

Программная инженерия

Том 7
№ 2
2016
Пр
ИН

Учредитель: Издательство "НОВЫЕ ТЕХНОЛОГИИ"

Издается с сентября 2010 г.

DOI 10.17587/issn. 2220-3397

ISSN 2220-3397

Редакционный совет

Садовничий В.А., акад. РАН
(председатель)
Бетелин В.Б., акад. РАН
Васильев В.Н., чл.-корр. РАН
Жижченко А.Б., акад. РАН
Макаров В.Л., акад. РАН
Панченко В.Я., акад. РАН
Стемпковский А.Л., акад. РАН
Ухлинов Л.М., д.т.н.
Федоров И.Б., акад. РАН
Четверушкин Б.Н., акад. РАН

Главный редактор

Васенин В.А., д.ф.-м.н., проф.

Редколлегия

Антонов Б.И.
Афонин С.А., к.ф.-м.н.
Бурдонов И.Б., д.ф.-м.н., проф.
Борзовс Ю., проф. (Латвия)
Гаврилов А.В., к.т.н.
Галатенко А.В., к.ф.-м.н.
Корнеев В.В., д.т.н., проф.
Костюхин К.А., к.ф.-м.н.
Махортов С.Д., д.ф.-м.н., доц.
Манцивода А.В., д.ф.-м.н., доц.
Назирова Р.Р., д.т.н., проф.
Нечаев В.В., д.т.н., проф.
Новиков Б.А., д.ф.-м.н., проф.
Павлов В.Л. (США)
Пальчунов Д.Е., д.ф.-м.н., доц.
Петренко А.К., д.ф.-м.н., проф.
Позднеев Б.М., д.т.н., проф.
Позин Б.А., д.т.н., проф.
Серебряков В.А., д.ф.-м.н., проф.
Сорокин А.В., к.т.н., доц.
Терехов А.Н., д.ф.-м.н., проф.
Филимонов Н.Б., д.т.н., проф.
Шапченко К.А., к.ф.-м.н.
Шундеев А.С., к.ф.-м.н.
Щур Л.Н., д.ф.-м.н., проф.
Язов Ю.К., д.т.н., проф.
Якобсон И., проф. (Швейцария)

Редакция

Лысенко А.В., Чугунова А.В.

Журнал издается при поддержке Отделения математических наук РАН, Отделения нанотехнологий и информационных технологий РАН, МГУ имени М.В. Ломоносова, МГТУ имени Н.Э. Баумана

СОДЕРЖАНИЕ

- Загоруйко Ю. А., Загоруйко Г. Б., Боровикова О. И.** Технология создания тематических интеллектуальных научных интернет-ресурсов, базирующаяся на онтологии 51
- Потапов В. П., Попов С. Е.** Высокопроизводительный алгоритм роста регионов для развертки интерферометрической фазы на базе технологии CUDA 61
- Васенин В. А., Пирогов М. В., Чечкин А. В.** Основной оператор радикального моделирования на демонстрационном примере 75
- Орлова Е. В.** Механизм, модели и алгоритмы управления производственно-экономическими системами на принципах согласования критериев заинтересованных агентов 86

Журнал зарегистрирован

в Федеральной службе

по надзору в сфере связи,

информационных технологий

и массовых коммуникаций.

Свидетельство о регистрации

ПИ № ФС77-38590 от 24 декабря 2009 г.

Журнал распространяется по подписке, которую можно оформить в любом почтовом отделении (индексы: по каталогу агентства "Роспечать" — 22765, по Объединенному каталогу "Пресса России" — 39795) или непосредственно в редакции.

Тел.: (499) 269-53-97. Факс: (499) 269-55-10.

Http://novtex.ru/prin/rus E-mail: prin@novtex.ru

Журнал включен в систему Российского индекса научного цитирования.

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

© Издательство "Новые технологии", "Программная инженерия", 2016

SOFTWARE ENGINEERING

PROGRAMMAYA INGENERIA

Vol. 7

N 2

2016

Published since September 2010

DOI 10.17587/issn. 2220-3397

ISSN 2220-3397

Editorial Council:

SADOVNICHY V. A., Dr. Sci. (Phys.-Math.),
Acad. RAS (*Head*)
BETELIN V. B., Dr. Sci. (Phys.-Math.), Acad. RAS
VASIL'EV V. N., Dr. Sci. (Tech.), Cor.-Mem. RAS
ZHIZHCHEKNO A. B., Dr. Sci. (Phys.-Math.),
Acad. RAS
MAKAROV V. L., Dr. Sci. (Phys.-Math.), Acad.
RAS
PANCHENKO V. YA., Dr. Sci. (Phys.-Math.),
Acad. RAS
STEMPKOVSKY A. L., Dr. Sci. (Tech.), Acad. RAS
UKHLINOV L. M., Dr. Sci. (Tech.)
FEDOROV I. B., Dr. Sci. (Tech.), Acad. RAS
CHETVERTUSHKIN B. N., Dr. Sci. (Phys.-Math.),
Acad. RAS

Editor-in-Chief:

VASENIN V. A., Dr. Sci. (Phys.-Math.)

Editorial Board:

ANTONOV B.I.
AFONIN S.A., Cand. Sci. (Phys.-Math)
BURDONOV I.B., Dr. Sci. (Phys.-Math)
BORZOV JURIS, Dr. Sci. (Comp. Sci), Latvia
GALATENKO A.V., Cand. Sci. (Phys.-Math)
GAVRILOV A.V., Cand. Sci. (Tech)
JACOBSON IVAR, Dr. Sci. (Philos., Comp. Sci.),
Switzerland
KORNEEV V.V., Dr. Sci. (Tech)
KOSTYUKHIN K.A., Cand. Sci. (Phys.-Math)
MAKHORTOV S.D., Dr. Sci. (Phys.-Math)
MANCIVODA A.V., Dr. Sci. (Phys.-Math)
NAZIROV R.R. , Dr. Sci. (Tech)
NECHAEV V.V., Cand. Sci. (Tech)
NOVIKOV B.A., Dr. Sci. (Phys.-Math)
PAVLOV V.L., USA
PAL'CHUNOV D.E., Dr. Sci. (Phys.-Math)
PETRENKO A.K., Dr. Sci. (Phys.-Math)
POZDNEEV B.M., Dr. Sci. (Tech)
POZIN B.A., Dr. Sci. (Tech)
SEREBRJKOV V.A., Dr. Sci. (Phys.-Math)
SOROKIN A.V., Cand. Sci. (Tech)
TEREKHOV A.N., Dr. Sci. (Phys.-Math)
FILIMONOV N.B., Dr. Sci. (Tech)
SHAPCHENKO K.A., Cand. Sci. (Phys.-Math)
SHUNDEEV A.S., Cand. Sci. (Phys.-Math)
SHCHUR L.N., Dr. Sci. (Phys.-Math)
YAZOV Yu. K., Dr. Sci. (Tech)

Editors: LYSENKO A.V., CHUGUNOVA A.V.

CONTENTS

- Zagorulko Yu. A., Zagorulko G. B., Borovikova O. I.** Technology
for Building Subject-Based Intelligent Scientific Internet Resources
Based on Ontology 51
- Potapov V. P., Popov S. E.** High-Performance Region-Growing
Algorithm for InSAR Phase Unwrapping Based on CUDA 61
- Vasenin V. A., Pirogov M. V., Chechkin A. V.** The Main Operator
of Radical Modeling on a Demonstration Example 75
- Orlova E. V.** Mechanism, Models and Control Algorithms
for Productive and Economic System under Harmonization Criteria
of Interested Agents 86

Information about the journal is available online at:
<http://novtex.ru/prin/eng>, e-mail: prin@novtex.ru

Ю. А. Загорулько, канд. техн. наук, зав. лаб., e-mail: zagor@iis.nsk.su,
Г. Б. Загорулько, науч. сотр., e-mail: gal@iis.nsk.su,
О. И. Боровикова, мл. науч. сотр., e-mail: olesya@iis.nsk.su,
Институт систем информатики им. А. П. Ершова СО РАН, г. Новосибирск

Технология создания тематических интеллектуальных научных интернет-ресурсов, базирующаяся на онтологии

Предложена технология разработки тематических интеллектуальных научных интернет-ресурсов, обеспечивающих содержательный доступ к систематизированным научным знаниям и информационным ресурсам определенной области знаний и средствам их интеллектуальной обработки. Особенностью технологии является использование формализмов онтологий и семантических сетей в качестве средства представления и систематизации знаний, а также ориентация на экспертов — специалистов в тех областях знаний, для которых создаются ресурсы.

Ключевые слова: интеллектуальный научный интернет-ресурс, онтология, семантическая сеть, тезаурус, технология, метапоиск, извлечение информации

Введение

К настоящему времени в мире накоплено огромное количество информации по различным областям знаний. Однако, несмотря на то что значительная ее часть представлена непосредственно в сети Интернет, задача обеспечения эффективного, удобного для потенциального пользователя доступа к ней пока не имеет удовлетворительного решения. Сложившееся положение объясняется особенностями представления научных знаний в сети Интернет, которые слабо формализованы, недостаточно систематизированы и распределены по всей метасети Интернет. Кроме того, современные информационные системы используют довольно ограниченный набор методов представления и интерпретации информации. Как правило, данные и знания в них представляются в виде текстовых документов или каталогов информационных ресурсов, в то время как для человека наиболее естественной и удобной формой подачи информации является ее представление в виде сети взаимосвязанных фактов.

Нерешенным остается и вопрос предоставления удобного доступа к средствам обработки информации, разработанным в различных областях знаний. Даже уже реализованные и представленные в сети Интернет в виде web-сервисов методы обработки информации остаются недоступными широкому кругу пользователей в силу их плохой систематизации и отсутствия содержательных описаний, по которым их можно было бы найти.

Для отмеченных выше задач предлагается концепция их решения на основе тематического интеллектуального научного интернет-ресурса (ИНИР) [1]. Такой ресурс способен обеспечивать содержательный доступ к систематизированным научным знаниям и информационным ресурсам определенной области знаний и к средствам их интеллектуальной обработки. Интернет-ресурс ИНИР назван интеллектуальным потому что не только систематизация и представление информации в этом ресурсе, но и все его функциональные возможности базируются на формализмах онтологий [2] и семантических сетей [3].

Принимая во внимание большую потребность в системах подобного класса, была разработана технология создания и сопровождения ИНИР, ориентированная не на программистов и инженеров знаний, а непосредственно на экспертов — специалистов в тех областях знаний, для которых такие ресурсы разрабатывают.

В статье рассмотрены основные положения технологии разработки тематических ИНИР. Заметим, что эта технология является развитием технологии построения порталов научных знаний, разработанной авторами ранее и успешно применявшейся при создании интернет-ресурсов для ряда научных дисциплин [4].

Статья структурирована следующим образом. Первый раздел посвящен описанию концепции и архитектуры системы знаний ИНИР. Здесь же представлена архитектура ИНИР. Второй раздел посвящен описанию технологии разработки ИНИР.

В третьем разделе дан пример использования предлагаемой технологии. В четвертом разделе содержится небольшой обзор работ по рассматриваемой тематике. В заключении обсуждены основные особенности и преимущества технологии.

1. Концепция и архитектура ИНИР

Интернет-ресурс ИНИР предназначен для информационной и аналитической поддержки научной и производственной деятельности в заданной области знаний. Он представляет собой доступную через сеть Интернет информационную систему. Эта система предназначена для систематизации и интеграции научных знаний и информационных ресурсов определенной области знаний, а также для обеспечения содержательного эффективного доступа к ним, к методам обработки информации, используемым в данной области знаний, и к методам решения типовых для данной области задач. Для этого система предоставляет пользователям соответствующие интерфейсы и сервисы.

1.1. Система знаний ИНИР

Как было сказано выше, тематический ИНИР базируется на формализмах онтологий и семантических сетей. Онтология при этом составляет ядро системы знаний ИНИР (рис. 1) и наряду с описанием различных аспектов моделируемой области знаний содержит соотношенное с ним описание структуры и типологии интегрируемых информационных ресурсов и методов интеллектуальной обработки информации. В то время как в семантической сети, структура которой определяется онтологией ИНИР, накапливается информация об основных сущностях

моделируемой области знаний, о релевантных ей научных информационных ресурсах и web-сервисах, реализующих использующиеся в данной области методы обработки данных.

Онтология ИНИР состоит из трех взаимосвязанных онтологий, отвечающих за представление указанных выше компонентов знаний, а именно онтологии области знаний ИНИР, онтологии научных интернет-ресурсов и онтологии задач и методов.

Онтология области знаний задает систему понятий и отношений, предназначенных для детального описания области знаний ИНИР и выполняемой в ее рамках научной и исследовательской деятельности.

Онтология научных интернет-ресурсов служит для описания представленных в сети Интернет информационных ресурсов, релевантных области знаний ИНИР.

Онтология задач и методов кроме описания задач, для решения которых предназначен ИНИР, и методов их решения включает также описания web-сервисов, реализующих как указанные методы решения задач, так и методы обработки информации, разработанные в моделируемой области знаний.

Система знаний ИНИР также включает тезаурус [5], содержащий термины моделируемой области знаний, а именно слова и словосочетания, с помощью которых понятия онтологии представляются в текстах и пользовательских запросах. Тезаурус задает смысл понятий, причем не столько с помощью традиционных определений, сколько посредством соотношения одних понятий с другими, используя для этого различные семантические отношения. Благодаря этому тезаурус может применяться как при обработке пользовательских запросов, так и при поиске и аннотировании информационных ресурсов, интегрируемых в ИНИР.

Таким образом, система знаний ИНИР включает формальное описание области знаний ИНИР, задает типологию релевантных ей информационных ресурсов и решаемых в ней задач и методов их решения (посредством онтологии). Эта система знаний описывает с помощью тезауруса смысл используемых в этой области знаний понятий. Средствами семантической сети обеспечивается эффективное представление информации о реальных объектах моделируемой области знаний, интегрируемых в ИНИР информационных ресурсах и средствах их интеллектуальной обработки.

1.2. Архитектура ИНИР

Интернет-ресурс ИНИР имеет традиционную для информационных систем трехуровневую архитектуру, включающую уровень представления информации; уровень обработки информации; уровень хранения и доступа к информации (базовый уровень).

Уровень представления информации обеспечивается пользовательским интерфейсом, главными функциями которого являются представление пользовательских запросов и результатов поиска и решений задач, а также обеспечение управляемой онто-



Рис. 1. Система знаний тематического ИНИР

логией навигации в информационном пространстве ИНИР. При этом пользовательский интерфейс обеспечивает доступ не только к контенту ИНИР, но и к средствам аналитической обработки информации.

На втором уровне обеспечиваются различные виды поиска и обработки информации, а также ее передача между уровнями. Для этих целей данный уровень включает модуль поиска информации в контенте ИНИР, а также средства ее аналитической обработки, реализованные в том числе в виде web-сервисов.

Базовый уровень поддерживает выполнение функций хранения и управления знаниями (онтологией и тезаурусом) и данными (контентом ИНИР) с использованием средств реляционных СУБД, технологий Semantic Web [6] и семантических web-сервисов [7].

В качестве хранилища данных используется RDF-хранилище Jena Fuseki [8], поддерживающее стандартный язык запросов SPARQL [9], изменение данных и логический вывод. В этом хранилище данные представляются в виде множества триплетов (*triples*), представляющих утверждения вида "субъект—предикат—объект", отвечающих известной модели данных RDF [10]. Такая структура обладает большой гибкостью представления данных и знаний, благодаря чему позволяет хранить вместе как описания онтологий и тезаурусов, выполненные на языке OWL [11], так и контент ИНИР.

Программные компоненты базового уровня ИНИР, обеспечивающие управление знаниями и данными, реализуются средствами языка запросов SPARQL. Благодаря тому, что взаимодействие с хранилищем осуществляется с помощью SPARQL HTTP-клиента, достигается независимость программных компонентов от конкретной реализации хранилища.

2. Технология построения ИНИР

Построение тематического ИНИР в рамках предлагаемого подхода фактически сводится к созданию его системы знаний, т. е. включает разработку онтологии и тезауруса, а также наполнение контента ИНИР. Так как онтология составляет основу системы знаний ИНИР, важнейшим компонентом рассматриваемой технологии является методология построения онтологий, поддержанная инструментальными средствами (редакторами онтологий и данных), а также представительным набором базовых онтологий, на основе которых эксперты строят онтологии и тезаурусы тематических ИНИР (рис. 2, см. вторую сторону обложки).

Технология построения ИНИР поддерживается также пользовательским интерфейсом; подсистемой сбора информации из сети Интернет, автоматизирующей наполнение контента ИНИР; хранилищем данных, обеспечивающим универсальные структуры для согласованного хранения онтологии, тезауруса и контента ИНИР; программными средствами для доступа к ним.

2.1. Методология построения онтологии и тезауруса ИНИР

Онтология конкретного ИНИР строится в соответствии с методологией, главными принципами которой являются:

- структурирование онтологии ИНИР на ряд относительно независимых онтологий;
- использование базовых онтологий, включающих только самые общие сущности, не зависящие от конкретных областей знаний ИНИР;
- построение всех онтологий ИНИР на основе базовых онтологий путем их достройки и развития.

Использование такой методологии значительно упрощает создание онтологии ИНИР и ее дальнейшее сопровождение.

Как было отмечено выше, онтология ИНИР состоит из трех относительно независимых, но связанных онтологий: онтология области знаний ИНИР; онтология научных интернет-ресурсов; онтология задач и методов. Все эти онтологии строятся на основе базовых онтологий.

В качестве базовых предложены следующие онтологии: онтология научной деятельности; онтология научного знания; базовая онтология задач и методов; базовая онтология научных информационных ресурсов; онтология представления тезауруса. На основе первых двух онтологий строится онтология области знаний ИНИР, на основе третьей — онтология задач и методов, на основе четвертой — онтология научных интернет-ресурсов. Тезаурус области знаний ИНИР строится на основе онтологии представления тезауруса. Рассмотрим эти онтологии подробнее.

Онтология научной деятельности базируется на онтологии, предложенной в работе [12] для описания научно-исследовательских проектов и расширенной для применения к более широкому классу задач. Эта онтология включает классы понятий, относящиеся к организации научной и исследовательской деятельности, такие как *Персона, Организация, Событие, Научная деятельность, Проект, Публикация* и др. Онтология научной деятельности включает также отношения, позволяющие связывать понятия данной онтологии не только между собой, но и с понятиями онтологии научного знания.

Онтология научного знания содержит классы, задающие структуры для описания понятий конкретных областей знаний, такие как *Раздел науки, Метод исследования, Объект исследования, Предмет исследования, Научный результат* и др. Эта онтология также включает отношения, связывающие между собой объекты указанных выше классов. Используя эти классы, можно выделить и описать значимые для моделируемой области знаний разделы и подразделы, задать типизацию методов и объектов исследования, описать результаты научной деятельности.

Базовая онтология задач и методов включает такие классы, как *Задача, Метод решения* и *Web-сервис*. С помощью понятий и отношений данной онтологии могут быть описаны задачи, для решения

которых предназначен ИНИР, методы их решения и реализующие их web-сервисы, а также web-сервисы, реализующие методы обработки информации, использующиеся в области знаний ИНИР.

Базовая онтология научных информационных ресурсов включает класс *Информационный ресурс* в качестве основного класса. Набор атрибутов и связей этого класса основан на стандарте Dublin core [13]. Его атрибутами являются: *название ресурса, язык ресурса, тематика ресурса, тип ресурса, дата создания ресурса* и др. Для представления информации об источниках ресурса и его создателях, а также связанных с ним событиях, организациях, персонах, публикациях и других сущностях вводят специальные отношения, связывающие класс *Информационный ресурс* с классами других онтологий ИНИР.

Описания web-сервисов базируются на онтологии OWL-S [14], предназначенной для описания семантических web-сервисов. Благодаря этому с web-сервисом связываются не только описание его интерфейса в терминах типов входных и выходных данных, но и описания его семантики, т. е. того, что сервис делает, его предметной области, ограничений на область применения и т. п. Причем все его декларативные свойства, функциональные возможности и интерфейсы кодируются в однозначной подающейся машинной обработке форме.

Наличие содержательных описаний у web-сервисов создает предпосылки для их успешной интеграции в ИНИР. При этом будет обеспечиваться содержательный доступ к ним не только для программных агентов, но и для пользователя ресурса, желающего найти необходимые для решения его задач средства интеллектуальной обработки информации.

Онтология представления тезауруса [15] базируется на российских и международных стандартах, регламентирующих структуру одноязычных и многоязычных информационно-поисковых тезаурусов, набор и свойства основных сущностей и отношений, устанавливаемых между ними. По этой причине она включает набор базовых понятий и отношений, присутствующих в любом тезаурусе. В частности, онтология представления тезауруса содержит классы, описывающие следующие его сущности: термины, которые подразделяют на дескрипторы (предпочтительные термины) и аскрипторы (текстовые входы, которые при поиске и индексировании документов могут быть заменены на соответствующие дескрипторы); источники терминов (web-ресурсы, текстовые документы или коллекции текстов, в которых встречаются или определяются термины) и области/подобласти знаний, с которыми в свою очередь могут быть соотнесены термины. В онтологии также представлены отношения, связывающие объекты перчисленных выше классов между собой.

2.2. Управление системой знаний ИНИР

Для поддержки процесса настройки ИНИР на область научных знаний и управления системой знаний этого ресурса разработаны редакторы онтологий и

данных. Эти редакторы имеют удобные графические интерфейсы. Они реализованы как web-приложения и обеспечивают удаленную настройку и сопровождение системы знаний ИНИР авторизованными экспертами через сеть Интернет. Для поддержки совместной разработки системы знаний коллективом экспертов в этих редакторах реализуется механизм делегирования прав экспертам разных уровней.

Для построения онтологий и управления ими служит редактор онтологий. Этот редактор проектировался таким образом, чтобы им могли пользоваться не только инженеры знаний, но и эксперты, не являющиеся специалистами в области информатики и математики. Редактор онтологий позволяет создавать, модифицировать и удалять любые элементы онтологии (классы понятий, атрибуты, отношения).

Для создаваемого класса может быть выбран родитель из ранее созданных классов, которые представляются пользователю в виде дерева. При этом от родительского класса наследуются не только все атрибуты, но и отношения, а сам родитель связывается с новым классом отношением "класс—подкласс". При создании отношений указываются их классы-аргументы, которые также выбираются из дерева классов.

Управление контентом ИНИР осуществляется с помощью редактора данных, который работает под управлением онтологий, что позволяет не только значительно облегчить корректный ввод данных, но и обеспечить их логическую целостность.

С помощью редактора данных можно создавать, редактировать и удалять информационные объекты (объекты введенных в онтологии классов) и связи между ними. При создании нового информационного объекта выбирается подходящий класс онтологий. После чего по представленному в онтологии описанию класса автоматически создается форма для ввода данных, включающая поля для ввода значений атрибутов объекта и его связей с другими объектами, уже существующими в контенте ИНИР. Типы таких связей и классы таких объектов определяются соответствующими отношениями онтологий. Основываясь на этом и текущем состоянии контента, редактор выдает список объектов, с которыми может быть связан редактируемый объект данным отношением.

Построение тезауруса и редактирование его содержания осуществляется с помощью редактора данных, работающего под управлением онтологий представления тезауруса, что позволяет обеспечить логическую целостность его терминологической системы.

2.3. Автоматизация сбора информации для ИНИР

Для того чтобы ИНИР был полезным ресурсом, его система знаний должна содержать актуальную и достаточно полную информацию о моделируемой области знаний и выполняемой в ее рамках научной деятельности. Создание и сопровождение такого ресурса — довольно сложная и трудоемкая задача, требующая значительных усилий разработчиков. Слож-

ность данной задачи обуславливается большим разнообразием видов собираемой информации и способов ее представления в сети Интернет, а трудоемкость — большими объемами такой информации.

Так, для ИНИР необходимо собирать информацию о проектах, организациях, персонах, конференциях и публикациях (т. е. обо всех объектах базовых классов онтологии научной деятельности) из таких источников, как информационные порталы, сайты организаций, ассоциаций, проектов и конференций, порталы знаний, социальные научные сети и др.

Для решения этой задачи служит подсистема, автоматизирующая сбор информации об основных сущностях области знаний ИНИР и релевантных ей интернет-ресурсах. Эта подсистема объединяет методы метапоиска и извлечения информации, которые базируются на онтологиях и тезаурусах.

Сбор информации для ИНИР включает следующие этапы (рис. 3, см. вторую сторону обложки): поиск релевантных области знаний ИНИР интернет-ресурсов и документов; извлечение информации из найденных ресурсов; занесение полученной информации в контент ИНИР. В соответствии с этими этапами подсистема сбора информации включает модуль поиска релевантных интернет-ресурсов, модуль извлечения информации из интернет-ресурсов, модуль занесения информации в контент ИНИР, а также базу данных ссылок на интернет-ресурсы (БД СИР).

На первом этапе на основе онтологии и тезауруса генерируются поисковые запросы, которые используются модулем поиска для нахождения релевантных интернет-ресурсов. Модуль поиска с этими запросами обращается к поисковым системам Google, Яндекс и Bing через их программные интерфейсы, т. е. использует механизм метапоиска с последующей фильтрацией результатов поиска, включая удаление дубликатов и нерелевантных ссылок. Релевантность ссылки при этом определяется на основе косинусной меры сходства [16], вычисляемой между векторами весов терминов запроса и страницы, скачанной по ссылке. Заметим, что БД СИР может заполняться не только автоматически, но и вручную — экспертами, которые могут вводить в БД СИР ссылки на релевантные, по их мнению, интернет-ресурсы при настройке ИНИР на область знаний.

На втором этапе выполняется анализ найденных интернет-ресурсов и извлечение из них информации. Особенностью реализуемого метода сбора инфор-

```

<Class Name= "Проект" engine = "FragmentSearch" >
  <Marker Term = "О проекте" Ptype = "Menu/Head" FragType = "Page/Block" />
  <Marker Term = "Проект" Ptype = "Head" FragType = "Block" />
  <Attr Name = "Название" type = "string" engine = "NameEntity" >
    <Marker Term = "Проект" Ptype = "link" FragType = "LinkText" />
    <Marker Term = "Проект" Ptype = "sentence" FragType = "QuoteText" />
    <Marker Term = "Проект" Ptype = "Head" FragType = "Head" />
  </Attr>
  <Attr Name = "Аннотация" type = "text" >
    <Marker Term = "Аннотация/Содержание проекта/Описание
    проекта/ О проекте " PType = "Head" FragType = "Block/Page" />
  </Attr>
  <Relation Name = "Публикация_о_Проекте" >
    <Marker Term = "Публикации" Ptype = "Menu/Head" FragType = "Page/Block" />
    <Marker Term = "Список публикаций" Ptype = "Menu" FragType = "Page" />
    <Marker Term = "Литература" Ptype = "Menu/Head" FragType = "Page/Block" />
    <Marker Term = "Библиография" Ptype = "Menu/Head" FragType = "Page/Block" />
    <Object Name = "Публикация" engine = "PublicationsList" />
  </Relation>
  <Relation Name = "Участник проекта" >
    <Marker Term = "Об участниках" Ptype = "Menu" FragType = "Page" />
    <Marker Term = "Список участников" Ptype = "Head" FragType = "Block"/>
    <Marker Term = "Исполнители" PType= "Head" FragType = "Block"/>
    <Marker Term = "Участники" Ptype = "Head" FragType = "Block" />
    <Object Name = "Персона" engine = "PersonList" />
  </Relation>
</Class>

```

Рис. 4. Фрагмент шаблона для извлечения информации о проектах

мации является разработка для каждого типа сущностей (класса онтологии) своего метода извлечения информации, который настраивается на область знаний, на типы интернет-ресурсов и документов.

Каждый такой метод извлечения информации включает набор шаблонов. В этих шаблонах для каждого типа извлекаемой информации указываются маркеры, определяющие место расположения извлекаемой информации, а также обработчики, реализующие алгоритмы анализа соответствующих фрагментов интернет-страниц (документов) и извлечения из них требуемой информации. Шаблоны также генерируются на основе онтологии и при необходимости могут быть отредактированы экспертом. Для повышения полноты извлечения информации увеличивают вариативность шаблонов за счет использования в них альтернативных терминов из тезауруса (синонимов и гипонимов).

На рис. 4 представлен фрагмент шаблона для извлечения информации о проекте с сайта научно-исследовательского проекта. Этот шаблон позволяет извлекать такую информацию, как название проекта, его аннотацию, а также данные об участниках проекта и публикациях по теме проекта.

На настоящее время разработаны методы извлечения информации о проектах, организациях и событиях, включая сопутствующие шаблоны и обработчики, реализующие извлечение информации о персонах и публикациях.

На третьем этапе извлеченная из сети Интернет информация заносится в контент тематического ИНИР.

3. Пример использования технологии ИНИР для поддержки исследований в энергетике

Рассматриваемая технология разработки ИНИР была опробована при создании прототипа научного интернет-ресурса для информационно-аналитической поддержки системных исследований в энергетике. Данный ресурс содержит систематизированную информацию об области знаний "исследования в энергетике", о решаемых в ее рамках стратегических задачах и методах их решения, о ресурсах, используемых в данных исследованиях.

