ветвления и объединения AndFork и AndJoin.

На верхнем пути использован элемент WP (WaitingPlace) с условием 'var==1' для ожидания потоком момента его выполнения. На нижнем пути переменная 'var' изменяет свое значение дважды. Сначала ей присваивается значение '1', затем '8'.

С помощью инструмента UCM2MSC проводится генерация набора диаграмм, покрывающего UCM проект по критерию ветвей. Результатом работы являются две эвристики (рис. 5), которые обходят верхнюю и нижнюю ветви рассматриваемого примера.

На следующем шаге генерируются трассы по эвристикам. После запуска трассового генератора инструмент VRS создает файл вердикта. Вердикт показывает, что обе эвристики остались непокрытыми. Генератор, пытаясь покрыть эвристику, сохраняет трассу на каждом шаге генерации, т. е. после каждого успешного применения очередного базового протокола. Это позволяет анализировать сохраненные непокрытые трассы, в которых зафиксировано расхождение.

Используя механизм сравнения, реализованный в инструменте UCM EVA, проводится анализ эвристики и трассы по предложенному методу. На рис. 6. показана вкладка анализа инструмента EVA, предназначенная для поиска элемента расхождения эвристики и трассы.

Несложно увидеть, что элементом расхождения между эвристикой и трассой является элемент 'WP'. Это означает, что не выполняется условие 'var==1' при попытке пройти через этот элемент. Теперь можно определить, какие другие элементы используют данную переменную и в каком отношении они находятся с элементом 'WP'.

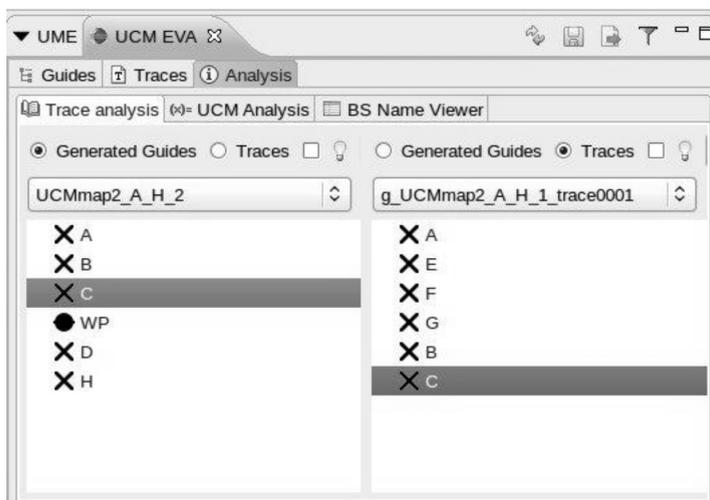


Рис. 6. Анализ эвристики и трассы для поиска элемента расхождения

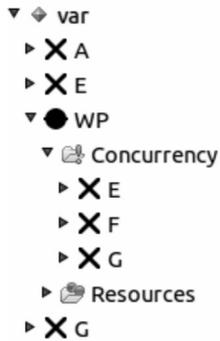


Рис. 7. UCM элементы, использующие переменную 'var'

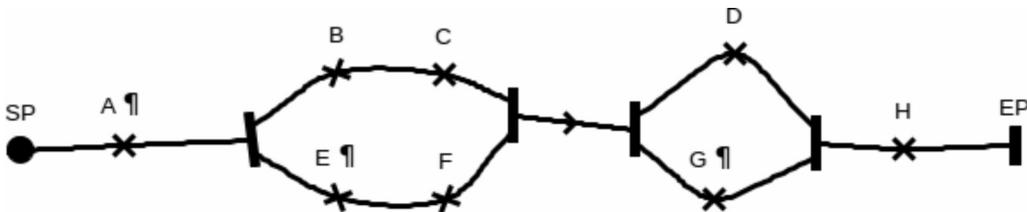


Рис. 8. UCM проект с синхронизацией элементов, использующих разделяемый ресурс 'var'

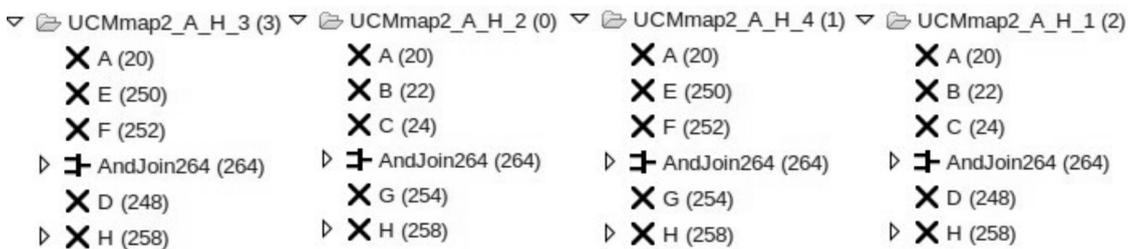


Рис. 9. Последовательности, покрывающие все ветки UCM диаграмм

Из рис. 7 видно, что переменная 'var' присутствует на четырех элементах проекта, в то время как два из них принадлежат параллельной ветви с 'WP'. Более детальное изучение UCM показывает, что переменная 'var' не использует синхронизацию

на параллельных ветвях UCM, что приводит к непредсказуемому результату исполнения системы и потенциальным тупикам, связанным с гонкой данных. Чтобы исправить ошибку, нужно скорректировать UCM, добавив синхронизацию (рис. 8).

После повторной генерации эвристик на выходе VRS получаются четыре последовательности, покрывающие все ветви проекта (рис. 9).

Предложенный в работе метод итерационного анализа расхождения трасс и эвристик позволяет сократить количество шагов анализа с общего числа UCM элементов в трассе, до числа элементов, в метаданных которых используется

переменная, участвующая в условиях расхождения. Предложенный метод визуализирует и существенно упрощает процесс создания и отладки тестовых сценариев.

Работа поддержана грантом РФФИ 11-07-90412-Укр_ф_а.

СПИСОК ЛИТЕРАТУРЫ

1. **Веселов, А.О.** Автоматизация тестирования в области телекоммуникаций [Текст] / А.О. Веселов, В.П. Котляров // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2010. –№4 (103). –С. 180–185.
2. **Никифоров, И.В.** Использование Use Case Map в современной технологической цепочке разработки программного обеспечения [Текст] / И.В. Никифоров, Ю.В. Юсупов // XXXVIII Неделя науки СПбГПУ: Матер. межвуз. науч.-техн. конф. –СПб.: Изд-во Политехнического ун-та, 2009. –С. 92–93.

3. Recommendation ITU-T Z.151 [Электронный ресурс] / User requirements notation (URN), 11/2008.
4. **Никифоров, И.В.** Генерация формальной модели системы по требованиям, заданным в нотации USE CASE MAP [Текст] / И.В. Никифоров, А.В. Петров, Ю.В. Юсупов // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2010. –№ 4 (103). –С. 191–195.
5. **Никифоров, И.В.** Преобразование высокоуровневого дизайна программного продукта в нотации Use Case Map в модель на формальном языке [Текст] / И.В. Никифоров, Ю.В. Юсупов // Технологии Microsoft



в теории и практике программирования. 16-17 марта 2010. –СПб.: Изд-во Политехнического ун-та, 2010. –С. 161–162.

6. **Летичевский, А.А.** Спецификация систем с помощью базовых протоколов [Текст] / А.А. Летичевский, Ю.В. Капитонова, А.А. Летичевский (мл.) [и др.] // Кибернетика и системный анализ. –2005. –№4. –С. 3–21.

7. **Воинов, Н.В.** Применение метода эвристик для создания оптимального набора тестовых сцена-

риев [Текст] / Н.В. Воинов, В.П. Котляров // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2010. –№ 4 (103). –С. 169–174.

8. **Дробинцев, П.Д.** Автоматизация тестирования на основе покрытия пользовательских сценариев [Текст] / П.Д. Дробинцев, В.П. Котляров // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. –2012. –№ 4 (152).

УДК 004.4'22

Б.В. Тютин, И.В. Никифоров, В.П. Котляров

ПОСТРОЕНИЕ СИСТЕМЫ АВТОМАТИЗАЦИИ СТАТИЧЕСКОЙ И ДИНАМИЧЕСКОЙ ПРОВЕРКИ ТРЕБОВАНИЙ К ПРОГРАММНОМУ ПРОДУКТУ

При создании программного продукта по мере детализации его архитектуры объем требований к нему увеличивается. Детализация спецификации требований, ее изменение и расширение приводят к существенным временным затратам на поддержание целостности и однозначности содержащейся в ней информации. Методы статического анализа проверяют сами требования, тогда как методы динамического анализа проверяют соответствие реализации программного продукта требованиям, их адекватность поставленной задаче и существующим ограничениям. Полнота и качество анализа требований играют ключевую роль в успехе проекта [1]. К статическим методам контроля требований принадлежат верификация и контроль семантики формализации, к динамическим – тестирование.

Таким образом, возможность использования тестовых наборов различной детализации и их параллельное выполнение позволят проводить тестирование на всех этапах разработки продукта. Автоматизация тестирования в данной ситуации существенно снижает затраты на его реализацию. Поддержка взаимосвязи между тестовыми сценариями и требованиями дает возможность в режиме реального времени адаптировать процесс тестирования в случаях изменения спецификации.

Цель работы. Основная цель работы состоит в создании технологии автоматизированного

тестирования многокомпонентного программного продукта с помощью системы тестирования, имеющей распределенную структуру и позволяющей производить автоматическое исполнение тестов с учетом архитектуры тестируемой системы. Тестовый набор создается путем проверки формальной модели системы с помощью символического верификатора, тем самым обеспечивается его корректность и покрытие требуемых свойств.

В рамках данной технологии тестовые наборы получаются на основе формального представления системы, создаваемого вручную с помощью инструмента UCM Specification Translator (UST) и Verificator of Requirement Specification (VRS). Система тестирования TestCommander используется для автоматической генерации исполняемого кода и управления тестированием.

Формализация требований. Для обеспечения возможности однозначной интерпретации требований при создании тестового набора необходимо из неявного или неформального вида преобразовать их в нотацию, обеспечивающую смысловую инвариантность при работе со спецификацией системы. При этом требуется, чтобы она была удобна для восприятия и редактирования человеком. Выбранная категория задач тестирования также требует возможности описывать параллельные взаимодействия. Этим требованиям отвечает Use Case Map (UCM) нотация,

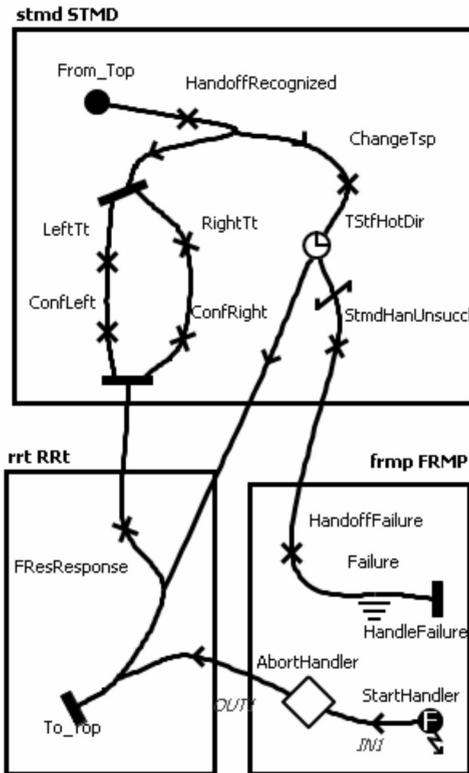


Рис. 1. UCM диаграмма для телекоммуникационного проекта

стандартизованная в ITU-T в 2003 г. [2]. UCM представляет собой набор связанных и структурированных диаграмм, каждая из которых состоит из последовательности UCM элементов. В совокупности набор диаграмм задает возможные поведения системы, описанные в требованиях [4]. На рис. 1 приведен пример UCM диаграммы, описывающий часть телекоммуникационного протокола.

