

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 24
2018
№ 1

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ
Издательство "Новые технологии"

СОДЕРЖАНИЕ

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Сорокин А. Б., Лобанов Д. А. Концептуальное проектирование интеллектуальных систем 3
- Сухов А. О., Медведева Е. Ю. Подход к разработке языкового инструментария для создания текстовых предметно-ориентированных языков 10

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Сирота А. А., Гончаров Н. И. Моделирование конфликтного взаимодействия систем с использованием формализма гибридных автоматов 17
- Буравцев А. В., Цветков В. Я. Анализ структуры сложной организационно-технической системы 27
- Резчиков А. Ф., Яндыбаева Н. В., Кушников В. А., Иващенко В. А., Богомолов А. С., Филимонок Л. Ю. Информационно-советующая система для моделирования и прогнозирования динамики показателей национальной безопасности на основе причинно-следственных комплексов и уравнений Форрестера 33

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

- Разумовский А. И. Информационная избыточность в разработке программных систем 41

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

- Загидуллин Р. Р., Занин Д. С. Определение интегральной оценки профессиональной квалификации студента вуза на различных этапах обучения . . . 49
- Чирцов А. С., Козуненко Д. А. Использование технологий создания стереоскопических изображений при численном моделировании сложных физических систем для лекционных демонстраций и учебных исследований 55

ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

- Ле Тхань Тунг Нгуен, Кравец А. Г., Нгок Зыонг Буй. Анализ средств и моделей взаимодействия между компонентами в системе управления корпоративной мобильностью 64

Главный редактор:

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

Зам. главного редактора:

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

Редакционный совет:

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

Редакционная коллегия:

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

Редакция:

БЕЗМЕНОВА М. Ю.
ГРИГОРИН-РЯБОВА Е. В.
ЛЫСЕНКО А. В.
ЧУГУНОВА А. В.

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

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

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

INFORMATION TECHNOLOGIES INFORMACIONNYYE TEHNOLOGII

Vol. 24
2018
No. 1

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

ISSN 1684-6400

CONTENTS

INTELLIGENT SYSTEMS AND TECHNOLOGIES

- Sorokin A. B., Lobanov D. A.** Conceptual Design of Intelligent Systems 3
- Sukhov A. O., Medvedeva E. Yu.** An Approach to Language Workbench Development
for Creating Textual Domain-Specific Languages 10

MODELING AND OPTIMIZATION

- Sirota A. A., Goncharov N. I.** Modeling of Conflict Interaction of Systems Using the
Formalism of Hybrid Automata 17
- Buravtsev A. V., Tsvetkov V. Ya.** Analysis of the Structure of a Complex Organiza-
tional and Technical System 27
- Rezhnikov A. F., Yandybaeva N. V., Kushnikov V. A., Ivashenko V. A., Bogomolov A. S.,
Filimonyuk L. Y.** Information and Advisory System for Modeling and Forecasting
the Dinamik of National Security Indicators on the Basis of Cause-Effect Complexes
and Forrester Equations 33

SOFTWARE ENGINEERING

- Razumowsky A. I.** Information Redundancy in the Development of Software Systems . . . 41

INFORMATION TECHNOLOGIES IN EDUCATION

- Zagidullin R. R., Zanin D. S.** Determination of Integral Assessment of Professional
Qualifications University Student at Different Stages Training 49
- Chirtsov A. S., Kozunen D. A.** The Use of Technology of Stereoscopic Images
Creation in Numerical Modeling of Complex Physical Systems for Lecture Demon-
strations and Educational Researches 55

APPLIED INFORMATION TECHNOLOGIES

- Le Thanh Tung Nguyen, Kravets A. G., Ngoc Duong Bui.** Analysis of Interaction Method
and Models Between Components of Enterprise Mobility Management System 64

Editor-in-Chief:

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

Deputy Editor-in-Chief:

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

Chairman:

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

Editorial Board Members:

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

Editors:

Bezmenova M. Yu.
Grigorin-Ryabova E. V.
Lysenko A. V.
Chugunova A. V.

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

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

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

INTELLIGENT SYSTEMS AND TECHNOLOGIES

УДК 004.023/004.825

А. Б. Сорокин, канд. техн. наук, доц., e-mail: ab_sorokin@mail.ru,
Д. А. Лобанов, магистрант, e-mail: tty66@mail.ru,
Московский технологический университет (МИРЭА)

Концептуальное проектирование интеллектуальных систем

Рассмотрена методика выделения концептуальных планов из структуры акта деятельности для проектирования систем, основанных на знаниях. На графическом уровне предложены четыре плановых представления: функциональной структуры, процессов, контекста и закономерностей. Данные планы соотносятся с определенными методами искусственного интеллекта. Реализован программный комплекс для выделения этих представлений в виде текстовых файлов.

Ключевые слова: планы концептуальной структуры акта деятельности, план функциональной структуры, план процессов, план контекста, план закономерностей

Введение

При возрастании сложности процессов управления человеческий разум не в состоянии познать многомерность проблемы, что приводит к упрощению реальности человеком и способствует принятию неадекватных решений. Наиболее упрощенной картиной реального мира является эмоциональное мышление, которое поддается всякого рода манипуляциям, ориентирует на краткосрочный результат и приводит к "легким" решениям, не требующим доказательств и логического обоснования.

Проектирование систем, основанных на знаниях (систем искусственного интеллекта — далее СИИ) и, соответственно, их комбинаций должно основываться на постоянном концептуальном осмыслении управленческих процессов. При этом центральным компонентом таких систем является база знаний экспертов, которая может быть извлечена из концептуальной структуры акта деятельности. Таким образом, реализуется экспертная система продукционного типа [1].

Элементы экспертных систем не только используются практически в каждом программном продукте, но и тесно связаны с другими подходами искусственного интеллекта. Примером могут служить интегрированные экспертные системы [2]. Поскольку такое сближение приводит к идее выделения из концептуальной структуры акта деятельности других представлений для проектирования СИИ, данную статью можно назвать продолжением работы [1].

1. Постановка задачи для выделения концептуальных планов

Общеизвестно, что в CASE-технологии (*Computer-Aided Software Engineering* — автоматизированная разработка программного обеспечения) используют два принципиально разных подхода к проектированию: структурный, основанный на общелогическом методе познания — анализе контекстного представления (контекстной диаграммы), и объектно-ориентированный, основанный на синтезе различных диаграмм для создания UML-модели (*Unified Modeling Language* — унифицированный язык моделирования).

Рассмотрим подход к проектированию СИИ, основанный на методе дедукции, как логический прием для выделения нового знания (планы акта деятельности) из ранее полученных знаний (концептуальная структура акта деятельности) [см. 3, с. 139].

Предложено выделять динамические и статические представления концептуальной структуры акта деятельности. Динамические представления определены двумя планами, относительно которых имеются следующие утверждения.

Утверждение 1. Функциональный план позволяет разрабатывать дискретно-событийные модели.

Утверждение 2. План процессов позволяет разрабатывать диаграмму потоков и уровней для имитационно-динамических моделей.

Статические представления также определены двумя планами, обусловленными следующими утверждениями.

Утверждение 3. План закономерностей служит своего рода интерпретатором графического представления в аналитическое представление в виде простейших математических уравнений.

Утверждение 4. План контекста позволяет разрабатывать различные модели, основанные на идеях когнитивного подхода.

Под концептуальным планом понимается определенная часть концептуальной структуры акта деятельности, которая рассматривается как проект для разработки моделей, основанных на знаниях. Задача реализации концептуальных планов на графическом уровне требует не только понимания методов искусственного интеллекта, но также больших временных затрат. Поэтому предлагается создать программное обеспечение "Интерпретатор" для выделения планов в виде текстовых файлов.

2. Функциональный план для проектирования дискретно-событийных моделей

Дискретно-событийная модель рассматривается как глобальная схема обслуживания заявок. Аналитические результаты таких моделей используются в теории массового обслуживания, как переход системы из одного состояния в другое дискретным образом в виде события [4].

Для проектирования сложных дискретно-событийных систем используют SADT-диаграммы (*Structured Analysis and Design Technique* — метод структурного анализа и проектирования), которые дают промежуточное представление, помогающее разработчику избежать ошибок при проектировании [5, с. 91].

Методология SADT представляет собой совокупность методов, правил и процедур, предназначенных для построения функциональной структуры сложных иерархических систем в виде блочной модели [6, с. 19]. Основным концептуальным

принцип методологии SADT — представление любой изучаемой системы в виде набора взаимодействующих и взаимосвязанных функциональных блоков (рис. 1).

В рамках методологии все, что происходит в системе и ее элементах, принято называть функциями, которые соответствуют определенному блоку. Функция F представлена процессом или действием Xa в зависимости от рассматриваемого уровня иерархии, и объектом Xao , на который воздействует данная активность.

$$F = \langle Xa, Xao \rangle. \quad (1)$$

При этом каждая сторона функционального блока (рис. 1) имеет стандартное значение [6, с. 16]: входная и выходная стрелки отображают данные или материальные объекты, управляющие стрелки — условия, влияющие на выполнение функции, а стрелки механизма — средства для исполнения работы. По аналогии функциональный план акта деятельности должен быть представлен: входными характеристиками (Vx), которые определены исходным материалом; выходными характеристиками (Vyx) — продуктом акта деятельности (далее продукт); управлением — требованием к продукту акта деятельности; механизмами — средствами акта деятельности [7, с. 20] (рис. 2).

Согласно графической нотации акта деятельности, изложенной в работе [1], функция образует структуру из двух возможных типов связующих звеньев: Lac (КД) — связь между компонентами процесса и процессом; Lp (ОС) — связь между элементами структуры и их свойствами (характеристиками) P .

Одним из важных моментов при разбиении является точная согласованность типов связей между функциями. Введем типизацию связей между функциями.

- Пересечение по исходному материалу, который является входной характеристикой нескольких функций. Функции группируются вследствие того, что они используют одни и те же входные данные.
- Пересечение по продукту и исходному материалу. Выход одной функции служит входными данными для следующей функции с меньшим доминированием. При последовательной связности имеет место обратная связь, когда выход функции становится входом другой функции с большим доминированием. Данная связь используется для описания циклов.
- Пересечение по продукту и свойству одного из компонентов. Выход одной функции служит свойством компоненты для другой менее доминирующей функции. При этом функции направлены на достижение одной цели. Поэтому реализуется полная зависимость одной функции от другой.



Рис. 1. Контекстная диаграмма SADT-модели

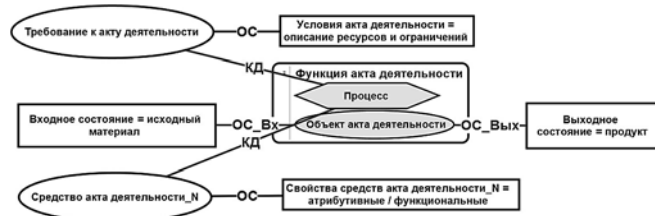


Рис. 2. Функциональный план акта деятельности

План функциональной структуры акта деятельности описывает функцию верхнего уровня, которая редуцирована на частные планы функциональных структур единичных решений. Функциональный план можно использовать не только для разработки дискретно-событийных моделей, но и для проектирования архитектуры гибридных интеллектуальных систем.

3. План процессов для проектирования имитационно-динамических моделей

Методология построения имитационно-динамического моделирования включает качественную и количественную стадии. Базовая структура количественной стадии содержит четыре существенных элемента: уровень (накопитель), функция решения, материальные и информационные потоки [8, с. 212] (рис. 3). Уровни представлены количественными показателями исследуемого объекта и характеризуют возникающие накопления внутри системы, они представляют собой те значения переменных в данный момент, которые они имеют в результате накопления вследствие разности между входящими и исходящими потоками. Содержимое накопителей может иметь любую природу. Материальный поток, вливаясь в уровень или вытекая из него, определяет изменение уровня. Кроме того, различают информационные потоки, с помощью которых принимается решение по управляющим воздействиям, т. е. определяется значение темпа потока на следующий интервал времени (функция решений) [9, с. 94].

Для реализации базовой структуры количественной стадии предлагается использовать процессный план акта деятельности, который обу-

словлен: процессом как совокупностью действий; объектом акта деятельности; его состояниями — исходными материалами или продуктами; средствами достижения цели и их свойствами (рис. 4).

Согласно графической нотации акта деятельности в процессный план добавлена связь *Lao* (ОД) между объектом процесса *Xao* и процессом *Xa* [1].

Процессный план акта деятельности описывает процессы верхнего уровня, которые могут быть редуцированы на частные планы единичных решений. При этом одним из важных моментов при разбиении должна быть точная согласованность типов связей между действиями. Введем типизацию связей между процессными планами структур единичных решений.

- Тип последовательной связности — исходное действие должно полностью завершиться, прежде чем начнется выполнение другого действия. Данный тип связи моделирует причинно-следственные зависимости трансформации исходного материала.
- Тип последовательной связности с обратной связью — реализует цикл процесса.
- Тип логической связности. Возможна ситуация, когда завершение одного действия может инициировать начало выполнения сразу нескольких других действий. Данный тип связи моделирует логические зависимости.
- Тип обратной логической связности. Определенное действие может требовать завершения нескольких других действий для начала своего выполнения.

Процессный план оперирует множеством действий, определяется, в какое количественное состояние перейдет в будущем объект из заданного текущего состояния. Тогда уровень накопителя представлен как объект действия *Xao*, определенный через связь *Lp* (ОС) одним из количественных состояний — исходный материал или продукт. Поток определен непрерывной активностью *Xa*, которая представлена действием, перемещающим содержимое между накопителями. Используя гидродинамическую метафору, отметим, что функция решений есть вентиль, управляющий потоком, который обусловлен средствами акта деятельности $\{Xac_1, Xac_2, \dots, Xac_N\}$ и связью *Lp* (ОС) с его функциональными свойствами *P*.

Функции решений есть уравнение темпов, отражающее понимание факторов, определяющих действия, которые будут совершаться непосредственно в следующий момент времени. Закон изменения темпа задается функциональной зависимостью [9, с. 94]:

$$V(t) = F(p_1(t), p_2(t), \dots, p_k(t)), \quad (2)$$

где $V(t)$ — входящий поток (исток) или выходящий поток (сток); F — произвольная функция от



Рис. 3. Базовая структура количественной стадии



Рис. 4. Процессный план акта деятельности

числа аргументов k ; $p_i(t)$ — параметры модели, значения которых в момент t известны.

Таким образом, функция решений не является столь очевидной и простой. Поэтому она должна быть определена другой деятельностью и планом зависимости. Также качественная стадия будет рассмотрена в рамках плана контекста как исследования структуры причинно-следственных связей.

На основе плана процессов разрабатывается диаграмма потоков и уровней для имитационных динамических моделей [10].

4. План закономерностей для проектирования аналитических моделей

Статические аспекты описывают концептуальную структуру акта деятельности в определенный момент времени (до или после действия). При этом отношения и соотношения являются основными связующими "строительными блоками" таких концептуальных структур.

Аналитические модели представляют собой уравнения или системы уравнений, записанные в виде алгебраических, интегральных, дифференциальных, конечно-разностных и иных соотношений и логических условий. Такая модель, как правило, имеет статическое представление, которое подходит лишь для очень простых и сильно идеализированных задач и объектов.

План закономерностей определяется множественностью проявляемых свойств объектов концептуальной структуры, которые характеризуются определенными отношениями ($R_{oc} = 1$, $R_{cc} = 1$) и соотношениями (R_p), тем самым фиксируются закономерности преобразования исходного материала в продукт.