Онтология рассматриваемого ресурса, в соответствии с методологией, описанной в подразд. 2.1, строится на основе базовых онтологий путем их дальнейшей структуризации и дополнения. Так, например, *объектом исследования* данной области знаний является топливно-энергетический комплекс (ТЭК). Фрагмент онтологии, описывающий ТЭК [17], включен в онтологию научного знания (рис. 5).

Как видно на рис. 5, ТЭК включает в себя следующие отраслевые системы: газоснабжающую (ГСС), углеснабжающую (УСС), электроэнергетическую (ЭЭС), нефтеснабжающую (НСС), теплоснабжающую (ТСС) и ядерноэнергетическую (ЯЭС). Каждая из систем энергетики в свою очередь добывает и использует те или иные топливно-энергетические ресурсы (ТЭР), включает процессы, поддерживающие ее полный производственный цикл от добычи до потребления соответствующего ресурса, и описывается отдельным фрагментом онтологии.

Базовая онтология задач и методов дополняется понятиями научных задач, решаемых в рамках исследований ТЭК и ее отраслевых систем. Так, одной из важнейших задач является задача обеспечения энергетической безопасности, включающая в себя анализ различных угроз, в качестве которых рас-

сматриваются угроза похолодания, угроза землетрясения, угроза наводнения и т. п.

Одним из основных методов исследований энергетики является моделирование. Онтология задач и методов включает описания большого набора как традиционных методов математического моделирования, так и интеллектуальных методов, позволяющих решать слабоформализованные задачи. Например, разработанные в Институте систем энергетики им. Л. А. Мелентьева СО РАН методы семантического моделирования (к нему относятся онтологическое, когнитивное, событийное и вероятностное моделирование) применяют для решения задач качественного анализа проблемных ситуаций, результаты которого могут быть использованы для формирования вариантов расчетов вычислительного эксперимента [18].

Методы решения задач, применяемые в исследованиях энергетики, предоставляются пользователям разрабатываемого ресурса в виде web-сервисов. На примере методов линейного программирования, относящихся к методам математического моделирования, была опробована методика разработки реализующих их web-сервисов. На языке С# был разработан web-сервис с использованием протокола передачи данных SOAP, работающего поверх протокола HTTP, и языка WSDL. Данный web-сервис может использоваться как для знакомства пользователей с работой реализованных в нем методов линейного программирования, так и для его встраивания в сторонние приложения. Для этого он снабжен двумя пользовательскими интерфейсами, доступ к которым осуществляется со страницы рассматриваемого интернет-ресурса, содержащей описание интересующего пользователя метода. Первый интерфейс является своего рода "песочницей", позволяющей пользователю задать входные данные, запустить исполнение

выбранного метода и просмотреть результаты его работы. Второй интерфейс позволяет пользователю получить описание метода и его входных данных, а также примеры использования сервиса сторонними приложениями.

Данный ресурс дает целостное представление об энергетике и ее отраслевых системах, о задачах исследований энергетики, методах и средствах их решения. Ресурс предоставляет удаленный доступ к приложениям, поддерживающим тот или иной метод. Также пользователи ресурса могут получить как неформальное, так и формализованное описание метода, сведения о группах, коллективах, развивающих данный метод, ссылки на интернет-ресурсы, относящиеся к представленной на ресурсе информации.

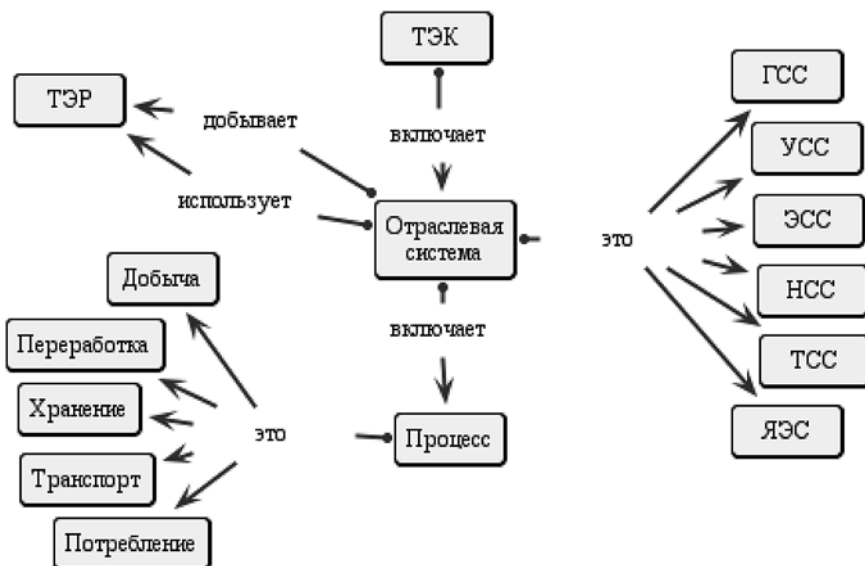


Рис. 5. Фрагмент онтологии ТЭК

4. Близкие работы

Большое число работ посвящено решению проблемы систематизации и интеграции информационных ресурсов, относящихся к определенной области знаний или некоторому сообществу, и обеспечения к ним удобного доступа.

Прежде всего следует отметить такой класс интернет-ресурсов, как семантические порталы (*Semantic Web Portal*) [19], которые базируются на онтологиях и других компонентах технологии *Semantic Web*. В рамках этого направления исследований было создано множество семантических порталов. Наиболее типичными их представителями являются *Esperanto* [20] и *OntoWeb* [21].

Портал *Esperanto* обслуживает информационные потребности участников одноименного европейского проекта. Он включает пять онтологий (Проект, Документ, Персона, Организация, Встреча), служащих для описания научно-исследовательских проектов. Недостатком *Esperanto* является то, что на нем возможен поиск объектов только по атрибутам классов, представленных в рамках одной онтологии. Кроме того, он ориентирован только на представление информации о научно-исследовательских проектах и не может быть настроен на конкретную область знаний.

Портал *OntoWeb* относится к так называемым порталам сообществ (*community portal*) и предназначен для облегчения передачи наработок по онтологиям и технологии *Semantic Web* от представителей научной общественности в индустрию. На портале может быть размещена любая информация об организациях, событиях, проектах, персонах и документах (научных статьях). Недостатком этого портала является то, что используемые в нем принципы организации информации не всегда интуитивно понятны всем пользователям, что затрудняет поиск требуемых данных.

Из отечественных разработок необходимо отметить систему ИСТИНА [22], созданную в МГУ имени М. В. Ломоносова. Эта система предназначена для сбора, систематизации, хранения, анализа и выдачи по запросу информации, характеризующей результаты деятельности научных и образовательных организаций. В этой системе для хранения и наглядного представления всех сущностей и связей, характеризующих различные аспекты научной деятельности, используются решения, базирующиеся на онтологиях. В частности, в систему ИСТИНА включены онтология научной деятельности и онтологии отдельных областей научного знания. Онтология научной деятельности содержит такие сущности, как Человек, Организация, Публикация, Конференция, Проект. Онтология каждой конкретной области научного знания (например, математики, информатики или физики) содержит основные разделы и понятия этой области знаний. Совместное использование этих двух типов онтологий позволяет проводить тематическую классификацию сущностей системы по терминам онтологий заданных областей научно-

го знания, а также расширяет возможности анализа деятельности подразделений и организаций по отдельным тематикам.

Система ИСТИНА нацелена на представление и анализ результатов научной деятельности как отдельных ученых, так и целых организаций в различных областях знаний. При этом сами области знаний (в силу их большого числа) описаны в ней очень кратко. В отличие от системы ИСТИНА тематический ИНИР ориентирован на детальное представление определенной области научных знаний и выполняемой в ней научной деятельности, включая используемые методы и полученные результаты.

Что касается технологии построения научных интернет-ресурсов, то прежде всего следует отметить программную оболочку для автоматической генерации и управления семантическими порталами *ODESeW* [20]. Созданные в этой оболочке порталы (например, *Esperanto*) обеспечивают для своих пользователей представление и визуализацию контента, базирующуюся на онтологии навигацию по контенту, поиск информации по ключевым словам и на основе онтологии. С помощью *ODESeW* разработчик может довольно быстро создать новый портал, построив нужные онтологии, либо импортировав ранее разработанные. К недостаткам *ODESeW* можно отнести то, что эта оболочка не предоставляет средств для автоматического или хотя бы автоматизированного заполнения контента портала конкретными фактами из предметной области, что затрудняет создание порталов знаний с большим по объему контентом. *ODESeW* не позволяет интегрировать внешние источники данных. Кроме того, разработчик портала должен освоить довольно сложный инструмент для создания онтологий.

OntoWebber [23] — еще один управляемый моделями и основанный на онтологиях инструмент для построения семантических порталов сообществ. Достоинством *OntoWebber* является то, что он может интегрировать как структурированные, так и слабо-структурированные источники данных, описываемые различными предметными онтологиями. К достоинствам этого инструмента можно также отнести то, что все модели, которыми управляется портал, описываются в терминах онтологий и могут быть переиспользованы в других приложениях. Недостатком *OntoWebber* является то, что пользователь должен сам описывать средствами довольно сложного языка все модели, включая ограничения целостности, правила поддержки актуальности информации, ее представления пользователю и навигации по контенту портала (в виде карт визуализации). Создание пользовательского интерфейса автоматизировано лишь частично, вследствие чего на пользователя возлагается достаточно много работы по формированию содержания и внешнего вида страниц сайта.

Рассмотрим еще один подход [24], обеспечивающий разработку структурированных порталов сообществ, в которых информация, извлеченная из веб-страниц, интегрируется и представляется в виде гра-

фа понятий и отношений, принятых в сообществе. Данный подход предлагает строить портал сверху вниз и пошагово. Сначала эксперты отбирают небольшое число важных для сообщества источников информации. Эти источники упорядочиваются по релевантности с использованием метрик PageRank и TF-IDF. Затем из источников, стоящих вверху списка, извлекается информация, которая и представляется на портале в виде понятий и отношений. Причем для каждой сущности (понятия и отношения) с использованием ограниченного набора простых операторов строится свой план (метод) извлечения информации. Пополнение контента портала сообщества выполняется за счет источников, ссылки на которые извлекаются из источников, отобранных на первом этапе (в дальнейшем — на предыдущем этапе). Такой подход позволяет постепенно строить порталы, содержащие важную информацию для сообщества. Недостатком данного подхода являются высокие требования к квалификации создателей порталов, которые должны уметь строить методы извлечения информации с использованием описанной выше методики.

Завершая обзор технологий построения научных интернет-ресурсов, отметим, что ни одна из рассмотренных технологий не обеспечивает содержательного доступа к методам обработки информации и решения типичных задач, характерных для моделируемой области знаний.

Что касается проблемы сбора информации из сети Интернет, то ей посвящено также много исследований. Однако, как показывает обзор [25], большая их часть направлена на извлечение информации, необходимой для решения задач электронной коммерции или анализа новостного потока и социальных сетей [26], и лишь незначительная часть — на извлечение информации для нужд научной деятельности [24, 27, 28]. При этом большинство исследователей используют популярные в настоящее время методы извлечения информации, основанные на обучении на примерах (см., например, работу [29]). Однако в связи с большим разнообразием видов извлекаемой информации для нужд ИНИР и способов ее представления в сети Интернет применение этих методов оказалось нецелесообразным. Поэтому в рассматриваемой технологии используется подход, базирующийся на онтологии, в соответствии с которым для каждого типа сущностей разрабатывается свой метод сбора и обработки информации, настраиваемый на предметную область и типы интернет-ресурсов.

Заключение

Представлены основные положения технологии построения тематических интеллектуальных научных интернет-ресурсов, ориентированной на экспертов. Эта технология позволяет им собирать и систематизировать в рамках единого информационного пространства обширные знания и данные в требу-

емой области знаний, а также используемые в ней средства интеллектуальной обработки информации.

Использование онтологий в качестве концептуальной основы ИНИР делает его удобным и эффективным средством представления и систематизации всех видов информации о моделируемой области знаний, в которых нуждаются исследователи. Причем эта информация представляется в виде сети знаний и данных, по связям которой поддерживается удобная навигация и содержательный поиск.

Использование онтологий также повышает уровень технологии построения ИНИР, создавая предпосылки для того, чтобы ею могли воспользоваться специалисты в областях знаний, для которых такие ресурсы разрабатываются, т. е. эксперты. С одной стороны, базирование средств построения и сопровождения системы знаний ИНИР на онтологии делает их доступными для использования непосредственно экспертами, так как принятое в ней представление знаний и данных в виде объектов и отношений между ними является наиболее естественным для человека. С другой стороны, онтология является удобным средством формирования и фиксации общего разделяемого всеми экспертами-разработчиками знания о данной области знаний, обеспечивая при этом возможность их переиспользования, что упрощает и ускоряет разработку новых приложений.

Кроме того, наличие в составе технологии методологии построения онтологии ИНИР и довольно представительного набора базовых онтологий, на основе которых она строится, а также удобных для экспертов редакторов знаний и данных значительно снижает сложность и трудоемкость процесса создания ИНИР.

Включение в состав технологической поддержки подсистемы, автоматизирующей сбор информации о научной деятельности, выполняемой в области знаний ИНИР, значительно снижает трудоемкость создания и сопровождения научных интернет-ресурсов.

Работа выполнена при финансовой поддержке РФФИ (проект № 13-07-00422).

Список литературы

1. Загорюлько Ю. А., Загорюлько Г. Б., Шестаков В. К., Кононенко И. С. Концепция и архитектура тематического интеллектуального научного интернет-ресурса // Труды XV Всерос. науч. конф. RCDL'2013. Ярославль, 14–17 октября 2013 г. Ярославль: ЯрГУ, 2013. С. 57–62.
2. Guarino N. Formal Ontology in Information Systems // Proceedings of FOIS'98. Trento, Italy, 1998. Amsterdam: IOS Press, 1998. P. 3–15.
3. Deliyanni A., Kowalski R. A. Logic and Semantic Networks // Comm. ACM. 1979. Vol. 22, N. 3. P. 184–192.
4. Загорюлько Ю. А. Технология разработки порталов научных знаний // Программные продукты и системы. 2009. № 4. С. 25–29.
5. Лукашевич Н. В. Тезаурусы в задачах информационного поиска. М.: Издательство МГУ, 2011. 512 с.
6. Hitzler P., Krötzsch M., Rudolph S. Foundations of Semantic Web Technologies. Chapman & Hall/CRC, 2009. 455 p.
7. McIlraith S. A., Son T. C., Zeng H. L. Semantic Web Services // IEEE Intelligent Systems. 2001. Vol. 16, N. 2. P. 46–53.

8. **Fuseki**: serving RDF data over HTTP. [Электронный ресурс]. URL: http://jena.apache.org/documentation/serving_data/ (дата обращения 30.10.2015).
9. **SPARQL Query Language for RDF**. W3C Recommendation 15 January 2008. [Электронный ресурс]. URL: <http://www.w3.org/TR/rdf-sparql-query/> (дата обращения 30.10.2015).
10. **Resource Description Framework (RDF)**. [Электронный ресурс]. URL: <http://www.w3.org/2001/sw/wiki/RDF> (дата обращения 30.10.2015).
11. **Antoniou G., Harmelen F.** Web Ontology Language: OWL. Handbook on Ontologies. Berlin: Springer Verlag, 2003. P. 67–92.
12. **Benjamins V. R., Fensel D.** Community is Knowledge! // Proc. of 11th Banff Knowledge Acquisition for Knowledge-based Systems workshop KAW'98. Banff, Canada, April 1998. Calgary: SRDG Publications, Department of Computer Science, University of Calgary, 1998. P. KM.2-1–KM.2-18.
13. **Hillmann D.** Using Dublin Core. [Электронный ресурс]. URL: <http://dublincore.org/documents/usageguide/> (дата обращения 30.10.2015).
14. **OWL-S: Semantic Markup for Web Services**. [Электронный ресурс]. URL: <http://www.w3.org/Submission/OWL-S/> (дата обращения 30.10.2015).
15. **Загорюлько Ю. А., Боровикова О. И.** Программная оболочка для построения многоязычных тезаурусов предметных областей, ориентированная на экспертов // Труды 13-й Нац. конф. по искусственному интеллекту с междунар. участием КИИ-2012. Белгород: Изд-во БГТУ, 2012. Т. 4. С. 76–83.
16. **Маннинг К. Д., Рагхаван П., Шютце Х.** Введение в информационный поиск. М.: Вильямс, 2011. 528 с.
17. **Копайгородский А. Н.** Интеграция данных в исследованиях энергетики на основе онтологий // Информационные и математические технологии в науке и управлении. Труды XVII Байкальской Всерос. конф. "Информационные и математические технологии в науке и управлении": Часть III. Иркутск: ИСЭМ СО РАН, 2012. С. 62–68.
18. **Массель Л. В., Ворожцова Т. Н., Копайгородский А. Н.** и др. Применение онтологий в исследованиях и поддержке принятия решений в энергетике // Материалы Всерос. конф. с междунар. участием "Знания — Онтологии — Теории". Новосибирск: ИМ СО РАН, 2013. Том 2. С. 29–38.
19. **Lausen H., Stollberg M., Hernández R. L.** et al. Semantic Web Portals — state of the art survey. Technical Report TR-2004-04-03. DERI, 2004.
20. **Corcho O., Gómez-Pérez A., López-Cima A.** et al. ODESeW. Automatic Generation of Knowledge Portals for Intranets and Extranets // The Semantic Web — ISWC 2003. Second Int. Semantic Web Conf., Sanibel Island, FL, USA, October 20–23, 2003. Berlin: Springer-Verlag, 2003. LNCS, Vol. 2870. P. 802–817.
21. **Spyns P., Oberle D., Volz R.** et al. OntoWeb a Semantic Web Community Portal // Proc. Fourth Int. Conf. on Practical Aspects of Knowledge Management (PAKM), Vienna, Austria, December 2002. Berlin: Springer-Verlag, 2002. LNCS, Vol. 2569. P. 189–200.
22. **Интеллектуальная система тематического исследования научно-технической информации (ИСТИНА)** / Под ред. акад. В. А. Садовниченко. М.: Изд-во МГУ, 2014. 262 с.
23. **Jin Y., Decker S., Wiederhold G.** OntoWebber: model-driven ontology-based web site management // Proc. of SWWS'01. The first Semantic Web Working Symposium. Stanford University, California, USA, July 30 — August 1, 2001. P. 529–547.
24. **DeRose P., Shen W., Chen F.** et al. Building Structured Web Community Portals: A Top-Down, Compositional, and Incremental Approach // Proc. of VLDB'07. Vienna, Austria, September 23–28, 2007. P. 399–410.
25. **Ferrara E., De Meo P., Fiumara G., Baumgartner R.** Web Data Extraction, Applications and Techniques: A Survey // Knowledge-Based Systems. 2014. Vol. 70. P. 301–323.
26. **Bernabe-Moreno J., Tejada-Lorente A., Porcel C.** et al. CARESOME: A system to enrich marketing customers acquisition and retention campaigns using social media information // Knowledge-Based Systems. 2015. Vol. 80. P. 163–179.
27. **Cobo M. J., Martínez M. A., Gutiérrez-Salcedo M.** et al. 25 years at Knowledge-Based Systems: A bibliometric analysis // Knowledge-Based Systems. 2015. Vol. 80. P. 3–13.
28. **Ланин В. В., Мальцев П. А., Лядова Л. Н.** Технологии сбора и анализа информации для исследовательского портала // Материалы Четвертой междунар. науч.-техн. конф. "Инфокоммуникационные технологии в науке, производстве и образовании" (Инфоком 4): Часть I. Ставрополь: Сев.-Кавк. гос. техн. ун-т, 2010. С. 218–222.
29. **Zhai Y., Liu B.** Extracting Web Data Using Instance-Based Learning // Proc. of 6th Int. Conf. on Web Information Systems Engineering (WISE-05) November 20–22, 2005. New York, NY, USA, P. 318–331.

Technology for Building Subject-Based Intelligent Scientific Internet Resources Based on Ontology

Yu. A. Zagorulko, zagor@iis.nsk.su, **G. B. Zagorulko**, gal@iis.nsk.su, **O. I. Borovikova**, olesya@iis.nsk.su,
A. P. Ershov Institute of informatics systems of Siberian Branch of Russian Academy of Sciences,
Novosibirsk, 630090, Russian Federation

Corresponding author:

Zagorulko Yury A., Head of Laboratory, A. P. Ershov Institute of informatics systems of Siberian Branch of Russian Academy of Sciences, Novosibirsk, 630090, Russian Federation
e-mail: zagor@iis.nsk.su

Received on September 28, 2015

Accepted on November 19, 2015

The paper discusses a technology of the development of subject-based Intelligent Scientific Internet Resources which provides content-based access to systematized scientific knowledge and information resources (ISIR) related to a certain knowledge area and to their intelligent processing facilities. An important merit of ISIR is its ability to reduce appreciably the time required to access and analyze information thanks to the accumulation of the semantic descriptions of the basic entities of a knowledge area being modeled, the Internet resources relevant to this area, and

the information processing facilities used in it directly in the ISIR content. The specific features of the technology are the use of ontology and semantic network formalisms as well as orientation to experts. For the considered technology ontology is used not only for formalization and systematization of above-mentioned kinds of knowledge, information resources and processing facilities, but also for organization of convenient access to them.

Keywords: intelligent scientific Internet resource, ontology, semantic network, thesaurus, technology, metasearch, information extraction

Acknowledgements: This work was supported by the Russian Foundation for Basic Research, project no. 13-07-00422.

For citation:

Zagorulko Yu. A., Zagorulko G. B., Borovikova O. I. Technology for Building Subject-Based Intelligent Scientific Internet Resources Based on Ontology, *Programmnyaya Ingeneria*, 2016, vol. 7, no. 2, pp. 51–60.

DOI: 10.17587/prin.7.51-60

References

1. Zagorul'ko Ju. A., Zagorul'ko G. B., Shestakov V. K., Kononenko I. S. Konceptcija i arhitektura tematičeskogo intelektual'nogo nauchnogo internet-resursa (Concept and architecture of thematic intelligent scientific Internet resource). *Trudy XV Vseros. nauch. konf. RCDL'2013*. Jaroslavl', 14–17 October, 2013, Jaroslavl', JarGU, 2013, pp. 57–62 (in Russian).
2. Guarino N. Formal Ontology in Information Systems. *Proceedings of FOIS'98*, Trento, Italy, 1998, Amsterdam: IOS Press, 1998, pp. 3–15.
3. Deliyanni A., Kowalski R. A. Logic and Semantic Networks, *Comm. ACM*, 1979, vol. 22, no. 3, pp. 184–192.
4. Zagorul'ko Ju. A. Tehnologija razrabotki portalov nauchnyh znanij (Technology of development of scientific knowledge portals). *Programmnye Produkty i Sistemy*, 2009, no. 4, pp. 25–29 (in Russian).
5. Lukashevich N. V. *Tezaurusy v zadachah informacionno-poiska* (Thesauri in information retrieval tasks). Moscow, Izd-vo MGU, 2011. 512 p. (in Russian).
6. Hitzler P., Krotzsch M., Rudolph S. *Foundations of Semantic Web Technologies*. Chapman & Hall/CRC, 2009. 455 p.
7. McIlraith S. A., Son T. C., Zeng H. L. Semantic Web Services, *IEEE Intelligent Systems*, 2001, vol. 16, no. 2, pp. 46–53.
8. Fuseki: serving RDF data over HTTP [Electronic resource], available at: http://jena.apache.org/documentation/serving_data/ (date of access 30.10.2015).
9. SPARQL Query Language for RDF. W3C Recommendation 15 January 2008 [Electronic resource], available at: <http://www.w3.org/TR/rdf-sparql-query/> (date of access 30.10.2015).
10. Resource Description Framework (RDF) [Electronic resource], available at: <http://www.w3.org/2001/sw/wiki/RDF> (date of access 30.10.2015).
11. Antoniou G., Harmelen F. *Web Ontology Language: OWL. Handbook on Ontologies*. Berlin, Springer Verlag, 2003, pp. 67–92 (in Russian).
12. Benjamins V. R., Fensel D. Community is Knowledge! *Proc. of 11th Banff Knowledge Acquisition for Knowledge-based Systems workshop KAW'98* (Banff, Canada, April 1998). Calgary: SRDG Publications, Department of Computer Science, University of Calgary, 1998, pp. KM.2-1–KM.2-18.
13. Hillmann D. Using Dublin Core [Electronic resource], available at: <http://dublincore.org/documents/usageguide/> (date of access 30.10.2015).
14. OWL-S: Semantic Markup for Web Services [Electronic resource], available at: <http://www.w3.org/Submission/OWL-S/> (date of access 30.10.2015).
15. Zagorul'ko Ju. A., Borovikova O. I. Programmnyaya obolochka dlja postroenija mnogojazychnyh tezaurusov predmetnyh oblastej, orientirovannaja na jekspertov (Expert-oriented program shell for building multilingual thesauri of the subject domains). *Trudy 13-j nac. konf. po iskusstvennomu intellektu s mezhdunar. uchastiem KII-2012*. Belgorod, Izd-vo BGTU, 2012, vol. 4, pp. 76–83 (in Russian).
16. Manning K. D., Raghavan P., Shjute H. *Vvedenie v informacionnyj poisk* (Introduction to Information Retrieval), Moscow, Williams, 2011, 528 p. (in Russian).
17. Kopajgorodskij A. N. Integracija dannyh v issledovanijah jenergetiki na osnove ontologii (Data integration of power research based on ontologies), *Informacionnye i matematicheskie tehnologii v nauke i upravlenii. Trudy XVII Bajkal'skoj Vseros. konf. "Informacionnye i matematicheskie tehnologii v nauke i upravlenii"*: Chast' III, Irkutsk, ISEM SO RAN, 2012, vol. 3, pp. 62–68 (in Russian).
18. Massel' L. V., Vorozhova T. N., Kopajgorodskij A. N., Makagonova N. N., Skripkin S. K. Primenenie ontologii v issledovanijah i podderzhke prinjatija reshenij v jenergetike (Application of the ontologies in studies and decision support in energetics), *Materialy Vseros. konf. s mezhdunar. uchastiem "Znanija — Ontologii — Teorii"*, Novosibirsk, IM SO RAN, 2013, vol. 2, pp. 29–38 (in Russian).
19. Lausen H., Stollberg M., Hernández R. L., Ding Y., Han S.-K., and Fensel D. Semantic Web Portals - state of the art survey. Technical Report TR-2004-04-03. DERI, 2004.
20. Corcho O., Gomez-Perez A., López-Cima A., Lopez-Garcia V., Suarez-Figueroa M. C. ODESeW. Automatic Generation of Knowledge Portals for Intranets and Extranets, *The Semantic Web — ISWC 2003*. Second Int. Semantic Web Conf., Sanibel Island, FL, USA, October 20–23, 2003. Berlin, Springer-Verlag, 2003, LNCS, vol. 2870, pp. 802–817.
21. Spyns P., Oberle D., Volz R., Zheng J., Jarrar M., Sure Y., Studer R., Meersman R. OntoWeb a Semantic Web Community Portal, *Proc. Fourth Int. Conf. on Practical Aspects of Knowledge Management (PAKM)*, Vienna, Austria, December 2002. Berlin, Springer-Verlag, LNCS, vol. 2569, 2002, pp. 189–200.
22. *Intellektual'naja sistema tematičeskogo issledovanija nauchno-tehnicheskoi informacii (ISTINA)* (Intellectual system of thematic analysis of scientific and technical information (ISTINA)) / Ed. V. A. Sadovnichiy. Moscow, Izd-vo MGU, 2014, 262 p. (in Russian).
23. Jin Y., Decker S., Wiederhold G. OntoWebber: model-driven ontology-based web site management, *Proc. of SWWS'01. The first Semantic Web Working Symposium*, Stanford University, California, USA, July 30 – August 1, 2001, pp. 529–547.
24. DeRose P., Shen W., Chen F., Doan A. H., Ramakrishnan R. Building Structured Web Community Portals: A Top-Down, Compositional, and Incremental Approach, *Proc. of VLDB'07*. Vienna, Austria, September 23–28, 2007, pp. 399–410.
25. Ferrara E., De Meo P., Fiumara G., Baumgartner R. Web Data Extraction, Applications and Techniques: A Survey, *Knowledge-Based Systems*, 2014, vol. 70, pp. 301–323.
26. Bernabe-Moreno J., Tejeda-Lorente A., Porcel C., Fujita H., Herrera-Viedma E. CARESOME: A system to enrich marketing customers acquisition and retention campaigns using social media information, *Knowledge-Based Systems*, 2015, vol. 80, pp. 163–179.
27. Cobo M. J., Martínez M. A., Gutiérrez-Salcedo M., Fujita H., Herrera-Viedma E. 25 years at Knowledge-Based Systems: A bibliometric analysis, *Knowledge-Based Systems*, 2015, vol. 80, pp. 3–13.
28. Lanin V. V., Mal'cev P. A., Ljadova L. N. Tehnologij sbora i analiza informacii dlja issledovatel'skogo portal (Technology of collection and analysis of information for research portal), *Materialy. Četvertoj mezhdunar. nauch.-tehn. konf. "Infokommunikacionnye tehnologii v nauke, proizvodstve i obrazovanii" (Infokom 4)*: Chast' I. Stavropol', Sev.-Kavk. gos. tehn. un-tet, 2010, pp. 218–222 (in Russian).
29. Zhai Y., Liu B. Extracting Web Data Using Instance-Based Learning. *Proc. of 6th Int. Conf. on Web Information Systems Engineering (WISE-05)*, November 20–22, 2005, New York, NY, USA, 2005, pp. 318–331.