В отличие от стандартных средств редактирования UCM, UST поддерживает возможность работы с потоком данных, задания точек отправки и приема сигналов и добавления условий ветвления и прерываний посредством работы с грамматикой метаданных. Информация из метаданных используется при формализации и полностью отображается в модели системы. Так же была добавлена возможность описания окружения системы, которое содержит начальные значения и атрибуты компонент.

Для автоматического перехода от высокоуровневого дизайна, заданного в нотации UCM, к формальному описанию на языке базовых прото-

колов [6], используется транслятор UST (рис. 2). К особенностям UST можно отнести: анализ UCM на наличие ошибок; оптимизации генерируемой модели в нотации базовых протоколов; структурирование модели по компонентам, позволяющее проводить верификацию и генерацию тестов, абстрагируясь от отдельных частей системы [7].

С помощью данного инструмента происходит построение UCM модели системы, внесение необходимых метаданных в модель и ее трансляция в нотацию базовых протоколов. Разработанный UCM проект проверяется модулем UST для анализа формальной модели на наличие ошибок, по-

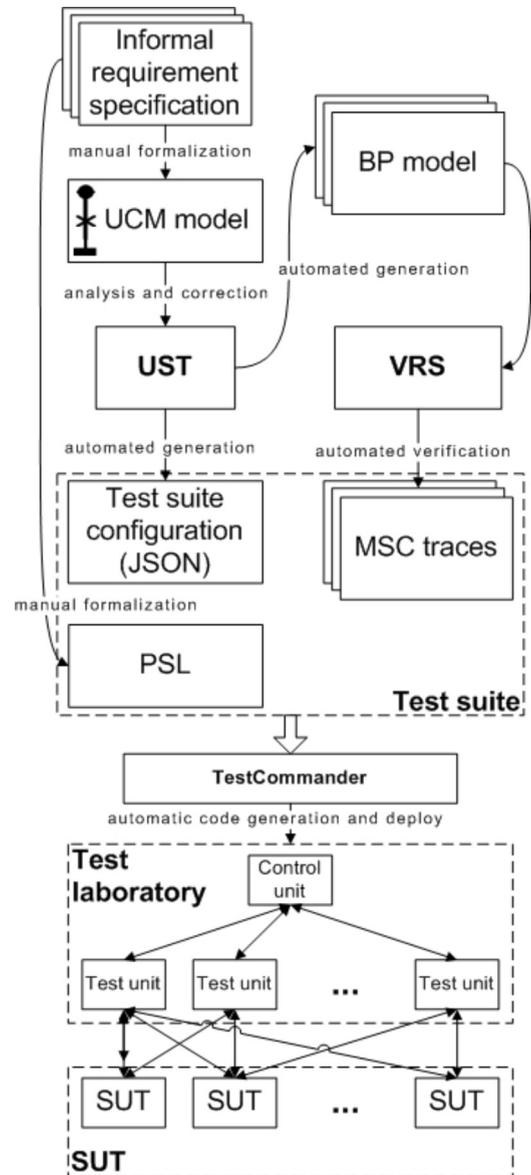


Рис. 2. Общий вид технологической цепочки



сле чего происходит их исправление. Затем, используя UST, пользователь проводит настройку опций трансляции и выбор метода генерации. После трансляции полученный набор базовых протоколов импортируется в символьный верификатор VRS, в котором происходит верификация модели и генерация символьных сценариев, покрывающих модель по заданному критерию [10]. Полученный таким образом набор MSC диаграмм конкретизируется с помощью внесения реальных значений данных и может использоваться в качестве описания тестового набора или для визуализации свойств и поведения системы.

Автоматизация тестирования. Для осуществления проверки системы на соответствие требованиям необходимо обеспечить контролируемый процесс запуска тестов и анализа их результатов [5]. Для этих целей создана система автоматизации тестирования TestCommander. С ее помощью на основании тестовых сценариев, записанных в нотации MSC [3], автоматически создаются приложения на одном из целевых языков программирования (C++, Java, скриптовые языки). На этапе исполнения они взаимодействуют с тестируемой системой и друг с другом посредством заранее определенных интерфейсов, воспроизводя заложенные в тестовый набор сценарии поведения. Для автоматизированного построения тестового набора в рамках данной работы предлагается технологическая схема, представленная на рис. 2.

Являясь объединением методов формализации требований и генерации поведенческих трасс на основе модели базовых протоколов [9], данный подход позволяет объединить управление требованиями, верификацию и тестирование. При этом автоматически созданные поведенческие сценарии описывают взаимодействие всех сущностей системы и ее окружения.

В TestCommander допускается использование любого способа создания MSC диаграмм. Рассматриваемый в статье метод обладает следующими преимуществами:

- автоматизацией рутинных этапов работы;
- простотой и ориентированностью на системного аналитика используемых формальных нотаций;
- применением методов и средств верификации для проверки требований [6].

Тестовый набор состоит из одного управляющего и нескольких тестирующих модулей. Тестирующий модуль взаимодействует с тестируемой системой (**system under test** или **SUT**) согласно ло-

гике теста и обменивается контрольными сигналами с управляющим модулем, руководящим тестированием и сбором результатов. Для определения протокола взаимодействия между элементами тестирующей системы, а также самой тестирующей системы с окружением, используется язык спецификации протоколов (Protocol specification language, PSL). Он позволяет задать вид сообщений, которыми обмениваются участвующие в тестировании сущности. В рамках рассматриваемого в данной статье подхода предполагается ручное создание спецификации на языке PSL.

Для конфигурирования тестового набора, задания параметров генерации кода и развертывания процесса тестирования применяется конфигурационный файл в формате JSON [8]. Он позволяет определить физическое расположение модулей тестового набора и SUT, а также сконфигурировать тестовый набор, заменив часть элементов тестируемой системы тестирующими модулями. При этом проверяется поведение не всей системы, а только отдельных ее компонент. Данный файл создается автоматически на основании представления системы в UCM, и на этапе создания тестового набора пользователь может вручную задать необходимые значения.

После определения параметров тестового набора (участвующих в тестировании элементов системы, физическое расположение тестового набора и т. п.), автоматически создается код тестового набора на языке целевой платформы и происходит развертывание среды тестирования. Запуск теста осуществляется путем старта процесса управляющего модуля тестового набора, который самостоятельно осуществляет запуск и останов тестирующих модулей и SUT. Результаты формируются в виде журнала событий и MSC диаграмм хода тестирования.

Возможность гибкой настройки позволяет располагать тестирующие модули на различных рабочих станциях в рамках тестовой лаборатории. Ввиду независимости отдельных сценариев тестирования при наличии свободных мощностей возможно разбиение тестового набора на сегменты, исполняемые параллельно. Вместе эти свойства обеспечивают необходимые условия для организации высокопроизводительной виртуальной лаборатории тестирования в облачной или кластерной инфраструктуре.

Ограничения использования технологии. Рассмотренный подход чрезвычайно эффективен

ввиду практически полной автоматизации рутинных операций по созданию кода и внесению изменений в спецификацию. Однако используемые технологии и архитектура системы тестирования накладывают ряд ограничений:

формальное представление требований в расширенной нотации UCM;

генерацию тестовых сценариев средствами верификатора VRS;

распределенность архитектуры.

Стандартный UCM позволяет описывать только функциональные свойства системы, характеризующие поток событийного управления. Для описания данных вычислительного процесса и семантики событий потока управления UCM расширен дополнительными смысловыми элементами – метаданными, а информация метаданных активно использована в алгоритме UST.

Инструмент верификации VRS позволяет построить модель системы в виде машины состояний в нотации базовых протоколов. Артефактом процесса верификации требований в данной системе являются MSC диаграммы, используемые в качестве тестовых сценариев для генерации кода тестового набора. Процесс обеспечения полного покрытия требований тестами для реальных промышленных систем сложен, но в данной технологии он решается путем автоматической генерации тестовых сценариев на основе формальной спецификации эвристик, задающих актуальные режимы поведения системы, и используемых в VRS при анализе модели системы и генерации трасс.

Еще один ограничивающий фактор – архитек-

тура системы тестирования. Она ориентирована на удаленное взаимодействие элементов, жесткую спецификацию протоколов, наличие определенных прав доступа к аппаратным средствам и подготовленную лабораторию тестирования.

Перечисленные ограничения не препятствуют успешному применению разработанной технологии для телекоммуникационных систем.

Главный аспект данной работы – интеграция управления требованиями, верификации и тестирования. Это существенно снизило влияние, оказываемое изменением спецификации программного продукта на процесс тестирования. Рассмотренный метод позволяет осуществлять проверку системы на соответствие требованиям в едином производственном цикле. Тестирование осуществляется на основании набора тестовых сценариев, корректность которых доказана при верификации модели системы. При этом существенно снижаются затраты, связанные с регрессионным тестированием при изменении требований или их дополнении. Этапы данного технологического процесса частично или полностью автоматизированы. Программные инструменты, относящиеся к различным этапам работы, взаимонезависимы; использующиеся форматы представления данных стандартизованы. Разработанные методики и программные средства опробованы в проектах тестирования протокола SIP и отдельных компонентах телекоммуникационных систем, использующих протоколы стандартов WiMax и LTE.

Работа поддержана грантом РФФИ 11-07-90412-Укр_ф_а

СПИСОК ЛИТЕРАТУРЫ

1. **Леффингуелл, Д.** Принципы работы с требованиями к программному обеспечению [Текст] / Д. Леффингуелл, Д. Уидриг. – М.: Вильямс, 2002.

2. **Buhr, R.J.A.** Use Case Maps for Object-Oriented Systems [Text] / R.J.A. Buhr, R.S. Casselman. – London: Prentice Hall, 1996.

3. ITU-T Recommendation Z.120: Message sequence chart (MSC) [Электронный ресурс] / Geneva, Switzerland, Oct. 1996 // Режим доступа: <http://eu.sabotage.org/www/ITU/Z/Z0120e.pdf>

4. ITU-T Recommendation Z.151 : User requirements notation (URN) [Электронный ресурс] / Language definition. Geneva, Switzerland, Sept. 2003 // Режим доступа: <http://www.itu.int/rec/T-REC-Z.151-200811-I/en>

5. **Kaner, C.** Testing Computer Software [Text] / C. Kaner, J. Falk, H.Q. Nguyen; 2nd ed. – New York: Wiley, 1999.

6. **Letichevsky A.** Basic protocols, message sequence charts, and the verification of requirements specifications [Text] / A. Letichevsky [et al.] // Computer Networks: The International J. of Computer and Telecommunications Networking. – 5 Dec. 2005. – Vol. 49. – № 5. – P. 661–675.

7. **Никифоров, И.** Генерация формальной модели системы по требованиям, заданным в нотации USE CASE MAPS [Текст] / И. Никифоров, А. Петров, Ю. Юсупов // Научно-технические ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. – 2010. – № 4 (103). – С. 191–195.

8. RFC 4627: The Application/JSON Media Type for JavaScript Object Notation (JSON), July 2006 [Электронный ресурс] / Режим доступа: <http://www.ietf.org/rfc/rfc4627.txt>

9. **Veselov, A.O.** Testing automation of projects in telecommunication domain [Text] / A.O. Veselov, V.P. Kotlyarov // Proc. of the 4th Spring/Summer Young Researchers' Colloquium on Software Engineering. –Nizhny Novgorod, June 1-2, 2010. –P. 81–86.