При построении плана закономерностей единичного решения необходимо учитывать следующие правила.

- Аналитическая закономерность может быть выявлена объектной частью концептуальной структуры единичного решения. При этом компонент "Требование к объекту действия" в плане закономерностей не учитывается (рис. 5).
- Состояние объекта действия (продукт) ассоциируется с решением, а средства действий — с внутренними параметрами уравнения.
- Отношения в плане закономерностей однопавлены и единичны, т. е. из одного элемента может выходить или входить в него одно отношение.
- Отношение между объектом и орудием действия R_{oc} и отношения между средствами R_{cc} представлены отношениями зависимостей и обусловлены типами "увеличивает" или "уменьшает", отношение R_{oc} — типом "определяет" (см. таблицу).
- Соотношения являются ассоциациями отношений зависимостей, которые определены сле-



Рис. 5. План закономерностей единичного решения

Ассоциативные правила

Тип отношения	Тип соотношения	Тип арифметического действия
Увеличивает	Больше на Больше в	Сложение Умножение
Уменьшает	Меньше на Меньше в	Вычитание Деление

дующими типами: "больше на", "меньше на", "больше в" и "меньше в". Соотношения ассоциаций, в свою очередь, могут быть отображены как арифметические действия (см. таблицу). Отношение R_{oc} , обусловленное типом "определяет", отображается как знак равенства.

На основе плана закономерностей могут определяться различные аналитические представления, в том числе функции решений (интенсивность потока) для имитационно-динамической модели [10].

5. План контекста для проектирования когнитивных моделей

Когнитивное моделирование ориентировано на помощь управленцу, оно помогает фиксировать свое представление о проблемной ситуации в виде формальной модели. В качестве такой модели обычно используется так называемая когнитивная карта ситуации, которая представляет известные субъекту основные законы и закономерности наблюдаемой ситуации в виде ориентированного знакового графа, в котором вершины графа — это факторы (признаки, характеристики ситуации), а дуги между факторами — причинно-следственные связи между факторами [11, с. 464] (рис. 6).

В когнитивной карте выделяют два типа причинно-следственных связей: положительные (+) и отрицательные (−). При положительной связи увеличение значения фактора-причины приводит к увеличению значения фактора-следствия, а при отрицательной связи увеличение значения фактора-причины приводит к уменьшению значения фактора-следствия.

При этом под понятием "фактор" понимается движущая сила, причина какого-нибудь процесса, определяющая его характер или отдельные его черты. Таким образом, когнитивная структура предметной области — это выявление

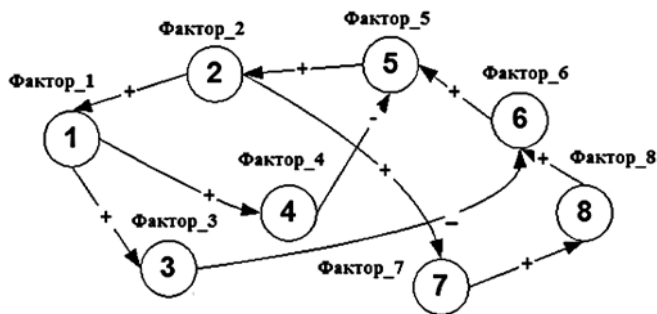


Рис. 6. Пример когнитивной карты

будущих целевых и нежелательных состояний объекта управления и наиболее существенных (базисных) факторов управления и внешней среды, влияющих на переход объекта в эти состояния, а также установление на качественном уровне причинно-следственных связей между ними, с учетом взаимовлияния факторов друг на друга [11, с. 465].

Таким образом, в плане контекста должны быть определены следующие элементы: субъект акта деятельности Xas , который взаимодействует с объектом действия Rso и различными компонентами действий $\{Rsc_1, Rsc_2, \dots, Rsc_N\}$; объект акта деятельности Xao , который существует во взаимодействии с субъектом Ros и с различными отношениями-компонентами $\{Roc_1, Roc_2, \dots, Roc_N\}$; компоненты действия в виде требований и средств акта деятельности $\{Xac_1, Xac_2, \dots, Xac_N\}$, которые взаимодействуют с субъектом $\{Rcs_1, Rcs_2, \dots, Rcs_N\}$ и существуют в отношениях между собой Rcc_{jN} . При этом необходимо учитывать двойную направленность взаимодействий (Rcs или Rsc) и отношений (Roc или Rco) (рис. 7).

В данном случае под термином "контекст" понимается структура внешней среды, в пределах которой выявляются факторы управления. Статистическое представление плана контекста определено взаимодействиями и отношениями, равными 1, и типами "увеличивает" или "уменьшает".

На уровне концептуальных структур единичных решений возможны следующие виды пересечений.

- По субъекту акта деятельности. Когда субъект взаимодействует с различными объектами действия как в одном акте деятельности,



Рис. 7. Контекстный план акта деятельности

так и в разных актах. При этом связь Las (СД) между субъектом действия Xas и действием Xa обрывается.

- По объекту акта деятельности. Когда один и тот же объект в одном акте деятельности представлен как "Продукт_1", а в другом — как "Продукт_2".
- По одному из компонентов действия (требованию или средствам акта деятельности). Когда один и тот же компонент используется в различных актах деятельности.

Соответственно, возможны различные сочетания данных пересечений. Например, по субъекту и средствам, по субъекту и объектам акта деятельности и т. д.

План контекста предназначен не только для проектирования когнитивных моделей, но для лучшего осмысления определенной деятельности, а также как дополнение к остальным концептуальным планам, например, для определения качественной стадии в имитационном динамическом моделировании.

6. Программная реализация концептуальных планов

Концептуальные планы по своей сути являются базой знаний для проектирования интеллектуальных систем [12]. Соответственно, задача выделения таких знаний из концептуальной структуры акта деятельности может быть решена на программном уровне.

Для разработки программного обеспечения (ПО) "Интерпретатор" использовался фреймворк Qt 5.3, в котором предоставлен класс `QXmlStreamReader`. Данный класс представляет собой быстрый синтаксически корректный XML (*eXtensible Markup Language* — расширяемый язык разметки) — анализатор с простым потоковым интерфейсом прикладного программирования. Базовая концепция потокового чтения состоит в представлении XML-документа в виде потока маркеров и позволяет создавать рекурсивные спускаемые анализаторы [13, с. 660]. Это означает, что можно разделить код анализа XML на различные классы.

Таким образом, происходит считывание структуры дерева и классификация его объектов по связям (СД, ОД, КД и ОС). После анализа XML-документа и формирования начального набора данных осуществляется вызов функции поиска объектов. Выполнение этих действий приводит к структуризации данных, необходимых для поиска пересечений, и выделению необходимых плановых представлений. На основе найденных пересечений выводятся необходимые отчеты о концептуальных планах в HTML-файле (*HyperText Markup Language* — язык гипертекстовой разметки).

На основе данного алгоритма разрабатываются четыре модуля программы "Интерпретатор", которые разделяются по двум блокам: динамических и статических знаний (рис. 8).

В отчете динамической базы знаний функционального плана определяются следующие элементы: функция — [*Действие_N Объект_N*]; входное свойство функции — [*Свойство объекта_N до действия_N*]; выходное свойство функции — [*Свойство объекта_N после действия_N*]; механизмы функции — [*Средства действия_N*]; инструкции по управлению функцией — [*Требования к акту деятельности_N*]. В HTML-файле функционального плана указываются выявленные пересечения и предлагаются рекомендации по его построению в виде различных конструктов. Например: Вход <*Свойство объекта_N до действия_N*> → Функция <*Действие_N Объект_N*> → Выход <*Свойство объекта_N после действия_N*>.

В отчете основными элементами базы знаний процессного плана [14] являются: уровень — [*Объект_N Количество: = <Свойство объекта до действия>*] и [*Объект_N Количество: = <Свойство объекта после действия>*]; поток — [*Действие_N*]. В HTML-файле процессного плана указываются выявленные пересечения и предлагаются рекомендации по его построению в виде следующего конструкта: Уровень_1 <*Объект_N Количество: = <Свойство объекта_N до действия_N>*> → поток <*Действие_N*> → Уровень_1 <*Объект_N Количество: = <Свойство объекта_N после действия_N>*>.

В отчете основными элементами базы знаний плана закономерностей являются: уравнение — [*Объект_N*]; параметр — [*Объект_N Свойство средства_N после действия_N*]; математический знак — [*Соотношение*]; знак равенства — [*Отношение <Определяет>*]. В HTML-файле плана закономерностей предлагаются рекомендации по его построению в виде следующего конструкта: Уравнение <*Объект_N*> Знак равенства <*Отношение 'Определяет'*> Параметр <*Объект_1 Свойство средства_1 после действия_1*> Математический знак <*Соотношение*> Параметр <*Объект_2 Свойство средства_2 после действия_2*> и т. д.

В отчете основными элементами базы знаний контекстного плана являются: факт взаимодействия — [*Субъект Взаимодействие Объект_N*], [*Субъект Взаимодействие Средства_N*] и [*Субъект Взаимодействие Требования_N*]; факт отношения — [*Средство_N-1 Отношение Средства_N*], [*Требование_N-1 Отношение Требования_N*] и [*Средство_N Отношение Требования_N*]; факт появления продукта — [*Объект_N Свойство объекта_N после действия_N*]. В HTML-файле контекстного плана указываются выявленные пересечения, и установленные факты являются рекомендациями к разработке когнитивных моделей.

Таким образом, ПО "Интерпретатор" расширяет возможности программного комплекса (ПК) "Оформитель + Решатель" [1]. Появляется новая составляющая данного ПК, которая обуславливает иную организацию совокупностей программ.

Создается ПК "Оформитель + Решатель + Интерпретатор", который позиционируется как интеллектуальная надстройка для проектирования автономных интеллектуальных моделей (рис. 9).

Использование программного комплекса определено следующим алгоритмом:

- после определения актов деятельности [1] в программе "Оформитель" реализуются их концептуальные структуры единичных решений, которые определены как целостная концептуальная модель принятия решений;

- в программном комплексе "Решатель" целостная концептуальная модель принятия решений проверяется на полноту и адекватность (при необходимости генерируется отчет о базе знаний в виде продукционных правил);

- после подтверждения полноты и адекватности модели в ПО "Интерпретатор" генерируются отчеты о базах знаний концептуальных планов; синтез баз знаний соответствует определенным моделям и системам искусственного интеллекта.



Рис. 8. Структурная схема программного обеспечения "Интерпретатор"



Рис. 9. Программный комплекс "Оформитель + Решатель + Интерпретатор"

Заключение

Практика использования интеллектуальных систем и их синергетических комбинаций показывает, что в настоящее время в ее теоретическом обосновании не существует:

— методики и концептуальной структуры, объединяющей ситуационные, имитационные, экспертные, эволюционные и другие подходы к моделированию СИИ;

— целостной модели знаний и единого языка графического описания для моделирования в области разработки СИИ;

— программного обеспечения представления баз знаний для проектирования СИИ.

Эти обстоятельства указывают на существование в области поддержки принятия решений проблем, состоящих в отсутствии единой концептуальной структуры обоснованных решений относительно управления сложными системами и в программном извлечении различных знаний из концептуальной модели для проектирования СИИ. Предложенный в работе метод и программный инструментарий позволяют эффективно решить данные проблемы.

Список литературы

1. Сорокин А. Б., Смольянинова В. А. Концептуальное проектирование экспертных систем // Информационные технологии. 2017. Том. 23, № 9 (23). С. 634—641.
2. Рыбина Г. В. Интегрированные экспертные системы: некоторые итоги и перспективы // Искусственный интеллект и принятие решений. 2008. № 1. С. 22 — 46.

3. **Философский** энциклопедический словарь / Гл. ред.: Л. Ф. Ильичев, П. Н. Федосеев, С. М. Ковалев, В. Г. Панов. М.: Сов. Энциклопедия, 1983. 840 с.

4. Боев В. Д., Сыпченко Р. П. Компьютерное моделирование. М.: ИНТУИТ.РУ, 2010. 349 с.

5. Емельянов В. В., Ясиновский С. И. Введение в интеллектуальное имитационное моделирование сложных дискретных систем и процессов. Язык РДО. М.: АНВИК, 1998. 427 с.

6. Марка Д., МакГоуэн К. Методология структурного анализа и проектирования SADT. М.: МетаТехнология, 1993. 240 с.

7. Сорокин А. Б. Полиаспектный анализ при проектировании систем поддержки принятия решений // Научно-техническая информация. Серия 2. Информационные процессы и системы. 2014. № 8. С. 10—23.

8. Теория систем и системный анализ в управлении организациями: Справочник: Учеб. пособие / Под ред. В. Н. Волковой и А. А. Емельянова. М.: Финансы и статистика, 2006. 848 с.

9. Форрестер Дж. Основы кибернетики предприятия. М.: Прогресс, 1971. 340 с.

10. Sorokin A. B., Bolotova L. S. The evolutionary model as the projection methodology situationally — activity analysis and its realization on the example of the model against the development of infectious diseases // Collection of scientific papers "Interactive systems: Problems of Human-Computer Interaction". Ulyanovsk, 2015. P. 120—130.

11. Болотова Л. С. Системы искусственного интеллекта: модели и технологии, основанные на знаниях. М.: Финансы и статистика, 2012. 663 с.

12. Сорокин А. Б. Свидетельство о государственной регистрации базы данных № 2015621317 "Интеллектуальная база данных для проектирования систем основанных на знаниях". Дата государственной регистрации в Реестре баз данных 26 августа 2015 г.

13. Шлее М. Qt 4.8. Профессиональное программирование на C++. СПб.: БХВ-Петербург, 2012. 912 с.

14. Сорокин А. Б., Лобанов Д. А. Свидетельство о государственной регистрации программы для ЭВМ № 2015662075 "Интерпретатор — процессный план". Дата государственной регистрации в Реестре программ для ЭВМ 17 ноября 2015 г.

A. B. Sorokin, Ph. D., Associate Professor, e-mail: ab_sorokin@mail.ru,

D. A. Lobanov, Master Student, e-mail: tty66@mail.ru, Moscow Technological University (MIREA)

Conceptual Design of Intelligent Systems

A methodology for distinguishing conceptual plans from the structure of an activity act for the design of intelligent systems is considered. In conceptual plan is understood as a certain part by conceptual structure of the act of activities which is considered as the project for development of the models based on knowledge. At the graphical level, four planned views are proposed: the functional structure, processes, context and regularities. The functional plan is designed for the design of discrete-event models and architectures of hybrid intelligent systems. The process plan is designed to develop a flow diagram and levels for system dynamics simulation modeling. On the basis of the plan of regularities different analytical representations, including functions of decisions (intensity of a flow) for an imitative dynamic model can be defined. The plan of a context is intended not only for design of cognitive models, but also to better comprehend certain activities and as an addition to the rest of the conceptual plans. However the graphic construction of the conceptual model of a dynamically complex environment and its particular representations is a complex and nontrivial task. Therefore, the program complex "Designer + Solver + Interpreter" is developed, which verifies the conceptual model of a dynamically complex environment on completeness and adequacy, and also generates the necessary knowledge base for the design of intelligent models in the form of text files.

Keywords: plans of conceptual structure of the act of activities, functional structure plan, process plan, plan of a context, plan of regularities

References

1. Sorokin A. B. Smol'janinova V. A. Konceptual'noe proektirovanie jekspertnyh sistem, *Informacionnye tehnologii*, 2017, vol. 23, no. 9, pp. 634—641 (in Russian).