В. П. Потапов, д-р техн. наук, проф., зам. директора, e-mail: potapov@ict.sbras.ru,
С. Е. Попов, канд. техн. наук, ст. науч. сотр., e-mail: popov@ict.sbras.ru,
Федеральное государственное бюджетное учреждение науки Институт вычислительных технологий Сибирского отделения Российской академии наук, г. Новосибирск

Высокопроизводительный алгоритм роста регионов для развертки интерферометрической фазы на базе технологии CUDA

Представлена высокопроизводительная версия алгоритма, именуемая алгоритмом роста регионов для развертки фазы интерферометрических данных. Создан метод упаковки/распаковки значений индексов точек снимка на основе битовых наборов. Разработаны быстрый алгоритм создания кольца развертки и алгоритм обновления карты номеров регионов в процедуре их объединения. Программная реализация алгоритма предусматривает возможность ее запуска на многопроцессорных платформах в мультипоточковом режиме с использованием графических устройств, поддерживающих технологию NVIDIA CUDA.

Ключевые слова: параллельные вычисления, технология CUDA, развертка фазы, алгоритм роста регионов

Введение

Ключевым этапом интерферометрической обработки изображений является операция восстановления (развертки) фазы. Развертка фазы является наиболее сложным этапом интерферометрической технологии, поскольку на получаемых интерферограммах практически всегда присутствуют разрывы, положение которых неизвестно [1–3]. Для установления положения разрывов фазы необходима дополнительная информация, такая как значения интенсивности принятого сигнала на двух снимках, когерентности между двумя снимками, априорная информация о рельефе и др. После включения в постановку такой информации задача развертки становится нелинейной и приобретает большую размерность.

Задаче развертки фазы в космической радиолокационной интерферометрии посвящено большое число работ [4–7], в которых описаны различные алгоритмы ее решения. Однако во многих публикациях на эту тему не уделяется должного внимания вопросам практической реализации описанных алгоритмов. Математические модели, обсуждаемые в работах, как правило, имеют общие решения без возможности их программной реализации. Интерферометрические данные (снимки), используемые в расчетах, зачастую имеют малый размер, а результаты работы программной реализации алгоритма представляются в текстовом виде.

Однако следует отметить, что существует ряд работ, направленных на оптимизацию вычислительной составляющей алгоритмов развертки фазы в целях улучшения производительности. Так, в работе [8] рассматривается метод для изображений с многоканальным каналом и мультибазовыми линиями. Основная идея представленного в этой работе алгоритма состоит в применении метода наименьших квадратов над полем разностей оцениваемых и абсолютных фазовых градиентов для всех интерферограмм в частотной области одновременно. Расчеты проводят с использованием быстрого двумерного преобразования Фурье на взаимно ортогональных фазовых градиентах. Показано, что для развертки фазы снимков размером 5000×5000 точек требуется порядка 6 мин.

Интересной, по мнению авторов, является работа [9], в которой методы развертки фазы применены для ОQ-микроскопии (*Optical Quadrature Microscopy*). Авторы работы [9] используют технологию CUDA для улучшения производительности алгоритма сопряженных градиентов и метода наименьших квадратов с весовыми коэффициентами. Показано, что процесс развертки фазы снимка размером 640×480 точек занял порядка 3 с.

В предложенных подходах, также как и в других им подобных, используются так называемые методы минимальной нормы. Эти методы относятся к классу глобальных, работают сразу со всеми значениями интерферограммы и относительно легко переносятся

на параллельно-вычислительные платформы. Однако с увеличением размеров интерферометрических снимков сильно страдает точность значений разворачиваемой фазы, что, например, подтверждается размерами изображений для ОQ-микроскопии [9]. Для данных с большой размерностью, низким коэффициентом когерентности и областями, где значения абсолютной фазы терпят разрыв, подобные методы показывают большую погрешность. Уединенный разрыв абсолютной фазы дает в решении локальную ошибку, значение которой убывает обратно пропорционально расстоянию до разрыва [10, 11]. Более того, остается вопрос эффективной производительности, что, в частности, вытекает из рассмотренных примеров [8, 9].

Проведенный анализ работ в области восстановления абсолютной фазы показал, что наиболее популярные оптимизации проводят для двух классических алгоритмов. Это алгоритм роста регионов (*Region-Growing Algorithm*) и алгоритм наименьшей стоимости потока (*Minimum Cost Flow Algorithm*). Именно эти алгоритмы реализованы в популярном программном продукте Exelis ENVI. Данные алгоритмы относятся к классу локальных и для формирования оценки абсолютной фазы в точке используют лишь значения на интерферограмме, лежащие в окрестности этой точки, что существенно снижает вероятность ошибки.

Существует большое число различных расширений и вариаций этих методов, улучшающих их производительность и точность [12, 13]. Так, в работе [12] описан алгоритм функций наименьшей стоимости потока, разделяющий целевое изображение на небольшие сцены, которые разворачиваются независимо. Объединение сцен происходит за счет суперпозиций вейвлет-преобразований. Показано, что снимок размером 5167×3400 точек при количестве независимых сцен, равным шести, разворачивается около 8 мин.

Однако, как и в случае с глобальными методами, время исполнения локальных алгоритмов увеличивается нелинейно относительно размера изображения.

С учетом изложенного выше, задача уменьшения времени работы алгоритмов развертки фазы на интерферометрических данных за счет применения параллельных процедур для отдельных функциональных составляющих уже существующих методов восстановления фазы является весьма актуальной. Ее решение востребовано в современных программных системах процессинга интерферометрических изображений, включая обработку радарных снимков.

Развертка интерферометрической фазы. Общие понятия

Интерферограмма строится путем комплексного перемножения основного изображения (*master*) и изображения, комплексно-сопряженного к вспомогательному (*slave*) [1, 14]. Одновременно с подсчетом интерферограммы проводится подсчет когерентности. Когерентность — это параметр, характеризующий согласованность (корреляцию) значений основного

и вспомогательного изображений. Близкое к единице значение когерентности говорит о хорошем соотношении сигнал/шум для интерферограммы. Значения когерентности задаются матрицей ($M \times N$) и лежат в диапазоне от 0 до 1.

Цифровая интерферограмма является матрицей ($M \times N$) главных значений фазы Φ , каждое из которых лежит в диапазоне $[-\pi; \pi]$ радиан. Реальные же значения фазы, имеющие непосредственный физический смысл, могут лежать в более широком диапазоне. Эти значения фазы называются абсолютными Ψ :

$$\Psi = \{\psi_{m,n}, m = 0, \dots, M-1; n = 0, \dots, N-1\};$$

$$\Phi = \{\phi_{m,n}, m = 0, \dots, M-1; n = 0, \dots, N-1\}.$$

Абсолютные значения фазы, различающиеся на целое число фазовых оборотов ($2\pi k$), имеют одинаковое главное значение фазы, т. е. неразличимы на интерферограмме, что составляет фазовую неоднозначность.

Задачей развертки фазы называется задача реконструкции матрицы абсолютной фазы Ψ по заданной матрице главного значения фазы Φ с точностью до произвольной аддитивной постоянной:

$$W[\psi_{m,n}] = \phi_{m,n}, m = 0, \dots, M-1; n = 0, \dots, N-1, \quad (1)$$

где $W[\cdot]$ — оператор свертки по модулю 2π радиан.

Значения абсолютной фазы $\psi_{m,n}$ удовлетворяют уравнению (1) тогда и только тогда, когда:

$$\psi_{m,n} = \phi_{m,n} + 2\pi k_{m,n}, \quad (2)$$

$$m = 0, \dots, M-1; n = 0, \dots, N-1$$

при условии

$$|\psi_{m+1,n} - \psi_{m,n}| < \pi, |\psi_{m,n+1} - \psi_{m,n}| < \pi, \quad (3)$$

$$m = 0, \dots, M-2; n = 0, \dots, N-2,$$

где $k_{m,n}$ — число неоднозначности.

Задача развертки фазы (1), (2) при условии (3) имеет единственное решение, в предположении непрерывности значений абсолютной фазы. Действительно, свойство (3) позволяет определить конечные разности $\Delta_x = \psi_{m+1,n} - \psi_{m,n}$, $\Delta_y = \psi_{m,n+1} - \psi_{m,n}$, соответствующие абсолютной фазе. Положив в какой-либо точке (m_0, n_0) значение абсолютной фазы равным произвольно выбранной величине $\psi_{m_0, n_0} = 0$, значение абсолютной фазы $\psi_{m,n}$ в любой другой точке (m, n) можно вычислить суммированием (интегрированием) конечных разностей вдоль любого пути P из (m_0, n_0) точки в точку (m, n) .

В большинстве практических случаев интерферограмма всегда содержит разрывы фазы (нарушение условия (3)), обусловленные многими факторами (шумы радиолокационной аппаратуры, канала распространения радиоволн и т. п.), поэтому определение непрерывного пути интегрирования становится нетривиальной и неоднозначной задачей. Непротиворечивое решение развертки фазы перестает существовать.

Алгоритм роста регионов

Существует много алгоритмов развертки фазы, каждый из которых по-своему устраняет имеющуюся противоречивость, имеет свои преимущества и недостатки. На практике выбирается такой метод, который обеспечивает наиболее высокую точность решения задачи при заданных требованиях к объему оперативной памяти ЭВМ, вычислительных ресурсов и времени поиска решения. Одним из таких алгоритмов является алгоритм роста регионов [5].

Алгоритм роста регионов минимизирует ошибки за счет выбора начальных точек в областях с высокой когерентностью и проходит по пути интегрирования, где вероятность успеха тестов надежности наиболее высока. Алгоритм не ведет к распространению ошибок и устойчив по отношению к шуму малой интенсивности. Точки областей, которые трудно развернуть в силу высокой зашумленности, "проходятся" из разных направлений. Более того, основные процедуры (этапы) алгоритма роста регионов [5] могут быть переложены на парадигму параллельных вычислений, что обосновано, следующими его свойствами.

1. Процедура развертки целевого пиксела основана на линейном предсказании значения абсолютной фазы элемента интерферограммы на основании развернутых ранее значений. На каждом этапе итерации целевые пиксели образуют "кольцо роста", построенное для всех соседних точек, ранее развернутых. Это обстоятельство позволяет оценить изменение фазы целевого пиксела, находящегося между соседними, значения фаз которых больше π .

2. Проверка надежности на основе последовательности "фазовых предсказаний" применяется на каждой итерации с целью подтвердить или отклонить предлагаемое абсолютное значение фазы.

3. Когда регионы встречаются в общих точках, предпринимается попытка их объединения. Проверка согласованности чисел неоднозначности проводится в каждой точке, которые объединяют регионы. Во всех точках присоединяемого региона происходит замена его номера на номер присоединяемого региона [5].

Свойство 1 означает необходимость поиска целевых точек снимка для построения предсказанных значений абсолютной фазы [6] на каждой итерации алгоритма по всей области изображения. Для уменьшения времени работы данной процедуры на больших снимках (от 16 млн точек) были **использованы методы параллельной обработки потоков с применением функций атомарных битовых операторов на базе технологии NVIDIA CUDA.**

Процедура построения "кольца развертки" в памяти GPU и расчета главных значений фаз с проверкой их надежности (свойство 2) накладывает требования интенсивного обмена данными CPU/GPU. В целях уменьшения объемов передаваемой информации была применена **методика упаковки/распаковки индексов точек в/из битовые значения чисел типа Long.**

Процедура объединения регионов, характеризующая свойство 3, порождает необходимость обновления чисел неоднозначности присоединяемого региона в общих точках, а также номеров региона в ранее развернутых. В данной процедуре были использованы **методы работы с "paged-pinned memory" технологии NVIDIA CUDA для уменьшения времени трансфера данных между памятью CPU/GPU.**

В разделе "Улучшенный алгоритм роста регионов" подробно представлены все этапы действий алгоритма, отражающие данные свойства, а также приведено математическое описание расчетов значений ключевых переменных и тестов надежности.

Постановка задачи

Разработать улучшенный алгоритм роста регионов, позволяющий существенно (в 5 и более раз) снизить время развертки интерферометрической фазы для снимков большого размера (от 16 млн точек) относительно базового алгоритма программного комплекса Exelis ENVI за счет применения технологии параллельных вычислений NVIDIA CUDA и многопоточного программирования на языке Java multithreading на отдельных этапах обработки радарных данных.

Программное обеспечение и средства разработки

Программный код алгоритма разрабатывался с применением языков программирования Java 8 (компилятор javac версии 1.8.0_65), C++ 2013/NVIDIA CUDA (компиляторы cl версии 18.00.40629 и nvcc версии 7.5.17). Были использованы стандартные Java-пакеты из поставки Java SE Development Kit 8, а также сторонний пакет Jcudalib-0.7.5 для работы со стандартными функциями пакета NVIDIA cuda_7.5.18_windows.

Разработка велась в среде IDE Eclipse Mars с использованием компилятора C++ из поставки IDE Microsoft Visual Studio 2013 Community Edition под управлением операционной системы Microsoft Windows 8.1 Professional Edition.

Улучшенный алгоритм роста регионов

В контексте разрабатываемого алгоритма понятия "развертка фазы" и "развертка точки/пиксела" будем считать равнозначными. В табл. 1 даны пояснения к элементам языков программирования Java и C++, используемым в псевдокоде далее в статье.

В псевдокоде, представленном ниже, полужирным курсивом будем выделять конструкции, программные операторы и функции языков Java и C++ (CUDA), определяемые пользователем процедуры и объекты — полужирным прямым шрифтом, переменные, индексы и элементы массивов — курсивом.

На рис. 1 и 2 приведены графические пояснения терминов, применяемых на дальнейших этапах.

Ключевые слова, операторы, объекты и методы псевдокода алгоритма [16, 17]

Название ключевого слова, оператора, объекта или метода	Описание
<<, >>, , &, →	Битовые операторы сдвига влево, сдвига вправо, OR, AND, соответственно (Java/C++), и лямбда-оператор языка Java 8
%, (int)	Операторы языков Java/C++ получения остатка и получения ближайшего целого от деления двух переменных, соответственно
Float, Integer, Short, Boolean	Ключевые слова для обозначения типов переменных Java
if, else, for, while, continue, break	Ключевые слова языка Java/C++ для обозначения условия, альтернативного условия, цикла, цикла с условием "пока", продолжения цикла со следующей итерации и останова цикла, соответственно
atomicOr(x,y)	Функция языка C++ (CUDA), которая выполняет атомарную операцию логического OR двух аргументов
blockIdx.x, blockDim.x, threadIdx.x	Переменные языка C++ (CUDA), содержащие значения индекса блока в сетке, размер блока, индекс точки в блоке, соответственно
class, implements, new	Ключевые слова языка Java для обозначения класса, указания его интерфейса и создания экземпляра объекта, соответственно
HashMap <type key, type value>	Ключевое слово языка Java для обозначения объекта-контейнера, хранящего пары ключ-значение указанного типа
ArrayList<value>	Ключевое слово для обозначения динамического массива в языке Java
Callable	Параметризованный интерфейс языка Java, инкапсулирующий объект, методы которого могут выполняться асинхронно
get(key), put(key,value), remove(key), size()	Методы объектов-контейнеров типов HashMap, Set и пр. в языке Java, позволяющие получить значение по ключу, записать пару ключ-значение в объект-контейнер, удалить ее по значению ключа и получить количество пар, соответственно
get(value), add(value), remove(value), length()	Методы динамического массива в языке Java, позволяющие получить значение, записать, удалить его и получить длину массива, соответственно.
keySet()	Метод объектов-контейнеров типа HashMap в языке Java, позволяющий получить объект (Set), содержащий список всех ключей
iterator()	Метод объекта Set в языке Java, возвращающий объект Iterator, позволяющий последовательно проводить итерации по каждому ключу
hasNext(), next()	Методы объекта Iterator в языке Java. Первый метод возвращает значение типа Boolean, показывающее, не достигнут ли конец списка ключей объекта Set. Второй метод возвращает следующий ключ из списка
stream(), parallelStream(), parallel()	Методы объектов-контейнеров типа HashMap, Set, Arrays в языке Java 8. Методы возвращают объект типа Stream, представляющий элементы объекта-контейнера как последовательный или параллельный поток, соответственно
forEach()	Метод объекта Stream в языке Java 8, который позволяет итерировать в цикле элементы потока

+	+	+
+	+	+
$c < 0,2$	$c > 0,2$	$c < 0,2$
$c < 0,2$	$c < 0,2$	$c < 0,2$

а)

$c < 0,2$	$c < 0,2$	$c < 0,2$
$c < 0,2$	$c < 0,2$	$c < 0,2$
$c < 0,2$	$c > 0,2$	$c < 0,2$
$c < 0,2$	$c < 0,2$	$c < 0,2$

б)

+	+	+
+	+	+
+	$c > 0,2$	+
+	+	+
+	+	+

в)

Рис. 1. Варианты размещения "неудачных" точек ($c > 0,2$). Плюсом помечены уже развернутые точки; c — значение элемента массива C (когерентность)

(-2N-2)		-2N		(-2N+2)
	(-N-1)	-N	(-N+1)	
-2	-1	<i>i</i>	1	2
	(N-1)	N	(N+1)	
(2N-2)		2N		(2N+2)

Рис. 2. Значения, прибавляемые к *i*-му индексу по направлениям D1 и D2

Входными данными алгоритма являются одномерные массивы значений интерферограммы ($\Phi = \{\varphi_i, i = 0, \dots, MN - 1\}$) и когерентности ($C = \{c_i, i = 0, \dots, MN - 1\}$) размерностью $M \times N$. Данные массивы создаются в оперативной памяти CPU.

Ниже представлены основные этапы улучшенного алгоритма роста регионов.

Этап 1. В глобальной памяти GPU создаем: массив, аналогичный массиву C , $U = \left\{u_c, c = 0, \dots, \frac{MN}{64} - 1\right\}$ — массив 64-битных чисел, содержащий индексы развернутых точек, и $F = \left\{f_c = 0, c = 0, \dots, \frac{MN}{64} - 1\right\}$ — массив 64-битных чисел, содержащий индексы "неудачных" точек. На рис. 1 представлены некоторые варианты "неудачных" точек.

В случае на рис. 1, *a* и рис. 1, *в*, если точка (выделена полужирным курсивом) не была развернута на текущей итерации, то она не будет развернута никогда, так как нет других направлений для расчета предсказываемых значений фазы. В случае на рис. 1, *б* точка (выделена полужирным курсивом) с приемлемым значением когерентности (кандидат на развертку) оказалась среди соседних точек, игнорируемых процедурой построения кольца роста.

Создаем массив $G = \left\{g_c = 0, c = 0, \dots, \frac{MN}{64} - 1\right\}$, содержащий индексы точек роста регионов, и массив $RM = \{rm_i = 0, i = 0, \dots, MN - 1\}$, содержащий номера регионов для развернутых точек с индексом *i*. Массивы G и RM создаются в *pinned memory* [15], которая дает возможность прямого доступа к памяти компьютера с GPU по запросу трансфера данных в/из ячейки без участия CPU.

В оперативной памяти CPU создаем: массив $\hat{F} = \{\hat{f}_i = 0, i = 0, \dots, MN - 1\}$, содержащий индикатор "неудачной" точки ($\hat{f}_i = 8$), и $\Psi = \{\psi_i = -10^6, i = 0, \dots, MN - 1\}$ — массив абсолютных значений фаз развернутых точек.

Определяем массивы констант $D1 = \{-N, -N + 1, 1, N + 1, N, N - 1, -1, -N - 1\}$, $D2 = \{-2N, -2N + 2, 2, 2N + 2, 2N, 2N - 2, -2, -2N - 2\}$, прибавляемых к *i*-му индексу точки для определения соседних по восьми направлениям (рис. 2).

Для массивов U, F, G вводим понятие значимого индекса точки. Точка *x* с индексом *i* в процессе работы алгоритма сохраняется в данных массивах сле-

дующим образом: $x_{(i \gg 6)} = x_{(i \gg 6)} | (1 \ll i)$, т. е. в 64-битном числе значимый индекс *i* кодируется битом, равным 1, незначимый индекс — битом, равным 0.

Типы данных в массивах U, F, G — *Long*; в массивах $D1, D2, RM$ — *Integer*; Φ, C, Ψ — *Float*; \hat{F} — *Short*.

Также на данном этапе создаются Java-объекты и переменные, используемые в последующих этапах алгоритма. Объект-контейнер `ovrlRegionPairs` типа `HashMap<String, HashMap<integer, float[2]>>`, содержащий пары ключ-значение (номера двух регионов ("*r1 r2*") и значения развернутых фаз их общих точек, соответственно их индексам). Объект `UnwrapRegionPixel`, наследуемый от интерфейса `Callable`, позволяющий получить развернутое значение фазы в *i*-й точке указанного региона. Объект-контейнер `tasks` типа `HashMap<short, UnwrapRegionPixel>`, содержащий пары ключ-значение (номер региона и объект `UnwrapRegionPixel`, см. этап 5). Переменная `isPixelUnwrap` типа `Boolean` указывает, была ли развернута *i*-я точка или нет.

Отметим, что массивы U и F изменяются единожды в каждой точке, т. е. значимый индекс не меняет своего состояния, будучи установленным один раз. Следовательно, можно передавать на сторону GPU только координаты измененных точек, и запускать процедуру обновления этих массивов только на сетке размером, равной количеству этих точек. С этой целью на стороне CPU создаются счетчики точек (`u_counter` и `f_counter`), изменивших свое состояние и массивы $\bar{U} = \{\bar{u}_w, w = 0, \dots, u_counter\}$ и $\bar{F} = \{\bar{f}_z, z = 0, \dots, f_counter\}$ типа *Long*, содержащие изменившиеся элементы массивов U и F , соответственно. А значения *i*-х индексов, соответствующих местам элементов \bar{u}_w и \bar{f}_z в массивах U и F , будем размещать в дополнительных объектах-контейнерах $\dot{U} = \left\{(\dot{u}_w, w), w = 0, \dots, u_counter, \dot{u}_w \in \left[0, \frac{MN}{64} - 1\right]\right\}$ и $\dot{F} = \left\{(\dot{f}_z, z), z = 0, \dots, f_counter, \dot{f}_z \in \left[0, \frac{MN}{64} - 1\right]\right\}$ типа

размещать в `HashMap<Integer, Integer>`. Такое решение позволит значительно снизить время работы алгоритма, так как трансфер массивов $\bar{U}, \bar{F}, \dot{U}, \dot{F}$ в память GPU на каждой итерации будет осуществляться частично, объемом $4u_counter$ и $4f_counter$ байт, соответственно.

Этап 2. Определяем начальные точки с индексами $S = \{s_l, l = 0, \dots, P - 1, s_l \in [0, MN - 1]\}$. Существует несколько критериев выбора начальных точек, в представленном алгоритме начальными выбирались те точки, где значения когерентности *C* наиболее высоки (от 0,97 до 1,0) [5]. Число *P* начальных точек определяется индивидуально по размеру интерферограммы. В представленном алгоритме поиск точек с максимальными значениями когерентности осуществлялся в равномерно распределенных окнах размером (M_x, M_y), где главные значения фазы φ , являются локально сглаженными [5], а также выбираются исходя из условия — не менее 200 точек на окно размером 4096×4096 точек.

```

// Запускаем на стороне CPU
Arrays.stream(S).parallel().forEach( $s_l \rightarrow \psi_{s_l} = \varphi_{s_l}$ )
for (l = 0 ... P - 1) {
     $rm_{s_l} = l;$ 
     $blockDim.x = 256; gridDim.x = (int) \frac{P}{blockDim.x} + P \% blockDim.x$ 
    if (P < 256) {  $blockDim.x = P, gridDim.x = 1;$ 
}

// Запускаем на стороне GPU
updateSeeds <<<blockDim.x, gridDim.x>>> ({ $s_l$ }) {
    l = blockDim.x*blockDim.x + threadIdx.x
    atomicOr( $u_{(s_l \gg 6)}, (l \ll (s_l \& 63))$ ) }

```

Рис. 3. Фрагмент псевдокода процедуры инициализации начальных точек и регионов

Начальные точки определяют регионы. Каждому региону присваивается свой номер ($r_l, l = 0, \dots, P - 1$). Нулевой номер присваивается региону с максимальным значением когерентности начальной точки, ($P - 1$)-й — с минимальным.

Абсолютные значения фазы считаются равными главным значениям фазы в начальных точках. Начальные точки считаются развернутыми. Число неоднозначности k_s , начальной точки равно 0.

Этап 3. Копируем массив **C** в глобальную память GPU. Заполняем массивы **Ψ**, **RM**, **U** согласно выбранным начальным точкам (рис. 3).

Этап 4. Для каждой развернутой точки региона ищем ближайшие соседние, неразвернутые точки (точки роста). Совокупность точек роста образует "кольцо роста". Данную процедуру выполняем на стороне GPU. Размер вычислительного блока ($blockDim.x$) устанавливаем равным 256 потоков, размер сетки

$gridDim.x = (int) \frac{MN}{blockDim.x} + (MN) \% blockDim.x$. Таким

образом, для каждого потока в блоке выполняем процедуру *kernel function*, описанную на рис. 4.

За счет того, что все инструкции на стороне GPU выполняются в SIMT-стиле (*single instruction multiply threads*), то на каждом шаге ветвления алгоритма те потоки, которые не удовлетворили условию, просто отсекаются. Таким образом удается проходить цикл не полностью, что значительно уменьшает время выполнения функции **createGrowthRing**.

Этап 5. Для каждой точки роста массива **G** выполняем процедуру развертки фазы. Если одна и та же точка более восьми раз не прошла тесты надежности [5], ее значение в массиве **F** устанавливается равным 8, а в массиве **F** соответствующий ее индексу бит — равным 1. Такие точки больше не будут участвовать в процессе развертки фазы. В контексте разрабатываемого алгоритма назовем их "неудачные" точки (*fail pixels*).

На данном этапе вводим вспомогательные массивы

$\hat{\Phi}^P = \{\hat{\varphi}_i, i = 0, \dots, 7\}$ — массив предсказанных значений по направлениям **D1** и **D2** [5], $\Omega = \{\omega_i, i = 0, \dots, 7, \omega_i = 0,5|1,0\}$ — массив коэффициентов предсказанных значений по направлениям **D1** и **D2** [5], константы

```

createGrowthRing <<<blockDim.x, gridDim.x>>> () {
    i = blockDim.x*blockDim.x + threadIdx.x //получаем i-й индекс точки
    j = (i >> 6); z = (i & 63); //получаем значение i-го индекса в 64-битных массивах
    if (((u_j >> z) & 1) = 0) { //если i-й индекс не значимый в массиве U
        if (c_i > t_c) { //если значение i-го индекса в массиве C больше порогового
            if (((f_j >> z) & 1) = 0) { //если i-й индекс не значимый в массиве F
                for (w = 0 ... 7) { //для каждого из восьми направлений
                    v = j + D1_w //вычисляем v-е направление
                    if (v > 0 AND v < M*N) { // проверка границ индекса согласно размерам снимка
                        if (((u_{(v >> 6)} >> (v & 63)) & 1) ≠ 0) { //если v-й индекс значимый в массиве U
                            atomicOr(g_j, (1 << z)) //устанавливаем индекс v значимым в массиве G
                        }
                    }
                }
            }
        }
    }
}

```

Рис. 4. Процедура построения кольца развертки createGrowthRing. Выполняется на стороне GPU

```

class UnwrapRegionPixel implements Callable<Float> {

    nC = -1;  $\hat{\Phi}^p$ ,  $\Omega$ ,  $\hat{v}$ ,  $\varphi^u$ ;

    makePrediction( $\hat{u}1, \hat{u}2, i$ ) { // Вычисляем предсказываемые значения развернутой фазы по
        // направлениям  $\hat{u}1, \hat{u}2$  [5]
        nC = nC + 1;  $\hat{\varphi}_{nC}^p = \left(1 + \left(1 - (\text{int}) \frac{\hat{u}2}{-10^6}\right)\right) * \hat{u}1 - \hat{u}2 * \left(1 - (\text{int}) \frac{\hat{u}2}{-10^6}\right)$ ;
         $\omega_{nC} = 0.5 + 0.5 * \left(1 - (\text{int}) \frac{\hat{u}2}{-10^6}\right)$ ;
         $\hat{v} = i$ 
    }

    call() { // Рассчитываем значение развернутой фазы  $\varphi^u$  согласно предсказанным значениям  $\hat{\varphi}_{nC}^p$ .
        // Проводим тесты надежности [5]
         $\hat{\varphi}^p = \frac{\sum_{l=0}^{nC} \hat{\varphi}_l^p * \omega_l}{\sum_{l=0}^{nC} \omega_l}$ ;  $k = (\text{int}) \left(\frac{\hat{\varphi}^p - \varphi_{\hat{v}}}{2 * \pi}\right)$ ;  $\varphi^u = \varphi_{\hat{v}} + 2 * \pi * k$ ;
         $d_p = \frac{\sum_{l=0}^{nC} |\hat{\varphi}_l^p - \hat{\varphi}^p| * \omega_l}{\sum_{l=0}^{nC} \omega_l}$ ;  $d_u = |\varphi^u - \hat{\varphi}^p|$ ;
        if( $d_p < t_p$  &&  $d_u < t_u$ ) { nC = -1; // Если тесты надежности пройдены }
        else { nC = -1; isPixelUnwrap = false // Иначе устанавливаем индикатор «не развернута» }
    }
}

```

Рис. 5. Объект-класс UnwrapRegionPixel, наследуемый от интерфейса Callable, содержащий методы для получения значения развернутой фазы в точке

$t_p = \frac{\pi}{2}$, $t_u = \frac{\pi}{2}$, $t_c = 0,97$ [5], необходимые для проверки тестов надежности на этапе 5, и константы $t_m = 3$, $t_r = \frac{3}{4}$, применяемые в процедуре объединения регионов на этапе 6.