10. **Baranov, S.** The technology of Automation Verification and Testing in Industrial Projects [Text] / S. Baranov, V. Kotlyarov, A. Letichevsky, P. Drobintsev // Proc. of St.Petersburg IEEE Chapter, International Conf., SPb., May 18-21, 2005 –P. 81–86.

УДК 004.415

П.Д. Дробинцев, В.П. Котляров, И.Г. Черноруцкий

АВТОМАТИЗАЦИЯ ТЕСТИРОВАНИЯ НА ОСНОВЕ ПОКРЫТИЯ ПОЛЬЗОВАТЕЛЬСКИХ СЦЕНАРИЕВ

Программное обеспечение (ПО) активно используется в различных областях деятельности человека. При этом следует отметить, что сложность разрабатываемых программных комплексов неуклонно растет, что приводит к появлению существенных проблем при анализе их поведения. В то же время требования к качеству постоянно ужесточаются в силу того, что ПО все чаще используется в областях, критичных к ошибкам и сбоям. Подобная ситуация приводит к необходимости разработки и внедрения новых методов анализа поведения программного продукта на фазе тестирования. Одно из наиболее активно развивающихся направлений в этой области – использование формальных моделей при разработке программ. К данному направлению относится огромное количество как средств разработки (CASE-средств), так и методов, используемых для построения моделей программ. Это, например, нотация языка UML [1] и поддерживающие ее средства разработки таких фирм, как IBM (линейка Rational) [6], Microsoft, Altova [7] и др. или нотации для описания программ в терминах процессов, такие, как BPMN [8] и IDEF [9], поддерживаемые в инструментах компаний Visual Paradigm и CA Technologies. Представленные средства позволяют проводить анализ функционирования программ на различных этапах разработки, начиная с самых ранних стадий, что позволяет существенно сократить издержки на выявление дефектов и таким образом повысить качество разрабатываемого ПО.

Следует отметить, что использование CASE средств в процессе разработки ПО связано с дополнительными трюдзатратами на построение

формальных моделей. Существенно снизить данные издержки возможно путем применения простой для инженерной практики нотации описания моделей. В данной статье рассматривается подход, основанный на использовании стандартизированной нотации UCM [5] для построения формальной модели системы с последующей генерацией поведенческих сценариев для целей тестирования.

Нотация UCM. Язык UCM является стандартом описания поведения системы в виде определения совокупности взаимодействующих процессов. Отличительные особенности языка: графическая нотация, использующая небольшое множество основных элементов, достаточное для описания функционирования сложных программных комплексов; использование высокоуровневых абстракций для создания многоуровневых описаний сложных программных комплексов, сохраняющих семантические особенности; объединенное представление потоков управления и данных в моделях. Следует отметить, что представленные особенности играют существенную роль, т. к. подход, описанный в настоящей статье, предполагает тесное взаимодействие с заказчиком ПО при формализации UCM сценариев поведения проектируемого ПО.

Пример простой UCM диаграммы изображен на рис. 1. На диаграмме приведена спецификация системы, описывающая подмножество функций медиаплеера. Для уменьшения сложности приложения в описании введены элементы – Stub, абстрагирующие часть описываемого поведения для упрощения представления функциональности системы в целом.

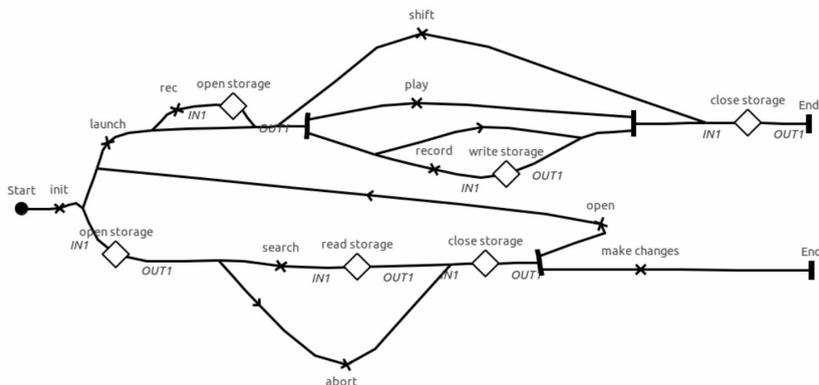


Рис. 1. Пример UCM диаграммы

Детальное описание языка UCM приведено в работах [3, 5].

Подход к проверке качества. Использование формальных моделей при разработке программного обеспечения дает возможность применить как формальные методы проверки правильности ПО, так и различные методы автоматической генерации, нацеленные на получение целевого кода приложения и тестов. UCM модель представляет полную совокупность сценариев поведения системы и позволяет получить подмножество сценариев, достаточное для покрытия поведения модели тестами в соответствии с выбранными критериями. В предлагаемом подходе UCM модель используется для получения двух типов сценариев:

- с полным покрытием поведения системы по ветвям;

- с покрытием путей, определенных пользователем.

Получение множества сценариев первого типа связано с полным обходом UCM диаграммы и автоматической генерацией путей, ведущих в каждую ветвь. Детальное описание метода обхода может быть найдено в [3]. Однако опыт тестирования реальных программных проектов показывает, что применение только критерия ветвей недостаточно, т. к. при этом не рассматривается ряд сценариев поведения системы, важных для пользователя. Например, конечное прохождение циклов с определенным количеством повторов или различные варианты поведения параллельных взаимодействующих процессов. Отсутствие в полученном автоматически тестовом наборе тестов, направленных на проверку подобных поведений, приводит к ухудшению качества проверяемого ПО. Таким образом, возникает проблема описания и последующего тестирования сценариев с

покрытием путей, определенных пользователем.

Трудоемкость создания пользовательского поведенческого сценария связана в первую очередь со степенью его детализации. В случае недостаточной детализации в сгенерированном тесте будут отсутствовать необходимые тестовые воздействия на систему, что чревато невозможностью исполнения пользовательского сценария, тогда как при избыточно детальном описании сценария трудозатраты на его создание будут достаточно велики. Для уменьшения трудозатрат при детализации пользовательских сценариев предлагается использовать подход, основанный на формировании гида, т. е. ручного указания для генерации сценария только тех точек (элементами UCM диаграммы) поведенческого сценария, которые интересны пользователю. Последующая детализация сценария осуществляется автоматически генератором/верификатором сценариев. Схема предложенного подхода изображена на рис. 2.

На первом этапе генератор сценариев по заданному пользователем точкам проводит генерацию детального сценария. В случае если расстояние между указанными пользователем точками больше одного UCM элемента, инструмент генерации вычисляет путь максимальной длины между двумя точками, и эта длина используется генератором для детализации сценария между этими точками.

На втором этапе инструментарий верификации [4] осуществляет конкретизацию сценария путем подбора такого пути между точками, который может быть пройден с учетом начальных установок в системе и ограничений, указанных в метаданных каждого элемента UCM диаграммы.

Язык описания пользовательских сценариев. Для задания пользовательских сценариев в предложенном подходе используется специали-

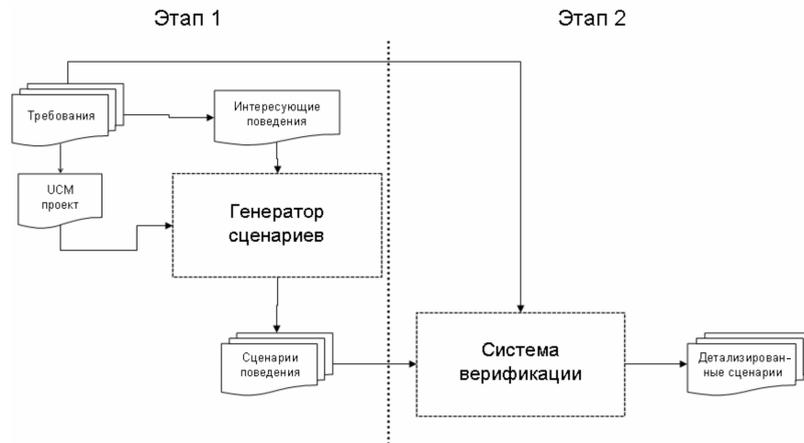


Рис. 2. Технологическая цепочка генерации пользовательских сценариев

зированной язык GDL (guides definition language). Данный язык построен на основе определения пользователем формул гидов четырех различных типов.

- формулы подмодели M определяются как $M = U \subseteq AU$, где AU – множество всех элементов UCM диаграммы. Формулы данного типа позволяют определить подмодель путем указания множества элементов UCM диаграммы. Таким образом, формула M описывает элементы, которые будут использоваться при обходе UCM диаграммы;

- формулы альтернативного выбора S определяются как $S = u \in AU$, где AU – множество всех элементов UCM диаграммы. Данные формулы позволяют задать подмножество M указанием одного элемента. При этом в определяемое множество M попадут все элементы, достижимые из элемента u , а также все элементы, лежащие на путях из точки s UCM диаграммы в точку u ;

- определение формулы пользовательских последовательностей R : $R = u_1 * n \rightarrow u_2 * n \rightarrow \dots \rightarrow u_n * n, \{u_1, \dots, u_n\} \in AU$, где AU – множество всех элементов UCM диаграммы и n – количество повторений рассматриваемого элемента u_i при описании цикла. Таким образом, формулы определяют последовательности UCM элементов u , которые должны быть представлены в генерируемом сценарии;

- формулы T определяются как $T = \{M^*, S^*, R^*\}$. Формулы данного типа определяют совокупность правил, описанных в формулах типа M , S , R , которая будет использована при генерации конкретного тестового набора.

Рассмотрим пример использования языка

GDL для спецификации. Предположим, что поведение системы описывается спецификацией, представленной на рис. 3.

С точки зрения покрытия по ветвям в автоматическом режиме будет сгенерировано четыре тестовых сценария, где « \rightarrow » определяет последовательную композицию протоколов, « (n) » определяет количество протоколов между двумя элементами UCM диаграммы:

- A, B – для покрытия B;
- A, F – для покрытия A и F;
- A, C(2), D – для покрытия C и D;
- A, C(2), E – для покрытия E.

Как видно из представленных тестов, каждая ветвь системы будет покрыта как минимум один раз. При этом в процессе уточнения трассы все дерево поведения будет пройдено.

Предположим, что пользователь хочет сократить время генерации и не рассматривать полный обход дерева поведения при работе с большими программными проектами. Для решения этой задачи могут использоваться формулы M и S . Например, уберем из рассмотрения ветку F, для этого необходимо задать формулу $M1 = A, B, C, D, E$. При этом последовательность элементов не важна, т. к. данной формулой формируется подмодель нашей системы. И определим формулу для тестового набора $T1 = M1$. В результате будет сгенерировано два теста: A, B, C, D и A, B, C, E, которые в совокупности покроют все элементы диаграммы за исключением элемента F. Аналогично можно решить данную задачу с использованием формулы $S1 = B$, данный вариант более предпочтителен в силу меньших трудозатрат в определении формул со стороны пользователя. Таким образом, форму-

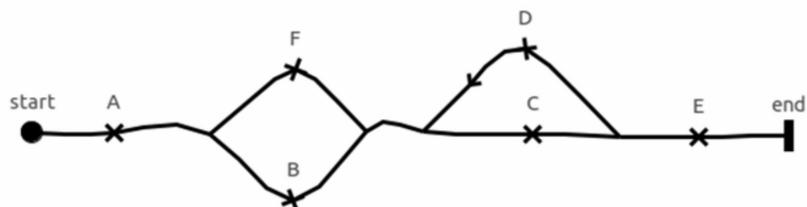


Рис. 3. Пример модели поведения системы

лы M и S позволяют решить проблемы, связанные с необходимостью частичного рассмотрения модели и дают возможность сокращения как обхода дерева поведения системы при уточнении сценария, так и уменьшения тестового набора.