2. Rybina G. V. Integrirovannye jekspertnye sistemy: nekotorye itogi i perspektivy, *Iskusstvennyj intellekt i prinjatие reshenij*, 2008, no. 1, pp. 22—46 (in Russian).

3. **Философский** jenciklopedicheskij slovar' / L. F. Il'ichev, P. N. Fedoseev, S. M. Kovalev, V. G. Panov. Moscow, Sov. Jenciklopedija, 1983, 840 p. (in Russian).

4. Boev V. D., Sytchenko R. P. Komp'yuternoe modelirovanie. Moscow, INTUIT.RU, 2010. 349 p. (in Russian).

5. Emel'janov V. V., Jasinovskij S. I. Vvedenie v intellektual'noe imitacionnoe modelirovanie slozhnykh diskretnykh sistem i processov. Jazyk RDO, Moscow, ANVIK, 1998, 427 p. (in Russian).

6. Marka D., MakGoujen K. Metodologija strukturnogo analiza i proektirovanija SADT, Moscow, MetaTehnologija, 1993, 240 p. (in Russian).

7. Sorokin A. B. Poliaspektnyj analiz pri proektirovanii sistem podderzhki prinjatija reshenij, Nauchno-tehnicheskaja informacija. Serija 2. Informacionnye processy i sistemy, 2014, no. 8, pp. 10–23 (in Russian).

8. Teorija sistem i sistemnyj analiz v upravlenii organizacijami: Spravochnik: Ucheb. posobie / Eds. V. N. Volkovoj, A. A. Emel'janova. Moscow, Finansy i statistika, 2006, 848 p. (in Russian).

9. Forrester Dzh. Osnovy kibernetiki predpriyatija, Moscow, Progress, 1971, 340 p. (in Russian).

10. Sorokin A. B., Bolotova L. S. The evolutionary model as the projection methodology situationally — activity analysis and its

realization on the example of the model against the development of infectious diseases, *Collection of scientific papers "Interactive systems: Problems of Human-Computer Interaction"*, Ulyanovsk, 2015. pp. 120–130.

11. Bolotova L. S. Sistemy iskusstvennogo intellekta: modeli i tehnologii, osnovannye na znaniyah, Moscow, Finansy i statistika, 2012, 663 p. (in Russian).

12. Sorokin A. B. Svidetel'stvo o gosudarstvennoj registracii bazy dannyh № 2015621317 "Intellektual'naja baza dannyh dlja proektirovanija sistem osnovannyh na znaniyah". Data gosudarstvennoj registracii v Reestre baz dannyh 26 avgusta 2015 goda.

13. Shlee M. Qt 4.8. Professional'noe programmirovanie na C++, Saint-Petersburg, BHV-Peterburg, 2012, 912 p. (in Russian).

14. Sorokin A. B., Lobanov D. A. Svidetel'stvo o gosudarstvennoj registracii programmy dlja JeVM № 2015662075 "Interpretator — processnyj plan". Data gosudarstvennoj registracii v Reestre programm dlja JeVM 17 nojabrja 2015 goda.

УДК 004.434

А. О. Сухов, канд. физ.-мат. наук, доц. каф., e-mail: ASuhov@hse.ru,

Е. Ю. Медведева, студент, e-mail: medvedevaeyu@mail.ru,

Национальный исследовательский университет "Высшая школа экономики", г. Пермь

Подход к разработке языкового инструментария для создания текстовых предметно-ориентированных языков

Описан подход к созданию языкового инструментария, новизна которого заключается в том, что в процессе функционирования языковой инструментальной системы выполняется интерпретация моделей различных уровней иерархии, а не генерация на их основе исходного кода. Это позволяет проводить настройку разработанных языков без регенерации кода редактора, выполнять многоуровневое моделирование, а также определять правила преобразования написанных программ в код на целевом языке.

Ключевые слова: предметно-ориентированные языки, текстовые языки, языковой инструментарий, мета-язык, многоуровневое моделирование, метамодель, трансформация моделей, модельно-ориентированный подход

Введение

Одной из движущих сил развития языков программирования и инструментальных средств разработки программного обеспечения является уменьшение трудовых затрат программистов за счет сокращения семантического разрыва между конструкциями языков программирования и объектами предметной области, которые требуется описать с помощью этих языков. Переход от машинных языков к языкам высокого уровня позволил существенно сократить трудозатраты и время разработки, а также повысить качество создаваемых систем. С начала 1970-х годов прошлого века стали активно развиваться проблемно- и предметно-ориентированные языки программирования, которые предназначены для решения определенного класса задач в конкретной предметной области. Эти языки позволили существенно сократить трудозатраты при раз-

работке приложений, поскольку одной команде такого языка соответствуют десятки строк кода, написанных на языке высокого уровня [1–4]. Примерами предметно-ориентированных языков являются Prolog, VHDL, HTML, GPSS и др. С течением времени такие языки стали все чаще применяться при разработке программного обеспечения. На сегодняшний день разработка любой информационной системы немыслима без использования языка запросов SQL, языка проектирования интерфейса пользователя, интегрированного в IDE (Integrated Development Environment).

Предметно-ориентированные языки (Domain-Specific Languages, DSL) — языки программирования (моделирования), предназначенные для решения определенного круга задач в конкретной предметной области.

Поскольку такие языки оперируют терминами предметной области, при разработке и сопро-

вождении программных систем использовать их могут не только программисты, но и эксперты в предметной области, бизнес-аналитики, конечные пользователи. DSL могут иметь визуальный или текстовый синтаксис [5]. *Визуальные предметно-ориентированные языки*, как и любые графические схемы и модели, более наглядны, чем текстовые DSL. Это делает их понятными и выразительными: одна конструкция языка способна заменить несколько строк кода. *Текстовые предметно-ориентированные языки* позволяют программисту более точно и подробно выразить проектные решения, так как являются более формализованными.

При создании программных систем могут использоваться ранее разработанные предметно-ориентированные языки, однако не для всех предметных областей и задач подходят существующие решения, поэтому появляется необходимость создания новых текстовых DSL, что является довольно трудоемким процессом, поскольку помимо самого языка необходимо разработать текстовый редактор для работы с ним, генератор и отладчик кода.

Языковой инструментарий — это вид программного обеспечения, предназначенный для создания и сопровождения предметно-ориентированных языков. Он позволяет упростить процесс разработки DSL за счет автоматизированного построения среды разработки, которая включает редактор кода с подсветкой синтаксиса, браузер проектов, трансформатор для преобразования созданных на DSL программ в код на целевом языке программирования и отладчик кода.

Целью исследования является разработка подхода к созданию языкового инструментария, который позволит выполнять построение текстовых предметно-ориентированных языков, написание программ с помощью этих языков и их преобразование в код на целевом языке.

Языковой инструментарий должен быть прост в использовании и понятен различным категориям пользователей, поскольку участие представителей заказчика является неотъемлемой частью современного процесса разработки программного обеспечения.

Анализ языковых инструментариев

В процессе разработки и сопровождения программного обеспечения часто возникает необходимость внесения изменений в ранее созданное решение. Причины этих изменений могут быть различны: в процессе анализа или проектирования были допущены неточности, произошло изменение предметной области, заказчик модифицировал требования к разрабатываемой системе и др. Все это может привести к необходимости внесения изменений в описание предметно-ори-

ентированного языка, которые повлекут за собой соответствующие модификации в редакторе языка. Для того чтобы существовала возможность внесения изменений в DSL без регенерации исходного кода редактора, необходимо, чтобы языковой инструментарий в процессе своего функционирования проводил интерпретацию описания языка, а не генерацию на его основе исходного кода анализаторов, как это делает большинство генераторов лексических и синтаксических анализаторов Lex, YACC, ANTRL и др.

Для описания синтаксиса DSL в языковых инструментариях используется метаязык. Большинство инструментариев предоставляет пользователю заранее предопределенный неизменяемый набор конструкций, "вшитый" в реализацию, что накладывает ряд ограничений на возможность гибкой настройки языкового инструментария и разрабатываемых DSL на предметную область. Возможность изменения описания метаязыка делает языковой инструментарий более гибким и позволит пользователю самостоятельно настраивать метаязык, применяемый для построения предметно-ориентированных языков. Кроме того, такая возможность позволит выполнять многоуровневое моделирование. Определив предметно-ориентированный язык, можно загрузить его в языковой инструментарий и использовать в качестве метаязыка. Это предоставит возможность создавать предметно-ориентированные метаязыки, что еще в большей степени адаптирует систему к особенностям конкретной предметной области.

Для преобразования написанных с помощью DSL программ в код на целевом языке программирования необходимо предоставить пользователю удобные средства описания такой трансформации. Данные средства должны позволять определять правила преобразования конструкций DSL в целевой язык, поскольку для каждого DSL и целевого языка эти правила будут уникальными.

При всем при этом языковой инструментарий не должен быть перегружен лишней функциональностью, которая не востребована при разработке и эксплуатации предметно-ориентированных языков, так как это может затруднить процесс использования языкового инструментария пользователями, которые не являются профессиональными IT-специалистами.

Существуют различные языковые инструментарии для создания текстовых DSL: OpenArchitectureWare, Meta Programming System, IDE Meta-Tooling Platform, MontiCore, Spoofox Language Workbench и др. Выполним сравнение наиболее развитых из них по следующим критериям:

- возможность создания DSL для широкого спектра предметных областей;
- возможность многоуровневого моделирования;
- возможность изменения описания DSL без регенерации кода его редактора;

- возможность определения правил трансформации конструкций DSL в целевой язык;
- наличие избыточной функциональности языкового инструментария, не используемой при создании DSL.

Eclipse OpenArchitectureWare (oAW) — это инструментарий с открытым исходным кодом для реализации модельно-ориентированной разработки приложений, основанный на платформе Eclipse [6].

Для описания синтаксиса разрабатываемых DSL используется компонент xText [7], поэтому возможность изменения метаязыка и многоуровневого моделирования отсутствует. На основе построенной грамматики выполняется генерация лексического и синтаксического анализаторов языка. При каждом изменении описания языка требуется повторная генерация кода его редактора.

Для описания трансформаций создаваемых программ в целевое представление используется язык xPanda. Генерация кода проводится только на языке Java.

Система *Meta Programming System* (MPS) разработана компанией JetBrains и используется совместно со средой разработки Java-приложений IntelliJ-IDEA [8, 9].

Для описания DSL данный языковой инструментарий предоставляет в распоряжение пользователя отдельные языки: язык структуры, язык редактора, базовый язык, язык шаблонов, каждый из которых применяется для определения отдельного аспекта DSL.

Абстрактный синтаксис языка описывается с помощью встроенного метаязыка, ключевым элементом которого является концепт. Концепт в MPS — это тип узла абстрактного синтаксического дерева, содержащий информацию о свойствах, методах, потомках и отношениях узла. Возможность многоуровневого моделирования в инструментарии отсутствует.

Язык редактора позволяет описать конкретный синтаксис DSL. Далее с помощью языка шаблонов следует определить правила преобразования конструкций создаваемого языка в код на языке Java. После этого требуется выполнить генерацию кода редактора языка. Для внесения изменений в язык необходимо повторно выполнить все этапы описания языка и регенерировать код редактора, т. е. возможность динамического изменения описания языка отсутствует.

Языковой инструментарий создан на базе среды разработки IntelliJ-IDEA, которая предназначена для решения более широкого круга задач, чем создание DSL, поэтому языковой инструментарий обладает излишней функциональностью, не используемой в процессе разработки DSL.

IDE Meta-Tooling Platform (IMP) — это плагин к среде Eclipse, разработанный компанией IBM в целях упрощения процесса создания IDE для новых языков программирования [10].

При использовании данного плагина необходимо определить только конкретный синтаксис языка, представляющий собой контекстно-свободную грамматику, а абстрактный синтаксис строится автоматически с помощью генератора синтаксического анализатора LPG. Причем изменить средства описания конкретного синтаксиса невозможно; как следствие, отсутствует возможность многоуровневого моделирования.

Описание правил трансформации моделей в исходный код на целевом языке выполняется на языке Java. После внесения изменений в описание языка следует выполнить повторную генерацию кода редактора DSL.

Поскольку данный языковой инструментарий является плагином к среде Eclipse, то его использование затруднено дополнительной функциональностью данной IDE.

MontiCore (MC) — языковой инструментарий, базирующийся на платформе Eclipse [11–13].

Данный программный продукт для определения DSL использует метаязык для описания конкретного и абстрактного синтаксиса, который подобен входному формату системы ANTLR. Возможности изменения описания метаязыка и многоуровневого моделирования отсутствуют.

По окончании описания языка необходимо выполнить компиляцию проекта, после чего будет запущен новый экземпляр Eclipse, в котором в качестве инструмента будет выступать разработанный DSL, поэтому возможность внесения изменений в описание предметно-ориентированного языка без регенерации исходного кода редактора отсутствует.

На основе построенной с помощью DSL модели может быть сгенерирован код на языке Java, изменение правил генерации кода невозможно.

Инструментарий MC является плагином к среде Eclipse, поэтому он перегружен широкими возможностями данной IDE.

Spoofax Language Workbench (SLW) — языковой инструментарий, который позволяет итеративно разрабатывать текстовые предметно-ориентированные языки и выполнять запуск написанных с их помощью программ в экземпляре среды Eclipse [14–16].

SLW предлагает пользователю среду разработки DSL, включающую в себя компоненты описания и анализа синтаксиса, трансформации и генерации объектного кода.

Для описания синтаксиса языка SLW использует собственный метаязык SDF3, который объединяет конкретный и абстрактный синтаксисы в едином шаблоне.

Для выполнения преобразований моделей в текстовое представление на целевом языке применяются язык Stratego, который позволяет определить текстовые шаблоны для генерации кода.

Результаты сравнения языковых инструментариев

Критерий сравнения	oAW	MPS	IMP	MC	SLW
Возможность создания DSL для широкого спектра предметных областей	+	+	+	+	+
Возможность многоуровневого моделирования	—	—	—	—	—
Возможность изменения описания DSL без регенерации кода его редактора	—	—	—	—	—
Возможность определения правил трансформации	+	+	+	—	+
Наличие избыточной функциональности	+	+	+	+	+

К ограничениям системы следует отнести отсутствие возможности многоуровневого моделирования и перегруженность среды разработки Eclipse функциями, которые не используются при создании DSL.

В таблице приведены сводные сравнительные характеристики рассмотренных языковых инструментариев.

Подводя итог, можно говорить о том, что все рассмотренные языковые инструментарии являются лишь надстройкой над более мощной средой программирования (Eclipse или IntelliJ-IDEA). Это приводит к тому, что данные инструментарии сложны для освоения из-за наличия избыточной функциональности. Кроме того, "отчуждение" созданных DSL от среды разработки для их дальнейшего использования при разработке и сопровождении программного обеспечения становится невозможным.

Ни один из языковых инструментариев не предоставляет возможности многоуровневого моделирования, что усложняет их настройку на специфику предметной области. Также ни один инструментарий не позволяет динамически изменять описание языка без повторной генерации исходного кода редактора; это затрудняет процесс модификации языка на этапе эксплуатации, так как для этих целей требуется иметь IDE.

Большинство рассмотренных языковых инструментариев позволяют выполнять преобразование созданных моделей только в код на языке Java, что также существенно ограничивает возможность их применения.