Используем класс UnwrapRegionPixel для развертки фазы в i -й точке. Его программная структура и псевдокод методов приведены на рис. 5.

Если точка разворачивается только в одном регионе, то объект tasks будет содержать только один объект с ключом-номером этого региона. Асинхронный запуск с помощью метода UnwrapRegionPixel.submit() [18] установит значение развернутой фазы равное φ^u . Если тесты надежности пройдены, происходит обновление значения i -й точки в массиве Ψ ($\psi_i = \varphi^u$) и точка считается развернутой. Если переменная-индикатор isPixelUnwrap = false, увели-

```

createOvrlpRegionPairs (tasks, i) {
    itr_task = task.keySet.iterator(); //Получаем список номеров регионов
    while(itr_task.hasNext()) { //Пока не достигли конца списка
        r1 = task.get(itr_task.next()) //Получаем номер первого региона,
        //итератор переводится в следующее положение
        while(itr_task.hasNext()) { //Продолжаем итерацию по списку
            r2 = task.get(itr_task.next()) //Получаем номер второго региона в списке
            " $\hat{r}_1_{\hat{r}_2}$ " = min( $r_1, r_2$ ) + "_" + max( $r_1, r_2$ ) //Формируем индекс пары
            if(!ovrlRegionPairs.contains(" $\hat{r}_1_{\hat{r}_2}$ ")) { //Если нет такой пары
                ovrlRegionPairs.add(" $\hat{r}_1_{\hat{r}_2}$ ") }
             $\varphi_1^u = \text{tasks.get}(\hat{r}_1). \varphi^u$ ;  $\varphi_2^u = \text{tasks.get}(\hat{r}_2). \varphi^u$  //Получаем значения
            //развернутых фаз для каждого региона в паре
            ovrlRegionPairs.get(" $\hat{r}_1_{\hat{r}_2}$ ").add(i, [ $\varphi_1^u, \varphi_2^u$ ]) //Добавляем общую точку
        }
        tasks.remove(r1) // Удаляем объект UnwrapRegionPixel с ключом равным первому региону
        itr_task = task.keySet.iterator(); //Заново получаем список номеров регионов,
        //продолжаем итерации пока все пары не будут созданы }}
}

```

Рис. 6. Процедура формирования пар перекрывающихся регионов в виде объекта HashMap

чиваем на 1 соответствующее значение в массиве \hat{F} ($f_i = f_i + 1$).

Если точка является общей для нескольких регионов, то объект **tasks** будет содержать несколько объектов **UnwrapRegionPixel**, каждый с ключом, равным номеру региона. Асинхронный запуск сформирует значения переменной φ^u для каждого **UnwrapRegionPixel** \in **tasks**. Если любой из объектов **UnwrapRegionPixel** установил

переменную $isPixelUnwrap = \text{false}$, то для i -й точки, аналогично, считаем $(\hat{f}_i = \hat{f}_i + 1) \in \hat{F}$. Иначе формируются пары перекрывающихся регионов, которые добавляют/обновляют текущие значения в объект **ovrlpRegionPairs** (рис. 6).

Порядок выполнения команд в процедуре развертки фазы представлен на рис. 7.

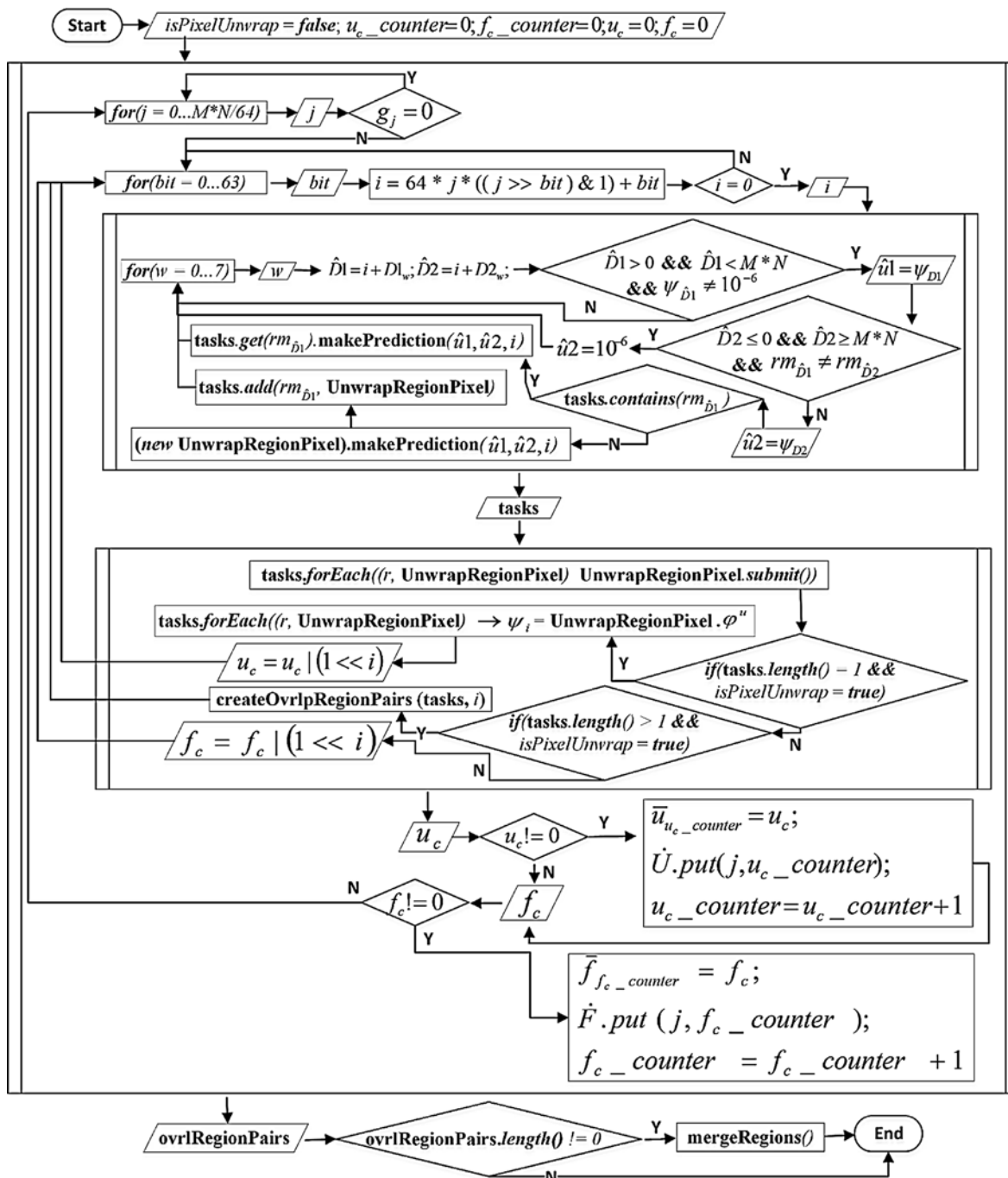


Рис. 7. Блок-схема процедуры развертки фазы

```

// Запускается на стороне CPU
blockDim.x = 256; gridDim.x = (int)  $\frac{M*N}{blockDim.x} + (M * N) \% blockDim.x$ 

// Запускается на стороне GPU
updateRegionsMap <<<blockDim.x, gridDim.x>>> ( $\hat{r}_1, \hat{r}_2$ ) {
    i = blockDim.x*blockDim.x + threadIdx.x
    if (rmi =  $\hat{r}_2$ ) { rmi =  $\hat{r}_1$  } // Если найден заменяемый номер, обновляем
                                     соответствующий элемент массива RM
}

```

Рис. 8. Фрагмент псевдокода процедуры замены индексов в карте регионов (массив **RM**)

Этап 6. Согласно алгоритму, описанному в работе [5], регионы разворачиваются независимо друг от друга, встречаясь в общих точках, в которых запускается процедура объединения. Узким местом в производительности данной процедуры является пост-процесс замены номера присоединяемого региона в массиве **RM** на номер присоединяющего региона. Обновляемые элементы могут быть распределены неравномерно по всему массиву **RM**, поэтому их поиск необходимо осуществлять перебором всех индексов.

Аналогично этапу 4, каждый элемент массива **RM** будем обрабатывать в вычислительном потоке (*thread*) на стороне GPU. На вход функции ядра передаем два параметра \hat{r}_1 и \hat{r}_2 типа *Integer*, номер заменяющего и заменяемого региона, соответственно. Так как массив **RM** на этапе 1 был создан в так называемой *pinned memory*, то осуществлять его трансфер в направлении CPU \leftrightarrow GPU нет необходимости. Функция ядра получает доступ к элементам данного массива через PCI Express-шину. Таким образом, процедура замены номеров регионов в массиве **RM** будет выглядеть так, как это представлено на рис. 8.

Если во время развертки точек были обнаружены общие точки как минимум для двух регионов, то по завершению этапа 5 получим сформированный объект-контейнер *ovrlRegionPairs* с соответствующими парами номеров регионов. Следовательно, необходимо выполнить процедуру их объединения.

В процессе выполнения данного этапа необходимо рассчитать модуль разностей чисел неоднозначности общих точек двух объединяемых регионов. И если отношение числа точек, составляющих модуль, к числу всех общих точек двух объединяемых регионов больше, чем константа t_{rr} [5], регионы считаются объединенными. Те точки, которые "не согласились" с модой, считаются "неудачными", что влечет за собой модификацию массивов \bar{F}, \hat{F} и переменной $f_c_counter$. Аналогично для точек, составляющих модуль, обновятся массивы \bar{U}, \hat{U} и переменная $u_c_counter$. Для этих же точек значение развернутой фазы берется из региона с меньшим номером и этот же номер записывается в соответствующие позиции в массиве **RM**. Процедура *updateRegionsMap* обновит номер присоединяемого региона \hat{r}_2 на номер \hat{r}_1 , где были только

его точки в позициях элементов массива **RM**. Если мода разностей чисел неоднозначности не существует или выборка получилась мультимодальной, то все точки в паре двух регионов объявляются "неудачными". После объединения одной пары регионов, все их общие точки удаляются из других пар объединяемых регионов, если таковые там имеются.

Процедуру объединения двух регионов назовем *mergePair*. В данной процедуре создается вспомогательный объект-контейнер *diffIdx* типа *HashMap<Integer, ArrayList<Integer>>*, содержащий пары ключ-значение (разность чисел неоднозначности и массив индексов пар точек, которые дают эту разность), переменная-индикатор *isMultiMode*, показывающая, является ли выборка мультимодальной или нет, и переменная *modeIdxCounter*, сохраняющая максимальное значение числа точек, образующих разность в процессе определения моды выборки. Порядок действий в процедуре объединения двух регионов *mergePair* представлен на рис. 9.

Тогда процедура объединения всех пар регионов будет иметь вид, представленный на рис. 10.

Этап 7. На данном этапе осуществляем трансфер вспомогательных массивов $\bar{U}, \bar{F}, \hat{U}, \hat{F}$ на сторону GPU и обновление соответствующих индексов массивов **U, F**. На рис. 11 представлена соответствующая процедура для массива **U**.

Этап 8. На данном этапе происходит проверка возможности развертки любого из регионов. Если после процедуры создания "кольца роста" ни одна из его точек не была развернута, то происходит релаксация параметра $t_c = t_c - 0,1$ [5] и алгоритм переходит снова на этап 4. Данный цикл продолжается, пока параметр $t_c > 0,2$.

Результаты тестирования алгоритма и его программная реализация

Тестирование высокопроизводительного алгоритма развертки интерферометрической фазы методом роста регионов проводилось на реальных данных (Ленинск-Кузнецкий район Кемеровской области). В тестах были использованы снимки аппарата Cosmo-SkyMed уровня L1A, полученные для данного района на даты 11.06.2014 и 19.06.2014.

Выборка получилась многомодальной, либо определить моду не удалось, тогда аналогично, уже для всех элементов объекта контейнера *diffIdx*, обновляем только элементы массивов \mathbf{F} , $\hat{\mathbf{F}}$ и переменную $f_c_counter$. Массивы \mathbf{U} , $\hat{\mathbf{U}}$ и переменная $u_c_counter$ остаются неизменными, так как все общие точки объявляются «неудачными».

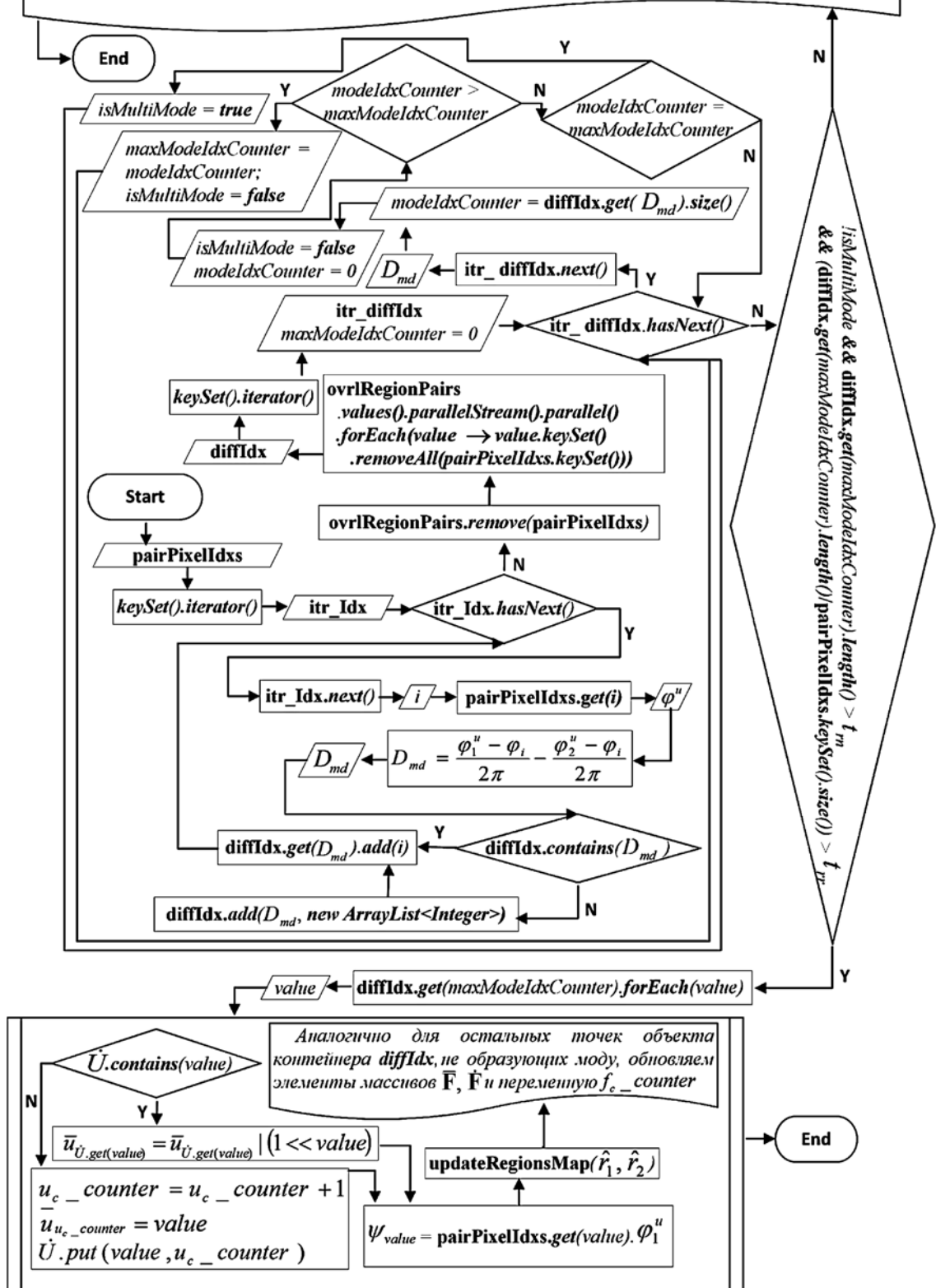


Рис. 9. Блок-схема процедуры объединения двух регионов mergePair

```

mergeRegions(){
    itr_pairs = ovrlRegionPairs.keySet.iterator() //Получаем список пар объединяемых регионов

    // Для каждой пары из списка
    while(itr_pairs.hasNext()){ // Пока не достигнут конец списка
        pairKey = itr_pairs.next() // Получаем ключ пары типа String, ("r1_r2")
        pairPixelIdxs = ovrlRegionPairs.get(pairKey) //Получаем объект типа HashMap,
                                                    содержащий индексы общих точек и значения
                                                    их развернутых фаз в регионах с номерами r1 и r2,
                                                    соответственно

        if(pairPixelIdxs.size() > t_m) { // Если число общих точек больше чем t_m
            mergePair();
        }
        itr_pairs = ovrlRegionPairs.keySet.iterator() //Получаем обновленный список пар объединяемых
                                                    регионов
    }
}

```

Рис. 10. Фрагмент псевдокода процедуры объединения всех пар регионов, находящихся в объекте-контейнере ovrlRegionPairs

Посредством программного комплекса Exelis ENVI 5.1 проведено совмещение снимков и формирование интерферограммы, компенсация фазового набега с помощью опорной цифровой модели рельефа, выбрана область размером 16 384×16 384 пиксела, проведена фильтрация методом Гольдштейна [18], получены данные когерентности. Развертка интерферометрической фазы осуществлялась двумя алгоритмами: базовым, представленном в программном комплексе Exelis ENVI 5.1, и разработанным авторами на базе технологии NVIDIA CUDA и многопоточного подхода Java-multithreading.

Тест проводился на четырех разных архитектурах. Их конфигурации представлены в табл. 2.

На рис. 12 приведены результаты сравнительного теста производительности (времени исполнения) двух алгоритмов.

Сравнительные тесты двух алгоритмов на различных архитектурах показали, что разработанный алгоритм довольно интенсивно использует вычислительные ресурсы графического адаптера. Число вычислительных ядер GPU увеличивается от архи-

тектуры 1 до 3 включительно, что уменьшает время работы алгоритма. В то время как базовый алгоритм практически не изменяет свое быстродействие. Архитектура 4 показывает, что на производительность алгоритма, использующего технологию CUDA, также оказывает влияние шина передачи данных (Архитектура 3, PCI Express 3.0 против Архитектура 4, PCI Express 2.0).

Результаты тестирования сравнивались с различными методами развертки интерферометрической фазы [8, 9, 12]. Особенное внимание уделялось попыткам оптимизации уже существующих методов. Хотелось бы отметить, что многие представленные алгоритмы показали не сопоставимое время работы даже с базовыми алгоритмами программного комплекса Exelis ENVI на снимках малого размера. Так, в упомянутых во введении работах [8, 9, 12] описанные в них методы показали на тестовых примерах время выполнения 6 мин, 8 мин и 3 с соответственно для снимков размером 5000×5000, 5167×3400 и 640×480 точек, что является не лучшим показателем производительности. Так, например, даже базовый

```

// Запускается на стороне CPU
if (u_c_counter ≤ 256) { blockDim.x = 256, gridDim.x = u_c_counter }
else { blockDim.x = 256; gridDim.x = (int)  $\frac{u_c\_counter}{blockDim.x} + u\_c\_counter \% blockDim.x$  }

// Запускается на стороне GPU
cIdx = U.keySet().toArray() // Получаем массив индексов вновь развернутых точек
updateU <<<blockDim.x, gridDim.x>>> (U, cIdx) {
    c = blockDim.x*blockDim.x + threadIdx.x
    u_cIdx_c = u_cIdx_c;
}

```

Рис. 11. Фрагмент псевдокода процедуры обновления массивов U, F на стороне GPU

Аппаратное и программное обеспечение, используемое в тестах алгоритмов

Архитектура	Конфигурация	
	Аппаратное обеспечение	Программное обеспечение
Архитектура 1	Intel Core i7-3770k (4.6GHz Quad core 8 Thread), SSD 256 Gb (Read-Write 545 Мбайт/с-535 Мбайт/с), 8Gb RAM (1600 Mhz), Nvidia GTX Titan X (Clock speed 1075 MHz, 12 Gb RAM, 3072 Cuda cores) (Compute capability version 5.2)	Microsoft Windows 8.1, CUDA Toolkit version 7.5
Архитектура 2	Intel Core i7-3770k (4.6GHz Quad core 8 Thread), SSD 256 Gb (Read-Write 545 Мбайт/с-535 Мбайт/с), 8Gb RAM (1600 Mhz), Nvidia GTX 780 Ti (Clock speed 1075 MHz (boost), 3 Gb RAM, 2880 Cuda cores) (Compute capability version 3.5)	
Архитектура 3	Intel Core i7 5700HQ (3.5GHz Quad core, 8 Thread), SSD 256 Gb (Read-Write 545 Мбайт/с-535 Мбайт/с), 8Gb RAM (1600 Mhz), Nvidia GTX 980M (Clock speed 1075 MHz (boost), 4Gb RAM, 1536 Cuda cores) (Compute capability version 5.2)	
Архитектура 4	AMD Phenom II (3.4 GHz Quad core 4 Thread), SSD 256 Gb (SATA II) (Read-Write 78 Мбайт/с-65 Мбайт/с), 8Gb RAM (600 Mhz), Nvidia GTX 780 Ti (Clock speed 875 MHz, 3 Gb RAM, 2880 Cuda cores, PCI Express 2.0) (Compute capability version 3.5)	

алгоритм роста регионов или минимальной стоимости потока для снимка размером 4096×4096 точек показывает время 28 с в программе комплекса Exelis ENVI.

Нелинейность возрастания показателя времени работы алгоритмов развертки фазы накладывает определенные трудности. К примеру, в программном комплексе Exelis ENVI время работы алгоритма роста регионов для изображения размером 4096×4096 точек составило 28 с, в то время как для снимка $16\,384 \times 16\,384$ точек это время составило уже 36 мин.

Однако использование технологии параллельных вычислений для функциональных частей алгоритмов развертки фазы позволяет значительно снизить общее время исполнения. Например, для изображения

размером 8192×8192 точек общее время алгоритма поиска соседних точек на базе технологии CUDA составило в среднем 62 с, в то время как базовый алгоритм программного комплекса Exelis ENVI показал результат 306 с.

Заключение

Разработан высокопроизводительный алгоритм роста регионов для развертки фазы интерферометрических данных (*High Performance Region-Growing Algorithm*).

По сравнению с базовым алгоритмом развертки фазы, применяемым в программном комплексе Exelis ENVI, были проведены следующие усовершенствования функциональной части.

- Разработан эффективный алгоритм упаковки/распаковки значений индексов координат точек интерферометрического изображения и вспомогательных массивов данных на базе битовых полей.

- Создан высокопроизводительный алгоритм поиска соседних точек (*unwrapping rings*) уже развернутых пикселей с применением технологии NVIDIA CUDA.

- Разработан многопоточный алгоритм обновления карты номеров регионов (*region index map*) в процедуре их объединения.

Программная реализация алгоритма предусматривает возможность ее запуска на многопроцессорных платформах в мультипоточном режиме с использованием графических устройств, поддерживающих технологию NVIDIA CUDA.

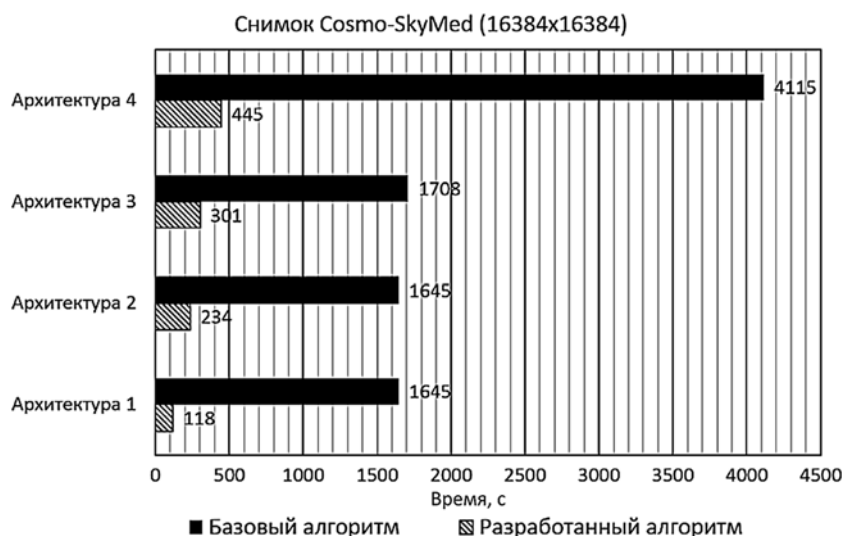


Рис. 12. Результаты сравнительного теста производительности разработанного алгоритма развертки фазы и базового алгоритма Exelis ENVI

Разработанный алгоритм прошел апробацию на реальных интерферометрических данных и показал улучшенное до 5 раз время работы по сравнению с базовым алгоритмом программного комплекса Exelis ENVI.

Список литературы

1. **Елизветин И. В., Шувалов Р. И., Буш В. А.** Принципы и методы радиолокационной съемки для целей формирования цифровой модели местности // Геодезия и картография. 2009. № 1. С. 39–45.
2. **Rosen P., Hensley S., Joughin I., et al.** Synthetic aperture radar interferometry // Proceedings of the IEEE. 2000. Vol. 88, N. 3. P. 333–382.
3. **Chen C. W.** Statistical-cost network-flow approaches to two-dimensional phase unwrapping for radar interferometry: PhD thesis, Stanford University, 2001.
4. **Martinez-Espla J. J., Martinez-Marin T., Lopez-Sanchez J. M.** Using a grid-based filter to solve InSAR phase unwrapping // IEEE Geosci. Remote Sens. Lett. 2008. Vol. 5, N. 2. P. 147–151.
5. **Xu W., Cumming I.** A region-growing algorithm for InSAR phase unwrapping // IEEE Trans. Geosci. Remote Sens. 1999. Vol. 37, N. 1. P. 124–134.
6. **Costantini M.** A method for phase unwrapping in the presence of dense fringe patterns // Proceedings of ESA Fringe' 99 Meeting. November 10–12, Liege, Belgium (CD-ROM, ESA Special Publication SP-478). ESA, Noordwijk, The Netherlands.
7. **Huadong Haol, Guolin Liu, Xianlei Chenl, Zhentan Cao.** InSAR Kalman Filter Phase Unwrapping Algorithm Based on SRTM DEM // Journal of Earth Science and Engineering. 2012. Vol. 2. P. 247–252.
8. **Haifeng Huang.** A fast multi-baseline and multi-frequency band phase-unwrapping algorithm // Measurement. 2014. Vol. 49. P. 401–406.
9. **Mistry P., Braganza Sh., Kaeli D., Leeser M.** Accelerating phase unwrapping and affine transformations for optical quadrature microscopy using CUDA // Proceedings of 2nd Workshop on General Purpose Processing on Graphics Processing Units, GPGPU 2009. Washington, DC, USA, March 8, 2009. New York, NY, USA, ACM, GPGPU-2. 2009. P. 28–37.
10. **Bamler R., Hartl P.** Synthetic aperture radar interferometry // Inverse Problems. 1998. Vol. 14, N. 4. P. R1–R54.
11. **Soni N. K.** Phase Unwrapping Algorithm Using Edge Detection and Statistical Cost Functions. Master thesis. Civil Engineering and Geosciences, 2012. 74 p.
12. **Jiang Ting-Chen.** Ameliorative Minimum Cost Flow Algorithm for Phase Unwrapping // Procedia Environmental Sciences. 2011. Vol. 10. P. 2560–2566.
13. **Martinez-Espla J. J., Martinez-Marin T., Lopez-Sanchez J. M.** An Optimized Algorithm for InSAR Phase Unwrapping Based on Particle Filtering, Matrix Pencil, and Region-Growing Techniques // IEEE Geoscience and Remote Sensing Letters. 2009. Vol. 6, N. 4. P. 835–839.
14. **Ferretti A., Monti-Guarnieri A., Prati C.** et al. InSAR Principles: Guidelines for SAR Interferometry Processing and Interpretation. ESA Publications. (2007). TM-19.
15. **Cook S.** CUDA Programming: A Developer's Guide to Parallel Computing with GPUs (1st ed.). Morgan Kaufmann Publishers Inc. 2013. P. 334–335.
16. **CUDA C Programming Guide.** NVIDIA developer zone. URL: <http://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html#bitwise-functions> (дата обращения 30.10.2015).
17. **Java Doc.** Java Platform, Standard Edition 7 API Specification. URL: <https://docs.oracle.com/javase/7/docs/api/> (дата обращения 02.11.2015).
18. **Lee J.-S., Papathanassiou K. P., Ainsworth T. L.** et al. A New Technique for Noise Filtering of SAR Interferometric Phase Images // IEEE Transactions on geoscience and remote sensing. 1998. Vol. 36, N. 5. P. 1456–1465.