В тестах, полученных для рассматриваемого примера, обеспечено покрытие по ветвям, однако не рассмотрены проблемы множественного прохождения цикла, содержащего элемент D . Следует отметить, что рассмотрение этого вопроса может быть важным при обеспечении качества. Так, например, пользователя может интересовать поведение системы при максимальном количестве прохождений цикла, определяемом метаданными, скрытыми в элементах диаграммы (предположим, что в этом примере это число равно пяти). Для создания подобного тестового набора необходимо воспользоваться формулой $R = C, D*5, E$, задающей пятикратное повторение цикла и выход из него в точку E . В результате применения этой формулы в процессе генерации тестового набора будет получено две тестовых последовательности: $A, B, C, D, D(2), C, E$ и $A, F, C, D, D(2), C, E$, покрывающие пятикратное прохождение цикла по всем возможным путям, начинающимся в стартовой точке. В отличие от тестовых наборов, нацеленных на покрытие поведения системы по ветвям,

использование данных тестов при тестировании представленной системы поможет выявить дефекты, связанные с некорректной обработкой данных при множественном прохождении цикла.

Таким образом, использование формул в процессе создания тестовых наборов позволяет решить следующие проблемы, возникающие при создании специфических тестовых сценариев:

определение пользовательских сценариев путем задания интересующих точек поведения на UCM диаграмме, при этом путь от точки к точке будет вычислен с использованием системы верификации, что гарантирует его корректность;

сокращение дерева поведения при его обходе с целью покрытия поведения по ветвям;

определение специфических сценариев для циклических конструкций и параллельного поведения системы.

В общем случае практика применения подхода, основанного на использовании покрытия пользовательских сценариев, дает возможность существенного увеличения тестового покрытия и за счет этого повышения качества разрабатываемого программного обеспечения.

Работа частично поддержана международным грантом РФФИ 11-07-90412-Укр_ф_а.

СПИСОК ЛИТЕРАТУРЫ

1. Грейди, Буч. Язык UML. Руководство пользователя = The Unified Modelling Language user guide [Текст] / Буч Грейди, Джеймс Рамбо, Айвар Джекобсон; 2-е изд. – М.: – СПб.: ДМК Пресс; Питер, 2004. – 432 с.
2. Recommendation ITU-T Z.120. Message Sequence Chart (MSC), 11/2000 [Электронный ресурс].
3. Никифоров, И.В. Использование Use Case Map в современной технологической цепочке разработки программного обеспечения [Текст] / И.В. Никифоров, Ю.В. Юсупов // Технологии Microsoft в теории и практике программирования. Матер. межвуз. конкурса-конф. студентов, аспирантов и молодых ученых Северо-Запада. 16-17 марта 2010. – СПб.: Изд-во Политехнического ун-та, 2010. – С. 107–108.

4. Колчин, А.В. Разработка инструментальных средств для проверки формальных моделей асинхронных систем: Дис. ... канд. физ.-мат. наук [Текст] / А.В. Колчин. – Киев, 2009. – 140 с.
5. Z.151 : User requirements notation (URN) – Language definition [Электронный ресурс] / Режим доступа: <http://www.itu.int/rec/T-REC-Z.151-200811-I/en>
6. [Электронный ресурс] / Режим доступа: <http://www-01.ibm.com/software/ru/rational/>
7. [Электронный ресурс] / Режим доступа: <http://www.altova.com/umodel.html>
8. [Электронный ресурс] / Режим доступа: <http://www.bpmn.org/>
9. [Электронный ресурс] / Режим доступа: <http://www.idef.com/IDEF0.htm>



НАУЧНО-ТЕХНИЧЕСКИЕ ПЕРЕВОДЫ

“С точностью до буквы...”



ООО «Иновационные Лингвистические Технологии – РЭС»,

Группа Компаний «ABC-Translations»™

<http://www.ilt-res.com>

Научно-исследовательская организация «Иновационные Лингвистические Технологии – РЭС» (Россия, г. Нижний Новгород), входит в состав международной Группы Компаний «ABC-Translations»™ и успешно функционирует в России и за рубежом. Одно из основных направлений деятельности компании – оказание услуг в сфере лингвистики и высококачественного профессионального перевода научной и технической литературы и документации различных тематик по 42 языкам мира.

Мы предлагаем авторам, редакциям и издательствам:

- высококачественный перевод любой информации, включая книги, монографии, статьи, дайджесты, интервью;
- редактирование и вычитку готовых переводов;
- тематический подбор статей в иностранной прессе, с подготовкой краткого резюме для последующей подготовки к публикации;
- подбор информации и составление дайджестов зарубежной прессы;
- копирайтинг.

Специфика научно-технического перевода состоит в точной передаче специальных отраслевых терминов и аутентичности содержания. Мы сформировали команду высокопрофессиональных технических переводчиков, компетентных специалистов – каждый из них специализируется не только на техническом переводе в целом, но и на определенной узкой тематике. Благодаря этому, а также собственной системе двухступенчатой рекурсивной проверки и верификации текста, мы гарантируем качество и аутентичность перевода.

Доверяя научно-технический перевод профессионалам, Вы экономите время, силы и средства, и можете быть уверены в качестве выполнения заказа.

Контакты: Телефон/Факс: +7 (831) 433-77-14
E-mail: info@ilt-res.com, alexander.elin@ilt-res.com, abc-translations@bk.ru
Сайт: <http://www.ilt-res.com>



ISSCS 2013

11-th International Symposium on Signals, Circuits and Systems
Iasi, Romania, July 11 - 12, 2013

organized by:

Faculty of Electronics, Telecommunications and Information Technology "Gheorghe Asachi" Technical University Iasi
IEEE Romania Section & CAS Chapter in cooperation with IEEE CAS Society

Call for Papers

Honorary Chair

Bogdan Ioan

Technical University of Iasi,
Iasi, Romania

General Co-Chairs

Liviu Goras

Technical University of Iasi,
Iasi, Romania

Sanjit Mitra

University of Southern California,
Los Angeles, CA, USA

Technical Program Chair

Majid Ahmadi

University of Windsor,
Ontario, Canada

Technical Program Committee

Adnan Al-Alaoui, Lebanon
Canhui Cai, China
Marco Carli, Italy
Paul Dan Cristea, Romania
Paulo Diniz, Brazil
Karen Egiazarian, Finland
Daniel Foty, U.S.A.
Georges Gielen, Belgium
Gordana Jovanovic-Dolecek, Mexico
Pedro Julián, Argentina
André Kaup, Germany
Alexander Korotkov, Russia
Franco Maloberti, Italy
Rui Paulo da Silva Martins, Macau
Yoshikazu Miyanaga, Japan
Asoke Nandi, UK
Georgi Nenov, Bulgaria
Branimir Reljin, Serbia
Salil Sanyal, India
Mihaela van der Schaar, U.S.A.
Mohamad Sawan, Canada
Erchin Serpendin, U.S.A.
Thanos Stouraitis, Greece
Mani Soma, U.S.A.
Wonyong Sung, Korea
M.N.S. Swamy, Canada
Alle-Jan van der Veen, The Netherlands
Lars Wanhammar, Sweden
Siddik Yarman, Turkey

ISSCS 2013 aims at bringing together scientists and researchers from academia and industry to present and discuss some of their latest ideas and results.

Prospective authors are invited to submit original papers in the following areas:

- Linear and Nonlinear Circuits & Systems
- Circuits & Systems for Communications
- Power Electronic Circuits
- Analog and Digital Signal Processing
- Image and Video Processing
- VLSI Circuits & Systems

The official language of the symposium will be English.

AUTHOR'S SCHEDULE:

Proposals for special sessions	December 3, 2012
Submission of full paper	February 11, 2013
Notification of acceptance	April 8, 2013
Submission of camera-ready version and registration form with signed copyright form	May 6, 2013

The papers should be written in IEEE/CAS A4 format (double column including paper title, authors' names and affiliation, short abstract). They should not exceed 4 pages (including figures and references).

Authors are expected to present their papers at the Symposium. At least one author of each accepted paper must register in order to have the paper included in the program and Proceedings.

ISSCS 2013

<http://scs.etti.tuiasi.ro/isscs2013>

Faculty of Electronics, Telecommunications and Information Technology
Bd. Carol I no.11, Iasi
700506, ROMANIA

Conference email address:
isscs2013@etti.tuiasi.ro



СВЕДЕНИЯ ОБ АВТОРАХ

КОНТАКТНЫЕ ДАННЫЕ

АКИМОВ Валерий Петрович – профессор кафедры радиофизики Санкт-Петербургского государственного политехнического университета, доктор физико-математических наук.

195251, Санкт-Петербург, ул. Политехническая, д. 29. E-mail: akimov@radio.rphf.spbstu.ru

БАРМИН Максим Анатольевич – старший преподаватель кафедры прикладной информатики Пензенской государственной технологической академии.

440605, г. Пенза, проезд Байдукова/ул. Гагарина, д. 1а/11

БОНДАРЧУК Сергей Сергеевич – старший научный сотрудник Томского государственного университета систем управления и радиоэлектроники, доктор физико-математических наук.

634050, г. Томск, пр. Ленина, д. 40.

БОРИСЕВИЧ Алексей Валерьевич – доцент кафедры автоматов Санкт-Петербургского государственного политехнического университета, кандидат технических наук.

195251, Санкт-Петербург, ул. Политехническая, д. 29. E-mail: alex.borysevych@gmail.com

ВАСИЛЬЕВ Владимир Андреевич – аспирант кафедры информатики Санкт-Петербургского государственного университета.

198504, Санкт-Петербург, Старый Петергоф, Университетский пр., д. 28.

ВЕРШИНИНА Лилия Павловна – профессор, заведующий кафедрой информатики и информационных технологий Санкт-Петербургского государственного университета культуры и искусств, доктор технических наук.

191186, Санкт-Петербург, Дворцовая наб., д. 2/4. E-mail: zk-inf@yandex.ru

ВОИНОВ Никита Владимирович – научный сотрудник кафедры информационных и управляющих систем факультета технической кибернетики Санкт-Петербургского государственного политехнического университета, кандидат технических наук.

195291, Санкт-Петербург, ул. Политехническая, д. 21.

ВОЛВЕНКО Сергей Валентинович – доцент кафедры радиоэлектронных средств защиты информации Санкт-Петербургского государственного политехнического университета, кандидат технических наук.

195291, Санкт-Петербург, ул. Политехническая, д. 29.

ГАНДЖА Василий Викторович – аспирант кафедры моделирования и основ теории цепей Томского государственного университета систем управления и радиоэлектроники.

634050, г. Томск, пр. Ленина, д. 40. E-mail: vasivik@gmail.com

ГЛЫБОВСКИЙ Станислав Борисович – аспирант кафедры радиофизики Санкт-Петербургского государственного политехнического университета.

195251, Санкт-Петербург, ул. Политехническая, д. 29. E-mail: sgbs@mail.ru

ГОЛУБ Юлия Александровна – аспирант кафедры системного анализа и управления Санкт-Петербургского государственного политехнического университета.

195251, Санкт-Петербург, ул. Политехническая, д. 29. Тел. 8(812)591-66-58.

ГРИГОРЬЕВА Анастасия Викторовна – аспирант кафедры информатики Санкт-Петербургского государственного университета.

198504, Санкт-Петербург, Старый Петергоф, Университетский пр., д. 28.