Существующие проблемы описания и использования текстовых предметно-ориентированных языков, ограничения языковых инструментариев и подходов к созданию DSL было решено устранить при разработке языкового инструментария, удовлетворяющего следующим требованиям:

- универсальность — возможность построения языков для широкого спектра предметных областей;
- наличие возможности многоуровневого моделирования;
- наличие возможности внесения изменений в описание языков без регенерации кода редакторов DSL;
- единообразие средств представления, описания и использования моделей различных уровней иерархии: построение моделей и работа с ними должны проводиться унифицировано с помощью одних и тех же программных средств;
- наличие возможности задания правил трансформации программ, написанных с помощью DSL, в код на целевом языке;
- удобство работы с системой для различных категорий пользователей.

Построение текстовых DSL

На основе требований, предъявляемых к языковому инструментарию, разработаем подход к созданию текстовых предметно-ориентированных языков.

Процесс создания DSL начинается с построения его *метамодели* (задания абстрактного синтаксиса), для описания которой используется *метаязык*. Именно благодаря наличию метаязыка языковой инструментарий позволяет разрабатывать DSL для широкого спектра предметных областей.

После построения метамодели пользователь получает в распоряжение язык, с помощью которого может выполнять разработку программы — описание *моделей предметной области*, содержащей основные объекты, связи между ними и операции над объектами.

Во время исполнения программы на основе модели предметной области строится *модель состояния*, объекты которой — экземпляры элементов модели предметной области и связей между ними.

Таким образом, во время разработки DSL создается целая иерархия моделей, представленная на рис. 1.

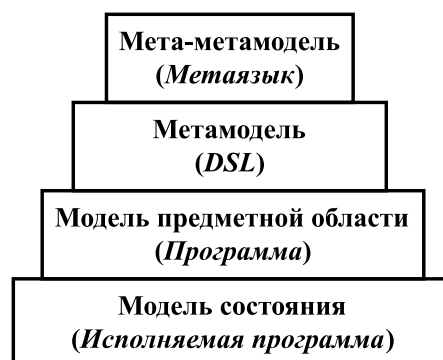


Рис. 1. Иерархия моделей при разработке DSL

Построение иерархии моделей и унифицированная обработка моделей различных уровней иерархии позволяют выполнять многоуровневое моделирование, которое предоставляет возможность в большой степени настроить разрабатываемый язык на специфику конкретной предметной области. Предположим, что необходимо разработать текстовый предметно-ориентированный язык для описания обслуживания на автозаправочной станции. Для этого в качестве метаязыка могут быть использованы формы Бэкуса — Наура (БНФ). Однако применять такой метаязык смогут только IT-специалисты. Если требуется предоставить конечному пользователю возможность самостоятельно определять, изменять DSL, то необходимо, чтобы метаязык был ему понятен. Для этого с помощью БНФ может быть разработан текстовый предметно-ориентированный язык для систем массового обслуживания, далее он должен быть загружен в языковой инструмент и использован в качестве предметно-ориентированного метаязыка при создании DSL для описания работы автозаправочной станции.

По окончании создания программы (модели предметной области) необходимо, чтобы ее код, написанный на DSL, был преобразован в код на целевом языке. Правила трансформации будут задаваться с помощью текстовых шаблонов, содержащих статическую и динамическую части. Статическая часть правила трансформации не зависит от модели и является неизменной. Динамическая часть содержит изменяемые фрагменты кода, которые зависят от значения атрибутов элементов модели.

Метаязык языкового инструментария

Метаязык языкового инструментария должен быть лаконичным, понятным различным категориям пользователей, а также позволять описывать метамодели для различных предметных областей, не ограничивая уровня детализации.

Для описания синтаксиса языка, с одной стороны, можно использовать стандартные метаязыки, например, расширенные формы Бэкуса — Наура (РБНФ), а с другой стороны, можно сделать систему более гибкой и настраиваемой за счет предоставления пользователю возможности самостоятельно модифицировать конструкции метаязыка. Однако в любом случае следует задать набор базовых терминальных метасимволов и обозначить правила построения метамodelей.

За основу формального описания синтаксиса текстовых DSL были взяты РБНФ. При разработке языка в описание его синтаксиса автоматически включаются predefined нетерминалы "буква", "цифра", "идентификатор", которые используются практически в любой предметной области.

Для возможности изменения встроенного метаязыка необходимо, чтобы языковой инструмент в процессе своего функционирования выполнял интерпретацию созданных моделей, а не генерацию на их основе программного редактора.

Архитектура языкового инструментария

В процесс разработки DSL могут быть вовлечены различные категории пользователей (рис. 2). IT-специалисты совместно с бизнес-аналитиками выполняют настройку метаязыка на предметную область (в случае необходимости), создают текстовые языки, определяют правила преобразования конструкций языка в код на целевом языке. Бизнес-аналитики совместно с конечными пользователями выполняют построение моделей с использованием разработанного DSL и генерацию кода на целевом языке.

Для построения текстовых предметно-ориентированных языков необходимо предоставить пользователю удобную среду разработки, в которой он мог бы создавать новые языки, строить с их помощью модели и генерировать код на целевом языке. Архитектура языкового инструментария представлена на рис. 3.

Единым хранилищем информации обо всех проектах является *репозиторий*. Он хранит как отдельные метамодели (предметно-ориентированные языки), так и готовые решения (DSL и созданные на его основе программы). Также вместе с языком хранятся правила трансформации программ в целевые языки.

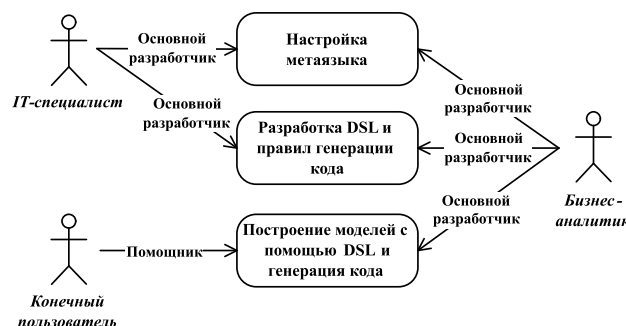


Рис. 2. Пользователи языкового инструментария

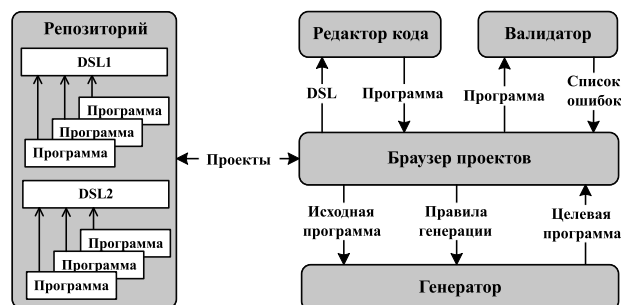


Рис. 3. Архитектура языкового инструментария

Браузер проектов предоставляет пользователю возможность загружать из репозитория ранее сохраненные проекты, добавлять, удалять, редактировать метамодели, созданные на их основе модели, определять правила трансформации моделей.

Редактирование моделей различных уровней иерархии осуществляется в *текстовом редакторе кода*, который выполняет подсветку синтаксиса, автодополнение конструкций языка в соответствии с заданными правилами, а для метамодели — еще и проверку корректности описания языка (все ли нетерминалы описаны, нет ли повторного описания нетерминала и др.).

Создание программы на основе построенного DSL контролируется *валидатором*, который преобразует метамодель в дерево синтаксического разбора и выполняет проверку модели на лексическую и синтаксическую корректность.

Преобразование созданной модели предметной области в код на целевом языке программирования осуществляется *генератором*, который на основе заданных в браузере проектов правил трансформации и исходной программы выполняет построение кода на целевом языке.

Благодаря тому, что хранение и обработка моделей различных уровней иерархии выполняются единообразно, появляется возможность выполнять многоуровневое моделирование, проводить объединение нескольких моделей, созданных на основе общей метамодели.

Заключение

Благодаря тому, что языковой инструментарий в процессе своего функционирования выполняет интерпретацию описания моделей различных уровней иерархии, существует возможность внесения изменений в описание DSL во время работы системы, без регенерации исходного кода редактора языка. Это делает языковой инструментарий более гибким и настраиваемым на изменившиеся условия эксплуатации без необходимости привлечения профессиональных IT-специалистов.

Унификация обработки моделей и метамоделей предоставляет возможность выполнять мультиязыковое программирование на основе объединения нескольких метамоделей в одну.

Возможность изменения описания метаязыка и многоуровневого моделирования позволяет приблизить метаязык к специфике предметной области.

Предоставление пользователю возможности определять правила трансформации позволяет выполнять генерацию кода на различных целевых языках, что также делает систему более гибкой.

Благодаря тому, что языковой инструментарий будет разрабатываться "с нуля", а не как надстройка над существующей IDE, он не будет загроможден излишней функциональностью, кото-

рая не используется в процессе разработки DSL. Это сделает систему понятной и простой в применении для различных категорий пользователей.

Теоретическая значимость работы заключается в том, что описанный подход к разработке языкового инструментария для создания текстовых динамически настраиваемых предметно-ориентированных языков позволяет устранить недостатки большинства существующих инструментов.

Практическая значимость подхода заключается в том, что он позволяет разработать языковой инструментарий для создания DSL, построения программ с использованием этих языков и генерации кода на целевом языке на основе определенных пользователем правил трансформации.

Список литературы

1. Кирсанова А. А. Проблематика использования текстовых DSL в информационных системах // Вестник ЮУрГУ. Сер. "Компьютерные технологии, управление, радиоэлектроника". 2015. Т. 15, № 3. С. 5—10.
2. Globalizing Domain-Specific Languages // Lecture Notes in Computer Science. Berlin: Springer. 2015. Vol. 9400. 88 p.
3. Drobintsev P. D. Model-Oriented Approach for Industrial Software Development // Modeling and Analysis of Information Systems. 2015. Vol. 22, № 6. P. 750—762.
4. Rumpe B. Agile Development with Domain Specific Languages // Modelling Foundations and Applications. Lecture Notes in Computer Science. Berlin: Springer. 2011. Vol. 6698. P. 387—388.
5. Сухов А. О. Классификация предметно-ориентированных языков и языковых инструментариев // Математика программных систем: межвузовский сборник научных статей. Пермь: Изд-во Пермского государственного национального исследовательского университета. 2012. Вып. 9. С. 74—83.
6. Haase A. Introduction to OpenArchitectureWare 4.1.2 // Model-Driven Development Tool Implementers Forum 2007. URL: <https://pdfs.semanticscholar.org/a6fa/d330ed634810f3f002658443ab361eb3c423.pdf> (дата обращения: 05.06.2017).
7. Жуков М. В. Реализация предметно-ориентированных языков средствами фреймворка Xtext // Изв. Пензенского государственного педагогического университета имени В. Г. Белинского. Сер. "Физико-математические и технические науки". 2009. № 13. С. 98—102.
8. Pech V., Shatalin A., Voelter M. JetBrains MPS as a Tool for Extending Java // Proceedings of the 2013 International Conference on Principles and Practices of Programming on the Java Platform: Virtual Machines, Languages, and Tools. NY. 2013. P. 165—168.
9. Voelter M. Lessons Learned from Developing Mbeddr: a Case Study in Language Engineering with MPS // Software & Systems Modeling. 2017. P. 1—46.
10. Charles Ph., Fuhrer R. M., Sutton S. M. IMP: a Meta-Tooling Platform for Creating Language-Specific IDEs in Eclipse // Proceedings of the Twenty-Second IEEE/ACM International Conference on Automated Software Engineering. NY. 2007. P. 485—488.
11. Krahn H., Rumpe B., Völkel S. MontiCore: a Framework for Compositional Development of Domain Specific Languages // International Journal on Software Tools for Technology Transfer. 2010. Vol. 12. Iss. 5. P. 353—372.
12. Krahn H., Rumpe B., Völkel S. MontiCore: Modular Development of Textual Domain Specific Languages // Objects, Components, Models and Patterns. Lecture Notes in Business Information Processing. Berlin: Springer. 2008. Vol. 11. P. 297—315.
13. Heim R. Compositional Language Engineering Using Generated, Extensible, Static Type-Safe Visitors // Modelling Foundations and Applications. Lecture Notes in Computer Science. Berlin: Springer. 2016. Vol. 9764. P. 67—82.

14. **Kats L. C. L., Visser E.** The Spoofox Language Workbench: Rules for Declarative Specification of Languages and IDEs // Proceedings of the Conference "Object-Oriented Programming, Systems, Languages and Applications" (OOPSLA 2010). 2010. P. 444–463.
15. **Vos B., Kats L. C. L., Pronk C.** EpiSpin: An Eclipse Plug-In for Promela/Spin Using Spoofox // Model Checking Software. Lecture Notes in Computer Science. Berlin: Springer. 2011. Vol. 6823. P. 177–182.

ware. Lecture Notes in Computer Science. Berlin: Springer. 2011. Vol. 6823. P. 177–182.

16. **Wachsmuth G. H., Konat G. D., Visser E.** Language Design with the Spoofox Language Workbench // IEEE Software. 2014. Vol. 31. P. 35–43.

A. O. Sukhov, Associate Professor, e-mail: ASuhov@hse.ru,

E. Yu. Medvedeva, Student, e-mail: medvedevaeyu@mail.ru,

National Research University Higher School of Economics, Perm, 614070, Russian Federation

An Approach to Language Workbench Development for Creating Textual Domain-Specific Languages

The usage of domain-specific languages at the software development process can reduce labor costs, because one command of this language corresponds to dozens of code lines written in a high-level language. However, at DSL creation it is also necessary to develop a convenient environment for working with it that includes a text editor, code generator and debugger. Language workbench is software designed to create and maintain domain-specific languages.

The paper describes an approach to the development of language workbench for creating of textual dynamically customizable domain-specific languages. The approach novelty is that in the process of functioning the language workbench interprets the models of different levels of hierarchy, rather than generates on their basis of the editor source code. This allows to configure the developed language for changed operating conditions without re-generation of the editor code, to perform multi-level modeling, and to determine the rules for conversion of written programs into code in the target language.

As the metalanguage for DSL construction, it is proposed to use the extended Backus-Naur forms. The architecture of the language workbench includes a repository, text code editor, project browser, validator, and generator.

Different categories of users can be involved at the DSL development process: IT-specialists, business analysts, end-users, etc.

Keywords: domain-specific languages, textual languages, language workbench, architecture, metalanguage, multi-level modeling, metamodel, model transformation, model-oriented approach, language tuning

References

1. **Kirsanova A. A.** Problematika ispolzovanija tekstovykh DSL v informacionnykh sistemakh (Problems of Using Textual DSL in Information Systems), *Vestnik JuUrGU. Seriya "Kompjuternye tehnologii, upravlenie, radioelektronika"*, 2015, vol. 15, no. 3, pp. 5–10 (in Russian).
2. **Globalizing Domain-Specific Languages**, *Lecture Notes in Computer Science*. Berlin: Springer, 2015, vol. 9400, 88 p.
3. **Drobintsev P. D.** Model-Oriented Approach for Industrial Software Development, *Modeling and Analysis of Information Systems*, 2015, vol. 22, no 6, pp. 750–762.
4. **Rumpe B.** Agile Development with Domain Specific Languages, *Modelling Foundations and Applications. Lecture Notes in Computer Science*. Berlin: Springer, 2011, vol. 6698, pp. 387–388.
5. **Suhov A. O.** Klassifikacija predmetno-orientirovannykh jazykov i jazykovykh instrumentarijev (Domain-specific Languages and Language Workbench Classification), *Matematika programnykh sistem: mezhvuzovskij sbornik nauchnykh statej*, Perm: Izdatelstvo Permskogo gosudarstvennogo nacionalnogo issledovatel'skogo universiteta, 2012, no. 9, pp. 74–83 (in Russian).
6. **Haase A.** Introduction to OpenArchitectureWare 4.1.2, Model-Driven Development Tool Implementers Forum — 2007. URL: <https://pdfs.semanticscholar.org/a6fa/d330ed634810f3f002658443ab361eb3c423.pdf> (Access at: 05.06.2017).
7. **Zhukov M. V.** Realizacija predmetno-orientirovannykh jazykov sredstvami frejmworka Xtext (The Realization of Object-oriented Languages by Means of Framework Xtext), *Izvestija Penzenskogo gosudarstvennogo pedagogicheskogo universiteta imeni V. G. Belinskogo. Seriya "Fiziko-matematicheskie i tehnicheckie nauki"*, 2009, no. 13, pp. 98–102 (in Russian).
8. **Pech V., Shatalin A., Voelter M.** JetBrains MPS as a Tool for Extending Java, *Proceedings of the 2013 International Conference*

on Principles and Practices of Programming on the Java Platform: Virtual Machines, Languages, and Tools. NY, 2013, pp. 165–168.