High-Performance Region-Growing Algorithm for InSAR Phase Unwrapping Based on CUDA

V. P. Potapov, potapov@ict.sbras.ru, **S. E. Popov**, popov@ict.sbras.ru, Institute of Computational Technologies SB RAS, Novosibirsk, 630090, Russian Federation

Corresponding Author:

Semen E. Popov, Chief Scientist, Institute of Computational Technologies SB RAS, Novosibirsk, 630090, Russian Federation
e-mail: popov@ict.sbras.ru

Received on September 28, 2015

Accepted on November 19, 2015

This paper presents a high-performance Region-Growing algorithm for InSAR phase unwrapping. Software implementation of the algorithm provides the ability to run on multiprocessor platforms in multi-threaded mode using the graphics devices that support the technology NVIDIA CUDA.

In order to reduce the volume of the data exchange between the CPU and GPU developed an effective method of packing / unpacking indexes of pixels using bit sets. It was possible to reduce up to 64 times the amount of traffic due to the conversion of the index points to a state of the bit variable of type Long.

Was improved constructing growth ring algorithm. By associating a detailed point of the image with one thread of the multiprocessor graphics unit, it was possible for one GPU computation cycle to process more than 20,000 pixels. To reduce iterations of the loop for each thread in the warp the boundary conditions at the corresponding pixel-candidate for the phase unwrapping were added.

To reduce the operating time of the "merge regions" procedure, which generates the need to update the "ambiguity numbers" of the joining region at the overlapping points and numbers of regions in previously unwrapped, "paged-pinned memory" were used to deploy region numbers map to reduce the data transfer time between CPU and GPU memories.

The developed algorithm has been tested on real interferometric data and showed the improved time (up to 5) compared to the basic algorithm of program complex Exelis ENVI.

Keywords: parallel computing, CUDA, phase unwrapping, region-growing algorithm

For citation:

Potapov V. P., Popov S. E. High-Performance Region-Growing Algorithm for InSAR Phase Unwrapping Based on CUDA, *Programmnaya Inzheneriya*, 2016, vol. 7, no. 2, pp. 61–74.

DOI: 10.17587/prin.7.61-74

References

1. **Elizavetin I. V., Shuvalov R. I., Bush V. A.** Printsipy i metody radiolokatsionnoi s'emki dlya tselei formirovaniya tsifrovoy modeli mestnosti (Principles and methods of radar imagery for forming a digital terrain model), *Geodeziya i kartografiya*, 2009, vol. 1, pp. 39–45 (in Russian).
2. **Rosen P., Hensley S., Joughin I., Li F., Madsen S., Rodriguez E., Goldstein R.** Synthetic aperture radar interferometry, *Proceedings of the IEEE*, 2000, vol. 88, no. 3, pp. 333–382.
3. **Chen C. W.** Statistical-cost network-flow approaches to two-dimensional phase unwrapping for radar interferometry. PhD thesis, Stanford University, 2001.
4. **Martinez-Éspla J. J., Martinez-Marin T., Lopez-Sanchez J. M.** Using a grid-based filter to solve InSAR phase unwrapping, *IEEE Geosci. Remote Sens. Lett.*, 2008, vol. 5, no. 2, pp. 147–151.
5. **Xu W., Cumming I.** A region-growing algorithm for InSAR phase unwrapping. *IEEE Trans. Geosci. Remote Sens.*, 1999, vol. 37, no. 1, pp. 124–134.
6. **Costantini M.** A method for phase unwrapping in the presence of dense fringe patterns. *Proceedings of ESA Fringe'99 Meeting*, November 10–12, Liege, Belgium (CD-ROM, ESA Special Publication SP-478), ESA, Noordwijk, The Netherlands.
7. **Huadong H., Guolin L., Xianlei Ch., Zhentan Cao.** InSAR Kalman Filter Phase Unwrapping Algorithm Based on SRTM DEM. *Journal of Earth Science and Engineering*, 2012, vol. 2, pp. 247–252.
8. **Haifeng H.** A fast multi-baseline and multi-frequency band phase-unwrapping algorithm, *Measurement*, 2014, vol. 49, pp. 401–406.
9. **Perhaad M., Sherman B., David K., Miriam L.** Accelerating phase unwrapping and affine transformations for optical quadrature microscopy using CUDA, *Proceedings of 2nd Workshop on General Purpose Processing on Graphics Processing Units, GPGPU 2009*, Washington, DC, USA, March 8, 2009, ACM, New York, NY, USA, GPGPU-2. 2009, pp. 28–37.
10. **Bamler R., Hartl P.** Synthetic aperture radar interferometry. *Inverse Problems*, 1998, vol. 14, no. 4, pp. R1–R54.
11. **Soni N.** Phase Unwrapping Algorithm Using Edge Detection and Statistical Cost Functions. Master thesis. Civil Engineering and Geosciences, 2012, 74 p.
12. **Jiang Ting-Chen.** Ameliorative Minimum Cost Flow Algorithm for Phase Unwrapping. *Procedia Environmental Sciences*, 2011, vol. 10, pp. 2560–2566.
13. **Juan J. M., Tomas M., and Lopez-Sanchez J.** An Optimized Algorithm for InSAR Phase Unwrapping Based on Particle Filtering, Matrix Pencil, and Region-Growing Techniques, *IEEE Geoscience and Remote Sensing Letters*, 2009, vol. 6, no. 4, pp. 835–839.
14. **Ferretti, A., Monti-Guarnieri A., Prati C., Rocca F., Massonnet D.** InSAR Principles: Guidelines for SAR Interferometry Processing and Interpretation. ESA Publications, 2007, TM-19.
15. **Cook S.** *CUDA Programming: A Developer's Guide to Parallel Computing with GPUs* (1st ed.). Morgan Kaufmann Publishers Inc. 2013, pp. 334–335.
16. **CUDA C Programming Guide.** NVIDIA DEVELOPER ZONE, available at: <http://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html#bitwise-functions>
17. **Java Doc.** Java™ Platform, Standard Edition 7 API Specification, available at: <https://docs.oracle.com/javase/7/docs/api/>
18. **Jong-Sen L., Papathanassiou P., Ainsworth L., Grunes R., Reigber R.** A New Technique for Noise Filtering of SAR Interferometric Phase Images, *IEEE Transactions on Geoscience and Remote Sensing*, 1998, vol. 36, no. 5, pp. 1456–1465.

ИНФОРМАЦИЯ

28 марта — 1 апреля 2016 г. в Северном (Арктическом) федеральном университете имени М. В. Ломоносова в г. Архангельске будет проходить международная научная конференция "Параллельные вычислительные технологии (ПаВТ) 2016" — десятая в серии ежегодных конференций, посвященных развитию и применению параллельных вычислительных технологий в различных областях науки и техники.

В первый день работы конференции будет объявлена 24-я редакция списка Top50 самых мощных компьютеров СНГ.

Во все дни конференции будет работать суперкомпьютерная выставка, на которой ведущие производители аппаратного и программного обеспечения представят свои новейшие разработки в области высокопроизводительных вычислений.

Сайт конференции: <http://agora.guru.ru/pavt2016/>

В. А. Васенин, д-р физ.-мат. наук, проф., e-mail: vassenin@msu.ru,
НИИ Механики МГУ им. М. В. Ломоносова, г. Москва,

М. В. Пирогов, канд. физ.-мат. наук, e-mail: pmv_mvpr@mail.ru,
НПО им. С. А. Лавочкина, Московская область, г. Химки,

А. В. Чечкин, д-р физ.-мат. наук, проф., e-mail: A.Chechkin@mail.ru

Военная академия РВСН имени Петра Великого, Московская область, г. Балашиха

Основной оператор радикального моделирования на демонстрационном примере

Обсуждены вопросы сопровождения сложных целенаправленных систем (ЦС), включая вопросы, связанные с их автоматизацией на основе применения программно-технических средств. Отмечено наличие в этой области важных с точки зрения практики, но еще не решенных задач. Для решения проблемы информационно-системной безопасности, которая рассматривается в качестве ключевой для ЦС, предложен подход, использующий методы и средства радикального моделирования и радикального программирования. Введено и на демонстрационном примере проиллюстрировано понятие основного оператора радикального моделирования, обеспечивающего реализацию целевых схем радикалов — решений задач жизненного цикла ЦС. Представлены соображения по созданию начального атласа схем радикалов ЦС, рассмотрен подход к созданию математического обеспечения процессов радикального моделирования, а также обоснована необходимость специализированного редактора схем радикалов.

Ключевые слова: системный подход, сложная система, целенаправленная система, критическая система, информационно-системная безопасность, интеллектуальное управление, среда радикалов, схема радикалов, универсальный язык информатики, язык RADICAL, язык диаграмм, программно-технические средства, программная инженерия, радикальное моделирование, радикальное программирование, радикальная модель, основной оператор радикального моделирования

Введение

Будем рассматривать сложные целенаправленные системы (ЦС), т. е. системы, для которых сформулирована цель (цели), определено их назначение. К таким ЦС относятся большинство технических систем в различных сферах общественной жизни. Положительные стороны ЦС очевидны, однако наряду с ними далеко не единичны случаи отказов ЦС, как следствие — материальные потери, экологические катастрофы, опасности для здоровья и жизни людей. Таких случаев, к сожалению, много даже только по материалам средств массовой информации. Положение дел еще более осложняется для критически важных ЦС, для которых характерны жесткие требования к их безопасности и безотказности функционирования [1].

В современных ЦС определяющую роль играют подсистемы автоматизации различных технологических процессов, программно-технические средства (ПТС), поддерживающие эти процессы, направленные для решения целевых, сенсорных и сертификационных задач. Значение ПТС для ЦС постоянно возрастает. Таким образом, ПТС становятся во все большей степени

"ответственными" за условия и результаты функционирования ЦС в целом. При этом следует отметить, что и сами ПТС, и подсистемы автоматизации также можно (и нужно) рассматривать как целенаправленные подсистемы ЦС. Программно-технические средства быстро развиваются, однако для них характерны и определенные проблемы, и целый ряд пока еще нерешенных задач [2]. К числу таких относятся, например, задачи обеспечения надежной верификации и безотказного функционирования ПТС. Несмотря на несомненные успехи использования большинства ЦС и, в частности, разнообразных ПТС, более детальный анализ позволяет сделать вывод о наличии системного неблагополучия. Современные ЦС зачастую конфликтны по отношению к среде окружения и даже опасны, в том числе для людей. Пригодных к широкому практическому использованию средств, кардинально улучшающих положение дел в ЦС и ПТС, на настоящее время нет. Как следствие, исследования на этом направлении представляются актуальными.

Ключевая проблема ЦС — это обеспечение их *информационно-системной безопасности* (ИСБ). Понятие ИСБ было введено А. В. Чечкиным в работе [3].

К числу главных факторов на этом направлении можно отнести следующие:

- должна быть создана избыточная — *радикальная* (более точное определение далее) модель (PM) широкой проблемной области ЦС, охватывающая все наиболее существенные факторы влияния на решения задач жизненного цикла ЦС;
- должна быть обеспечена защита аккумулированной и уже циркулирующей в системе информации, выделение на ее основе дополнительной информации в целях решения очередных штатных задач жизненного цикла ЦС;
- должны быть обеспечены эффективные попытки решения нештатных задач жизненного цикла ЦС и на этой основе самообучение и развитие ЦС;
- должны быть обеспечены мониторинг, учет и устранение негативных последствий решения каждой задачи в проблемной области ЦС на всех этапах жизненного цикла ЦС.

Для обеспечения ИСБ требуется интеллектуальное управление ЦС. Это означает, что система, как уже отмечалось ранее, должна не только обеспечивать решение штатных задач, но и должна обладать механизмами решения нештатных задач, обладать способностью к самообучению и развитию. Одним из подходов, ориентированных на такое управление ЦС, является радикальное моделирование и радикальное программирование [4, 5]. Радикальное моделирование и радикальное программирование основаны на концепции избыточной системы в форме среды радикалов А. В. Чечкина [6—8] и на формализме схем радикалов — языке RADICAL [3, 9].

Для широкой проблемной области сложной ЦС (понятие будет уточнено далее), представленной в радикальной модели ЦС, на всем ее жизненном цикле характерна изменчивость, появление новых требований к системе, ее перманентной модификации, которую необходимо учитывать. На ранних этапах создания ЦС многое объективно остается неизвестным. Здесь (да и на других этапах жизненного цикла ЦС) большое значение имеют опыт специалистов, умение оценить ситуацию, предвидеть будущие потребности и задачи. В расчете на них следует разработать перспективные математические модели, алгоритмы и реализующие их ПТС ЦС. Необходимо отметить, что процесс разработки ЦС, как правило, имеет сложный, итерационный характер. К сожалению, современные инструментальные средства в необходимом объеме его *не поддерживают*. Уточним это утверждение.

В настоящее время для ЦС отсутствуют единые образные и полные формальные описания, которые реализованы программно и пригодны для решения большинства реальных практических задач. Отсутствуют инструментальные средства, предоставляющие разработчикам и другим специалистам возможность в удобном для них виде и оперативно описывать, представлять все значимые составляющие широкой проблемной области, все их важные свойства и связи, все варианты таких составляющих на различных этапах жизненного цикла ЦС. Нет

средств обеспечения всесторонней, полной оценки и сравнения рассматриваемых вариантов, ситуаций, которые могут складываться в проблемной области. Основная нагрузка здесь ложится на специалистов по конкретной ЦС, работа которых, несмотря на прогресс в области инструментальных средств, должным образом этими средствами пока не охватывается. Разработанное программное обеспечение, созданное на традиционных подходах, требует многочисленных модификаций на всех этапах жизненного цикла ЦС. Безусловно, в широкой проблемной области любой ЦС всегда будут иметь место модификации, которые можно назвать объективно необходимыми, неизбежными. Причина в том, что далеко не все известно на ранних стадиях разработки, а многое "проявляется" только в ходе предпроектных исследований, проектирования, проведения испытаний и далее в ходе эксплуатации системы. В результате появляются нештатные вопросы, которые необходимо уметь разрешать, как правило, оперативно.

Однако наряду с причинами, которые приводят к модификациям ПТС, имеют место и другие. Они заключаются в несовершенстве современных средств разработки ПТС, методов решения задач автоматизации технологических процессов системы. Кроме того, у современных инструментальных средств отсутствуют возможности, обеспечивающие в полном объеме анализ корректности модификаций. Практически вся основная нагрузка здесь снова приходится на специалистов. Однако даже высококвалифицированный специалист, обладающий большим опытом решения задач в конкретной проблемной области, не застрахован от ошибок. Для успешной работы ему необходим соответствующий программный инструментарий, который мог бы поддерживать как можно более широкий круг задач всей проблемной области ЦС. Такой инструментарий в настоящее время отсутствует.

Предлагаемый в статье подход на основе радикального моделирования и программирования [4, 5] является одним из способов устранения отмеченных выше недостатков.

Радикальная модель широкой проблемной области целенаправленной системы и основной оператор радикального моделирования

Для более точного и полного понимания следующего далее материала представим краткие сведения о концепции среды радикалов и языке схем радикалов RADICAL. Основное понятие в этой концепции — понятие радикала. Радикал — это любая ЦС, которая может пребывать в двух видах состояний — активном и пассивном. Активное состояние — радикал "включен", используется по своему назначению. Пассивное состояние — радикал "выключен", находится в резерве.

Радикальная модель ЦС — это избыточная (более широкая по сравнению с традиционно декларируемой требованиями к системе) модель проблемной области в форме среды радикалов. Среда радикалов —

это принципиально избыточная система, состоящая из радикалов. Избыточность связана с включением в проблемную область ЦС не только составляющих с их свойствами, связями самой ЦС и ее окружения, но и с учетом штатных задач, обусловленных ее назначением (целевых, сенсорных, сертификационных) вместе с методами и ПТС их реализации. Избыточность нацелена на случай появления новых требований к системе, необходимости адекватной реакции на разнообразные штатные ситуации и возникновения нештатных ситуаций.

Конструкции языка RADICAL вводятся следующим образом. Фиксируется алфавит. Из символов алфавита строятся имена радикалов. Из построенных имен радикалов и символа стрелки строятся цепочки радикалов. Например, $NameA \rightarrow azb \rightarrow NameB$; цепочка радикалов (здесь ";" — завершающий символ цепочки). Такие цепочки реализуют информационные пути поиска нужного радикала в среде радикалов.

Схема радикалов определяется как множество некоторых цепочек радикалов и соответствует локальной семантической сети. Среда радикалов описывается полной схемой радикалов. Вводятся стандартные радикалы — уникамы и контейнеры. Уникамы соответствуют составляющим широкой проблемной области. Они представляют собой совокупность термов, предметных переменных логики предикатов. В именах уникамов используют символ "u". Контейнеры соответствуют свойствам и связям составляющих проблемной области системы. Таким образом, контейнеры соответствуют символам логики предикатов. В именах контейнеров используют символ "c". Служебные радикалы $d[1], d[2], \dots$ именуют направления в схемах радикалов. Пример символического представления схемы радикалов:

$c_db_tb \rightarrow \{d[1] \rightarrow uDB1; d[2] \rightarrow uTB11; d[3] \rightarrow uTB12;\}$
 $c_exe_source \rightarrow \{d[1] \rightarrow uA; d[2] \rightarrow uTB11;\}$
 $c_exe_receiver \rightarrow \{d[1] \rightarrow uA; d[2] \rightarrow uFILE1;\}$

Здесь в контейнере c_db_tb доступны уникамы — база данных $DB1$ и ее таблицы $TB11$ и $TB12$. В контейнер c_exe_source вложен запускаемый модуль A и таблица $TB11$ — источник данных для этого модуля. Контейнер $c_exe_receiver$ содержит запускаемый модуль A и файл-приемник данных $FILE1$, заполняемый работающим модулем A .

В языке RADICAL разработана алгебра схем радикалов с операциями объединения, пересечения, удаления, разметки и др.

Имеются различные представления (визуализации) схем радикалов. На рис. 1 изображено *геометрическое (векторное)* представление только что рассмотренной схемы радикалов.

На плоскости введена прямоугольная система координат. Ее горизонтальная ось H направлена вправо. Вертикальная ось V направлена вниз. Горизонтали соответствуют используемым типам контейнеров и уникаму. Вертикали — экземплярам контейнеров. На рис. 1 цифры 1, 2 и 3 — индексы служебных радикалов направлений $d[1], d[2], d[3]$.

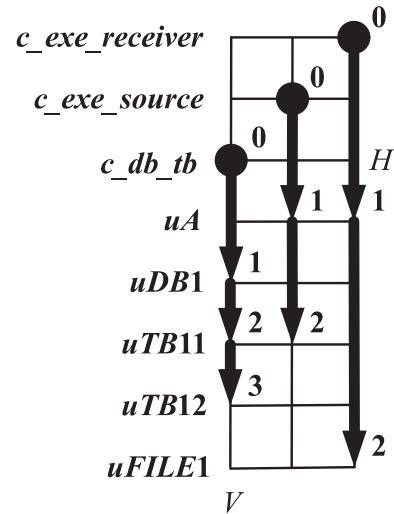


Рис. 1. Геометрическое представление схем радикалов

Язык RADICAL является языком диаграмм. На рис. 2 приведено *табличное, диаграммное* представление схемы радикалов рис. 1. Здесь поля A, D, E, F — целочисленные, B и C — строковые, первичный ключ составляет поле A .

В рассмотренных геометрическом и табличном представлениях схем радикалов можно удалять вертикали и горизонтали, склеивая оставшиеся фрагменты, что соответствует операциям алгебры схем радикалов. В рамках алгебры схем радикалов формируются выделенные части (сечения) среды радикалов (области обозрения среды радикалов).

Приведенные выше два способа представления среды радикалов — не единственные. Выразительность этих представлений схем радикалов может быть повышена путем использования отступов (по горизонтали вправо), цвета и движения. В представлениях могут быть добавлены пиктограммы, изображения, видеоматериалы. Изображения можно масштабировать, видеоматериалы — активировать (проигрывать). В автоматизированном режиме представления схемы радикалов можно преобразовать в диаграммы любого вида. Возможен быстрый пере-

A	B	C	D	E	F
-3	c	$c_exe_receiver$			0
-2	c	c_exe_source		0	
-1	c	c_db_tb	0		
1	u	uA		1	1
2	u	$uDB1$	1		
3	u	$uTB11$	2	2	
4	u	$uTB12$	3		
5	u	$uFILE1$			2

Рис. 2. Табличное представление схем радикалов

ход от одного представления к другому. Представления схем радикалов — основа пользовательского интерфейса среды радикалов. Рассмотренные представления демонстрируют возможность применения для схем радикалов многочисленных, разнообразных выразительных средств теории графов, диаграмм, таблиц, а также развитого инструментария баз данных.

С учетом изложенного выше можно констатировать, что среда радикалов РМ ЦС имеет многоуровневую структуру и должна быть программно реализована с учетом пользовательского интерфейса [4]. Так, организованная РМ широкой проблемной области ЦС является инструментальным средством обеспечения комплексного развития ЦС, в том числе ПТС ЦС любого назначения. Многоуровневая структура РМ — это ядро (начальное) и уровни модификаций РМ. Уровням РМ принадлежат стандартные радикалы (уникумы и контейнеры, рассмотренные выше), библиотеки радикалов, ультраконтейнеры, соответствующие правилам-продукциям и регулирующие требования (ограничения) к радикалам.

При решении задач на различных этапах жизненного цикла ЦС новые уникамы каждый раз реализуются на основе многоуровневых (в общем случае) структур, состоящих из библиотечных радикалов. Синтез новых уникамов осуществляется с помощью двунаправленного метода (ДН-метода), предназначенного для решения как штатных, так и нештатных задач [5]. Для всех уровней развития РМ ЦС имеется базовая процедура, которую назовем *основным оператором* радикального моделирования. Этот оператор является основным в процессе моделирования действий по инженерному сопровождению программ (программной инженерии) применительно к ЦС. Рассматриваемый далее на демонстрационном примере основной оператор представляет собой *самоорганизующуюся процедуру активации* РМ ЦС.

Начальный вид основного оператора на демонстрационном примере

Приведем предварительную формулировку для основного оператора радикального моделирования, которая далее будет уточнена. Пусть для проблемной области некоторой ЦС построена РМ. Выделим в РМ следующие схемы радикалов: *исходную, целевую, допустимые (промежуточные), библиотечные, ресурсные, схемы ДН-метода* синтеза уникама.

Требуется обеспечить достижение целевой схемы радикалов из исходной схемы с помощью схем ДН-метода синтеза уникама, используя имеющиеся библиотечные схемы радикалов, а также схемы, представляющие доступные ресурсы (например, временные ресурсы). При этом все промежуточные состояния исходной схемы радикалов "на пути" к целевой схеме должны принадлежать множеству допустимых состояний, т. е. не быть конфликтными. Важно, что достижение целевой схемы радикалов необходимо обеспечить при условии, что возможны появления новых схем, отражающих нештатные ситуации в проблемной области ЦС.

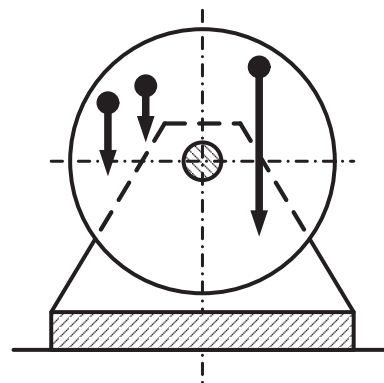


Рис. 3. Целенаправленная система демонстрационного примера

В терминах табличного представления схем радикалов основной оператор радикального моделирования — это оператор достижения целевой конфигурации ячеек таблицы из исходной конфигурации. В терминах геометрического представления схем — это оператор достижения целевой конфигурации векторов из исходной.

Теперь рассмотрим демонстрационный пример с расширением вида основного оператора, действующего в РМ. Рассмотрим ЦС, которая состоит из вертикально расположенного диска с осью вращения и опорой. Конструкция находится в поле силы тяжести Земли. Опора располагается на горизонтальной поверхности (рис. 3).

Возможны внешние воздействия на диск — штатные и нештатные, вследствие которых возникают вертикально направленные силы и соответствующие вращающие моменты. Целевая задача заключается в обеспечении равновесия диска на некоторых заранее задаваемых промежутках времени. Пусть библиотека стандартных радикалов содержит уникамы, контейнеры и ультраконтейнеры, представляющие необходимые для данной проблемной области знания по механике твердого тела. Пусть среди этих знаний имеется и условие равновесия твердого тела, имеющего ось вращения, — равенство нулю суммарного вращающего момента от действия всех приложенных к этому телу сил.

Для решения целевой задачи допускается использовать заранее заготовленные грузы-противовесы, представленные штатными радикалами из библиотеки стандартных радикалов. Пусть грузы-противовесы могут крепиться только в специальном образом конструктивно оформленных для этого местах на поверхности диска.

Пусть в процессе функционирования ЦС некоторое подмножество таких мест стало недоступно для целевого использования. Таким образом, часть внешних воздействий, которые ранее были бы классифицированы как штатные (противодействие им было подготовлено заранее), теперь уже нельзя считать таковыми. Теперь это будут нештатные воздействия в рамках демонстрационного примера. Заметим, что некоторые из них могут вновь стать штатными в результате применения ДН-метода синтеза уникама и самообучения.

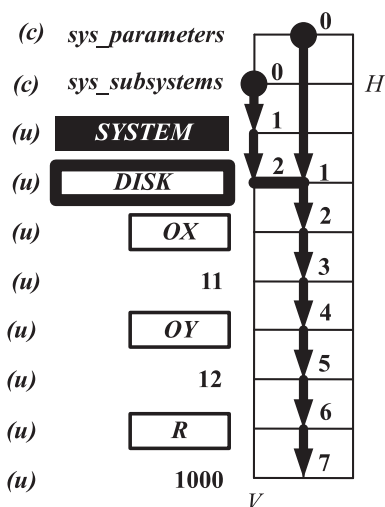


Рис. 4. Геометрическое представление РМ ЦС

Предположим, что для рассматриваемой ЦС создана радикальная модель в форме среды радикалов — РМ ЦС. На рис. 4 представлена схема среды радикалов, в которую попадает часть радикалов рассматриваемой РМ ЦС.

Видно, что в состав уникама ЦС *SYSTEM* входит уникам-подсистема *DISK* (используется контейнер *sys_subsystems*). Некоторые параметры диска и их значения вложены в контейнер *sys_parameters*. Это координаты центра диска — уникамы *OX* и *OY* (их значения 11 и 12) и радиус диска (его значение 1000). Здесь единицы измерения в целях упрощения схемы не показаны.

На рис. 5, 6 в одной из табличных форм представления схем радикалов приведены два сечения РМ ЦС.

На рис. 5 представлена таблица *TB_CASES* прогнозируемых ситуаций с равновесием диска. Поля таблицы *TB_CASES* следующие: *ID_CASE* — идентификатор ситуации (ключевое поле); *TIME* — прогнозируемое время наступления ситуации, с; *STATE* — прогнозируемое состояние диска (будет достигнут баланс с применением заготовленных грузов-противовесов или нет).

<i>ID_CASE</i>	<i>TIME</i>	<i>STATE</i>
1	10	<i>Balance</i>
2	100	<i>Balance</i>

Рис. 5. Таблица *TB_CASES*

<i>ID_CASE_LOADS</i>	<i>ID_LOAD</i>	<i>BALANCE_DIRECTION</i>	<i>M</i>
1	1	<i>FromBalance</i>	50
1	2	<i>ToBalance</i>	-20
1	3	<i>ToBalance</i>	-30
2	1	<i>FromBalance</i>	10
2	2	<i>ToBalance</i>	-10

Рис. 6. Таблица *TB_LOADS*

На рис. 6 представлена таблица *TB_LOADS* — таблица нагрузок — внешних и создаваемых грузами-противовесами. Поля таблицы *TB_LOADS* следующие: *ID_CASE_LOADS* — идентификатор ситуации (ключевое поле, значения — из *TB_CASES*); *ID_LOAD* — идентификатор нагрузки (ключевое поле); *BALANCE_DIRECTION* — "направление" нагрузки по отношению к целевому состоянию (для внешней нагрузки на диск значение этого поля всегда *FromBalance*, для груза-противовеса — всегда *ToBalance*); *M* — значение вращающего момента, создаваемого нагрузкой, учитывает направления вращения. Таблицы *TB_CASES* и *TB_LOADS* связаны между собой. На рис. 7 показано третье сечение РМ ЦС, в котором отражены структуры (состав полей) таблиц *TB_CASES* и *TB_LOADS* и связи между таблицами (полями таблиц).

На рис. 7 непосредственно под уникамами-таблицами расположены их поля. Некоторые контейнеры, например, контейнеры *tb_fields*, объединяющие таблицу и ее поля, в целях упрощения не показаны. Таблицы связаны между собой связью "главная—подчиненная", что фиксируется контейнером *tb_master_detail*. Контейнер *fld_master_detail* уточняет, связываются ли поля *ID_CASE* (*TB_CASES*) и *ID_CASE_LOADS* (*TB_LOADS*). Таким образом, как это было отмечено выше, источник значений для ячеек *ID_CASE_LOADS* (*TB_LOADS*) — ячейки *ID_CASE* (*TB_CASES*).