ДАРМАХЕЕВ Валерий Владимирович – аспирант кафедры систем информатики Восточно-Сибирского государственного университета технологий и управления.

670013, г. Улан-Удэ, ул. Ключевская, д. 40 в. E-mail: darmaheev@yandex.ru

ДЕРКАЧЕНКО Валентин Николаевич – профессор кафедры прикладной математики и исследования операций в экономике Пензенской государственной технологической академии, кандидат технических наук.

440605, г. Пенза, проезд Байдукова/ул. Гагарина, д. 1а/11.

ДМИТРИЕВ Вячеслав Михайлович – заведующий кафедрой моделирования и основ теории цепей Томского государственного университета систем управления и радиоэлектроники, доктор технических наук, профессор.

634050, г. Томск, пр. Ленина, д. 40. Тел. 8(382-2)41-39-15, e-mail: decan@toe.tusur.ru

ДМИТРИЕВ Павел Иванович – аспирант кафедры информатики и информационных технологий Санкт-Петербургского государственного университета культуры и искусств.

191186, Санкт-Петербург, Дворцовая наб., д. 2/4. E-mail: Pavel.Dmitriev@billing.ru

ДОДОХОВ Александр Леонидович – руководитель направления защиты баз данных ЗАО «Аладдин Р.Д.».

101000, Москва, Милютинский пер., д. 14. E-mail: a.dodokhov@aladdin-rd.ru

ДРОБИНЦЕВ Павел Дмитриевич – доцент кафедры информационных и управляющих систем факультета технической кибернетики Санкт-Петербургского государственного политехнического университета, кандидат технических наук.

195291, Санкт-Петербург, ул. Политехническая, д. 21.

ЗАБОРОВСКИЙ Владимир Сергеевич – заведующий кафедрой телематики Санкт-Петербургского государственного политехнического университета, доктор технических наук, профессор.

195251, Санкт-Петербург, ул. Политехническая, д. 29.

ЗАВЬЯЛОВ Сергей Викторович – ассистент кафедры радиоэлектронных средств защиты информации Санкт-Петербургского государственного политехнического университета.

195291, Санкт-Петербург, ул. Политехническая, д. 29.

ЗИКРАТОВ Игорь Алексеевич – заведующий кафедрой безопасных информационных технологий Санкт-Петербургского национально-исследовательского университета информационных технологий, механики и оптики, доктор технических наук, профессор.

197101, Санкт-Петербург, Кронверкский пр., д. 49. E-mail: zikratov@cit.ifmo.ru

ЗОЛОТОВ Олег Иванович – заведующий кафедрой процессов управления и информационных систем Санкт-Петербургского государственного горного университета, кандидат технических наук, профессор.

191186, Санкт-Петербург, ул. Миллионная, д. 5.

ЗУБКОВ Александр Фёдорович – декан факультета Института промышленной экономики, информатики и сервиса Пензенской государственной технологической академии, кандидат технических наук, профессор.

440605, г. Пенза, проезд Байдукова/ул. Гагарина, д. 1а/11. E-mail: zubkov@pgta.ru

ЗУДИЛОВА Татьяна Викторовна – заместитель директора Санкт-Петербургского национально-исследовательского университета информационных технологий, механики и оптики, доцент, кандидат технических наук.

197101, Санкт-Петербург, Кронверкский пр., д. 49.

ИВАНОВ Александр Сергеевич – магистр кафедры информационных и управляющих систем факультета технической кибернетики Санкт-Петербургского государственного политехнического университета.

195291, Санкт-Петербург, ул. Политехническая, д. 21. E-mail: al.s.ivanov@gmail.com

КОТЛЯРОВ Всеволод Павлович – профессор кафедры информационных и управляющих систем факультета технической кибернетики Санкт-Петербургского государственного политехнического университета, кандидат технических наук.

195291, Санкт-Петербург, ул. Политехническая, д. 21. E-mail: vpk@ics2.ecd.spbstu.ru

КОТЛЯРОВА Лина Павловна – старший преподаватель кафедры информационных и управляющих систем факультета технической кибернетики Санкт-Петербургского государственного политехнического университета.

195291, Санкт-Петербург, ул. Политехническая, д. 21.

КРИВОШЕЙКИН Анатолий Валентинович – заведующий кафедрой технической электроники и электротехники Санкт-Петербургского государственного университета кино и телевидения, доктор технических наук, профессор.

191119, Санкт-Петербург, ул. Правды, д. 13.

ЛЕТИЧЕВСКИЙ Александр Адольфович – академик НАНУ, Институт кибернетики имени В.М. Глушкова НАНУ, доктор физико-математических наук.

03680 ГСП, г. Киев-187, пр. Глушкова, д. 40.
E-mail: let@cyfra.net

ЛУКАШИН Алексей Андреевич – аспирант кафедры телематики Санкт-Петербургского государственного политехнического университета.

195251, Санкт-Петербург, ул. Политехническая, д. 29. E-mail: lukash@neva.ru

ЛЫЧАГИН Николай Иванович – заместитель генерального директора ОАО «Интелтех», доктор технических наук, профессор.

197342, Санкт-Петербург, ул. Кантемировская, д. 8. E-mail: nil@ntc1.inteltech.ru

ЛЯШЕНКО Александр Леонидович – доцент кафедры процессов управления и информационных систем Санкт-Петербургского государственного горного университета, кандидат технических наук.

191186, Санкт-Петербург, ул. Миллионная, д. 5.
E-mail: akuna_matata_kmv@mail.ru

МАКАРОВ Сергей Борисович – заведующий кафедрой радиоэлектронных средств защиты информации Санкт-Петербургского государственного политехнического университета, доктор технических наук, профессор.

195251, Санкт-Петербург, ул. Политехническая, д. 29. E-mail: makarov@cee.spbstu.ru

МАЛЫГИН Леонид Леонидович – директор ИМИТ (филиал СПбГПУ), доктор технических наук, профессор.

162600, Вологодская обл., г. Череповец, ул. Первомайская, д. 48. Тел. 8(8202)24-02-36.

МОЛЧАНОВ Александр Павлович – доцент кафедры технической электроники и электротехники Санкт-Петербургского государственного университета кино и телевидения, кандидат технических наук.

191119, Санкт-Петербург, ул. Правды, д. 13.

МОРЕВА Светлана Леонидовна – инженер-программист Ленинградской атомной электростанции, аспирант кафедры процессов управления и информационных систем Санкт-Петербургского государственного горного университета.

191186, Санкт-Петербург, ул. Миллионная, д. 5.
E-mail: sweetmoreva@yandex.ru

НИКИФОРОВ Игорь Валерьевич – аспирант кафедры информационных и управляющих систем Санкт-Петербургского государственного политехнического университета.

195251, Санкт-Петербург, ул. Политехническая, д. 29. E-mail: igor.nikiforovv@gmail.com

ОСИПОВ Никита Алексеевич – доцент кафедры программных систем Санкт-Петербургского национального исследовательского университета информационных технологий, механики и оптики, кандидат технических наук.

197101, Санкт-Петербург, Кронверкский пр., д. 49.

ПЕТРОВ Алексей Викторович – магистр Санкт-Петербургского государственного политехнического университета.

195251, Санкт-Петербург, ул. Политехническая, д. 29.

ПОДДУБНЫЙ Игорь Евгеньевич – магистр кафедры информационных и управляющих систем факультета технической кибернетики Санкт-Петербургского государственного политехнического университета.

195291, Санкт-Петербург, ул. Политехническая, д. 21.

РАЗУМОВСКИЙ Андрей Владимирович – ассистент кафедры безопасных информационных технологий Санкт-Петербургского национально-исследовательского университета информационных технологий, механики и оптики, кандидат технических наук.

197101, Санкт-Петербург, Кронверкский пр., д. 49.
E-mail: xrew@yandex.ru

РОСТОВ Николай Васильевич – доцент кафедры систем автоматического управления Санкт-Петербургского государственного политехнического университета, кандидат технических наук.

195291, Санкт-Петербург, ул. Политехническая, д. 21. Тел. 8(812) 297-59-60.

САБАНОВ Алексей Геннадьевич – заместитель генерального директора ЗАО «Аладдин Р.Д.», кандидат технических наук.

101000, Москва, Милютинский пер., д. 14. E-mail: a.sabanov@aladdin-rd.ru

САФОНОВ Владимир Олегович – профессор кафедры информатики Санкт-Петербургского государственного университета, доктор технических наук.

198504, Санкт-Петербург, Старый Петергоф, Университетский пр., д. 28.

СЕМЕНОВ Петр Константинович – аспирант кафедры распределенных вычислений и компьютерных систем Санкт-Петербургского государственного политехнического университета.

195291, Санкт-Петербург, ул. Политехническая, д. 21. E-mail: spk@dcn.ftk.spbstu.ru

СПИВАК Антон Игоревич – доцент кафедры безопасных информационных технологий Санкт-Петербургского национально-исследовательского университета информационных технологий, механики и оптики, кандидат технических наук.

197101, Санкт-Петербург, Кронверкский пр., д. 49. E-mail: anton.spivak@gmail.com

ТЮТИН Борис Викторович – аспирант кафедры информационных и управляющих систем Санкт-Петербургского государственного политехнического университета.

195291, Санкт-Петербург, ул. Политехническая, д. 21. E-mail: b.tyutin@gmail.com

ХОРУЖНИКОВ Сергей Эдуардович – директор Санкт-Петербургского национального исследовательского университета информационных технологий, механики и оптики, доцент, кандидат физико-математических наук.

197101, Санкт-Петербург, Кронверкский пр., д. 49.

ЦАРЕВ Владимир Александрович – заведующий кафедрой программного обеспечения вычислительной техники и автоматизированных систем ИМИТ (филиал СПбГПУ), кандидат технических наук.

162600, Вологодская обл., г. Череповец, ул. Гоголя, д. 33-А. Тел. 8(8202)23-13-40, e-mail: vats@imit.ru

ЧЕНЦОВ Александр Георгиевич – заведующий отделом Института математики и механики Уральского отделения РАН, член-корреспондент РАН, доктор физико-математических наук, профессор.

620990, г. Екатеринбург, ул. С. Ковалевской, д. 16.

ЧЕНЦОВ Павел Александрович – научный сотрудник Института математики и механики Уральского отделения РАН, кандидат физико-математических наук.

620990, г. Екатеринбург, ул. С. Ковалевской, д. 16.

ЧЕРКАС Павел Сергеевич – аспирант кафедры программного обеспечения вычислительной техники и автоматизированных систем ИМИТ (филиал СПбГПУ).

162600, Вологодская обл., г. Череповец, ул. Гоголя, д. 33-А. Тел. 8(8202)23-12-71, e-mail: pushka-k@mail.ru

ЧЕРНОРУЦКИЙ Игорь Георгиевич – профессор, заведующий кафедрой информационных и управляющих систем, декан факультета технической кибернетики Санкт-Петербургского государственного политехнического университета, доктор технических наук.

195291, Санкт-Петербург, ул. Политехническая, д. 21. E-mail: dean@ftk.spbstu.ru

ЧЕСНОКОВ Михаил Александрович – аспирант кафедры технической электроники и электротехники Санкт-Петербургского государственного университета кино и телевидения.

191119, Санкт-Петербург, ул. Правды, д. 13. E-mail: chesnokov.inc@gmail.com

ЩЕСНЯК Сергей Степанович – профессор научно-исследовательского центра прикладной электродинамики Санкт-Петербургского государственного университета аэрокосмического приборостроения, доктор технических наук.