9. **Voelter M.** Lessons Learned from Developing Mbeddr: a Case Study in Language Engineering with MPS, *Software & Systems Modeling*, 2017, pp. 1–46.

10. **Charles Ph., Fuhrer R. M., Sutton S. M.** IMP: a Meta-Tooling Platform for Creating Language-Specific IDEs in Eclipse, *Proceedings of the Twenty-Second IEEE/ACM International Conference on Automated Software Engineering*. NY, 2007, pp. 485–488.

11. **Krahn H., Rumpe B., Völkel S.** MontiCore: a Framework for Compositional Development of Domain Specific Languages, *International Journal on Software Tools for Technology Transfer*, 2010, vol. 12, iss. 5, pp. 353–372.

12. **Krahn H., Rumpe B., Völkel S.** MontiCore: Modular Development of Textual Domain Specific Languages, *Objects, Components, Models and Patterns. Lecture Notes in Business Information Processing*. Berlin: Springer, 2008, vol. 11, pp. 297–315.

13. **Heim R.** Compositional Language Engineering Using Generated, Extensible, Static Type-Safe Visitors, *Modelling Foundations and Applications. Lecture Notes in Computer Science*. Berlin: Springer, 2016, vol. 9764, pp. 67–82.

14. **Kats L. C. L., Visser E.** The Spoofox Language Workbench: Rules for Declarative Specification of Languages and IDEs, *Proceedings of the Conference "Object-Oriented Programming, Systems, Languages and Applications" (OOPSLA 2010)*. Nevada, 2010, pp. 444–463.

15. **Vos B., Kats L. C. L., Pronk C.** EpiSpin: An Eclipse Plug-In for Promela/Spin Using Spoofox, *Model Checking Software. Lecture Notes in Computer Science*. Berlin: Springer, 2011, vol. 6823, pp. 177–182.

16. **Wachsmuth G. H., Konat G. D., Visser E.** Language Design with the Spoofox Language Workbench, *IEEE Software*, 2014, vol. 31, pp. 35–43.

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 004.942

А. А. Сирота, д-р техн. наук, проф., зав. каф., e-mail: sir@cs.vsu.ru,
Н. И. Гончаров, аспирант, e-mail: nikigoncharov@ya.ru,
Воронежский государственный университет

Моделирование конфликтного взаимодействия систем с использованием формализма гибридных автоматов

Рассмотрены математические и компьютерные модели конфликтного взаимодействия систем на основе формализма гибридных автоматов (карт состояний). Предложены типовые математические схемы и реализующие их имитационные модели, основанные на описании смены состояний в условиях конфликта. Получены аналитические соотношения для оценки нижней границы вероятности выигрыша одной из сторон в конфликте, позволяющие абстрагироваться от конкретного вида плотностей распределений для времени пребывания сторон в своих возможных состояниях. Приведены результаты сравнения полученных оценок с результатами имитационного моделирования.

Ключевые слова: конфликтное взаимодействие систем, гибридные автоматы, карты состояний, моделирование конфликта коалиций систем

Введение

В современных условиях функционирование сложных систем характеризуется наличием различного рода конфликтных взаимодействий. Основные идеи современной теории конфликта [1–11] состоят в построении концептуальной модели, связывающей действующие объекты и факторы и направленной на установление закономерностей рационального поведения сторон в условиях конфликта. Для этого в подобной модели всегда в той или иной форме реализуется представление конфликта в виде графа, описывающего набор возможных состояний систем и допустимые переходы между ними. В известных работах рассматриваются различные подходы к математическому моделированию конфликта систем: на основе аппарата сетей Петри [6], теории игр [7], теории активных систем [8], вероятностных сетей [9], теории динамических систем [10] и др.

Плодотворным подходом для построения математической и компьютерной модели конфликта является использование аппарата полумарковских случайных процессов (ПСП), естественным образом интегрируемого с графовыми представлениями состояний систем [2–5, 11–13]. Тем не менее при построении моделей на основе ПСП действует принципиальное ограничение: необ-

ходимо в явном виде задавать плотности распределения вероятностей для времени нахождения систем в своих состояниях. Часто эти плотности являются неизвестными, что приводит к необходимости проведения анализа для нескольких вариантов распределений, которые также до конца не обоснованы с физической точки зрения [12]. Еще одним ограничением подхода ПСП к моделированию конфликта является сложность его применения для коалиций систем, поскольку это неминуемо ведет к возрастанию числа рассматриваемых состояний и делает затруднительным аналитическое описание.

Естественным выходом в этой ситуации представляется использование средств компьютерного имитационного моделирования (ИМ), основанных на использовании иерархических объектных представлений, обеспечивающих описание конфликта в его естественном виде и возможность простой модификации моделей при изменении структуры систем и состава их коалиций [14]. Такие возможности, на наш взгляд, предоставляет формализм гибридных автоматов (ГА) и его многочисленные реализации в современных средствах ИМ. Их главной особенностью является представление каждого объекта — участника конфликта в виде диаграммы (карты) состояний, отражающих поведение этого объекта. Причем для

описания процессов здесь необязательно вводить свойство полумарковости или использовать статистическое представление. Смена состояний или взаимодействие с другими объектами в ГА осуществляется путем генерации и передачи событий (сообщений), фиксирующих достижение того или иного результата. Такой подход обеспечивает более широкие возможности создания объектно-ориентированных моделей конфликта и, прежде всего, при участии с каждой стороны нескольких объектов (конфликт коалиций), а также введение в модели элементов управления, реализующих те или иные стратегии. При реализации моделей конфликта на основе ПСП требуется вводить так называемые конфликтно-обусловленные состояния, общие для всех объектов системы, что, как уже упоминалось выше, вызывает трудности реализационного характера при разрастании числа состояний.

Целью данной работы является обоснование типовых математических схем конфликтного взаимодействия систем и реализующих их компьютерных имитационных моделей, основанных на использовании формализма ГА. В рамках предложенных схем математического моделирования решается задача получения соотношений для оценки нижней границы вероятности выигрыша сторон в конфликте, исследование которых позволяет определить область их применения и обосновать необходимость перехода к ИМ в конкретных случаях.

1. Математическое описание конфликта на основе гибридных автоматов

Будем придерживаться стандартного представления процесса функционирования системы как процесса смены состояний, в каждом из которых система может находиться в течение определенного времени, затрачиваемого на выполнение некоторой "работы" (подпроцесса). Переходы из одного состояния в другое рассматриваются как события, которые происходят мгновенно (за бесконечно малое время) и фиксируют скачкообразное изменение состояний, т. е. завершение одной работы, сопровождающееся изменением "статуса" объекта, и переход к выполнению другой. Согласно [15–17] гибридным автоматом (ГА) называется кортеж

$$H = \langle T, S^D, S^C, SC, I_0, D_x, F, Pr\ ed, A\ lg \rangle. \quad (1)$$

В выражении (1) используются следующие обозначения: T — множество моментов времени; S^D — множество дискретных переменных, представляющих режимы функционирования ГА со значениями, принадлежащими Q , где Q — мно-

жество символов, определяющих возможные дискретные состояния (режимы работы ГА); $S^C = \{x_k, k = \overline{1, n}\}$ — множество вещественных переменных $x = (x_1, \dots, x_n)^T \in R^n$, являющихся аргументами или значениями непрерывных функций, описывающих локальное поведение ГА для каждого дискретного состояния; $SC = \{Q, E\}$ — граф с множествами узлов, представляющих режимы работы ГА $Q = \{q_i, i = \overline{1, N}\}$, и ориентированных дуг $E = \{(q_i, q_j), i, j = \overline{1, N}\}$, определяющих возможные переходы; $I_0 \subseteq Q \times T \times R^n$ — множество начальных состояний для инициализации локального поведения в узлах; $D_x \subseteq Q \times T \times R^n$ — множество, определяющее допустимую область изменения непрерывных переменных S^C для каждого режима; $F : T \times Q \times R^n \rightarrow R^n$ — функция дискретных и непрерывных переменных, определяющая характер изменения непрерывных переменных во времени и описывающая локальное поведение при нахождении в каждом дискретном состоянии; $Pr\ ed = \{pr_i\}$ — множество предикатов, каждый из которых есть отображение $pr_i : T \times Q \times R^n \rightarrow \{false, true\}$, определяющее условия перехода к новому локальному поведению; $A\ lg = \{a\ lg_j\}$ — множество алгоритмов, каждый из которых есть отображение вида $a\ lg_j : Q \times R^n \rightarrow Q \times R^n$, и которые реализуются мгновенно и трактуются как действия, необходимые для подготовки нового локального поведения.

Под состоянием ГА понимают пару $\langle q, x \rangle$, которая состоит из значения дискретной переменной $q \in Q$ и точки $x \in D_x(q)$, являющейся значением непрерывной переменной, описывающей локальное поведение. Иногда удобно представить непрерывное состояние, объединив в единый вектор время и другие параметры. Переходы между узлами графа $SC = \{Q, E\}$ определяются приписыванием дугам предикатов и алгоритмов. При описании пары $\langle pr_i(e), a\ lg_j(e) \rangle$ используется символическое представление [15]:

$$Jump(e) = \alpha[\beta]/\gamma, \quad (2)$$

где α — внешнее событие, порождающее переход; $[\beta]$ — "сторожевое" условие, определяющее возможность выполнения перехода; γ — действие или событие, сопровождающее выполнение перехода, включающее подготовку начального состояния для непрерывного локального поведения в новом дискретном состоянии. При задании переходов в (2) необязательно использовать все компоненты. Так, часто удобно выполнить начальную инициализацию непосредственно после завершения перехода при входе в новое состояние.

Моделирование ГА сводится к вычислению последовательности пар, отражающих смену дискретных состояний и интервалы времени пребы-

вания в каждом состоянии при выполнении соответствующей работы (локального поведения):

$$\begin{aligned} \langle q_k, \tau_k \rangle, k = 1, 2, \dots; q_k \in Q, \tau_k = t_k - t_{k-1}, \\ t_k = t : I(x(t) \in Z_x), \tau_k = t_k - t_{k-1}, \\ x(t) = f_k(x(t_{k-1}), x_{t_{k-1}}^t, t), t \in T, \end{aligned} \quad (3)$$

где $t_{k-1} = t_{imp}$ — момент времени очередного входа (возможно, неоднократного) в дискретное состояние q_k ; $t_k = t_{out}$ — момент выхода из дискретного состояния q_k , фиксируемый индикатором достижения вектором непрерывных параметров $I(x(t) \in Z_x)$ заданного подмножества значений Z_x и фактически определяющий условие выполнения перехода [β] в представлении (2); $f_k(x(t_{k-1}), x_{t_{k-1}}^t, t)$ — оператор, задающий локальное поведение в состоянии q_k ; $x_{t_{k-1}}^t$ — траектория, описывающая предшествующие значения вектора x , определяющая локальное поведение на интервале (t_{k-1}, t) , включая стохастические воздействия, описывающие любые возмущения и неопределенности внешнего и внутреннего характера. При моделировании локального поведения могут быть реализованы две стратегии [16—18]:

- детальное моделирование выполняемой в каждом дискретном состоянии работы q_k на основе задания оператора локального поведения (3) конкретного вида, что позволяет зафиксировать значение времени $t_k = t_{out}$ завершения локального поведения, а также достигнутый при выполнении работы результат;
- событийное моделирование на основе метода статистических эквивалентов, при реализации которого в соответствии с заданным законом распределения $p_k(\tau)$ генерируется интервал времени τ_k выполнения работы в дискретном состоянии q_k и событие, отражающее факт успешного или неуспешного завершения работы с вероятностью P_k .

Далее будем ориентироваться на второй способ моделирования, принципиально не ограничивая возможность перехода к детальному представлению. Следует отметить, что допускается использование сложных иерархических ГА, что подразумевает задание внутри дискретных состояний ГА вложенных подавтоматов. Также нет никаких ограничений для использования моделей с параллельно действующими несколькими ГА [16, 17].

1.1. Математическое моделирование конфликта систем. При использовании ГА для моделирования конфликта систем важно провести типизацию основных схем конфликтных взаимодействий. Первой из предлагаемых схем, которую целесообразно использовать как базовую для построения более сложных моделей, является представленная на рис. 1 модель конфликтного взаимодействия двух близких по функциональной структуре систем с антагонистическими интересами (модель "дуэли"). В ней, как в модели ГА, параллельно действуют два внутренних ГА: автомат A и автомат B . Для них множество $S^D = \{s_a, s_b\}$ представлено двумя дискретными переменными, каждая из которых принимает два значения $s_a \in Q_A = \{L_A, D_A\}$, $s_b \in Q_B = \{L_B, D_B\}$, определяющими состояние "жизни" и "смерти" A и B . Соответственно, формально $Q = Q_A \times Q_B$. Удобно также ввести множества $L = \{L_A, L_B\}$, $D = \{D_A, D_B\}$. Переход в D_A и D_B происходит под влиянием событий $attack_B$ и $attack_A$, формируемых при нахождении в состояниях L_B и L_A соответственно. Выигрыш одной из сторон в конфликте при ее нахождении в состоянии при выходе из L означает переход другой стороны в состояние из D . Состояния из D в данной модели являются поглощающими. Стрелки с черным кружком обозначают начальные состояния L_A, L_B .