Усложним задачу (что, как правило, и происходит на практике). Пусть теперь необходимо учитывать не только механические, но и тепловые явления. В этом случае в избыточной РМ активируется новая расширенная координатная система контейнеров (КСК), в ней появится подсистема КСК тепловых явлений. Пусть некоторые из грузов-противовесов создают не только вращающий момент, но и температурное (*T*) поле (рис. 8).

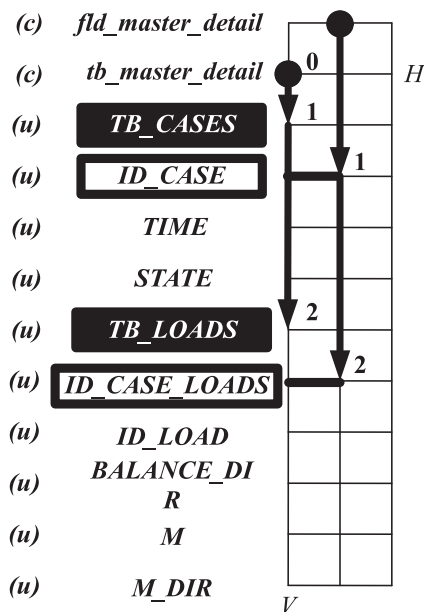


Рис. 7. Геометрическое представление связей таблиц *TB_CASES* и *TB_LOADS*

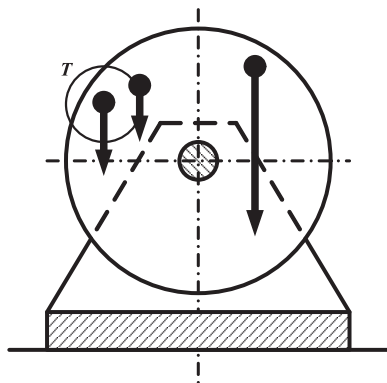


Рис. 8. Усложнение ситуации в проблемной области ЦС

Создаваемая температура при некотором воздействии температурного поля может быть недопустимой для некоторых других грузов-противовесов. Этот факт означает, что решение, обеспечивающее равновесие диска, может "проходить по механике", но "не проходить по теплофизике". Требуется учесть, что механические явления, вообще говоря, зависят от тепловых явлений. В силу этой зависимости механизмы могут заклинивать, неправильно спроектированные мосты могут разрушаться. Вследствие связей между подсистемами КСК возможны вторичные конфликты. Таким образом, в стремлении сохранить одно, можно разрушить другое. Здесь необходимы компромиссы, уступки, согласованные действия. Необходимо учесть последствия и устранить конфликты, используя связи между подсистемами КСК.

Для устранения в том числе и вторичных конфликтов необходима перманентная сертификация многоуровневой РМ ЦС на всех ее уровнях. Для штатных библиотечных радикалов (для соответствующих схем радикалов) должен быть обеспечен полный цикл работы с ними, а именно: хранение; сертификация (определение пригодности для использования при решении различных задач); поиск; подготовка к использованию в РМ ЦС — транспортировка, настройка, монтаж (встраивание); демонтаж. Неправильное использование (сборка) схем радикалов должно быть исключено самой их конструкцией. Не подлежащие (по условиям) сборке схемы должны быть деактивированы (исключены).

Должна быть обеспечена контролируемость собираемых (испытываемых, эксплуатируемых) схем радикалов (представляющих реальные составляющие проблемной области). Затраты, связанные с использованием библиотечных радикалов (соответствующих составляющих), могут быть различными. Отсюда — необходимость в оптимизации использования стандартных радикалов.

Теперь предположим, что ЦС должна решать задачи по обработке (например, съемке) внешних объектов. По этой причине еще усложним наш пример. Пусть на горизонтальной плоскости, в месте расположения нашей конструкции (а также на горизонтальной плоскости опоры, находящейся непосред-

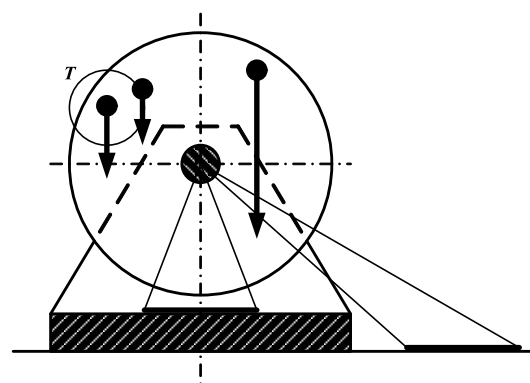


Рис. 9. Добавление объектов обработки в проблемную область ЦС

ственно под диском), время от времени появляются объекты, которые необходимо обработать заранее оговоренным способом (рис. 9).

Появление одних объектов обработки предсказуемо. Соответственно, задачи по их обработке штатные. Появление других — неожиданность, и задачи их обработки нештатные. Для появляющихся новых объектов обработки, как и для штатных радикалов, должен быть обеспечен полный цикл работы с ними в РМ. Такой цикл включает перечисленные далее действия: описание объектов обработки и средств их обработки в схемах радикалов; выбор объектов, для которых обработка допустима (необходима, желательна); определение возможности обработки одних объектов в зависимости от обработки других объектов; подготовка ЦС к обработке объектов (ЦС может работать в различных режимах, одни совместимы между собой, другие — нет). При необходимости должна быть обеспечена также группировка объектов для их совместной обработки или, наоборот, разбиение объекта на части для его последующей обработки.

Процесс расширения основного оператора на демонстрационном примере

Основываясь на задаче, которая поставлена на демонстрационном примере, проведем расширение основного оператора радикального моделирования. Пусть для некоторой ЦС построены следующие схемы радикалов: исходная; целевая; допустимые промежуточные; библиотечные; ресурсные; схемы ДН-метода синтеза уникама. Пусть в схемах представлены все подсистемы КСК ЦС и все связи между ними. Пусть в схемах представлен весь цикл действий со штатными радикалами, а также со всеми типами объектов обработки ЦС. По условиям задачи требуется обеспечить достижение целевой схемы радикалов из исходной, удовлетворив при этом всем имеющимся ультраконтентам (правилам-продукциям) при условии, что возможно появление новых схем, представляющих нештатные ситуации в проблемной области. Здесь под условием удовлетворения понимается отсутствие конфликтов в РМ. Заметим также, что в качестве причин возникновения не-

штатных ситуаций могут выступать не только внешние воздействия, но и изменения внутри самой ЦС.

Основной оператор радикального моделирования должен действовать оперативно, имея в виду отсутствие конфликтов по временным ресурсам. Действие оператора должно быть таким, чтобы в дальнейшем ЦС смогла успешно продолжить и завершить свой жизненный цикл. Это требование означает, что не должно быть конфликтов по состоянию ЦС, а также по условиям использования системой ее ресурсов. Последнее означает, что ресурсы не должны быть растрочены.

В терминах табличного представления схем радикалов основной оператор радикального моделирования — это действия по достижению целевой конфигурации ячеек таблицы из исходной конфигурации с определенными ограничениями. Аналогично, в терминах геометрического представления схем — это действия по достижению целевой конфигурации векторов с определенными ограничениями.

Описанный выше основной оператор радикального моделирования характерен для проблемной области любой ЦС, в том числе для таких ЦС, как ПТС. Во всех ЦС присутствуют (так или иначе) цели, ресурсы, конфликты (которых следует избегать), библиотеки, а также штатные и нештатные задачи, которые необходимо решать. Более того, во всех таких ЦС для реализации основного оператора применяется процедура синтеза (конструирования), когда целевой уникаму (решение задачи) конструируется из библиотечных уникамов.

Подчеркнем, что в радикальном моделировании и в радикальном программировании речь всегда идет о составляющих ЦС, о математических объектах (задачах, методах), о ПТС, используемых при моделировании широкой проблемной области. В ПТС частные модели ЦС конструируют (реализуют) из функций (процедур), управляющих структур, программных классов и т. д. Задача разработки ПТС относится к научно-практическому направлению программной инженерии [10]. Согласно подходу, основанному на радикальном моделировании и радикальном программировании, для реализации основного оператора требуется анализировать, проектировать, оценивать, реализовывать, тестировать, обслуживать и модернизировать ПТС ЦС в рамках инструментального средства обеспечения комплексного развития ЦС, основанного на РМ ЦС. При таком подходе основной оператор радикального моделирования ЦС является в то же время основным оператором, поддерживающим процедуры инженерии программ.

Как отмечено ранее, программная инженерия основывается на системном подходе. Системный подход, в свою очередь, определяется как направление методологии научного познания, в основе которого лежит рассмотрение объекта как системы — целостного комплекса взаимосвязанных элементов. Таким образом, можно рассматривать радикальное моделирование как универсальный инструментарий системного подхода, ориентированный на его применение при создании и сопровождении ПТС любого назначения.

Приведем теперь дополнительные замечания, поясняющие работу основного оператора, реализуемого

с помощью ДН-метода синтеза уникаму, который решает очередную задачу среды радикалов. Рассмотрим упрощенную нештатную задачу. Пусть имеется целевой уникаму $uGoal$. Требования к нему описываются некоторой системой контейнеров, приводить которые не будем. Требуется реализовать (построить) этот целевой уникаму с помощью библиотечных радикалов. Так как вначале $uGoal$ не реализован (еще не известно, удастся ли его реализовать), то оценим его как виртуальный с помощью уникаму $uVirtual$, вложенного в контейнер оценки $c_valuation$ (рис. 10, в схемах этого примера направления в контейнерах в целях упрощения не пронумерованы).

К $uGoal$ применяется основной оператор (уникаму $uMainOperator$, вложенный в контейнер $c_sys_operator$) (рис. 11), и на основе ДН-метода осуществляется работа с библиотекой стандартных радикалов.

Отметим последовательность действий основного оператора. На первом уровне иерархии уникамов, которая должна реализовать целевой уникаму $uGoal$, выполняются действия с помощью $u1$ и $u2$ (вложены в контейнер $c_sys_subsystems$ вместе с $uGoal$). Каждый из этих уникамов приносит с собой свою систему КСК, и с помощью соответствующих контейнеров уникаму $u1$ и $u2$ связываются между собой. Эти контейнеры на рис. 11 не показаны. Здесь важно отметить, что уникаму $u1$ — конкретный, как следствие, он оценивается как $uReal$. Уникаму $u2$ — обобщенная составляющая, представляющая класс составляющих ЦС, поэтому для нее оценка — $uVirtual$. Теперь надо реализовать промежуточный целевой уникаму $u2$. Для этого рекурсивно используется уникаму основного оператора $uMainOperator$ (соответствующее сечение среды радикалов, рис. 12).

Пусть результат успешен — $u2$ удалось реализовать с помощью уникамов $u21, u22$, причем оба

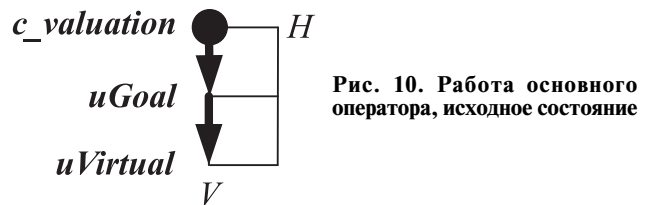


Рис. 10. Работа основного оператора, исходное состояние

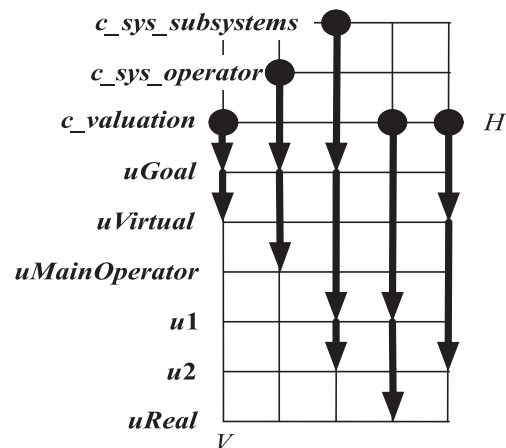


Рис. 11. Работа основного оператора, первый этап

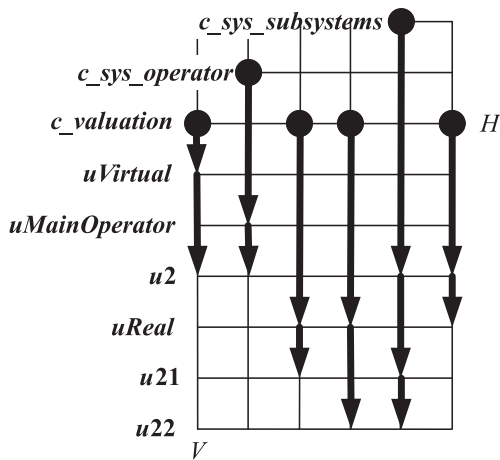


Рис. 12. Работа основного оператора, второй этап

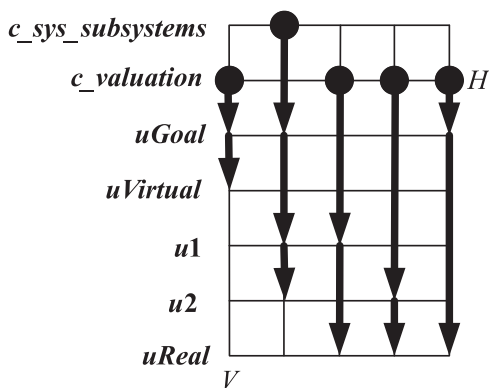


Рис. 13. Работа основного оператора, завершающий этап

они — конкретные (*uReal*). В связи с этим обстоятельством изменяется оценка для *u2*: была *uVirtual*, стала *uReal* — работает соответствующее правило, представленное ультраконтейнером. Теперь имеем: *uGoal* реализуется (на верхнем уровне) с помощью *u1* (оценка — *uReal* с самого начала) и *u2* (оценка — вначале *uVirtual*, затем заменена на *uReal* в соответствии с оценками для уникамов иерархии, реализующей *u2*), о чем иллюстрирует сечение среды радикалов, представленное на рис. 13.

Таким образом, и для самого *uGoal* следует заменить исходную оценку *uVirtual* на *uReal*.

Вопросы создания атласа схем радикалов и математическое обеспечение радикального моделирования

Неудовлетворительное состояние с разработкой средств автоматизации процессов в сложных ЦС, которое отмечалось выше, является следствием того, что эта проблема исследована еще недостаточно. Представляется, что одним из шагов на пути существенного улучшения состояния дел на этом направлении могла бы стать разработка начального атласа схем радикалов, характерных и общих для ЦС, из различных проблемных областей. С одной стороны,

атлас должен ориентироваться на понятие основного оператора и описывающие его схемы. С другой стороны, в основание атласа должна быть положена многоуровневая РМ ЦС [4], и таким образом должны быть в основном охвачены схемы радикалов, представляющие перечисленные далее сущности среды радикалов.

- *Уникумы, контейнеры и ультраконтейнеры* — стандартные радикалы в терминологии концептуальных положений среды радикалов и библиотеки стандартных радикалов.

- *Константы и переменные* — параметры составляющих ЦС.

- *Ультраоснащенные КСК* — координатные системы контейнеров, характерные для ЦС, для их ультраоснащения, связи между КСК.

- *Средства, используемые при преобразовании схем радикалов*: механизмы выделения (окраски) и активации радикалов; механизмы формирования запросов и поиска ответов на них.

- *Средства "развертывания" среды радикалов* (средства генерирования новых схем радикалов с помощью запросов и специализированной схемы поиска ответов на запросы — радикал-активатора, работающего в том числе с ультраконтейнерами).

- *Классификации схем радикалов*. Средства для классификации задач (и схем радикалов вообще).

- *Средства для представления конфликтов и ухода от них* (разрешения конфликтов), охватывающие следующие задачи: задачи прогнозирования и оценки конфликтов; задачи классификации преобразований среды радикалов с точки зрения приближения ее к конфликту (удаления от конфликта); задачи разрешения конфликта (ухода от конфликта).

- *Схемы ДН-метода* синтеза уникама разрешения штатных и нештатных конфликтов.

- *Средства модификации РМ ЦС*.

- *Средства представления широкой проблемной области ЦС, ее подсистем, средства решения соответствующих задач проблемной области*.

- *Датчики и исполнители ЦС*.

- *Средства представления широкой проблемной области взаимодействующих ЦС, средства решения соответствующих задач проблемной области*.

- *Виды обеспечения, характерные для автоматизированных систем согласно работам [11, 12]: математическое, программное, техническое, информационное, лингвистическое, методическое, организационное, эргономическое, правовое и метрологическое обеспечение*.

Работа над атласом схем радикалов предполагает анализ широких (в принятой интерпретации) проблемных областей ЦС различных типов с точки зрения их внутренней общности, применения концепции среды радикалов, выявления, обобщения схем радикалов, характерных для различных ЦС.

Основой предлагаемой технологии радикального моделирования является *математическое обеспечение РМ*, состоящее из двух частей. Назовем их, с определенной степенью условности, *базовое математическое обеспечение (БМО)* и *инструментальное математическое обеспечение (ИМО)*.

К БМО РМ отнесем наиболее общее математическое обеспечение, необходимое для ЦС любой проблемной области. Основное его содержание — реализованные с помощью ПТС базовые элементы и базовые операции (процедуры), позволяющие с помощью схем радикалов с высокой степенью адекватности осуществлять построение (конструирование) любых потенциально реализуемых с помощью ПТС математических объектов и методов, которые могут быть востребованы для разработки различных ЦС. Такое БМО служит средством объединения математических объектов и методов ЦС, которые используются при построении. Оно предоставляет механизмы для верификации реализующих эти методы программ, для оценки эффективности их использования при решении различных прикладных задач. В состав БМО РМ входит также основной оператор радикального моделирования.

Инструментальное математическое обеспечение РМ — это всевозможные математические объекты и методы (начиная с основных), разработанные с помощью БМО, а также с помощью других построенных ранее математических объектов и методов ИМО. Все математические объекты и методы ИМО при этом реализуются с помощью ПТС. Эффективность таких методов применительно к решению конкретных задач ЦС должна постоянно оцениваться, а сами методы — корректироваться по результатам такой оценки. Кроме того, программно-технические реализации методов должны подлежать перманентной (периодической) верификации на предмет их соответствия предъявляемым требованиям. Порядок такой верификации, содержание испытаний зависят, например, от формы представления метода, от сопровождающих его описаний, от специфики самого метода, от результатов предыдущих испытаний. Возможно также исключение методов из состава ИМО РМ, например, в результате выявления ошибок, которые принципиально нельзя исправить, несоответствия спецификациям и др.

Описанный выше подход к построению математического обеспечения РМ означает принципиальное изменение точки зрения на математические методы, которые используются для построения ЦС. В рамках предлагаемого подхода математические методы обязательно должны быть представлены в схемах радикалов, реализованы с помощью ПТС и в любое время должны быть готовы к использованию.

К созданию редактора схем радикалов и инструментальные средства решения задач широкой проблемной области целенаправленной системы

Как показывает опыт, применение радикального моделирования приносит существенную пользу на практике [13—16]. Проблемная область ЦС (ПТС ЦС) становится более обозримой и контролируемой, что облегчает принятие решений по самым разным вопросам создания, эксплуатации и модернизации таких систем.

Радикальное моделирование хорошо согласуется с методологией инструментального средства для решения задач на всех этапах жизненного цикла ЦС в рамках расширенной проблемной области ЦС.

Согласно этому подходу основная цель проводимых работ заключается в создании технологии разработки математического, программно-технического и других видов обеспечения целевых автоматизированных систем, предназначенных как для научных исследований, так и для решения практических задач. Программно-технические решения предоставляются пользователям в виде набора взаимосвязанных инструментальных средств. Все разрабатываемые инструментальные средства при этом должны иметь единую основу, которая базируется на концептуальных положениях среды радикалов и универсальном формализме схем радикалов — на языке RADICAL. Программно-технические решения для реализации этих средств могут изменяться, однако технология работы со схемами радикалов остается постоянной. Она основана на использовании стандартных радикалов (уникумов, контейнеров, ультраконтейнеров), средств активации схем, поиска ответов на запросы и на двунаправленном методе синтеза целевых уникалов. При таком подходе вполне естественным становится, например, использование *виртуальных* радикалов [5]. Если эксперту-разработчику не хватает имеющихся в библиотеке радикалов, он может ввести в библиотеку необходимые ему виртуальные радикалы, обладающие необходимыми свойствами (системами контейнеров), и строить целевые уникалы с их помощью.

Как следствие представленных выше соображений, для проведения работ по созданию начального атласа схем радикалов ЦС, для разработки и совершенствования инструментария радикального моделирования необходим *редактор* схем радикалов. Здесь в качестве основных представлений схем радикалов предлагается табличное и геометрическое представления, а также их комбинации. При этом должен быть обеспечен быстрый переход от одного представления к другому.

Представляется целесообразным, чтобы на начальном этапе функции редактора заключались в создании достаточно простых библиотечных радикалов и в построении с их помощью схем радикалов (также вначале простых), представляющих проблемную область, например, на языке диаграмм, перемещающихся в различных плоскостях 2D-прямоугольников. Редактор должен обеспечивать в том числе описание исходной, целевой и допустимых (недопустимых) промежуточных (на пути от исходной к целевой) систем, состоящих из 2D прямоугольников. Должно быть обеспечено проведение вычислительных экспериментов по автоматизированному достижению целевых систем прямоугольников из исходных, т. е. по реализации основного оператора радикального моделирования.

Должна быть обеспечена поддержка модификации создаваемых схем радикалов, их активации и оценки. Необходимо поддерживать формирование обозримых сечений среды радикалов (точек зрения на среду радикалов), поскольку соответствующие схемы для проблемной области ЦС обычно сложны и недоступны для обозрения с одной точки зрения целиком и во всех подробностях. Конечно, необходимо разработать систему *контрольных* (тестовых) примеров, на которой работа редактора будет проверяться.

Список литературы

Благодаря используемым представлениям схем радикалов, все соответствующие ситуации, складывающиеся в плоскостях КСК, все преобразования схем радикалов, все процессы, протекающие в среде радикалов, можно увидеть на экране дисплея, при этом снабженными численными оценками. Это увеличит *выразительность интерфейса РМ*.

Случай диаграммной визуализации проблемной области, состоящей из плоских прямоугольников, представляется принципиально важным. Видимо, с помощью связанных между собой систем прямоугольников, перемещающихся в разных плоскостях, можно представить проблемную область любой ЦС. Однако этот вопрос требует отдельного рассмотрения.

Относительно средств реализации редактора схем радикалов с помощью ПТС представляется возможным утверждать, что объектно-ориентированное, визуальное, компонентное, логическое программирование, языки программирования C++, Object Pascal, Prolog, системы управления базами данных (например, Microsoft SQL Server), другие инструментальные программные средства предоставляют достаточно богатые возможности, которые с успехом могут быть использованы для начала работы над редактором.

Заключение

На демонстрационном примере проиллюстрированы потенциальные возможности моделирования с использованием среды радикалов на всех этапах жизненного цикла сложных целенаправленных систем.

Отталкиваясь от базовых положений радикального моделирования представлены соображения авторов по развитию этой методологии, включая разработку начального атласа схем радикалов, математического обеспечения, по созданию редактора таких схем.

Отмечая определенный прогресс в области практического применения радикального моделирования и программирования на основе среды радикалов, авторы призывают заинтересованных исследователей активно включиться в эту работу.

1. **Критически** важные объекты и кибертерроризм. Часть 1. Системный подход к организации противодействия / Под ред. В. А. Васенина. М.: МЦНМ, 2008. 398 с.
2. **Васенин В. А.** Модернизация экономики и новые аспекты инженерии программ // Программная инженерия. 2012. № 2. С. 2—17.
3. **Чечкин А. В., Пирогов М. В.** Метод интеллектуализации критических систем с использованием таблиц радикалов // Нейрокомпьютеры: разработка, применение. 2012. № 2. С. 3—12.
4. **Васенин В. А., Пирогов М. В., Чечкин А. В.** Радиальное моделирование и инженерия сложных программных систем // Программная инженерия 2014. № 10. С. 3—10.
5. **Пирогов М. В.** Радиальное программирование // Программная инженерия. 2013. № 4. С. 2—15.
6. **Соболева Т. С., Чечкин А. В.** Дискретная математика. Третье издание. М.: Изд. центр Академия, 2014. 256 с.
7. **Чечкин А. В.** Математическая информатика. М.: Наука, 1991. 412 с.
8. **Чечкин А. В.** Обеспечение информационно-системной безопасности сложной системы на основе среды нейрорадикалов ее проблемной области // Нейрокомпьютеры: разработка, применение. 2008. № 7. С. 6—11.
9. **Чечкин А. В., Пирогов М. В.** Активаторы и регуляторы среды радикалов // Нейрокомпьютеры: разработка, применение. 2012. № 5. С. 3—8.
10. **Свободная энциклопедия** Википедия. Программная инженерия. URL: https://ru.wikipedia.org/wiki/Программная_инженерия.
11. **ГОСТ 34.003—90.** Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения.
12. **ГОСТ 23501.101—87.** Системы автоматизированного проектирования. Основные положения.
13. **Чечкин А. В., Евграфов А. Е., Рожков В. В.** и др. Применение схем радикалов для описания проблемной области автоматизированного комплекса планирования и управления // Информационно-измерительные и управляющие системы. 2009. Т. 7, № 3. С. 5—11.
14. **Губанов А. А., Пирогов М. В., Плещев Н. Н., Рожков В. В.** Метод формирования программы работы целевой аппаратуры космического аппарата с использованием визуализации // Нейрокомпьютеры: разработка, применение. 2015. № 2. С. 35—42.
15. **Губанов А. А., Пирогов М. В., Плещев Н. Н., Рожков В. В.** Управление знаниями и данными автоматизированного комплекса планирования и управления // Нейрокомпьютеры: разработка, применение. 2015. № 3. С. 51—58.
16. **Губанов А. А., Пирогов М. В., Плещев Н. Н., Рожков В. В.** Радиальное моделирование проблемной области планирования и управления беспилотного космического аппарата // Нейрокомпьютеры: разработка, применение 2015. № 12. С. 25—33.

The Main Operator of Radical Modeling on a Demonstration Example

V. A. Vasenin, vasenin@msu.ru, Moscow State University, 119234, Moscow, Russian Federation,
M. V. Pirogov, pmv_mvpm@mail.ru, NPO Lavochkin, 141400, Moscow Region, Russian Federation,
A. V. Chechkin, A.Chechkin@mail.ru, Military Academy by Great Pieter, 143900, Moscow Region, Russian Federation

Corresponding author:

Vasenin Valery A., Professor, Moscow State University, 119234, Moscow, Russian Federation
e-mail: vasenin@msu.ru

*Received on October 27, 2015
Accepted on November 31, 2015*

In this article questions of support of complex purposeful systems are discussed. Including the questions relating to subsystems of automation of various technological process and hardware and software for the solution of tasks at various stages of life cycle of purposeful systems is discussed. Despite certain achievements of purposeful systems

in this area there is a number of tasks important for practice which aren't solved now. The problem of information system security is considered as a key at creation, operation and modernization of purposeful systems. Authors of this article offer approach to permission of this problem. This approach consists in intellectual management of process of engineering of purposeful systems by means of methods and instruments of radical modeling and radical programming. Conceptual provisions of such approach are based on use of radical environment (multilevel model of problem area of purposeful systems) and also on systems application of language of schemes of radicals which is a universal remedy of the description of problem area. The concept of the main operator of the radical modeling providing implementation of the target scheme of radicals is entered. The concept the main operator is illustrated by a demonstration example. Reasons of authors according to the solution of the questions relating to creation of the initial atlas of schemes of radicals for problem areas of purposeful systems are presented. Approach to creation and structurization of mathematical support of radical modeling is stated. Need of the specialized program — the editor of schemes of radicals is proved.

Keywords: systems thinking, complex system, purposeful system, critical system, information system security, intellectual management, radical environment, scheme of radicals, universal language of informatics, "RADICAL", language of diagrams, hardware and software, software engineering, radical modeling, radical programming, radical model, main operator of radical modeling

For citation:

Vasenin V. A., Pirogov M. V., Chechkin A. V. The Main Operator of Radical Modeling on a Demonstration Example, *Programmnyaya Ingeneriya*, 2016, vol. 7, no. 2, pp. 75–85.