190103, Санкт-Петербург, Рижский пр., д. 26. E-mail: sergey.schesnyak@scaegroup.com



АННОТАЦИИ

КЛЮЧЕВЫЕ СЛОВА

Заборовский В.С., Лукашин А.А. СИСТЕМА КОНТРОЛЯ ДОСТУПА В СРЕДЕ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ.

Предложена система контроля доступа в среде облачных вычислений. Разработана формализация процесса фильтрации в терминах теории категорий. Развит сетцентрический подход к системе контроля доступа, основанный на целостности, информационной связанности и локальности принятия решений компонентами рассматриваемой системы

ЗАЩИТА ИНФОРМАЦИИ. КОНТРОЛЬ ДОСТУПА. ТЕЛЕКОММУНИКАЦИОННЫЕ СИСТЕМЫ. МЕЖСЕТЕВОЕ ЭКРАНИРОВАНИЕ. ОБЛАЧНЫЕ ВЫЧИСЛЕНИЯ.

Осипов Н.А., Зудилова Т.В., Хоружников С.Э. ОБЛАЧНЫЕ ТЕХНОЛОГИИ ПРИ ОБРАБОТКЕ ДАННЫХ БОЛЬШОГО ОБЪЕМА.

Рассмотрена специфика обработки данных, содержащих несколько миллионов записей на примерах задач моделирования инфокоммуникационных систем. Предложен способ обработки данных большого объема с помощью технологии PowerPivot и с использованием облачных технологий.

ДАННЫЕ БОЛЬШОГО ОБЪЕМА. МОДЕЛИРОВАНИЕ СЛОЖНЫХ СИСТЕМ. ОБЛАЧНЫЕ ТЕХНОЛОГИИ. ЧАСТНОЕ ОБЛАКО.

Дмитриев В.М., Ганджа В.В. АЛГОРИТМ АНАЛИЗА И СЖАТИЯ ДАННЫХ ПО ТОПОЛОГИЧЕСКОЙ СТРУКТУРЕ ИССЛЕДУЕМОГО ОБЪЕКТА.

Введено новое понятие для способов редактирования входной информации, связанное с приданием активных свойств существующим в системах компьютерного моделирования редакторам исходных данных. Определены структуры и функциональные характеристики такого редактора, приведены конкретные алгоритмы, реализующие эти характеристики.

ИНФОРМАЦИЯ. МОДЕЛИРОВАНИЕ. ЦЕПИ. РЕДУКЦИЯ КОМПОНЕНТЫ.

Борисевич А.В. СИНТЕЗ АЛГОРИТМОВ СТАБИЛИЗАЦИИ НЕЛИНЕЙНЫХ ОБЪЕКТОВ НА ОСНОВЕ СХЕМЫ ПРОДОЛЖЕНИЯ РЕШЕНИЯ ПО ПАРАМЕТРУ.

Рассмотрено использование метода продолжения по параметру для стабилизации выхода класса нелинейных систем, не линеаризуемых по обратной связи. Изучена связь методов продолжения по параметру с линеаризацией систем по обратной связи. Описан принцип действия регулятора, использующего нелинейное преобразование координат и метод предиктора-корректора.

АФФИННЫЕ НЕЛИНЕЙНЫЕ СИСТЕМЫ. СТАБИЛИЗАЦИЯ. ЛИНЕАРИЗАЦИЯ ПО ОБРАТНОЙ СВЯЗИ. МЕТОД ПРОДОЛЖЕНИЯ ПО ПАРАМЕТРУ. МЕТОД ПРЕДИКТОРА-КОРРЕКТОРА.

Волвенко С.В., Завьялов С.В., Макаров С.Б. ИМИТАЦИОННАЯ МОДЕЛЬ СИСТЕМЫ ПЕРЕДАЧИ СПЕКТРАЛЬНО-ЭФФЕКТИВНЫХ СИГНАЛОВ С OFDM.

Рассмотрена имитационная модель системы передачи OFDM сигналов при использовании на каждой поднесущей частоте оптимальных спектрально-эффективных сигналов. Изучены два метода приема данных сигналов: алгоритм когерентного оптимального поэлементного приема и алгоритм Витерби. Получены графики зависимостей вероятности ошибочного приема сигналов по этим методам от отношения сигнал/шум для канала с постоянными параметрами и аддитивным нормальным белым шумом.

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ. СПЕКТРАЛЬНО-ЭФФЕКТИВНЫЕ СИГНАЛЫ. СИГНАЛЫ С OFDM. ПИК-ФАКТОР. АЛГОРИТМ ПРИЕМА. ПОМЕХОУСТОЙЧИВОСТЬ.

Бондарчук С.С., Додохов А.Л., Сабанов А.Г. ТЕХНОЛОГИЯ ЗАЩИТЫ ПЕРСОНАЛЬНЫХ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ СУБД ORACLE.

Рассмотрена задача возможности применения СУБД Oracle для хранения и обработки персональных данных. Разработана технология шифрования конфиденциальной информации в соответствии с ГОСТ 28147-89.

ЗАЩИТА ИНФОРМАЦИИ. ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ. ШИФРОВАНИЕ. ПЕРСОНАЛЬНЫЕ ДАННЫЕ.

Семенов П.К. ПРАКТИЧЕСКИЕ АСПЕКТЫ АЛГОРИТМА ТАЛА–ВАРДИ СПИСОЧНОГО ДЕКОДИРОВАНИЯ ПОЛЯРНЫХ КОДОВ.

Предложена улучшенная реализация алгоритма Тала–Варди списочного декодирования полярных кодов, предоставляющая возможность эффективной аппаратной реализации на платформах с фиксированной запятой.

ПОМЕХОУСТОЙЧИВОЕ КОДИРОВАНИЕ. СПИСОЧНОЕ ДЕКОДИРОВАНИЕ. ПОЛЯРНЫЕ КОДЫ.

Акимов В.П., Глыбовский С.Б., Щесняк С.С. КОРРЕКЦИЯ ФАЗОВЫХ ОШИБОК ПАРАБОЛИЧЕСКОГО ЗЕРКАЛА С ПОМОЩЬЮ ВТОРИЧНОГО СЕТЧАТОГО РЕФЛЕКТОРА.

Разработан численно-аналитический метод расчета коррекции фазовых ошибок неровного параболического зеркала вторичным рефлектором на основе ламинированной сетки. С помощью математического моделирования случайных реализаций поверхностных неровностей проанализировано влияние неровностей на КНД при различных параметрах корректирующего рефлектора.

ФАЗОВЫЕ ОШИБКИ. КОРРЕКТИРУЮЩИЙ РЕФЛЕКТОР. ПАРАБОЛИЧЕСКАЯ АНТЕННА. СЕТЧАТЫЙ ЭКРАН. КНД.

Ляшенко А.Л. РАЗРАБОТКА МЕТОДИКИ СИНТЕЗА РАСПРЕДЕЛЕННОГО П-РЕГУЛЯТОРА ПО ПОКАЗАТЕЛЮ КОЛЕБАТЕЛЬНОСТИ.

Рассмотрена возможность использования аппарата расширенных частотных характеристик (РЧХ) для анализа систем с распределенными параметрами (СПП). Сформулирована и решена задача по разработке методики расчета настроек распределенного П-регулятора. Представленная методика позволяет производить синтез распределенных регуляторов по показателю колебательности.

СИСТЕМЫ С РАСПРЕДЕЛЕННЫМИ ПАРАМЕТРАМИ. РАСШИРЕННЫЕ ЧАСТОТНЫЕ ХАРАКТЕРИСТИКИ. РАСПРЕДЕЛЕННЫЙ РЕГУЛЯТОР. ПЕРЕДАТОЧНЫЕ ФУНКЦИИ РАСПРЕДЕЛЕННЫХ ОБЪЕКТОВ. ПОКАЗАТЕЛЬ КОЛЕБАТЕЛЬНОСТИ.

Ростов Н.В. МНОГОКРИТЕРИАЛЬНОЕ НОРМИРОВАНИЕ ЭТАЛОННЫХ МОДЕЛЕЙ В ЗАДАЧАХ СИНТЕЗА СИСТЕМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ.

Рассмотрены задачи нормирования параметров операторных и канонических векторно-матричных эталонных моделей замкнутых систем высокого порядка. Предложен метод многокритериального нормирования на основе определения Парето-оптимальных значений нормируемых параметров. Разработан сценарий поиска таких значений по разным векторным критериям с применением процедур зондирования на сетках параметров или генетического алгоритма. Приведены примеры нормирования параметров эталонных моделей третьего порядка.

СИСТЕМЫ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ. ЭТАЛОННЫЕ МОДЕЛИ. МНОГОКРИТЕРИАЛЬНОЕ НОРМИРОВАНИЕ. ЗОНДИРОВАНИЕ. ПАРЕТО-ОПТИМАЛЬНЫЕ ЗНАЧЕНИЯ ПАРАМЕТРОВ.

Лычагин Н.И., Вершинина В.П., Дмитриев В.П. УПРАВЛЕНИЕ ЗНАНИЯМИ В СИСТЕМЕ МЕНЕДЖМЕНТА КАЧЕСТВА ПРЕДПРИЯТИЯ – РАЗРАБОТЧИКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.

Рассмотрены базовые процессы системы менеджмента качества предприятия – разработчика программного обеспечения. Разработаны пути создания системы управления знаниями, основным ресурсом знаний которой является документация на программное обеспечение.

УПРАВЛЕНИЕ ЗНАНИЯМИ. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ. МЕНЕДЖМЕНТ КАЧЕСТВА.

Голуб Ю.А. МЕТОДЫ И МОДЕЛИ ИССЛЕДОВАНИЯ И РАЗРАБОТКИ АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ ИЗБИРАТЕЛЬНОГО РАСПРОСТРАНЕНИЯ ИНФОРМАЦИИ И ИНИЦИАТИВНОГО ОБСЛУЖИВАНИЯ.

Предложена многоуровневая модель структуры информационной системы на основе определения системы, реализующего системно-целевой подход. Также разработаны методы оценки взаимосвязей между стратами многоуровневой модели, базирующиеся на идее решающих матриц Г.С. Поспелова и информационного подхода к анализу систем А.А. Денисова.

СИСТЕМА. ИНФОРМАЦИОННЫЕ ОЦЕНКИ. МЕТОД РЕШАЮЩИХ МАТРИЦ. МОДЕЛЬ.



Деркаченко В.Н., Зубков А.Ф., Ковалерова Н.В., Бармин М.А. МАТЕМАТИЧЕСКИЕ МОДЕЛИ ПРОГНОЗИРОВАНИЯ ДИНАМИКИ РАЗВИТИЯ РЫНКА ЖИЛОЙ НЕДВИЖИМОСТИ.

Рассмотрена поквартальная динамика изменения цен на первичном и вторичном рынках жилой недвижимости и дана сравнительная оценка динамики цен на рынке. Методом пошагового регрессионного анализа выявлены факторы, оказывающие существенное влияние на стоимость недвижимости.

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ. РЫНОК НЕДВИЖИМОСТИ. статистический анализ.

Кривошейкин А.В., Молчанов А.П., Чесноков М.А. ДИНАМИЧЕСКАЯ ФИЛЬТРАЦИЯ ЭЛЕМЕНТОВ РЕЧЕВЫХ СИГНАЛОВ И СИГНАЛОВ С НЕПЕРИОДИЧЕСКИМ ЛИНЕЙЧАТЫМ СПЕКТРОМ.

Предложена и исследована система адаптивной фильтрации аддитивной смеси шума и речевого сигнала с периодическим и непериодическим спектром.

АДАПТИВНАЯ ФИЛЬТРАЦИЯ. МОДЕЛИРОВАНИЕ. РЕЧЬ. СЛУХОВОЙ АППАРАТ. ОСНОВНОЙ ТОН.