Для детального описания поведения сторон предлагается ввести внутреннее описание состояний из множества $L = \{L_A, L_B\}$ в виде вложенных

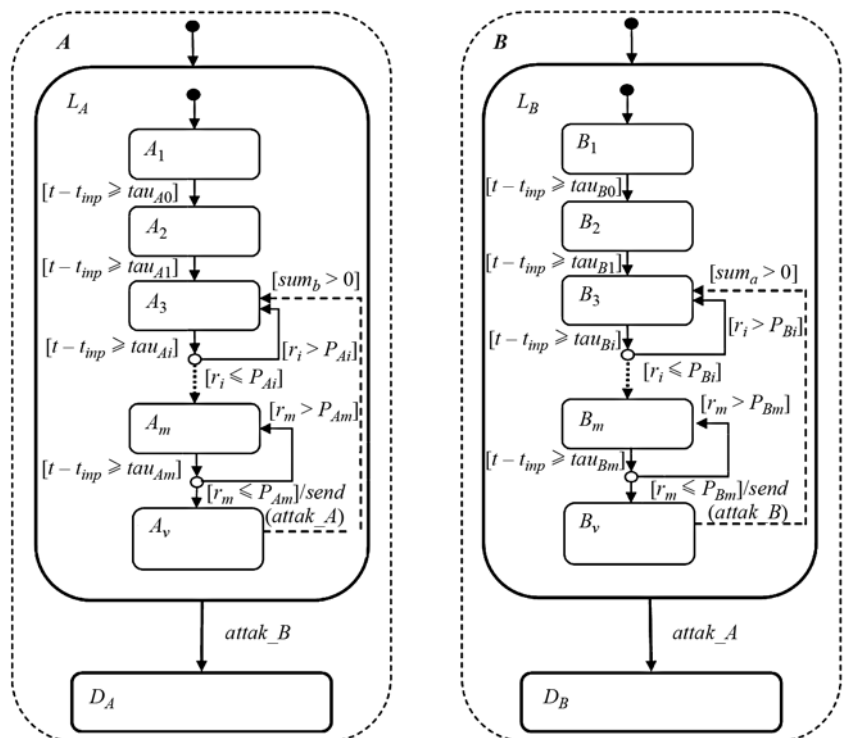


Рис. 1. Представление конфликта систем с использованием ГА

гибридных автоматов, которые будем называть гибридными автоматами активных элементов (ГА АЭ). Такой подход позволяет ввести агрегированное описание перехода в состояния из D . В противном случае, следовало бы ввести все возможные переходы в состояния из D для внутренних состояний ГА АЭ. Постулируется, что каждый АЭ в подобной модели конфликта определяется цепочкой последовательно выполняемых дискретных состояний с вероятностными переходами двух типов: в прямом направлении, задаваемом последовательностью состояний; в обратном направлении (возврат в предшествующие состояния для повторного выполнения работ). Можно показать, что любая последовательность действий в реальной системе может быть приведена к подобной схеме путем декомпозиции и/или агрегации выполняемых в ней действий.

Описание АЭ каждой из сторон (для определенности стороны A) предлагается ввести в рамках математической схемы ГА (1). Типовая структура графов ГА и вложенных ГА АЭ представлена на рис. 1. Для описания вложенных ГА определяются следующие множества и операторы.

1. Множество $S_A^D = \{J_A\}$, состоящее из целочисленной переменной (индекса состояний) J_A , принимающей конечное множество значений, принадлежащих множеству символов дискретных состояний $J_a \in Q_A^L$.

2. Множество Q_A^L — символов, определяющих дискретные состояния ГА АЭ и состоящее из следующих подмножеств $Q_A^L = Q_{Ad}^L \cup Q_{Ap}^L \cup Q_{Av}^L$.

Здесь подмножество $Q_{Ad}^L = \{A_1, \dots, A_p\}$ обозначает символы подготовительных состояний для организации в последующем активной деятельности ГА АЭ, которые выполняются однократно без возврата и повторения (на рис. 1 $p = 2$).

Подмножество $Q_{Ap}^L = \{A_{p+1}, \dots, A_m\}$ обозначает символы состояний активной деятельности, которые могут выполняться многократно с возвратом и повторением, что отражает вероятностный характер достижения успеха при выполнении соответствующих работ.

Подмножество $Q_{Av}^L = \{A_{m+1} = A_v\}$ состоит из символа одного состояния, называемого критическим (состояние выигрыша). Переход в критическое состояние сопровождается генерацией и пересылкой события $attack_A$, переводящего ГА стороны B из состояния L_B в состояние D_B . Состояние A_v в данной модели является поглощающим. При этом в общем случае для него допустимы переходы в другие внутренние состояния L_A . В любом случае при таком исходе сторона A остается в состояниях из L_A и выигрывает в данной реализации конфликтного взаимодействия.

3. Множество вещественных переменных $S_A^C = \{x_k, k = \overline{1, n}\}$ — $x = (x_1, \dots, x_n)^T \in R^n$, являющихся, наряду с переменной времени $t \in T$ аргументами или значениями непрерывных функций, описывающих локальное поведение ГА для каждого дискретного состояния. В качестве компонентов x предлагается использовать следующие переменные: $x = (\tau, r, u)^T \in R^3$, где τ — время выполнения работы в каждом дискретном состоянии ГА АЭ; r — переменная, характеризующая результат выполнения работы, которая может принимать как конечное число значений, так и значения на множестве вещественных чисел; u — переменная, характеризующая интервал времени, остающийся до завершения работы и нахождения в данном состоянии. При этом множество $D_x \subseteq R^4$.

4. Множество начальных состояний ГА $I_{0,A} = \{L_A\}$ и множество начальных состояний соответствующего ГА АЭ $I_{0,L_A} = \{A_1\} \times T \times R^n$. В иерархической схеме ГА переход в начальное состояние $I_{0,A} = \{L_A\}$ означает переход в начальное состояние вложенного ГА АЭ $I_{0,L_A} = \{A_1\} \times T \times R^n$.

5. Оператор локального поведения предлагается задавать для каждого состояния ГА АЭ на основе следующих соотношений:

$$x(t) = (\tau(t), r(t), u(t))^T = f(x(t_{k-1}), t);$$

$$t_{k-1} = t_{inp}, \quad x(t_{k-1}) = (\tau_k, r_k, t_{k-1} + \tau_k)^T, \quad (4)$$

$$(\tau_k, r_k)^T : P_{Ak}(\tau, r) = P_{Ak}(r/\tau)P_{Ak}(\tau), \quad \tau(t) = \tau_k,$$

$$r(t) = r_k, \quad u(t) = u(t_{k-1}) - t, \quad t \geq t_{k-1},$$

$$t_{out} = t_k = t : I(u(t) = 0),$$

где $P_{Ak}(\tau, r)$ — совместная плотность распределения τ и r , которые в общем случае являются статистически зависимыми случайными величинами. В соответствии с (4) сначала на основе плотности $P_{Ak}(\tau)$ генерируется интервал времени τ_k пребывания в данном состоянии. Далее на основе условной плотности $P_{Ak}(r/\tau_k)$ формируется значение r_k , определяющее результат, достигаемый после выполнения работы.

6. Для каждого из состояний подмножества $Q_{Ad}^L = \{A_1, \dots, A_p\}$ переход в следующее состояние при $t \geq t_{k-1}$ задается на основе сторожевого условия при $pr_k(e) : I(u(t) = 0) = true$ или в соответствии с (2) как $[t - t_{k-1} \geq \tau_k]$ (на рис. 1 обозначение $[t - t_{inp} \geq \tau_{Ak}]$). При этом действие γ , сопровождающее переход и предусматривающее согласно (2) выполнение алгоритма инициализации нового локального поведения для следующего состояния из $Q_{Ad}^L = \{A_1, \dots, A_p\}$, не задается. В данной схеме

его удобно выполнить сразу при входе в новое состояние путем задания начальных значений переменных в соотношениях (4).

Для каждого состояния подмножества $Q_{Ap}^L = \{A_{p+1}, \dots, A_m\}$ переход в следующее состояние при $t \geq t_{k-1}$ осуществляется при выполнении сторожевого условия $pr_k(e): I(u(t) = 0) \wedge I(r(t) \in R_k) = \text{true}$, где R_k — множество значений, принадлежность к которому отражает успешное завершение работы. Для каждого состояния подмножества $Q_{Ap}^L = \{A_{p+1}, \dots, A_m\}$ при $t \geq t_{k-1}$ осуществляется возврат для повторения работы при выполнении сторожевого условия $pr_k(e): I(u(t) = 0) \wedge I(r(t) \notin R_k) = \text{true}$. Удобно (см. рис. 1) представить подобные переходы как общий переход в точку ветвления, обозначенную кружком (состояние без времени), если $[t - t_{k-1} \geq \tau_k]$, из которой затем задаются два перехода: переход в следующее дискретное состояние и переход, возвращающий ГА к повторению работы в данном состоянии.

Далее будем рассматривать упрощенный вариант задания исхода $r(t) = r_k$ путем генерации равномерной случайной величины $r_k \in [0, 1]$. Тогда условие перехода в следующее состояние из точки ветвления определяется как $[r_k \leq P_{Ak}]$, а условие возврата как $[r_k > P_{Ak}]$, где P_{Ak} или $P_{Ak}(\tau)$ — вероятность успешного завершения работы, задаваемая в рамках (4).

Действие или событие γ , сопровождающее выполнение переходов для состояний подмножества $Q_{Ap}^L = \{A_{p+1}, \dots, A_m\}$, как и ранее, не задается, за исключением перехода в критическое состояние, который сопровождается генерацией события $attack_A$ (на рис. 1 обозначение $send(attack_A)$).

Аналогичным образом вводится описание и детализация множеств и операторов для ГА и ГА АЭ стороны B ; при этом необязательно точное совпадение множеств дискретных состояний и других элементов ГА АЭ.

1.2. Оценка нижней границы вероятности выигрыша. Покажем возможность получения нижней границы вероятности выигрыша на основе задания моментов для времени пребывания в каждом дискретном состоянии без знания конкретно вида распределений. Для этого запишем выражение для общего времени пребывания каждого ГА АЭ, например автомата стороны A , в группе состояний $Q_{Ad}^L \cup Q_{Ap}^L = \{A_1, \dots, A_p, A_{p+1}, \dots, A_m\}$, т. е. во всех состояниях за исключением критического. Это время определяется соотношением

$$\tau_a = \sum_{k=1}^p \tau_k + \sum_{k=p+1}^m \sum_{j=1}^{h_k} \tau_{k,j}, \quad (5)$$

где $h_k \in \{1, 2, \dots, \infty\}$ — случайная величина, характеризующая число циклов повторения работы,

выполняемой в состоянии $A_k \in Q_{Ap}^L$ при отсутствии положительного результата на предыдущем цикле; $\tau_{k,j}$ — случайное время выполнения работы на j -м цикле повторения работы.

С учетом (5), считая заданными вероятности $P_{Ak}(\tau) = p_{ak}$ успешного завершения работы в $A_k \in Q_{Ap}^L$, а также учитывая то обстоятельство, что случайные величины, определяющие время пребывания ГА АЭ в состояниях $Q_{Ad}^L \cup Q_{Ap}^L = \{A_1, \dots, A_p, A_{p+1}, \dots, A_m\}$, являются независимыми, запишем выражение для плотности распределения τ_a в виде:

$$\begin{aligned} P_A(\tau_a) &= \sum_{\{h_{p+1}, \dots, h_m\}} P(h_{p+1}, \dots, h_m) P(\tau_a/h) = \\ &= \sum_{h_{p+1}=1}^{\infty} \dots \sum_{h_m=1}^{\infty} \prod_{k=p+1}^m (1 - p_{ak})^{h_k-1} p_{ak} P(\tau_a/h). \end{aligned} \quad (6)$$

В (6) $P(\tau_a/h)$ есть свертка плотностей распределения, описывающая композицию случайных величин (5) для условия, определяемого значением вектора $h = (h_{p+1}, \dots, h_m)^T$, каждая компонента которого отражает число повторяющихся циклов работы в соответствующих дискретных состояниях. Для распределения $h = (h_{p+1}, \dots, h_m)^T$ в (6) используется следующее представление:

$$\begin{aligned} P(h_{p+1}, \dots, h_m) &= \prod_{k=1}^m P(h_k), \quad P(h_k) = (1 - p_{ak})^{k-1} p_{ak}, \\ \sum_{h_k=1}^{\infty} P(h_k) &= \frac{p_{ak}}{1 - (1 - p_{ak})} = 1. \end{aligned} \quad (7)$$

При использовании аналитического представления для плотностей распределения $P_{Ak}(\tau_k)$, $k = \overline{1, m}$, вычисление свертки $P(\tau_a/h)$ в выражении (6) возможно только приближенно и только численными методами. Далее для оценки вероятностей выигрыша в конфликте необходимо с использованием плотностей $P_A(\tau_a)$ для A и $P_B(\tau_b)$ для B определить:

$$\begin{aligned} P_{Va} &= \Pr(\tau_a < \tau_b) = \int_{-\infty}^{\infty} P_A(u) \left(\int_u^{\infty} P_B(v) dv \right) du, \\ P_{Vb} &= \Pr(\tau_a \geq \tau_b) = \int_{-\infty}^{\infty} P_B(v) \left(\int_v^{\infty} P_A(u) du \right) dv. \end{aligned} \quad (8)$$

Рассмотрим теперь возможность использования фундаментальных неравенств теории вероятностей для получения граничных оценок для вероятностей (7) на основе информации о первом и втором моментах распределений $P_{Ak}(\tau_k)$, $k = \overline{1, m}$, $P_{Bk}(\tau_k)$, $k = \overline{1, m}$. С этой целью определим математические ожидания

$m_a = M[\tau_a]$, $m_b = M[\tau_b]$ и дисперсии $d_a = D[\tau_a] = M[(\tau_a - m_a)^2]$, $d_b = D[\tau_b] = M[(\tau_b - m_b)^2]$, для τ_a и τ_b . Основываясь на выражениях (5)–(7), можно представить выражения для условного относительно $h = (h_{p+1}, \dots, h_m)^T$ математического ожидания $m_a(h)$ и безусловного математического ожидания m_a следующим образом:

$$\begin{aligned} m_a(h) &= M[\tau_a/h] = \sum_{k=1}^p m_{a\tau_k} + \sum_{k=p+1}^m h_k m_{a\tau_k}; \\ m_a &= M[\tau_a] = \sum_{\{h_{p+1}, \dots, h_m\}} m_a(h) \prod_{k=1}^m P(h_k) = \\ &= \sum_{k=1}^p m_{a\tau_k} + \sum_{k=p+1}^m m_{a\tau_k} \sum_{h_k=1}^{\infty} h_k (1 - p_{ak})^{h_k-1} p_{ak}. \end{aligned} \quad (9)$$

Для бесконечной суммы в (9) используем следующее преобразование:

$$\begin{aligned} \sum_{h_k=1}^{\infty} h_k (1 - p_{ak})^{h_k-1} p_{ak} &= \\ &= p_{ak} \frac{d}{dp_{ak}} \left(-(1 - p_{ak}) \sum_{h_k=1}^{\infty} (1 - p_{ak})^{h_k-1} \right) = \\ &= p_{ak} \frac{d}{dp_{ak}} \left(-\frac{1 - p_{ak}}{p_{ak}} \right) = \frac{1}{p_{ak}}. \end{aligned}$$

Тогда для m_a окончательно получим

$$m_a = M[\tau_a] = \sum_{k=1}^p m_{a\tau_k} + \sum_{k=p+1}^m \frac{m_{a\tau_k}}{p_{ak}}. \quad (10)$$

Выражение для d_a также получим на основе условной и безусловной дисперсии, которую, после громоздких, но несложных преобразований слагаемых можно представить в следующем окончательном виде:

$$d_a = \sum_{k=1}^p d_{a\tau_k} + \sum_{k=p+1}^m \frac{d_{a\tau_k}}{p_{ak}}. \quad (11)$$

Аналогичного вида соотношения получаются для m_b и d_b стороны B .

Рассмотрим возможность получения оценок вероятности выигрыша сторон. В качестве первой из таких оценок целесообразно рассмотреть гауссовское приближение для разности случайных величин $\delta\tau = \tau_a - \tau_b$, учитывая, что в соотношении (5) для τ_a (и аналогично для τ_b) выполняется суммирование независимых случайных величин, часть из которых имеет одинаковое распределение. Тогда такую оценку можно представить как

$$\begin{aligned} P_{Vag} &= \Pr(\tau_a < \tau_b) = \Pr(\delta\tau < 0) = \\ &= \int_{-\infty}^0 N(u, m_a - m_b, d_a + d_b) du = F\left(-\frac{m_a - m_b}{\sqrt{d_a + d_b}}\right); \end{aligned} \quad (12)$$

$$P_{Vbg} = \Pr(\tau_a \geq \tau_b) = 1 - P_{Va};$$

$$F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x N(v, 0, 1) dv,$$

где $N(u, m, d)$ — обозначение гауссовской плотности распределения вероятностей с соответствующими параметрами.