DOI: 10.17587/prin.7.75-85

References

1. **Kriticheski** vazhnye ob'ekty i kiberterrorizm. Part 1. *Systemnyy podhod k organizacii protivodejstviya* (Crucial objects and cyberterrorism. Part 1. System approach to the organization of counteraction) / Eds. by V. A. Vasenin. Moscow, MCNMO, 2008. 398 p. (in Russian).
2. **Vasenin V. A.** Modernizacija jekonomiki i novye aspekty inzhenerii programm (Modernization of economy and new aspects in engineering of programs), *Programmnyaya Ingeneriya*, 2012, no. 2, pp. 2–17 (in Russian).
3. **Chechkin A. V., Pirogov M. V.** Metod intellektualizacii kriticheskikh system s ispol'zovaniem tablic radikalov (Method of intellectualization of critical systems with use of tables of radicals), *Nejrokomputery: Razrabotka, Primenenie*, 2012, no. 2, pp. 3–11 (in Russian).
4. **Vasenin V. A., Pirogov M. V., Chechkin A. V.** Radikalnoe modelirovanie i inzheneriya slozhnykh programmnykh system (Radical modeling and engineering of difficult program systems), *Programmnyaya Ingeneriya*, 2014, no. 10, pp. 3–10 (in Russian).
5. **Pirogov M. V.** Radikal'noe programirovanie (Radical programming), *Programmnyaya Ingeneriya*, 2013, no. 4, pp. 2–15 (in Russian).
6. **Soboleva T. S., Chechkin A. V.** *Diskretnaja matematika* (Discrete mathematics), Moscow, Akademija, 2014. 256 p. (in Russian).
7. **Chechkin A. V.** *Matematicheskaja informatika* (Mathematical informatics), Moscow, Nauka, 1991. 412 p. (in Russian).
8. **Chechkin A. V.** Obespechenie informacionno-sistemnoj bezopasnosti slozhnoj sistemy na osnove sredy neyroradikalov ee problemnoj oblasti (Providing the information system security of complex system on the bases of the medium of neuroradicals of its problem area), *Nejrokomputery: Razrabotka, Primenenie*, 2008, no. 7, pp. 6–11 (in Russian).
9. **Chechkin A. V., Pirogov M. V.** Aktivatory i regulatory sredy radikalov (Activators and regulators of environment of radicals), *Nejrokomputery: Razrabotka, Primenenie*, 2012, no. 5, pp. 3–8 (in Russian).
10. **Svobodnaja** jenciklopedija Wikipedia. Programmnyaya inzheneriya (Software engineering), available at: https://ru.wikipedia.org/wiki/Software_engineering.
11. **GOST 34.003–90.** Informacionnaja tehnologija. Kompleks standartov na avtomatizirovannye sistemy. Avtomatizirovannye sistemy. Terminy i opredelenija (Information technology. Set of standards for automated systems. Automated systems. Terms and definitions) (in Russian).
12. **GOST 23501.101–87.** Sistemy avtomatizirovannogo projektirovanija. Osnovnye polozhenija (Computer-aided design systems. Main provisions) (in Russian).
13. **Chechkin A. V., Evgrafov A. E., Rozhov V. V., Loshhenkov V. I., Pirogov M. V.** Primenenie shem radikalov dlja opisaniya problemnoj oblasti avtomatizirovannogo kompleksa planirovanija i upravlenija (Application of schemes of radicals for the description of problem area of the automated complex of planning and management), *Informacionno-Izmeritel'nye i Upravljajushie Sistemy*, 2009, vol. 7, no. 3, pp. 5–11 (in Russian).
14. **Gubanov A. A., Pirogov M. V., Pleshhev N. N., Rozhov V. V.** Metod formirovanija programmy raboty celevoj apparatury kosmicheskogo apparata s ispolzovaniem vizualizacii (Method of formation of the program of work of the target spacecraft equipment using visualization), *Nejrokomputery: Razrabotka, Primenenie*, 2015, no. 2, pp. 35–42 (in Russian).
15. **Gubanov A. A., Pirogov M. V., Pleshhev N. N., Rozhov V. V.** Upravlenie znaniyami i dannymi avtomatizirovannogo kompleksa planirovanija i upravlenija (Knowledge management and data of the automated complex planning and control), *Nejrokomputery: Razrabotka, Primenenie*, 2015, no. 3, pp. 51–58 (in Russian).
16. **Gubanov A. A., Pirogov M. V., Pleshhev N. N., Rozhov V. V.** Radikalnoe modelirovanie problemnoj oblasti planirovanija i upravlenija bespilotnogo kosmicheskogo apparata (Radical modeling of problem area of planning and control of unmanned spacecraft), *Nejrokomputery: Razrabotka, Primenenie*, 2015, no. 12, pp. 25–33.

Е. В. Орлова, канд. техн. наук, доц., e-mail: ekorl@mail.ru, Уфимский государственный авиационный технический университет

Механизм, модели и алгоритмы управления производственно-экономическими системами на принципах согласования критериев заинтересованных агентов

Рассмотрены механизм, а также модели и алгоритмы их реализации при управлении процессами функционирования производственно-экономической системы (ПЭС) в условиях системной модернизации экономики с учетом согласования интересов ПЭС, агентов рыночной и институциональной сред. Предложена авторская концепция управления функционированием ПЭС, применение которой позволит повысить эффективность принимаемых решений за счет согласования противоречивых критериев заинтересованных агентов.

Ключевые слова: производственно-экономические системы, моделирование, управление, согласование интересов, многокритериальность и полисубъективность принятия решений

Введение

В сложившихся условиях неопределенности глобальных мировых экономических трендов и смены технологических укладов стратегия вывода экономики России из состояния рецессии должна базироваться на системном подходе и поиске ключевых системообразующих объектов, определяющих состояние и динамику развития экономики. Таким объектом является предприятие как производственно-экономическая система (ПЭС), интегрирующая в ходе целенаправленной экономической деятельности институциональные, капитальные, интеллектуальные, трудовые, финансовые и информационные ресурсы.

К числу факторов, существенно влияющих на деятельность предприятий, относят как рыночные факторы, способствующие адаптации и согласованию интересов микроэкономических агентов, так и нерыночные факторы, связанные с административными барьерами, определяющие взаимоотношения предприятий с федеральными, региональными и муниципальными структурами. Внутренняя среда около половины российских предприятий характеризуется отсутствием мероприятий по реструктуризации, внедрению инновационных технологий. Этот факт свидетельствует о необходимости организационно-экономической модернизации как внутренней, так и внешней сред функционирования предприятий. Магистральной должна стать траектория к консолидации и равноправному сотрудничеству всех самостоятельных микроэкономических агентов

экономики России. Это в равной степени можно отнести к горизонтальному (функциональному) и вертикальному (иерархическому) их взаимодействию.

Для повышения продуктивности экономики и создания системы устойчивого инновационного развития предприятий разных отраслей, форм собственности, организационно-правовых форм, размеров и территориального месторасположения необходимо разработать механизмы многостороннего и многоуровневого стратегического планирования и управления. Система управления, основанная на этих механизмах, должна включать в себя стратегическое планирование на уровне предприятий, на уровне агентов рынка и на уровне государства. На уровне предприятий система стратегического планирования должна быть единой в разрезе функциональной и управленческой вертикалей. На уровне агентов рынка процессы создания и реализации стратегических планов должны учитывать интересы всех участников деятельности предприятия, они призваны стать механизмами консолидации и сбалансированного управления предприятием. На уровне государства должна быть сформирована бюджетно-налоговая политика как наиболее активная составляющая экономической и промышленной политики, ориентированная на развитие потенциала предприятий и на их модернизацию.

В связи с изложенным выше задача создания инструментария для моделирования и управления функционированием ПЭС в условиях системной модернизации экономики, с учетом согласования интересов агентов самой ПЭС, рыночной и инсти-

туциональной сред, является актуальной. Решение этой задачи и применение результатов этого решения позволит повысить эффективность управленческих процессов ПЭС.

Анализ исследований в области моделирования и управления организационными системами [1—8] показал, что задачи, связанные с экономико-математическим моделированием и управлением процессами согласования экономических интересов участников производственно-экономического, рыночного и бюджетно-налогового процессов, направленных на развитие ПЭС и экономики в целом, не нашли своего достаточного отражения. Проведенные ранее исследования не затрагивают вопросы динамического ценообразования на продукцию предприятий в условиях конкурентного рынка, изменяющихся предпочтений потребителей и поведенческих стратегий конкурентов.

1. Описание ПЭС как объекта управления

Производственно-экономическая система представляет собой сложную организационную систему, объединяющую производство и реализацию продукции, воспроизводство ресурсов. Она характеризуется динамичностью, большим числом параметров, сильной их взаимосвязанностью, иерархичностью своей структуры, наличием обратных материальных и информационных связей, нелинейностью и нестационарностью. Это кибернетическая система, функционирующая в условиях неопределенности в многомерном пространстве ограничений по всем ресурсам, которая находится под воздействием случайных возмущений. Сложные ПЭС создаются человеком в целях преобразования ресурсов в готовую продукцию или услуги, удовлетворяющие потребности общества.

Схема взаимодействия ПЭС с макросистемами — государством и социумом — представлена на рис. 1, см. третью сторону обложки. К макросреде функционирования ПЭС относится государство, которое обладает полномочиями по регулированию экономики (и ПЭС как части экономики), обеспечивая безопасность, единство и территориальную целостность национальной экономики, возможности ее развития.

Являясь открытой системой, ПЭС в процессе получения ресурсов, проектирования, производства и реализации продукции активно взаимодействует с внутренней и внешней средами. Внутреннюю среду составляют производственно-технические, экономические и информационные факторы. Производственно-технические факторы включают совокупность машин, оборудования, инструментальных средств, сырья, материалов, а также технологий производства. Экономические факторы отражают совокупность экономических процессов, включающих движение капитала и денежных средств, экономические показатели деятельности предприятия (рентабельность, оборачиваемость). Информационные факторы при этом представляют собой совокупность программно-

технических средств, обеспечивающих ПЭС соответствующей информацией для эффективных коммуникаций в процессе управления ею. Внешнюю среду составляют субъекты деловой среды — поставщики, конкуренты и потребители, а также экономические и политические процессы, включая государственную политику в области налогового, кредитно-денежного и других инструментов государственного регулирования.

В целом, деятельность предприятия осуществляется в рамках ряда ограничений, к числу которых относят технологические ограничения (производственная функция); финансовые ограничения (стоимость факторов производства); спрос (емкость рынка); деятельность конкурентов; государственное регулирование, налоги и субсидии; этические правила и нормы (общественные нормы ведения бизнеса); время.

Структура взаимосвязей ПЭС и окружающих ее подсистем микро- и макроуровней может быть представлена функциональными и регулируемыми связями. Функциональные связи обеспечивают согласованные действия подсистем в процессе реализации своих функций, а регулирующие связи реализуют процессы координации, контроля и регулирования одной подсистемы со стороны другой. При этом должно обеспечиваться согласование интересов всех взаимодействующих подсистем.

Исходя из изложенного выше, проблемное поле можно представить как совокупность взаимосвязанных, подлежащих решению вопросов. Эти вопросы возникают в процессе взаимодействия ПЭС как основного звена экономики, производящей продукцию, с рынком как сферой активности агентов, оказывающих влияние на принятие экономических решений ПЭС, а также с институциональной средой государства, определяющей правила экономической деятельности ПЭС в части фискального регулирования. Можно выделить следующие основные вопросы:

- обеспечение развития ПЭС через налоговое стимулирование;
- обеспечение устойчивого функционирования ПЭС через балансирование ресурсов и результатов деятельности;
- согласование экономических интересов ПЭС с ее финансовыми возможностями;
- повышение эффективности управления ПЭС через согласованное с внешними агентами поведение на отраслевом рынке.

2. Концепция моделирования и управления ПЭС

Основным объектом анализа в настоящей работе являются предприятия как производственно-экономические системы, образующие один из слоев многоуровневой структуры национальной экономики. Предприятия рассматривают как целостные образования, функционирующие в пространственно-временном континууме. Они состоят из совокупности взаимосвязанных внутренних объектов, процессов их

взаимодействия, проектных решений и сред и ряда других атрибутов их деятельности. Процессы управления предприятием моделируют с учетом сложных взаимодействий разноплановых подсистем, обеспечивая согласованность экономических интересов участников хозяйственного, финансового, рыночного процессов.

Структура системы управления определена в виде трех блоков задач. К их числу относят блоки, обеспечивающие формирование: эффективной системы управления ресурсами предприятия, направленной на согласование стратегического и оперативного уровней управления (управление проектами, процессами); адаптационного механизма управления рынком, поддерживающего согласованность интересов производителей продукции и потребителей (управление процессами ценообразования); налоговой нагрузки и налоговых ставок как инструмента бюджетно-налоговой политики государства, направленной на повышение экономического роста экономики в целом (управление институциональной средой). Решение перечисленных задач осуществляется на основе системы моделирования, имеющей иерархическую структуру (рис. 2, см. третью сторону обложки), которая включает перечисленные далее составляющие.

- ◇ Моделирование налоговой нагрузки и налоговых ставок с учетом противоречивых интересов ПЭС (субъектов налогообложения) и экономики как инструмента налоговой политики, направленной на развитие ПЭС. Здесь выявляют наиболее общие закономерности, характеризующие развитие экономической системы в целом.
- ◇ Моделирование особенностей рыночных взаимодействий производителей и потребителей. Этот срез важен, так как именно рынок обуславливает формирование цен, являющихся одними из важнейших характеристик эффективности ПЭС.
- ◇ Моделирование ресурсной составляющей ПЭС. На этом уровне рассматривается преобразование ресурсов в результаты и целью являются обеспечение наиболее рационального распределения ресурсов для производства продукции и обеспечение эффективного развития ПЭС.

Иерархическая система моделирования строится следующим образом.

Первый уровень моделирования определяет модель оптимальной налоговой системы. На макроуровне моделируются не сами процессы, а их результаты. Речь идет о моделировании динамики ограничений, накладываемых на исследуемые процессы. Для анализа таких характеристик, как налоговая нагрузка, важны не сами детали процессов, которые протекают в системах. В большей степени важны объемы доступных ресурсов, емкости экономических ниш и макропропорции (макросоотношения) базовых процессов, обеспечивающих заполнение этих ниш в результате активной деятельности при использовании имеющихся технологий [9].

Модель оптимальной налоговой системы включает в себя модель определения эффективного уровня налоговой нагрузки. Последняя позволяет опреде-

лять области допустимой налоговой нагрузки по группам однородных субъектов налогообложения. Модель налоговой системы включает также модель определения оптимального уровня налоговых ставок по видам налогов и группам однородных субъектов налогообложения [10, 11]. На этом уровне целесообразно агрегированное рассмотрение производственно-экономических систем, которые объединяются в кластеры однородных по структуре налогооблагаемых баз субъектов налогообложения и имеют схожие налоговые возможности и близкие финансово-экономические портреты. Результатом моделирования на этом уровне являются области допустимой налоговой нагрузки по группам однородных субъектов налогообложения и обосновываются оптимальные налоговые ставки, удовлетворяющие интересам субъектов налогообложения и экономики в целом.

Второй уровень моделирования — модель динамики рыночного ценообразования. Она предназначена для управления деятельностью ПЭС в условиях нестационарности параметров внешней среды и зависимости сфер производства и потребления на основе механизма адаптивного ценообразования. Процессы формирования и изменения цен, их структура и уровень согласовываются как со стратегическими целями производителей, так и с изменяющимися предпочтениями потребителей [12].

Динамическая модель ценовой конкуренции отражает процессы стратегического взаимодействия фирм на рынке несовершенной конкуренции и описывает эти процессы с помощью теоретико-игрового подхода и методов нелинейной динамики. Динамика рыночных цен моделируется с помощью двумерного отображения. При этом согласование ценовых решений производителей осуществляется на основе мониторинга устойчивости равновесия цен по Нэшу. Результатом моделирования рыночных процессов являются равновесные по Нэшу цены на производимую продукцию, обеспечивающие достижение режима максимальной эффективности функционирования ПЭС. Результаты исследования на этом уровне задают внешние условия и ограничения для моделирования на третьем уровне — уровне отдельных производственно-экономических систем.

Третий уровень моделирования основан на моделях управления ресурсами ПЭС. Эти модели предназначены для формирования оптимальных комбинаций ресурсов производства в условиях ограничений, сформированных на первом и втором уровнях моделирования.

Цель управления ресурсами не производственная, а экономическая, т. е. поиск оптимальных сочетаний затрат производственных ресурсов. По этой причине задачи и соответствующие им функции управления ресурсами акцентированы на экономическом аспекте деятельности ПЭС. На этом уровне используются разнообразные модели, ориентированные на решение конкретных задач [13—15]. В рамках подзадачи формирования многовариантных сочетаний экономических параметров (ресурсов и объемов производства), обеспечиваю-

ших одинаковый уровень эффективности производства, основное внимание уделяют исследованию ресурсоотдачи и ее влиянию на производственную функцию, а также ситуационному анализу экономического поведения производителя. В подзадаче регулирования финансовых ресурсов определяют оптимальные финансовые стратегии ПЭС, обеспечивающие согласование ее экономических интересов с финансовыми возможностями.

На третьем уровне используют модели системной динамики, описывающие изменение результативности производственной и финансово-экономической деятельностей в зависимости от движения денежных потоков. Метод ситуационного сбалансированного управления предназначен для мониторинга текущей ситуации и формирования оптимальных в данных условиях сочетаний ресурсных потоков ПЭС. Результатом моделирования процессов ресурсообеспечения ПЭС являются значения затрат ресурсов, объемы производства и цены.

Представленный концептуальный подход предназначен для того, чтобы, с одной стороны, восходить к описанию долговременных макротенденций в области налогового регулирования производственно-экономической деятельности, а с другой стороны, спускаться к анализу перспектив отдельных предприятий с учетом складывающихся тенденций экономического развития страны.

Таким образом, система поддержки принятия управленческих решений является многоуровневой и направлена на повышение эффективности функционирования ПЭС. На нижних двух уровнях формируются данные о деятельности ПЭС, ее финансово-экономическом, технико-экономическом состоянии, а управление нацелено на мониторинг и корректировку показателей деятельности, таких как объем производства, цена, объем затрат, обеспечивающих рост эффективности ПЭС. На верхнем уровне осуществляется сбор данных о функционировании ПЭС в регионе, включая данные мониторинга налоговой отчетности ПЭС, затем происходит обработка этих данных с позиции обеспечения максимальной устойчивости и результативности деятельности ПЭС на основе регулирования показателей налоговой нагрузки и налоговых ставок по группам однородных ПЭС.

3. Механизм управления ПЭС и алгоритмы его реализации

Модель экономической эффективности производства формируется как отношение прибыли и затрат производства: $\alpha = \frac{\Pi}{C}$, где прибыль $\Pi = pq - (C_{\text{пост}} + qC_{\text{пер}}^{\text{уд}})$, а общие затраты производства C определяют как сумму переменных и постоянных затрат. Здесь α — рентабельность (относительная прибыль) при производстве товара объемом q ; Π — валовая прибыль, образуемая при реализации товара объемом q по цене p ; C — суммарные затраты, включающие постоянную часть $C_{\text{пост}}$ и переменную часть $C_{\text{пер}}^{\text{уд}} = qC_{\text{пер}}^{\text{уд}}$. Размер налога на прибыль, подлежащего уплате в бюджет, определяют как $T_{\text{НП}} = t_{\text{НП}}\Pi$, где

$T_{\text{НП}}$ — объем налога на прибыль, подлежащего уплате в бюджет; $t_{\text{НП}}$ — ставка налога на прибыль.

Обобщенно в основе моделирования налоговой нагрузки лежит предположение Лаффера о нелинейной связи объема производства, отражаемого значением валового выпуска продукции, с уровнем налоговой нагрузки. Для каждой группы однородных субъектов налогообложения строятся зависимости вида $\theta = T/X$, где θ — налоговая нагрузка, T — размер налоговых поступлений; X — объем производимой продукции. Следуя этой же теории Лаффера о нелинейной зависимости налоговых доходов от уровня налоговой нагрузки, положим, что объем производства также нелинейно связан с налоговой нагрузкой. Производственная функция аппроксимируется квадратичным полиномом: $X(\theta) = a\theta^2 + b\theta$ а налоговая функция $T(\theta)$ имеет вид $T(\theta) = a\theta^3 + b\theta^2$, где a и b — параметры функций. Идентификация производственной $X(\theta)$ и налоговой $T(\theta)$ функций позволяет рассчитать точки Лаффера первого и второго рода, при которых производственная и налоговая функции достигают своего максимума соответственно.

Согласование систем оперативного и стратегического финансового планирования обеспечивается за счет учета связи показателей оперативного и стратегического уровней управления. На оперативном уровне управления предприятия обеспечивается конкурентоспособность производимой продукции, отслеживается общее состояние ПЭС, ее финансовая и операционная эффективность, а на стратегическом уровне контроль ведется за инвестиционной привлекательностью предприятия и за ростом его стоимости в долгосрочной перспективе. Взаимосвязь показателей осуществляется на основе многофакторной модели:

$$ROE = \frac{\Pi - T_{\text{НП}}}{E} = \frac{\Pi}{A} \frac{A}{E} \frac{NI}{EBIT} = ROA \cdot LR \cdot B,$$

где ROE — рентабельность собственного капитала; NI — чистая прибыль; E — собственный капитал; $EBIT$ — прибыль до выплаты процентов по займам и налога на прибыль; A — активы; ROA — рентабельность активов; LR — коэффициент, определяющий эффект финансового рычага; B — коэффициент, отражающий снижение прибыльности предприятия при выплате процентов по используемому капиталу и налоговых отчислений.

Модель конкурентного взаимодействия фирм (в случае дуополии) описывается двумерной системой разностных уравнений:

$$\begin{cases} p_1(t+1) = \\ = p_1(t) + k_1 \frac{(-p_1^2(t) p_2(t) + 2c_1 p_1(t) p_2(t) + c_1 p_2^2(t))}{(p_1^2(t) + p_1(t) p_2(t))^2}, \\ p_2(t+1) = \\ = p_2(t) + k_2 \frac{(-p_1(t) p_2^2(t) + 2c_2 p_1(t) p_2(t) + c_2 p_1^2(t))}{(p_2^2(t) + p_1(t) p_2(t))^2}, \end{cases}$$

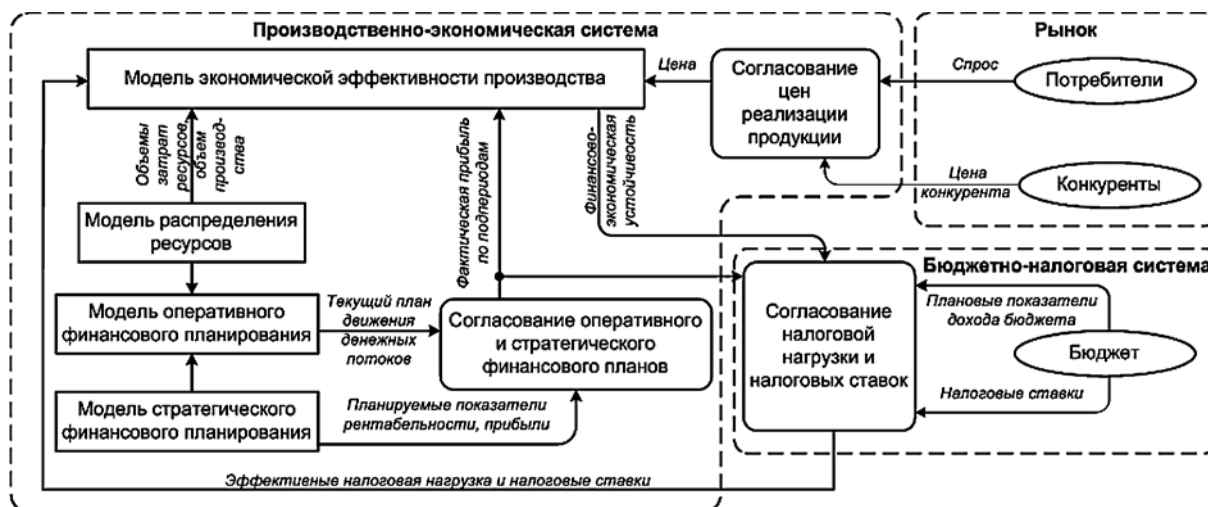


Рис. 3. Структурная схема механизма управления ПЭС

где $p_1(t)$, $p_2(t)$ — цены фирм, взятые через дискретные интервалы времени t ; вторые слагаемые в обоих уравнениях показывают, как осуществляется изменение цен в период t и каким образом это изменение повлияет на цену в следующем $(t + 1)$ периоде. Строго говоря, параметры k_1 и k_2 характеризуют прирост цен за счет изменения ценовой политики. Параметры c_1 и c_2 представляют собой себестоимости продукции первого и второго предприятий соответственно.

На рис. 3 представлена структурная схема механизма управления ПЭС, построенного на принципах учета интересов заинтересованных агентов, включая конкурентов, потребителей и бюджетно-налоговой системы. Механизм управления ПЭС состоит из четырех взаимосвязанных технологий: сбалансированной эффективности; финансового планирования; управления динамикой цен; синтеза оптимальных налоговых ставок.

Технология сбалансированной эффективности функционирования ПЭС базируется на использовании методов моделирования и ситуационного анализа. Технология моделирования и управления включает следующие этапы: анализ производственно-экономического и финансового состояния ПЭС; исследование эффективности производства товаров по видам и по стадиям их жизненного цикла; ситуационное сбалансированное управление экономическими параметрами — объемом выпуска, ценой, затратами по конкретному виду продукции на основе методов ситуационного моделирования и использования производственных правил; принятие решений об изменении экономических параметров, разработка вариантов управленческих решений.

Технология финансового планирования состоит из трех взаимосвязанных компонентов, логически соответствующих подсистемам управления финансовыми ресурсами. К числу таких компонентов относятся: моделирование платежного календаря; оценку кредитоспособности ПЭС; согласование систем оперативного и стратегического финансового пла-

нирования в виде модуля оценки влияния факторов оперативного уровня на факторы, отражающие стратегический уровень управления.

Технология управления динамикой цен в условиях стратегического взаимодействия фирм представлена в виде последовательности следующих пяти этапов. Этап 1 — оценка функции спроса на продукцию, которая определяется как решение задачи потребителя, максимизирующего свою "полезность" при бюджетном ограничении. Этап 2 — динамический анализ конкурентного взаимодействия. Определение цен, локально равновесных по Нэшу в условиях стратегического взаимодействия фирм. Этап 3 — введение адаптационного механизма для стабилизации хаотической динамики рыночных цен. Этап 4 — анализ устойчивости локально равновесных по Нэшу цен. Вычисление матрицы Якоби, вычисление старшего ляпуновского показателя, построение бифуркационных диаграмм в пространствах параметров модели, построение аттракторов системы. Этап 5 — реализация адаптационного механизма, основанного на изменении цен, пропорционального скорости изменения прибылей фирм.

Технология синтеза оптимальных налоговых ставок состоит из следующих структурных компонентов: классификация ПЭС как субъектов налогообложения по принципу однородности налогооблагаемых баз; построение налоговой и производственной функций на базе концепции Лаффера о нелинейной зависимости налоговых сборов от размера налоговой ставки; определение допустимой совокупной налоговой нагрузки на субъект налогообложения, обеспечивающей устойчивость его функционирования; оптимизация налоговых ставок по видам налогов и классам субъектов налогообложения, обоснование дифференцированных налоговых ставок, обеспечивающих максимальный выпуск продукции.