Морева С.Л., Золотов О.И., Ляшенко А.Л. РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ МОДЕЛИ ТЕМПЕРАТУРНЫХ ПОЛЕЙ АКТИВНОЙ ЗОНЫ РЕАКТОРА РБМК-1000.

Рассмотрены проблемы анализа систем с распределенными параметрами на примере активной зоны реактора РБМК-1000. Представлено описание активной зоны реактора. Изложена методика составления математической модели распределенного объекта управления.

РЕАКТОР. ГРАФИТОВАЯ КЛАДКА. ОБЪЕКТЫ С РАСПРЕДЕЛЕННЫМИ ПАРАМЕТРАМИ. МОДЕЛИРОВАНИЕ.

Дармахеёв В.В. ЭФФЕКТИВНОСТЬ ПРИМЕНЕНИЯ РЯДА ГЕНЕТИЧЕСКИХ ОПЕРАТОРОВ СЕЛЕКЦИИ.

Проведен анализ эффективности применения различных генетических операторов селекции. Рассмотрена возможность повышения сходимости генетического алгоритма за счет применения различных операторов селекции. В результате проведенного исследования определены критерии для оценки эффективности применения операторов селекции.

ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ. ОПЕРАТОРЫ СЕЛЕКЦИИ. ЭФФЕКТИВНОСТЬ. РЯД. РАЗНООБРАЗИЕ ПОПУЛЯЦИИ.

Ченцов А.Г., Ченцов П.А. МАРШРУТИЗАЦИЯ ПЕРЕМЕЩЕНИЙ С ОГРАНИЧЕНИЯМИ И НЕСТАЦИОНАРНЫМИ ФУНКЦИЯМИ СТОИМОСТИ.

Проведено исследование задачи маршрутизации перемещений с выполнением работ в пунктах следования. Построены теоретический метод решения и оптимальный алгоритм, реализованный на ПЭВМ.

МАРШРУТ. ДИНАМИЧЕСКОЕ ПРОГРАММИРОВАНИЕ. УСЛОВИЯ ПРЕДШЕСТВОВАНИЯ. МЕГАПОЛИС.

Зикратов И.А., Спивак А.И., Разумовский А.В. МЕТОД РАЗГРАНИЧЕНИЯ ДОСТУПА К ОБЪЕКТАМ В СИСТЕМЕ КОНТРОЛЯ ВЕРСИЙ.

Рассмотрены проблемы разграничения доступа к объектам в системах контроля версий. Изложены проблемы ограничения доступа на уровне отдельных версий объекта. Предложен метод разграничения доступа и представлена реализация метода.

ДОСТУП. РАЗГРАНИЧЕНИЕ. ВЕРСИЯ. СКВ. ОБЪЕКТ.

Григорьева А.В., Васильев В.А., Сафонов В.О. ПРИМЕНЕНИЕ ЭЛЕМЕНТОВ АСПЕКТНО-ОРИЕНТИРОВАННОГО ПОДХОДА ASPECT.NET ПРИ РАЗРАБОТКЕ ГЕОЛОКАЦИОННОЙ СИСТЕМЫ УПРАВЛЕНИЯ РЕСУРСАМИ ДЛЯ ОБЛАЧНОЙ ПЛАТФОРМЫ MICROSOFT WINDOWS AZURE.

Рассмотрено применение элементов аспектно-ориентированного подхода ASPECT.NET при разработке открытой геолокационной системы управления ресурсами, позволяющей получать данные о местоположении ресурсов в реальном времени, а также оперативно реагировать на изменения с помощью сложной бизнес-логики правил и отчетов, тем самым эффективно решая задачу управления ресурсами.

АСПЕКТНО-ОРИЕНТИРОВАННЫЙ ПОДХОД. УПРАВЛЕНИЕ РЕСУРСАМИ. ОБЛАЧНАЯ ПЛАТФОРМА.

Воинов Н.В., Котлярова Л.П., Поддубный И.Е. МЕТОДИКА РАЗРАБОТКИ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ ДЛЯ ПЛАТФОРМЫ WINDOWS PHONE 7.

Описана методика создания приложений для современных мобильных устройств. В качестве основной платформы выбрана новая операционная система компании Microsoft – Windows Phone 7. Приведены результаты пилотирования методики при работе со студентами разных курсов.

МОБИЛЬНАЯ ПЛАТФОРМА. МОБИЛЬНОЕ УСТРОЙСТВО. МОБИЛЬНОЕ ПРИЛОЖЕНИЕ. MICROSOFT WINDOWS PHONE.

Малыгин Л.Л., Царев В.А., Черкас П.С. АВТОМАТИЧЕСКОЕ УПРАВЛЕНИЕ ПАРАМЕТРАМИ СРЕДСТВ ФОРМИРОВАНИЯ ИЗОБРАЖЕНИЙ В СИСТЕМАХ РАСПОЗНАВАНИЯ ТЕКСТОВЫХ МЕТОК РЕАЛЬНОГО ВРЕМЕНИ.

Рассмотрен способ повышения эффективности систем распознавания текстовых меток реального времени за счет автоматического управления параметрами средств формирования изображений. Описаны особенности определения качества изображений, выбора средств формирования изображений и управления их параметрами с помощью автоматических регуляторов. Указаны возможные области практического применения представленных решений.

РАСПОЗНАВАНИЕ ТЕКСТОВЫХ МЕТОК. «УМНЫЕ» КАМЕРЫ. СИСТЕМЫ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ. КАЧЕСТВО ИЗОБРАЖЕНИЙ. УПРАВЛЕНИЕ СРЕДСТВАМИ ФОРМИРОВАНИЯ ИЗОБРАЖЕНИЙ.

Иванов А.С., Котляров В.П., Летичевский А.А. АНАЛИЗ ТРЕБОВАНИЙ РЕАКТИВНЫХ СИСТЕМ.

Описан метод верификации требований многокомпонентных реактивных систем, заданных формальными моделями в нотации базовых протоколов. Рассмотрены примеры применения данной технологии и возможности дальнейшего практического использования. Указаны преимущества, ограничения и промышленные аналоги данного подхода.

ФОРМАЛЬНАЯ ВЕРИФИКАЦИЯ. ФОРМАЛЬНАЯ НОТАЦИЯ. ЛОГИКА ХОАРА. УПРАВЛЕНИЕ ТРЕБОВАНИЯМИ.

Никифоров И.В., Петров А.В., Котляров В.П. СТАТИЧЕСКИЙ МЕТОД ОТЛАДКИ ТЕСТОВЫХ СЦЕНАРИЕВ, СГЕНЕРИРОВАННЫХ С ИСПОЛЬЗОВАНИЕМ ЭВРИСТИК.

Представлен подход к отладке автоматически сгенерированных тестовых сценариев по высокоуровневым Use Case Map-спецификациям с использованием эвристик. Рассмотрен интерактивный алгоритм поиска и исправлений ошибок как в тестах, так и в модели. Применение алгоритма продемонстрировано на примере UCM проекта с двумя потоками.

АВТОМАТИЗАЦИЯ. ТЕСТЫ. ОТЛАДКА. UCM.

Тютин Б.В., Никифоров И.В., Котляров В.П. ПОСТРОЕНИЕ СИСТЕМЫ АВТОМАТИЗАЦИИ СТАТИЧЕСКОЙ И ДИНАМИЧЕСКОЙ ПРОВЕРКИ ТРЕБОВАНИЙ К ПРОГРАММНОМУ ПРОДУКТУ.

Описан метод автоматического построения тестового набора, покрывающего требования к системе, представленные в формальной нотации Use Case Maps. Рассмотрена автоматизированная технология проверки требований, основанная на данном методе. Указаны преимущества и ограничения разработанного подхода.

АВТОМАТИЗАЦИЯ ТЕСТИРОВАНИЯ. ФОРМАЛЬНАЯ НОТАЦИЯ. АВТОМАТИЧЕСКАЯ ГЕНЕРАЦИЯ КОДА. ВЕРИФИКАЦИЯ. УПРАВЛЕНИЕ ТРЕБОВАНИЯМИ.

Дробинцев П.Д., Котляров В.П., Черноруцкий И.Г. АВТОМАТИЗАЦИЯ ТЕСТИРОВАНИЯ НА ОСНОВЕ ПОКРЫТИЯ ПОЛЬЗОВАТЕЛЬСКИХ СЦЕНАРИЕВ.

Представлен подход к автоматизации тестирования на основе пользовательских сценариев, заданных в формате Use Case Map. Рассмотрена технологическая цепочка генерации тестов с их последующей детализацией. Применение подхода продемонстрировано на примере элементарного UCM проекта.

АВТОМАТИЗАЦИЯ. ТЕСТЫ. ВЕРИФИКАЦИЯ. UCM.

ABSTRACTS

KEYWORDS

Zaborovsky V.S., Lukashin A.A. ACCESS CONTROL SYSTEM IN CLOUD COMPUTING ENVIRONMENT.

Paper proposes network access control system for private clouds. Introduced formalization of filtering process using category theory terms. Paper develops network centric approach to access control systems based on integrity, information cohesion and local decisions about access control.

INFORMATION SECURITY. ACCESS CONTROL. TELECOMMUNICATION SYSTEMS. FIREWALLS. CLOUD COMPUTING.

Osipov N.A., Zudilova T.V., Khoruzhnikov S.E. CLOUD TECHNOLOGIES WHILE WORKING WITH BIG DATA.

The article describes methods of processing big data in excel' format, which contains millions records of the modelling samples in info-communication systems. Using clouding technologies will allow degreasing costs of information working and increase efficiency of scientific research. We suggest the PowerPivot technology implementation for big data processing in clouding.

BIG DATA. MODELLING COMPLEX SYSTEMS. CLOUD TECHNOLOGIES. PRIVATE CLOUD.

Dmitriev V.M., Gandzha V.V. ALGORITHM ANALYSIS AND DATA COMPRESSION THE TOPOLOGICAL STRUCTURE OF EXAMINED OBJECT.

In this article we introduce a new concept for ways to edit the input information associated with giving active properties of current systems in the computer simulation of the editors of input data. Determined by the structure and functional characteristics of the editor are specific algorithms that implement these characteristics.

INFORMATION. MODELLING. CIRCUIT REDUCTION. COMPONENT.

Borisevich A.V. SYNTHESIS OF ALGORITHMS FOR NONLINEAR STABILIZATION BASED ON NUMERICAL CONTINUATION APPROACH.

Method of numerical continuation for output stabilization of nonlinear systems class is presented. Feedback linearization and numerical homotopy continuation considered as equivalent methods for output zeroing. Regulator based on nonlinear coordinate transformation and predictor-corrector continuation method reviewed.

AFFINE NONLINEAR SYSTEMS. STABILIZATION. OUTPUT ZEROING. FEEDBACK LINEARIZATION. NUMERICAL CONTINUATION. PREDICTOR-CORRECTOR METHOD.

Volvenko S.V., Zavalov S.V., Makarov S.B. SIMULATION MODEL OF SYSTEM OF TRANSFER OF SPECTRAL-EFFECTIVE SIGNALS WITH OFDM.

The simulating model of system of an OFDM broadcast of signals is considered when using on each subcarrier frequency of optimum spectral-effective signals. Two methods of reception of these signals are considered: algorithm of coherent optimum bit-by-bit reception and algorithm of Viterbi. Graphs of dependences of error probability of signals reception on these methods from the relation a signal noise for the channel with constant parameters and additive normal white noise are received.

SIMULATION MODELLING. SPECTRAL-EFFECTIVE SIGNALS. OFDM SIGNALS. PEAK FACTOR. RECEPTION ALGORITHM. ERROR-RATE PERFORMANCE.