Помимо этого, целесообразно рассмотреть возможность получения других оценок на основе фундаментальных неравенств теории вероятностей. Пусть для определенности $m_a < m_b$. Рассмотрим цепочку эквивалентных неравенств для разности случайных интервалов времени τ_a и τ_b :

$$\delta\tau = \tau_a - \tau_b < 0;$$

$$\delta\tau - (m_a - m_b) < -(m_a - m_b) = m_b - m_a.$$

Тогда для оценки вероятности события $\delta\tau = \tau_a - \tau_b < 0$ можно использовать неравенство Чебышева [20] в виде:

$$\Pr[|\delta\tau - (m_a - m_b)| < m_b - m_a] \geq 1 - \frac{d_a + d_b}{(m_b - m_a)^2}.$$

Поскольку для исходного неравенства $\delta\tau = \tau_a - \tau_b < 0$ диапазон значений снизу не ограничен, то можно записать следующее выражение для нижней границы вероятности выигрыша:

$$\begin{aligned} P_{Va} &= \Pr[\tau_a < \tau_b] \geq \\ &\geq \Pr[|\delta\tau - (m_a - m_b)| < m_b - m_a] \geq \\ &\geq P_{Vbc} = 1 - \frac{d_a + d_b}{(m_b - m_a)^2}. \end{aligned} \quad (13)$$

Уточнить оценку P_{Vbc} в (13) можно, если предположить, что плотность распределения композиции $\delta\tau$ является унимодальной. Для этого необходимо, чтобы плотности распределения $P_A(\tau)$ и $P_B(\tau)$ были унимодальными и симметричными. Тогда и плотность композиции $\delta\tau = \tau_a - \tau_b$ будет унимодальной. В этом случае можно усилить (13) на основе неравенства Высочанского — Петунина [20] и получить нижнюю границу для вероятности выигрыша:

$$\begin{aligned} P_{Va} &= \Pr[\tau_a < \tau_b] \geq \Pr[|\delta\tau - (m_a - m_b)| < m_b - m_a] = \\ &= \Pr[|\delta\tau - (m_a - m_b)| < \rho \sqrt{d_a + d_b}] \geq \\ &\geq P_{Vbn} = 1 - \frac{4}{9\rho^2} = 1 - \frac{4(d_a + d_b)}{9(m_b - m_a)^2}, \end{aligned} \quad (14)$$

$$\rho = \frac{m_b - m_a}{\sqrt{d_a + d_b}} \geq \sqrt{\frac{8}{3}} = 1,6329.$$

Подобная оценка позволяет уточнить результат при достаточных значениях отношения ρ .

1.3. Моделирование конфликта коалиций. Более сложной является задача математического моделирования конфликта коалиций. Данная задача подразумевает наличие групп конфликтно взаимодействующих объектов, действия которых с каждой стороны координируются некоторым органом управления на основе заранее заданного регламента поведения. Процесс конфликтного взаимодействия коалиций, как и ранее, может быть представлен как два параллельно функционирующих ГА: автомата A , состоящего из N_a иерархически вложенных элементарных ГА, и автомата B , состоящего из N_b иерархически вложенных элементарных ГА. Пусть структура каждого вложенного ГА с обеих сторон будет аналогична структуре, представленной на рис. 1. Под коалицией, описываемой автоматом A , будем понимать кортеж

$$A = \langle A^{(1)}, \dots, A^{(N_a)}, \text{disp}_a \rangle, \quad (15)$$

где $A^{(i)}, i = \overline{1, N_a}$ — элементарные гибридные автоматы, участвующие в коалиции с данной стороны; $\text{disp}_a: T \times Q \times R^n \rightarrow \{1, 2, \dots, N_b\}$ — функция координации, обеспечивающая диспетчерское управление ГА коалиции.

Аналогичным образом вводится коалиция противостоящей стороны, описываемая кортежем элементарных гибридных автоматов и своей функцией координации $B = \langle B^{(1)}, \dots, B^{(N_b)}, \text{disp}_b \rangle$. Укрупненное отображение конфликта коалиций с использованием ГА (карт состояний) представлено на рис. 2.

Пусть для автомата $A^{(i)}$ множество $S_i^D = \{s_{a,i}\}$ представлено одной дискретной переменной, каждая из которых принимает два значения $s_{a,i} \in Q_{A_i} = \{L_A, D_A\}$, где L_A определяет состояние "жизни" в условиях отсутствия или наличия конфликтного воздействия со стороны элементов противостоящей коалиции, а D_A определяет со-

стояние "гибели" в результате конфликтного воздействия со стороны элементов противостоящей коалиции. Переход в D_A происходит под влиянием события attak_B , генерируемого одним из участников коалиции B .

Все внутренние переходы для ГА АЭ автомата $A^{(i)}$ формируются как и для ранее описанных ГА в дуэльной ситуации (см. рис. 1). Кроме этого, дополнительно вводится переход, показанный на рис. 1 штриховой линией. Этот переход позволяет из состояния A_v (состояние выигрыша) вернуться в первое из состояний группы $Q_{Ap}^L = \{A_{p+1}, \dots, A_m\}$ — состояний активной деятельности для назначения нового противника из числа элементов противостоящей стороны. Данный переход выполняется при условии

$$[\text{sum}_b > 0], \text{sum}_b = \sum_{j=1}^{N_b} I_B^{(j)}(s_{b,i} \in L_B), \quad (16)$$

где $I_B^{(j)}(s_{b,i} \in L_B) = 1$, $I_B^{(j)}(s_{b,i} \notin L_B) = 0$ — индикаторы пребывания ГА $B^{(j)}$ в состоянии L_B , что позволяет фиксировать общее число оставшихся в "живых" элементов коалиции со стороны B .

При вычислении функции координации действий коалиции для каждого ГА АЭ должен быть определен индекс элементарного ГА противостоящей стороны. Данный индекс может формироваться как на основе выбранной стратегии распределения целей для ГА АЭ по результатам анализа дискретных состояний оставшихся в живых элементарных ГА противника, так и на основе анализа координат элементарных ГА в фазовом пространстве состояний, описываемых непрерывными переменными.

Определим для каждого элементарного ГА противостоящей стороны $B^{(i)}$ индикатор $i_{b,j} = I_B^{(j)}\left(s_{b,j} \in L_B\right)$ и дополнительно индикатор $i_{ab,j} = I_{AB}^{(j)}(A \rightarrow L_B)$, фиксирующий состояние $B^{(i)}$ как ранее назначенной цели активного воздействия со стороны одного из элементов A (индикатор события "находится под воздействием"). Без принципиального ограничения общности можно использовать функцию координации сторон как отображения вида

$$\text{disp}_a: I(S_A^D, S_B^D) \rightarrow \{1, 2, \dots, N_b\}, \quad (20)$$

$$I_{ab} = (i_{b,1}, \dots, i_{b,N_b}, i_{ab,1}, \dots, i_{ab,N_b}) \in I(S_A^D, S_B^D);$$

$$\text{disp}_b: I(S_B^D, S_A^D) \rightarrow \{1, 2, \dots, N_a\},$$

$$I_{ba} = (i_{a,1}, \dots, i_{a,N_a}, i_{ba,1}, \dots, i_{ba,N_a}) \in I(S_B^D, S_A^D),$$

т. е. как функции, использующие на входе данные только дискретных переменных, определяющих состояния ГА противоборствующих сторон.

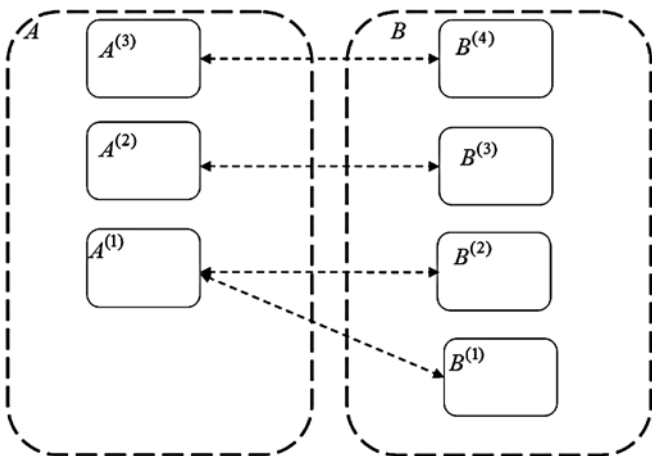


Рис. 2. Графическое отображение конфликта коалиций ГА

В данном случае получение аналитических оценок вероятностей выигрыша сторон существенно затрудняется в силу усложнения модели взаимодействия групп ГА, находящихся под управлением. Оценки, подобные (13)—(15) авторам удалось получить для допущения, согласно которому каждый ГА коалиции в текущее время может взаимодействовать только с одним или с двумя ГА противостоящей стороны, причем при начальном назначении целей в виде элементарных ГА противостоящей стороны, последующее переназначение ГА после успешного завершения "дуэли" не выполняется. В итоге использована бернуллиевская схема, в которой оценки вероятности выигрыша для пар или других групп (один с двумя) конфликтующих ГА получены на основе оценок аналогичных (12)—(14). Следует отметить, что общее решение задачи исследования конфликта в данной ситуации лежит, главным образом, в применении технологий ИМ с использованием объектно-ориентированного подхода для описания поведения участников коалиций.

2. Результаты компьютерного эксперимента

Проверку возможностей применения оценок (12)—(14) проводили методом статистического имитационного моделирования при использовании различных видов распределений для времени пребывания систем в своих состояниях. Для реализации ИМ с применением формализма гибридных автоматов была использована среда Matlab + Simulink + Stateflow. Здесь модель конфликта целесообразно строить с использованием

трех вложенных компонентов: главной программы, управляющей проведением статистического эксперимента (*m*-файл), запускаемой из нее *S*-модели (*Simulink*-модели), являющейся оболочкой для объектной *SF*-модели (*Stateflow*-модели) и, собственно, самой *SF*-модели, реализующей представление объектов в виде карт состояний Харела [14].

В рассмотренном ниже примере структура реализованной *SF*-модели конфликта (рис. 3) в целом совпадает со структурой предложенной математической модели и может быть однозначно понята исходя из стандартных обозначений среды. При этом используется следующая физическая интерпретация состояний. Состояния $A1$ (A_1) из $Q_{Ad}^L = \{A_1\}$ и $B1$ (B_1) из $Q_{Bd}^L = \{B_1\}$ соответствуют однократно выполняемым "работам" доставки и развертывания противоборствующих систем в зону активных действий. Состояния $A2$ (A_2) и $A3$ (A_3) из $Q_{Ap}^L = \{A_2, A_3\}$ и $B2$ (B_2), $B3$ (B_3) из $Q_{Bp}^L = \{B_2, B_3\}$ соответствуют "работам", выполняемым с возможным возвратом и повторением. При этом в $A2$ и $B2$ реализуется обнаружение противника в фазовом пространстве координат и высокоточная оценка этих координат, а в $A3$ и $B3$ — применение активных элементов для воздействия на противника (информационного или физического). Подобное представление характерно для радиоэлектронных систем в условиях информационного противоборства, например систем радиоэлектронной борьбы [2].

Время пребывания каждой из сторон в своих дискретных состояниях ($A1$, $A2$, $A3$ и $B1$, $B2$, $B3$) задается путем вызова *m*-функции *pro(...)*, формирующей значение случайной величины, распределенной по заданному закону. Переходы из одного состояния в другое осуществляются по условиям истечения времени пребывания с вероятностями $p_{a2} = pa2$, $p_{a3} = pa3$, $p_{b2} = pb2$, $p_{b3} = pb3$ (для возвратных состояний). Для управления процессами смены состояний вводятся внутренние переменные $t1$, at , $t2$, bt , смысл применения которых ясен из представленной на рис. 3 диаграммы.

Для множества статистических экспериментов, каждый из которых состоял из 1000 испытаний, рассматривались различные комбинации законов распределения, их параметров и вероятностей возврата для повторного выполнения работ. Для наглядности полученные результаты сведены к зависимостям вероятности выигрыша

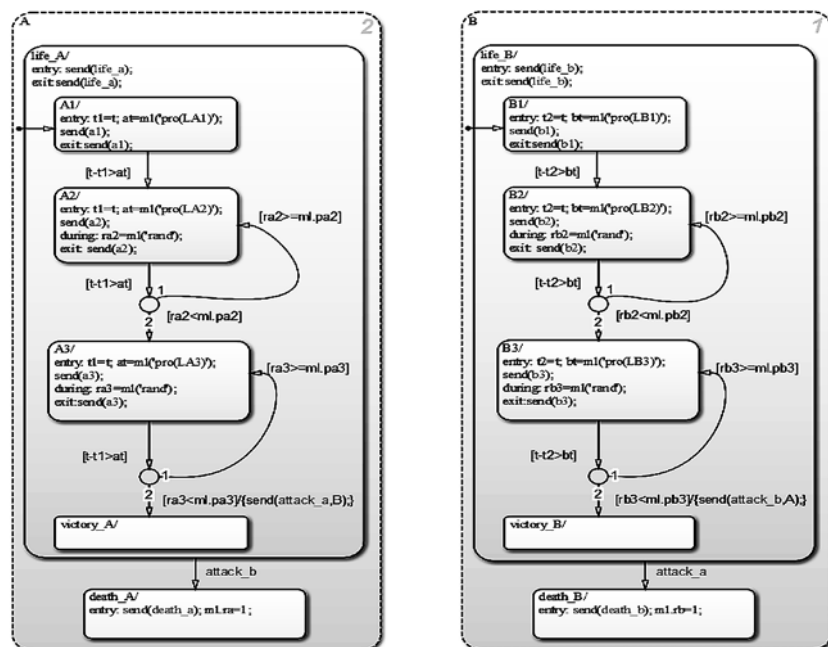


Рис. 3. Карты состояний для описания сторон А и В в *SF*-модели конфликта



Рис. 4. Результаты сравнения полученных оценок с результатами ИМ

от отношения $\rho = (m_b - m_a)^2 / (d_a + d_b)$. На рис. 4, а даны примеры типовых зависимостей, полученные при синхронном изменении вероятностей $p_{a2} = p_{a3}$ в диапазоне 0,70...0,90 с шагом 0,02 и вероятностей $p_{b2} = p_{b3}$ соответственно, меняющихся от 0,30 до 0,10. Математические ожидания и дисперсии для времени пребывания каждой из сторон в своих состояниях задавали одинаковыми $m_{ai} = m_{bi} = 1$, $i = \overline{1,3}$ и $d_{ai} = d_{bi} = 1$, $i = \overline{1,3}$ с одним и тем же законом распределения. На рис. 4, б приведены зависимости, полученные для значений математического ожидания m_{a1} , изменяющегося в диапазоне 0,9...6,3 с шагом 0,6. При этом значения вероятностей для возвратных состояний задавались одинаковыми $p_{a2} = p_{a3} = 0,70$ и $p_{b2} = p_{b3} = 0,30$. Математические ожидания и дисперсии для времени пребывания в других состояниях задавались равными $m_{ai} = m_{bi} = 1$, $i = 2,3$, $m_{b1} = 1$ и $d_{ai} = d_{bi} = 1$, $i = 2,3$ и также при одном и том же законе распределения.