Алгоритм реализации представленного экономического механизма изображен на рис. 4. До тех пор, пока не обеспечен необходимый уровень ROE , осуществ-

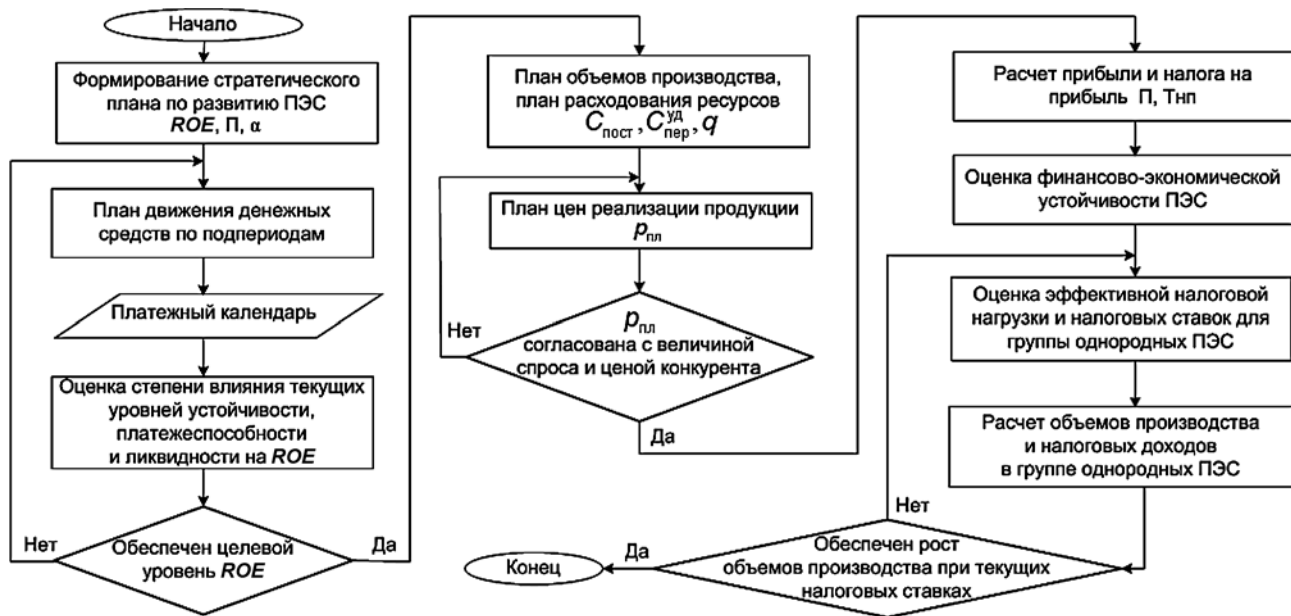


Рис. 4. Алгоритм реализации механизма управления

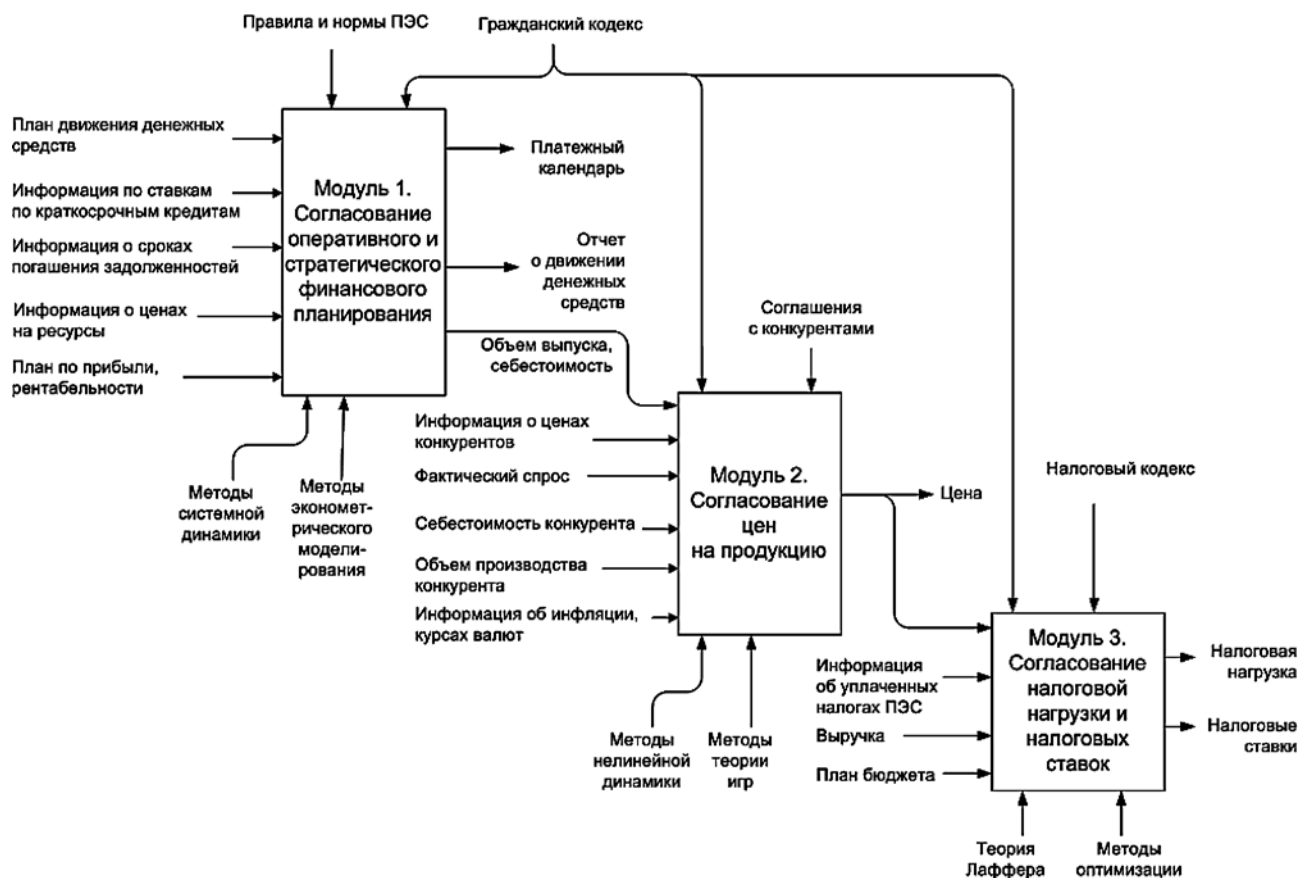


Рис. 5. Функциональная схема механизма согласованного управления ПЭС

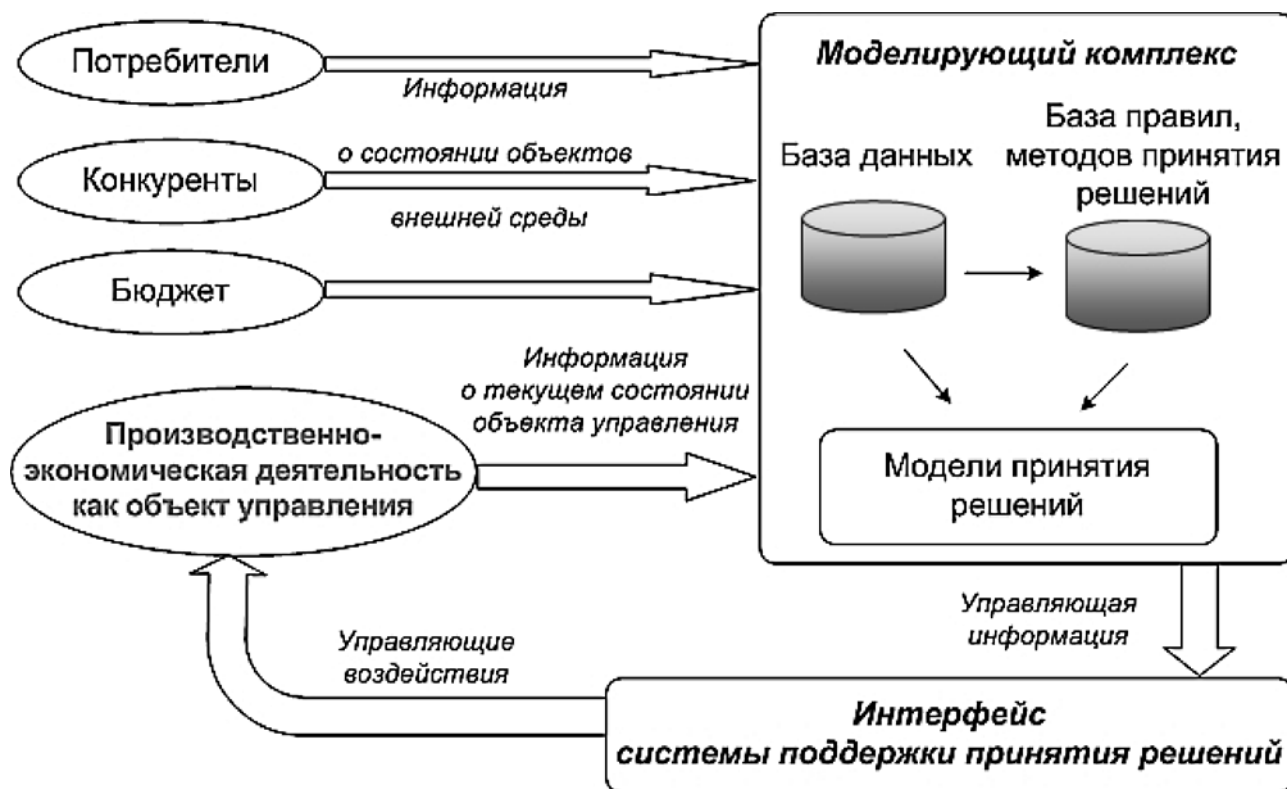


Рис. 6. Программно-инструментальный комплекс реализации механизма согласованного управления ПЭС

включают планирование и перераспределение оттоков денежных средств в соответствии с притоком денежных ресурсов со статьями затрат и по подпериодам, проводится обоснование необходимых сумм заемного финансирования. Далее осуществляется планирование объемов производства по видам продукции и объемов затрат, необходимых для этих ресурсов. Цена реализации $p_{пл}$ устанавливается исходя из объемов платежеспособного спроса и ценовой стратегии основных конкурентов. Затем для полученных объемов производства q и цен реализации p осуществляется планирование налоговых нагрузок и налоговых ставок, обеспечивающих рост объемов выручки производителя.

Функциональная схема реализации механизма согласованного управления ПЭС показана на рис. 5.

Схема отражает последовательность этапов, соответствующую каждому из этих этапов входной и выходной информации, а также методов управления.

Программно-инструментальный комплекс, реализующий механизм согласованного управления ПЭС, состоит из моделирующего комплекса и интерфейса поддержки принятия управленческих решений и показан на рис. 6.

Моделирующий комплекс включает модели и методы принятия решений, соответствующих описанным технологиям, базу данных о производимой продукции, ее свойствах, базы данных потребителей, договоров, данных о статьях расхода, определяющих платежный календарь. Модели принятия решений реализованы в виде программ, написанных с ис-

пользованием следующих инструментальных средств: MATLAB — оптимизация налоговых ставок, модели динамики цен; AnyLogic — модели платежного календаря, системно-динамическое моделирование влияния факторов оперативного уровня планирования на стратегические показатели деятельности ПЭС.

Описанный подход и механизм управления ПЭС по критерию оптимизации эффективности предназначен, с одной стороны, для предприятия, которое получает инструменты по анализу своей текущей деятельности, и поддержке принятия управленческих решений в области финансово-экономического планирования и регулирования ценовой политики. С другой стороны, инструмент полезен для государства, имеющего возможность осуществлять мониторинг и регулировать бюджетно-налоговую политику региональной социально-экономической системы, направленной на экономический рост. Такой подход можно рассматривать как управленческую инновацию, обеспечивающую ускорение процессов принятия решений как на уровне ПЭС, так и на уровне государственного института бюджетно-налогового планирования и регулирования.

4. Вычислительные эксперименты

Рассматривается вычислительный эксперимент, реализуемый для механизма согласования налоговой нагрузки и налоговых ставок. Моделирование проводится на основе использования статистических

данных за два года о значениях налогооблагаемых баз для отдельных налогов, а также значений выручки и совокупных налоговых доходов по каждому из пяти классов ПЭС — субъектов налогообложения региональной системы (крупных предприятий, обеспечивающих около 50 % налоговых доходов). Этапы проведения анализа следующие.

Этап 1. Построение зависимостей производственной $X_i(\theta)$ и налоговой $T_i(\theta)$ функций от совокупной налоговой нагрузки в группе субъектов налогообложения. Нахождение значений точек Лаффера первого и второго рода θ^* , θ^{**} по каждой i -й группе ($i = 1...5$).

Этап 2. Выявление интервала (области нечувствительности — области изменения совокупной налоговой нагрузки, удовлетворяющей интересам одновременно налогоплательщика и получателя налогов) для совокупной налоговой нагрузки θ по каждой группе объектов на основе исследования функций эластичности $E_\theta(X_i)$ и $E_\theta(T_i)$.

Этап 3. Нахождение фактического значения совокупной налоговой ставки θ на графиках функций $X_i(\theta)$ и $T_i(\theta)$.

Этап 4. Определение стратегии по изменению совокупной налоговой ставки θ и расчет эффективности от установления новой ставки для каждой группы объектов.

Этап 5. Определение стратегии по изменению ставок конкретных налогов по каждой группе объектов и расчет эффективности от их введения.

Далее представлены результаты моделирования совокупной налоговой нагрузки и налоговых ставок для первой группы налогоплательщиков. Первую группу налогоплательщиков, численное моделирование которой показано выше, можно охарактеризовать следующим образом. В исследуемой выборке предприятий-налогоплательщиков она самая многочисленная, имеет следующую структуру налогооблагаемых баз: доля налога на добавленную стоимость составляет 0,576, налога на прибыль — 0,125, единого социального налога — 0,098, налога на имущество — 0,08, прочие налоги — 0,121.

Производственная и налоговая функции, а также их точки экстремума имеют следующий вид:

$$X_1(\theta) = -141,5 \cdot 10^7 \theta^2 + 400,1 \cdot 10^6 \theta,$$

$$T_1(\theta) = -141,5 \cdot 10^7 \theta^3 + 400,1 \cdot 10^6 \theta^2, \theta^* = 0,13, \theta^{**} = 0,19.$$

Анализ фактической совокупной налоговой нагрузки для первой группы субъектов показывает, что в периодах t и $t - 1$ ее значение превышает обе точки Лаффера θ^* и θ^{**} и составляет соответственно 0,23 и 0,25. Построена область нечувствительности налоговой нагрузки, ее границы составляют от 0 до 0,09 (рис. 7, см. третью сторону обложки).

Детальный анализ производственной и налоговой функций для предприятий первой группы показал, что существующее налоговое бремя таково, что находится в третьей зоне. Это соответствует ситуации, при которой налоговое бремя находится правее обеих точек Лаффера. Выявленная картина соответствует тому, что

Темпы приростов производственной и налоговой функций при изменении налоговой нагрузки для субъектов первой группы

Изменение ставки $\Delta\theta$	Новая ставка θ	Темп прироста $(X(\theta - 0,01) - X(\theta))/X(\theta)$	Темп прироста $(T(\theta - 0,01) - T(\theta))/T(\theta)$
—	0,23	—	—
0,01	0,22	0,138	0,089
0,02	0,21	0,260	0,150
0,03	0,2	0,365	0,187
0,04	0,19	0,454	0,201
0,05	0,18	0,526	0,194
0,06	0,17	0,581	0,169
0,07	0,16	0,620	0,127
0,08	0,15	0,643	0,071
0,09	0,14	0,649	0,004
0,1	0,13	0,639	-0,074
0,11	0,12	0,612	-0,159
0,12	0,11	0,568	-0,250
0,13	0,1	0,508	-0,344
0,14	0,09	0,432	-0,440
0,15	0,08	0,339	-0,534
0,16	0,07	0,229	-0,626
0,17	0,06	0,103	-0,712
0,18	0,05	-0,040	-0,791
0,19	0,04	-0,199	-0,861
0,2	0,03	-0,374	-0,918
0,21	0,02	-0,566	-0,962
0,22	0,01	-0,775	-0,990
0,23	0	-1,000	-1,000

налогово-фискальная политика стимулирует падение производственной функции при одновременном неудовлетворении фискальных интересов. Следовательно, необходимы меры по снижению совокупной налоговой ставки, что приведет к увеличению значений производственной и налоговой функций одновременно.

Возможны следующие варианты по изменению (уменьшению) совокупной налоговой нагрузки для субъектов первой группы (см. таблицу). Также в таблице приведены расчетные значения темпов прироста значений производственной и налоговой функций, эластичностей производственной и налоговой функций, получаемых в результате реализации каждого из 22 вариантов снижения нагрузки.

Самый значительный прирост производственной функции достигается на уровне налоговой нагрузки 13...14 %, однако дальнейшее снижение налоговой ставки не даст дополнительного эффекта в области роста налоговых поступлений. Анализ эластичностей производственной и налоговой функций показал, что область нечувствительности может быть определена интервалом [0; 0,09], так как вне этого отрезка рост налоговой нагрузки приводит к большему снижению налоговых поступлений, чем достигаемый при данной нагрузке прирост объемов производства. Поэтому рекомендуемым для принятия решения является снижение совокупной налоговой ставки на 14 % до уровня 9 %.

Изменение совокупной налоговой нагрузки возможно путем изменения ставок налогов. Причем концептуальный анализ по видам налогов остается в рамках теории Лаффера, что сохраняет методологическое единство исследования. Поэтому для качественного и количественного анализа необходимо построить по каждой группе субъектов зависимости налогооблагаемых баз конкретных налогов (добавленная стоимость, чистая прибыль, фонд заработной платы, среднегодовая стоимость имущества) от соответствующих налоговых ставок.

Для налогоплательщиков первой группы производственные и налоговые функции для каждого вида налогов приведены ниже. Производственная и налоговая функции, рассчитанные для налога на добавленную стоимость, а также экстремальные точки этих функций и текущее значение налоговой нагрузки, имеют следующий вид:

$$\begin{aligned} X_{11}(\theta) &= -107,3 \cdot 10^6 \theta^2 + 548,5 \cdot 10^5 \theta, \\ T_{11}(\theta) &= -107,3 \cdot 10^6 \theta^3 + 548,5 \cdot 10^5 \theta^2, \\ \theta^* &= 0,28; \theta^{**} = 0,35, \theta = 0,37. \end{aligned}$$

Для налога на прибыль:

$$\begin{aligned} X_{12}(\theta) &= -273,2 \cdot 10^8 \theta^2 + 715,8 \cdot 10^7 \theta, \\ T_{12}(\theta) &= -273,2 \cdot 10^8 \theta^3 + 715,8 \cdot 10^7 \theta^2, \\ \theta^* &= 0,23; \theta^{**} = 0,34, \theta = 0,15. \end{aligned}$$

Для подоходного налога:

$$\begin{aligned} X_{13}(\theta) &= -368,7 \cdot 10^6 \theta^2 + 479,3 \cdot 10^5 \theta, \\ T_{13}(\theta) &= -368,7 \cdot 10^6 \theta^3 + 479,3 \cdot 10^5 \theta^2, \\ \theta^* &= 0,07; \theta^{**} = 0,09, \theta = 0,08. \end{aligned}$$

Для налога на имущество:

$$\begin{aligned} X_{14}(\theta) &= -273,2 \cdot 10^8 \theta^2 + 715,8 \cdot 10^6 \theta, \\ T_{14}(\theta) &= -273,2 \cdot 10^8 \theta^3 + 715,8 \cdot 10^6 \theta^2, \\ \theta^* &= 0,009; \theta^{**} = 0,016, \theta = 0,02. \end{aligned}$$

Множество комбинаций ставок конкретных налогов, реализующих каждое изменение совокупной налоговой ставки, можно представить следующей матрицей:

$$A = \begin{pmatrix} \Delta\theta_1 & \Delta\theta_2 & \Delta\theta_3 & \Delta\theta_4 & \Delta\theta_5 \\ \Delta X_1 & \Delta X_2 & \Delta X_3 & \Delta X_4 & \Delta X_5 \\ \Delta T_1 & \Delta T_2 & \Delta T_3 & \Delta T_4 & \Delta T_5 \end{pmatrix},$$

где $\Delta\theta_1, \Delta\theta_2, \Delta\theta_3, \Delta\theta_4, \Delta\theta_5$ — значение, на которое изменится ставка налога на добавленную стоимость, налога на прибыль, подоходного налога, налога на имущество, ставка "прочего" налога, соответственно; ΔX_i — изменение налогооблагаемой базы конкретного налога; ΔT_i — изменение налоговых сборов конкретного налога.

Для анализируемых предприятий определено частное решение и представлено в виде матрицы **A**. Элементы этой матрицы отражают необходимые преобразования ставок конкретных налогов в виде увеличения/уменьшения, реализующих первый вариант — уменьшение совокупной налоговой ставки θ на 1 %, рост производственной функции ΔX на 14 %, а рост налоговой функции ΔT на 9 %:

$$A = \begin{pmatrix} -0,02 & 0,05 & -0,01 & -0,01 & 0 \\ 0,07 & 0,11 & 0,05 & 0,23 & 0 \\ 0,02 & 0,33 & -0,08 & -0,53 & 0 \end{pmatrix}.$$

Необходимо отметить, что представленное изменение налоговых ставок реализует лишь один из возможных вариантов налоговой политики, позволяющий повысить эффективность налоговой системы. Формирование конкурентоспособной налоговой системы с определением оптимальных налоговых ставок по всем видам налогов и разным группам ПЭС представлено в работе [9].

Заключение

Предложенный подход к моделированию и управлению ПЭС отличается тем, что учитывает ее свойства, проявляющиеся в сложных нелинейных взаимодействиях с экономическими агентами внутренней и внешней сред, что позволяет моделировать процессы функционирования ПЭС во взаимодействии с экономическими агентами, согласовывая в процессе управления их противоречивые цели и критерии.

Разработана концептуальная модель управления ПЭС, определяющая с единой системно-методологической позиции структуру системы управления ПЭС в виде трех блоков задач: формирования эффективной системы управления ПЭС, обеспечивающей управление ресурсами и согласование процессов стратегического и оперативного управления ПЭС; формирования эффективной системы управления ценообразованием ПЭС как рыночного механизма стратегического взаимодействия с конкурентами и потребителями; формирования налоговой нагрузки и налоговых ставок как инструмента бюджетно-налоговой политики государства, направленной на стимулирование развития ПЭС и экономического роста за счет рационального управления взаимодействием субъектов экономики. Концептуальная модель составляет основу методологии и позволяет повысить эффективность принимаемых решений за счет согласованного принятия решений по управлению ПЭС.

Предложенная схема системы управления функционированием ПЭС, объединяющая разнородные экономико-математические методы и модели, отражает гетерогенные свойства ПЭС и обеспечивает синтез эффективных алгоритмов управления. Разработаны инструменты поддержки принятия решений по управлению ПЭС в виде механизмов управления, методов и моделей, которые реализованы в виде интегрированного программного комплекса.

Список литературы

1. Прангшвили И. В. Системный подход и общесистемные закономерности. М.: Синтег, 2000.
2. Бурков В. Н., Губко М. В., Коргин Н. А., Новиков Д. А. Теория управления организационными системами и другие науки об управлении организациями // Проблемы управления. 2012. № 4. С. 2—10.
3. Новиков Д. А. Теория управления организационными системами. М.: Физматлит, 2012. 604 с.
4. Клейнер Г. Б. Системная экономика как платформа развития современной экономической теории // Вопросы экономики. 2013. № 6. С. 4—28.
5. Лившиц В. Н. О нестационарности российской переходной экономики // Проблемы теории и практики управления. 2014. № 2. С. 8—13.
6. Занг В. Б. Синергетическая экономика. Время и процессы в нелинейной экономической теории. М.: Мир, 1999. 335 с.

7. Евстигнеева Л. П., Евстигнеев Р. Н. Экономика как синергетическая система. М.: Ленард, 2010. 272 с.

8. Макарова Е. А. Интеллектуальная поддержка принятия управленческих решений в многосекторных макроэкономических системах с учетом рыночных отношений на основе динамических моделей: дис. ... д-ра техн. наук 05.13.10. Уфа, 2011. 389 с.

9. Орлова Е. В. Экономико-математический инструментарий управления экономической системой в условиях неопределенности. Уфа: УГАТУ, 2012. 172 с.

10. Орлова Е. В., Исмагилова Л. А. Налоговая система и реальный сектор: оптимизация интересов // Journal of Economic Regulation (Вопросы регулирования экономики). 2014. № 2. С. 133—142.

11. Орлова Е. В. Инструментарий управления эффективностью налоговой системы // Экономический анализ: теория и практика. 2013. № 43 (436). С. 59—70.

12. Орлова Е. В. Механизм эффективного ценообразования на продукцию промышленных предприятий // Экономика и предпринимательство. 2013. Т. 7. № 12-1. С. 622—626.

13. Орлова Е. В. Имитационная модель управления стохастическими финансовыми потоками предприятия // Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Экономические науки. 2012. Т. 5. № 156. С. 185—189.

14. Исмагилова Л. А., Орлова Е. В. Управление производственно-экономическими системами на принципах сбалансированной эффективности // Матер. 12-го Всерос. совещания по проблемам управления. М.: ИПУ РАН, 2014. С. 8086—8095.

15. Орлова Е. Управление эффективностью предприятия // Проблемы теории и практики управления. 2014. № 6. С. 123—129.

Mechanism, Models and Control Algorithms for Productive and Economic System under Harmonization Criteria of Interested Agents

E. V. Orlova, ekorl@mail.ru, Ufa State Aviation Technical University, Ufa, 450008, Russian Federation

Corresponding author:

Ekaterina V. Orlova, Associate Professor, Ufa State Aviation Technical University, Ufa, 450008, Russian Federation
e-mail: ekorl@mail.ru

Received on September 1, 2015

Accepted on November 2, 2015

The problem of simulation and operational control in production and economic system (PES) under condition of economy's systematic modernization considering the coordination of interests of agents of the PES, agents of market and institutional environments is considered in the article. PES is a complex organizational system that integrates the processes of production, sales, and reproduction of resources, is characterized of the properties as dynamic character and large number of parameters, strong interconnectivity, hierarchical structure, an inverse of the material and information communications, non-linear and non-stationary. A conceptual model for the creation of the PES control system, which determines a unified system-methodological position structure of the PES control system as three blocks of problems is developed and includes: the formation of an effective system of control PES, ensuring resource management and coordination of the processes of strategic and operational management of the PES; the formation of an effective system of control PES as a market pricing mechanism for strategic cooperation with competitors and consumers; the formation of the tax burden and tax rates as an instrument of fiscal policy aimed at stimulating the development of PES and economic growth at the expense of sound management of the interaction of economic agents. The conceptual model is the basis of the control methodology in PES and aims to improve the effectiveness of the decisions making at the expense of a coherent management in PES. The proposed functional scheme of the

PES control system uniting disparate economic and mathematical methods and models is proposed. The scheme reflect the heterogeneous properties of PES and provides a synthesis of efficient control algorithms.

Keywords: production and economic systems, modeling, management, coordination of interests, multiple criteria decision making

For citation:

Orlova E. V. Mechanism, Models and Control Algorithms for Productive and Economic System under Harmonization Criteria of Interested Agents, *Programmnyaya Ingeneriya*, 2016, vol. 7, no. 2, pp. 86–96.

DOI: 10.17587/prin.7.86-96

References

1. **Prangishvili I. V.** *Systemnyi podhod i obshesistemnyy zakonornosti* (The systems approach and system-wide patterns). Moscow: SINTEG, 2000 (in Russian).
2. **Burkov V. N., Gubko M. V., Korgin N. A., Novikov D. A.** *Teoriya Upravleniya organizatsionnymi sistemami i drugie nauki ob upravlenii organizatsiyami* (Theory of control of organizational systems and other management science organizations), *Problemi Upravleniya*, 2012 no. 4, pp. 2–10 (in Russian).
3. **Novikov D. A.** *Teoriya upravleniya organizatsionnymi sistemami* (Theory of control of organizational systems). Moscow: FIZMATLIT, 2012. 604 p. (in Russian).
4. **Kleiner G. B.** *Sistemnaya ekonomika kak platforma rasvitiya sovremennoy ekonomicheskoy eorii* (The system platform for the development of the economy of modern economic theory), *Voprosi Ekonomiki*, 2013, no. 6, pp. 4–28 (in Russian).
5. **Livshits V. N.** *O nestacionarnosti Rossiyskoy perehodnoy ekonomiki* (On Unsteady Russian transition economy), *Problemi Teorii i Praktiki Upravleniya*, 2014, no. 2, pp. 8–13 (in Russian).
6. **Zang W. B.** *Differential Equations, Bifurcations, and Chaos in Economics*. World Scientific Publishing Company, 2005. 512 p.
7. **Evtigneeva L. P., Evtigneev R. N.** *Ekonomika kak sinergeticheskaya sistema* (Economics as a synergetic system). Moscow: Lenard, 2010. 272 p. (in Russian).
8. **Makarova E. A.** *Intellektualnaya podderzka Prinyatiya Resheniy v Mnogosekturnih Macro-Ekonomicheskikh sistemah na base dinamicheskikh modeley* (Intellectual support management decision making in multi-sector macro-economic systems based on market relations on the basis of dynamic models), *dis. ... Dr. in Eng.* 05.13.10, Ufa, 2011. 389 p. (in Russian).
9. **Orlova E. V.** *Ekonomiko-matematicheskii instrumentariy upravleniya Ekonomicheskoy sistemoy v usloviyah neopredelennosti* (Economic-mathematical tools for management in economic system under uncertainty). Ufa: UGATU, 2012. 172 p.
10. **Orlova E. V., Ismagilova L. A.** *Nalogovayja Sistema i Realnyy Sektor Ekonomiki: Optimizatsia Interesov* (The tax system and the real sector: optimization of interests), *Journal of Economic Regulation*, 2014, no. 2, pp. 133–142 (in Russian).
11. **Orlova E. V.** *Mehanizm Efektivnogo Cenoobrasovania na Produksiju Promishlennih Predpriyariy* (Pricing mechanism for industrial enterprises), *Ekonomika i Predprinimatelstvo*, 2013, no. 12-1, pp. 622–626 (in Russian).
12. **Orlova E. V.** *Imitatsionnaya Model Upravleniya Stokhasticheskimi Finansovimi Potokami Predpriyatiya* (Simulation of Stochastic Financial Flows in Enterprise), *Nauchno-tehnicheskie Vedomosti St. Petersburgskogo Gosudarstvennogo Polytechnicheskogo Universiteta. Ekonomika*, 2012, no. 5, pp. 185–189 (in Russian).
13. **Orlova E. V.** *Instrumentariy Upravleniya Effectivnostju Nalogovoy Sistemi* (Tools for Management of Efficiency Tax System), *Ekonomicheskii analiz: Teoria i Practica*, 2013, no. 43, pp. 59–70 (in Russian).
14. **Ismagilova L. A., Orlova E. V.** *Upravlenie proizvodstvenno-ekonomicheskimi sistemami na principah sbalansirovannoy effectivnosti* (Management in production and economic system on the balanced efficiency principles), *Trudi 12 Vserossiyskogo Soveshania po Problemam Upravleniya*. Moscow: IPU RAN, 2014, pp. 8086–8095 (in Russian).
15. **Orlova E. V.** *Upravlenie effectivnostju predpriyatiya* (Enterprise performance management), *Problemi Teorii i Praktiki Upravleniya*, 2014, no. 6, pp. 123–129 (in Russian).

ООО "Издательство "Новые технологии". 107076, Москва, Стромьинский пер., 4
Технический редактор *Е. М. Патрушева*. Корректор *Е. В. Комиссарова*

Сдано в набор 03.12.2015 г. Подписано в печать 22.01.2016 г. Формат 60×88 1/8. Заказ PI216
Цена свободная.

Оригинал-макет ООО "Авансед солюшнз". Отпечатано в ООО "Авансед солюшнз".
119071, г. Москва, Ленинский пр-т, д. 19, стр. 1. Сайт: www.aov.ru