Bondarchuk S.S., Dodokhov A.L., Sabanov A.G. TECHNOLOGY FOR PROTECTION OF PERSONAL DATA USING ORACLE DBMS.

The problem of the possibility of using oracle database for storage and processing of personal data. Developed a technology for encryption of confidential information in accordance with GOST 28147-89.

INFORMATION SECURITY. INFORMATION SECURITY. ENCRYPTION. PERSONAL DATA.

Semenov P.K. PRACTICAL ISSUES OF TAL-VARDY ALGORITHM OF POLAR CODE LIST DECODING.

Improved implementation of Tal-Vardy algorithm for polar codes list decoding is proposed. It enables efficient fixed-point implementation of the decoder.

ERROR-CORRECTING CODING. LIST DECODING. POLAR CODES.

Akimov V.P., Glybovskiy S.B., Shchesnyak S.S. A CORRECTION OF PARABOLIC MIRROR'S PHASE ERRORS BY MEANS OF SECONDARY MESHED REFLECTOR.

A hybrid numerical-analytical method for computing a correction of uneven parabolic mirror's phase errors by means of secondary meshed reflector is developed. Involving mathematical modelling of random surface distortions an influence of unevenness on antenna directivity has been examined for different correcting reflector's parameters.

PHASE ERRORS. CORRECTING REFLECTOR. PARABOLIC MIRROR. MESHED SCREEN. DIRECTIVITY.

Lyashenko A.L. METHOD OF DISTRIBUTED P-REGULATOR SYNTHESIS BY USING INDEX OF OSCILLATION.

In article the possibility of use of the device of the expanded response characteristics for the analysis of systems with the distributed parameters is considered. The problem on working out of a design procedure of adjustments of the distributed P-regulator is formulated and solved. The presented technique will allow to make synthesis of the distributed regulators on index of oscillation.

SYSTEMS WITH THE DISTRIBUTED PARAMETERS. THE EXPANDED RESPONSE CHARACTERISTICS. THE DISTRIBUTED REGULATOR. TRANSFER FUNCTIONS OF THE DISTRIBUTED OBJECTS. INDEX OF OSCILLATION.

Rostov N.V. MULTIOBJECTIVE NORMALIZATION OF REFERENCE MODELS IN SYNTHESIS OF AUTOMATIC CONTROL SYSTEMS.

The problems of parametrical normalization of reference models for automatic control systems are considered. The technique of multiobjective normalization are presented, which use direct sounding and genetic algorithm. The examples are given to Pareto-optimal parameter normalization for the third order reference models.

AUTOMATIC CONTROL SYSTEMS. REFERENCE MODELS. MULTIOBJECTIVE NORMALIZATION. SOUNDING. PARETO-OPTIMAL VALUES.



Lychagin N.I., Vershinina V.P., Dmitriev V.P. KNOWLEDGE MANAGEMENT IN SOFTWARE DEVELOPING COMPANY QUALITY MANAGEMENT SYSTEM.

In the article are examined the base processes of the system of the management of the quality of enterprise – the developer of software. Are developed the ways of creating control system by knowledge, by the basic resource of knowledge which is documentation on software.

KNOWLEDGE MANAGEMENT. SOFTWARE. QUALITY MANAGEMENT.

Golub Yu.A. METHODS AND MODELS OF RESEARCH AND DEVELOPMENT OF AUTOMATED INFORMATION SYSTEMS SELECTIVE DISSEMINATION OF INFORMATION AND SERVICES INITIATIVE.

The paper proposes a multi-level model of the structure of information systems based on the definition of a system that implements a system-oriented approach. Also developed methods for assessing the relationship between the strata of multilevel models based on the idea of solving the matrix G.S. Pospelov and information systems approach to the analysis of A.A. Denisov.

SYSTEM. ASSESSMENT INFORMATION. METHOD SOLVES MATRIX. MODEL.

Derkachenko V.N., Zubkov A.F., Kovalerova N.V., Barmin M.A. MATHEMATICAL MODELS OF THE REAL ESTATE MARKET DEVELOPMENT PROGNOSTICS.

Quarterly market trends and comparative trend evaluation are being introduced in respect to primary and pre-owned real estate market price change. With use of incremental regression analysis, factors affecting real estate costs are exposed.

MATHEMATICAL MODELS. REAL ESTATE. STATISTIC ANALYSIS.

Krivosheykin A.V., Molchanov, A.P., Chesnokov M.A. DYNAMIC FILTERING OF ELEMENTS OF SPEECH SIGNALS AND SIGNALS WITH NON-PERIODIC SPECTRUM.

In clause systems of adaptive filtering of additively combining of noise and speech signal both with periodic and with nonperiodic spectrum was presented and investigated.

ADAPTIVE FILTERING. MODELLING. HEARING AID. SPEECH. THE BASIC TONE.

Moreva S.L., Zolotov O.I., Lyashenko A.L. DEVELOPMENT OF MATHEMATICAL MODEL OF TEMPERATURE FIELDS OF THE ACTIVE ZONE OF THE RBMK-1000 REACTOR.

In article problems of the analysis of systems with the distributed parameters on an example of an active zone of the RBMK-1000 reactor are considered. The description of an active zone of the reactor is presented. The technique of drawing up of mathematical model of the distributed object of management is stated.

REACTOR. GRAPHITE LAYING. OBJECTS WITH THE DISTRIBUTED PARAMETERS. MODELLING.

Darmakheev V.V. EFFICIENCY OF USAGE OF GENETIC OPERATOR SELECTION OF SEQUENCE.

This article is devoted to analysis efficiency different genetic operators selection. The possibility of increasing convergence of genetic algorithm by using various operators of selection. The study defined the criteria to evaluate the efficiency of selection operators.

GENETIC ALGORITHMS. SELECTION OPERATORS. EFFICIENCY. SEQUENCE. POPULATION DIVERSITY.

Chentsov A.G., Chentsov P.A. ROUTING OF PERMUTATIONS WITH CONSTRAINTS AND NONSTATIONARY COST FUNCTIONS.

In article the problem of permutations routing with fulfillment of works in the sequence points is investigated. Theoretical method of solving and optimal algorithm realized on PC are constructed.

ROUTE. DYNAMIC PROGRAMMING. PRECEEDING CONDITIONS. MEGALOPOLIS.

Zikratov I.A., Spivak A.I., Razumovskiy A.V. METHOD OF DIFFERENTIATING ACCESS TO OBJECTS IN VERSION CONTROL SYSTEM.

The article considers the problem of access to objects in the version control systems. The problems of access restrictions at the level of the individual versions of an object. A method for access control and implementation of the method is presented.

ACCESS. DISTINCTION. VERSION. SCV. OBJECT.

Grigorjeva A.V., Vasilev V.A., Safonov V.O. USE ELEMENT ASPECT-ORIENTED APPROACH IN THE DRAFTING ASPECT.NET GEOLOCATION RESOURCE MANAGEMENT SYSTEM TO CLOUD PLATFORMS MICROSOFT WINDOWS AZURE.

This article discusses the use of elements of aspect-oriented approach in the development of open ASPECT.NET geolocation resource management system that provides data about the location of resources in real time and respond quickly to changes with complex business logic rules and reporting, thereby effectively solving the problem of resource management.

ASPECT-ORIENTED APPROACH. RESOURCE MANAGEMENT. CLOUD PLATFORMS.

Voinov N.V., Kotlyarova L.P., Poddubnyy I.E. METHODIC OF WINDOWS PHONE 7 MOBILE APPLICATIONS DEVELOPMENT.

The paper presents a methodic of an applications development for modern mobile devices. The new operating system of Microsoft Corporation – Windows Phone 7 – is chosen as the main platform. Also described are the results of the methodic piloting among students of different courses.

MOBILE PLATFORM. MOBILE DEVICE. MOBILE APPLICATION. MICROSOFT WINDOWS PHONE.

Malygin L.L., Tsarev V.A., Cherkas P.S. AUTOMATIC PARAMETERS CONTROL OF CAMERAS, LENSES AND ILLUMINATION IN REAL-TIME TEXT MARKER RECOGNITION SYSTEMS.

The article contains description of efficiency increase methods for text labels recognition real time systems by means of automatic control of camera and lenses parameters. Features of images quality calculation, camera and lenses selection and control of their parameters by means of automatic controller are described. Possible application fields of the provided methods are specified.

TEXT MARKER RECOGNITION. SMART CAMERAS. CONTROL SYSTEM. IMAGE QUALITY. CONTROL OF CAMERA PARAMETERS.

Ivanov A.S., Kotlyarov V.P., Letichevsky A.A. VERIFICATION OF REQUIREMENTS FOR REACTIVE SYSTEMS.

The article features an approach for verification of requirements for reactive systems represented in the basic model notation. An examples of use are reviewed. The benefits and restrictions of this technology are revealed and discussed.

FORMAL VERIFICATION. FORMAL NOTATION. HOARE LOGIC. REQUIREMENT MANAGEMENT.

Nikiforov I.V., Petrov A.V., Kotlyarov V.P. STATIC APPROACH FOR DEBUGGING OF TEST SCENARIOS GENERATED WITH USAGE OF HEURISTICS.

The paper presents a debugging approach for automatic generated tests from high level Use Case Map-specification with usage of heuristics. The incremental algorithm of bugs finding and fixing in generated tests as well as in the model is described. Usage of the algorithm is presented on UCM project with two threads.

AUTOMATIZATION. TESTS. DEBUG. UCM.



Tyutin B.V., Nikiforov I.V., Kotlyarov V.P. ELABORATION OF THE TOOLSUITE FOR AUTOMATION OF THE STATIC AND DYNAMIC SOFTWARE REQUIREMENT CHECKING.

The article features an automated test suite generation method based on formal system requirement representation using Use Case Maps notation. An automated requirement checking technology is reviewed. The benefits and restrictions of this technology are revealed and discussed.

TESTINGAUTOMATION.FORMALNOTATION.TESTCODEGENERATION.VERIFICATION.REQUIREMENT MANAGEMENT.

Drobintsev P.D., Kotlyarov V.P., Chernorutsky I.G. APPROACH FOR TESTING AUTOMATION BASED ON USER SCENARIOUS.

The paper presents an approach to testing automation based on user scenarios defined in UCM format. Technology chain for tests generation and detailed specification is described.

AUTOMATIZATION. TESTS. VERIFICATION. UCM.

НАУЧНО-ТЕХНИЧЕСКИЕ ВЕДОМОСТИ СПбГУ

№ 4 (152) 2012

Учредитель – Санкт-Петербургский государственный политехнический университет

Издание зарегистрировано в Госкомпечати РФ, свидетельство № 013165 от 23.12.94

Редакция журнала

канд. техн. наук, д-р экон. наук, профессор *А.В. Бабкин* – научный редактор

Е.А. Калинина – литературный редактор, корректор

Г.А. Пышкина – ответственный секретарь, выпускающий редактор

Телефон редакции 552-62-16, 297-18-21

E-mail: infocom@spbstu.ru

Компьютерная верстка *А.Н. Смирнов*

Директор Издательства Политехнического университета *А.В. Иванов*

Лицензия ЛР № 020593 от 07.08.97

Подписано в печать 27.08.2012. Формат 60×84 1/8. Бум. тип. № 1.
Печать офсетная. Усл. печ. л. 16,27. Уч.-изд. л. 16,27. Тираж 1000. Заказ

Санкт-Петербургский государственный политехнический университет
Издательство Политехнического университета
член Издательско-полиграфической ассоциации университетов России
Адрес университета и издательства: 195251, Санкт-Петербург, Политехническая ул., д. 29.