Анализ представленных зависимостей показывает, что достаточно точной оказалась нижняя граница, полученная на основе неравенства Высочанского — Петунина, а также оценка, основанная на гауссовском приближении. Физический смысл полученных зависимостей состоит в том, что чем больше значение параметра ρ , характеризующего относительное среднестатистическое различие для времени достижения участниками дуэли своих критических состояний, тем больше вероятность выигрыша. Это говорит о важности фактора времени при реализации упреждающего характера воздействия по сравнению с вероятностью "поражения" на конечном этапе по результатам целеуказания. Если, например, сторона A превосходит B по вероятности "поражения" на завершающем этапе в соотношении 0,95 к 0,70, но при этом B действует в 2 раза более оперативно при измерении координат A , то B имеет более высокую эффективность в усло-

виях конфликта: соотношение числа выигрышей для A и B равно примерно 0,40 и 0,60.

При моделировании конфликта коалиций реализована SF -модель, в которой основные состояния ГА-участников совпадают с показанными на рис. 3. Число систем с каждой стороны может варьироваться (см. рис. 2). Каждой из систем сторон A и B выдается целеуказание в виде индекса системы противоборствующей стороны, которая должна далее находиться под воздействием системы A . При выполнении перехода в состояние *victory_A* фиксируется факт поражения противоборствующей системы B . При проведении статистических экспериментов на основе данной модели также рассматривали различные комбинации исходных параметров. На рис. 4, в приведены примеры зависимостей для конфликта коалиций, состоящих из четырех систем с каждой стороны, полученные для значений вероятностей p_{a3} в диапазоне 0,70...0,90 с шагом 0,02 и вероятностей p_{b3} в диапазоне 0,30...0,14 с шагом 0,02. Значения других вероятностей принимали равными $p_{a2} = 0,70$ и $p_{b2} = 0,30$. Математические ожидания и дисперсии для времени пребывания сторон в своих состояниях задавали одинаковыми и равными единице при одном и том же законе распределения.

Заключение

В целом можно сделать вывод о том, что предложенные в рамках формализма гибридных автоматов математические схемы и реализующие их компьютерные имитационные модели конфликтного взаимодействия систем, а также коалиций систем, обеспечивают наглядные объектно-ориентированные представления, которые целесообразно использовать как базовые для составления более сложных моделей в интересах исследования возникающих на практике ситуаций. Для дуэльного конфликта систем полу-

ченные аналитические соотношения позволяют вычислить оценки вероятности выигрыша в условиях неопределенности вида плотностей распределений для времени пребывания сторон в своих возможных состояниях. Для коалиций систем подобные аналитические оценки могут быть получены для ограниченного круга ситуаций с использованием весьма существенных допущений. Кардинальное решение задачи исследования закономерностей конфликта в данной ситуации лежит, главным образом, в использовании технологий имитационного моделирования с использованием объектно-ориентированного подхода для представления участников коалиций и их поведения. При использовании формализма гибридных автоматов в предложенной реализации это снимает проблему трудоемкости разработки моделей при изменении состава и структуры пространства состояний участников коалиций.

Список литературы

1. Макаренко С. И., Михайлов Р. Л. Информационные конфликты — анализ работ и методологии исследования // Системы управления, связи и безопасности. 2016. № 3. С. 95—178.
2. Модели информационного конфликта средств поиска и обнаружения / Под. ред. Ю. Л. Козирацкого. М.: Радиотехника, 2013. 232 с.
3. Дружинин В. В., Конторов А. С., Конторов Д. С. Введение в теорию конфликта. М.: Радио и связь, 1989. 288 с.
4. Сухоруков Ю. С., Шляхин В. М. Принципы моделирования динамики взаимодействия сторон в условиях радиолокационного конфликта // Радиотехника. 1992. № 1—2.
5. Будников С. А. Оценка вероятностных показателей в конфликте информационно-управляющих систем // Системы управления и информационные технологии. 2009. № 3 (37). С. 27—31.
6. Радько Н. М., Мокроусов А. Н. Динамическая модель работы адаптированного к помехам радиосредства с использованием сетей Петри // Информация и безопасность. 2009. № 2. С. 257—262.

7. Семисошенко М. А. Управление автоматизированными сетями декаметровых связи в условиях сложной радиоэлектронной обстановки. СПб.: ВАС, 1997. 364 с.
8. Губанов Д. А., Новиков Д. А., Чхартишвили А. Г. Социальные сети: модели информационного влияния, управления и противоборства / Под ред. Д. А. Новикова. М.: Издательство физико-математической литературы, 2010. 228 с.
9. Коцыняк М. А., Осадчий А. И., Коцыняк М. М., Лаута О. С., Дементьев В. Е., Васюков Д. Ю. Обеспечение устойчивости информационно-телекоммуникационных систем в условиях информационного противоборства. СПб.: ЛО ЦНИИС, 2015. 126 с.
10. Вакуленко А. А., Шевчук В. И. Математическая модель динамики конфликта радиоэлектронных систем // Радиотехника. 2011. № 1. С. 56—59.
11. Радзиевский В. Г., Сирота А. А. Информационное обеспечение радиоэлектронных систем в условиях конфликта. М.: ИПРЖР, 2001. 456 с.
12. Андреев И. А., Будников С. А., Гладков А. В. Полумарковская модель оценки конфликтной устойчивости информационной инфраструктуры // Вестник Воронежского государственного университета. Серия "Системный анализ и информационные технологии". 2017. № 1. С. 10—17.
13. Вялых А. С., Вялых С. А., Сирота А. А. Оценка уязвимости информационной системы на основе ситуационной модели динамики конфликта // Информационные технологии. 2012. № 9. С. 16—21.
14. Алгазинов Э. К., Сирота А. А. Анализ и компьютерное моделирование информационных процессов и систем / Под общ. ред. А. А. Сироты. М.: Диалог-МИФИ, 2009. 416 с.
15. Harel D. Statecharts: a Visual Formalism for complex systems // Science of Computer Programming. 1987. Vol. 8. P. 231—274.
16. Колесов Ю. Б., Сениченков Ю. Б. Моделирование систем. Динамические и гибридные системы: учеб. пособие. СПб.: БХВ-Петербург, 2012. 224 с.
17. Шпаков В. М. Ситуационные спецификации имитационных моделей гибридных реактивных систем // Труды СПИИРАН. 2002. Вып. 1. Т. 2. С. 212—222.
18. Парийская Е. Ю. Сравнительный анализ математических моделей и подходов к моделированию и анализу непрерывно-дискретных систем // Дифференциальные уравнения и процессы управления. 1997. № 1. С. 91—120.
19. Колмогоров А. Н., Фомин С. В. Элементы теории функций и функционального анализа. Изд. четвертое, перераб. М.: Наука, 1976. 544 с.
20. Высочанский Д. Ф., Петунин Ю. И. Обоснование правила 3-sigma для одномодальных распределений // Теория вероятностей и мат. статистика. 1979. Вып. 21. С. 23—35.

A. A. Sirota, Professor, Head of Department, e-mail: sir@cs.vsu.ru,

N. I. Goncharov, Graduate Student, e-mail: nikigoncharov@ya.ru, Voronezh state University

Modeling of Conflict Interaction of Systems Using the Formalism of Hybrid Automata

Explores the mathematical and computer models of conflict interaction systems based on the formalism of hybrid automata (card state). Offers model mathematical schemes and implement them in a simulation model, based on the description of change of States in conflict. The obtained analytical equations for estimating lower bounds on the probability of winning of one of the parties to the conflict, allowing to abstract from the specific species densities distributions for the residence time of the parties in their possible States. The results of the comparison of the estimates with simulation results.

Keywords: conflict interaction systems, hybrid automata, maps of states, simulation of conflict of coalitions of systems

References

1. Makarenko S. I., Mihajlov R. L. Informacionnye konflikty — analiz rabot i metodologii issledovaniya (Information conflicts —

analysis of works and research methodology), *Sistemy upravleniya, svyazi i bezopasnosti*, 2016, no. 3, pp. 95—178 (in Russian).

2. *Modeli informacionnogo konflikta sredstv poiska i obnaruzheniya* (Models of information conflict of means of search

and discovery), ed. Yu. L. Kozirackogo, Moscow, Radiotekhnika, 2013, 232 p.

3. **Druzhinin V. V., Kontorov A. S., Kontorov D. S.** Vvedenie v teoriyu konflikta (Introduction to the theory of conflict). Moscow, Radio i svyaz', 1989, 288 p. (in Russian)

4. **Suhorukov Yu. S., Shlyahin V. M.** Principy modelirovaniya dinamiki vzaimodeystviya storon v usloviyah radiolokatsionnogo konflikta (Principles of modelling the dynamics of interaction between the parties in terms of radar conflict). *Radiotekhnika*, 1992, no. 1–2 (in Russian).

5. **Budnikov S. A.** Ocenka veroyatnostnykh pokazateley v konflikte informatsionno-upravlyayushchih sistem (Estimation of probabilistic factors in the conflict management information systems), *Sistemy upravleniya i informatsionnye tekhnologii*, 2009, no. 3 (37), pp. 27–31 (in Russian).

6. **Rad'ko N. M., Mokrousov A. N.** Dinamicheskaya model' raboty adaptirovannogo k pomekham radiosredstva s ispol'zovaniem setey Petri (A dynamic model adapted to noise radio resources using Petri nets), *Informatsiya i bezopasnost'*, 2009, no. 2, pp. 257–262 (in Russian).

7. **Semisoshenko M. A.** Upravlenie avtomatizirovannymi setyami dekametrovoj svyazi v usloviyah slozhnoy radioelektronnoy obstanovki (The management of the automated networks decimeter communication in complex electronic environment), Saint-Peterburg, VAS, 1997. 364 p. (in Russian).

8. **Gubanov D. A., Novikov D. A., Chkhartishvili A. G.** Social'nye seti: modeli in-formatsionnogo vliyaniya, upravleniya i protivoborstva (Social networks: models of informational influence, control and confrontation), Ed. D. A. Novikov, Moscow, Izdatel'stvo fiziko-matematicheskoy literatury, 2010, 228 p. (in Russian).

9. **Kocynyak M. A., Osadchij A. I., Kocynyak M. M., Laut O. S., Dement'ev V. E., Vasyukov D. Yu.** Obespechenie ustojchivosti informatsionno-telekommunikatsionnykh sistem v usloviyah informatsionnogo protivoborstva (Sustainability information and telecommunication systems in the conditions of information warfare), Saint-Peterburg, LO CNIIS, 2015, 126 p. (in Russian).

10. **Vakulenko A. A., Shevchuk V. I.** Matematicheskaya model' dinamiki konflikta radioelektronnykh sistem (A mathematical model of conflict dynamics electronic systems), *Radiotekhnika*, 2011, no. 1, pp. 56–59 (in Russian).

11. **Radzievskij V. G., Sirota A. A.** Informatsionnoe obespechenie radioelektronnykh sistem v usloviyah konflikta

(Information support of electronic systems in conflict), Moscow, IPRZHR, 2001, 456 p. (in Russian).

12. **Andreeshchev I. A., Budnikov S. A., Gladkov A. V.** Polumarkovskaya model' ocenki konfliktnoy ustojchivosti informatsionnoy infrastruktury (Semi-Markov model of conflict assessment the sustainability information infrastructure), *Vestnik Voronezhskogo gosudarstvennogo universiteta. Seriya Sistemnyy analiz i informatsionnye tekhnologii*, 2017, no. 1, pp. 10–17 (in Russian).

13. **Vyalyh A. S., Vyalyh S. A., Sirota A. A.** Ocenka uyazvimosti informatsionnoy si-stemy na osnove situatsionnoy modeli dinamiki konflikta (Vulnerability assessment of information systems based on situational model of conflict dynamics), *Informatsionnye tekhnologii*, 2012, no. 9, pp. 16–21 (in Russian).

14. **Algazinov E. K., Sirota A. A.** Analiz i komp'yuternoe modelirovanie in-formatsionnykh processov i sistem (Analysis and computer modeling of information processes and systems). Ed. A. A. Siroty, Moscow, Dialog-MIFI, 2009. 416 p. (in Russian).

15. **Harel D.** Statecharts: a Visual Formalism for complex systems, *Science of Computer Programming*, 1987, vol. 8, pp. 231–274.

16. **Kolesov Yu. B., Senichenkov Yu. B.** Modelirovanie sistem. Dinamicheskie i gi-bridnye sistemy. Uchebnoe posobie. (Modeling of systems. Dynamic and hybrid systems. Textbook), Saint-Peterburg, BHV-Peterburg, 2012, 224 p. (in Russian).

17. **Shpakov V. M.** Situatsionnye specifikatsii imitatsionnykh modeley gibridnykh reaktivnykh sistem (Situational specifications of simulation models of hybrid reactive systems), *Trudy SPIIRAN*, 2002, is. 1, vol. 2, pp. 212–222 (in Russian).

18. **Parijskaya E. Yu.** Sravnitel'nyy analiz matematicheskikh modeley i podhodov k modelirovaniyu i analizu nepreryvno-diskretnykh sistem (Comparative analysis of mathematical models and approaches to modeling and analysis of continuously-discrete systems), *Differentsial'nye uravneniya i processy upravleniya*, 1997, no. 1, pp. 91–120 (in Russian).

19. **Kolmogorov A. N., Fomin S. V.** EHlementy teorii funktsij i funktsional'nogo analiza (Elements of the theory of functions and functional analysis). Moscow, Nauka, 1976, 544 p. (in Russian).

20. **Vysochanskij D. F., Petunin Yu. I.** Obosnovanie pravila Z-sigma dlya odnomo-dal'nykh raspredelenij (Rationale for the rule 3-sigma for onodolnykh distributions), *Teoriya veroyatnostej i mat. statistika*, 1979, is. 21, pp. 23–35 (in Russian).

УДК 004.8

А. В. Буравцев, зам. директора, e-mail: mister_j@mail.ru,

Институт информационных технологий и автоматизированного проектирования,
Московский технологический университет (МИРЭА),

В. Я. Цветков, д-р техн. наук, проф., зам. руководителя, e-mail: cvj2@mail.ru,

Центр стратегического анализа и развития,

Научно-исследовательский и проектно-конструкторский институт информатизации,
автоматизации и связи на железнодорожном транспорте (НИИАС), Москва

Анализ структуры сложной организационно-технической системы

Выполнен анализ особенностей сложной организационно-технической системы и предложен метод по формированию ее структуры и иерархии. Показано место сложной организационно-технической системы в ряду сложных систем. Раскрываются содержание и особенности сложной организационно-технической системы. Отмечено включение когнитивных факторов в работу сложной организационно-технической системы. Описано применение дихотомического метода для построения структуры сложной организационно-технической системы. Дихотомический метод применен в сочетании с оппозиционным методом. Сочетание дихотомического и оппозиционного методов позволяет последовательно выделять качественные компоненты в виде структуры сложной системы. Дихотомический метод дополнен методом построения иерархий, основанном на теории графов.

Ключевые слова: системный анализ, сложная система, социальная система, сложная организационно-техническая система, аутопойезис, моделирование, структура системы, дихотомическое деление, моделирование иерархии

Введение

Системный подход рассматривается как научное направление, предметом которого является исследование объектов и систем [1, 2], включая анализ их структуры [3]. В современном понимании системный подход исследует материальные и абстрактные объекты различной сложности, масштаба и уровня вложенности, в основе описания которых можно применить понятие "система" [4]. Системный подход находит применение при решении разных задач от образования [5] до инновационной деятельности [6]. В широком смысле под системой понимают множество элементов, находящихся в отношениях и связях друг с другом, которые образуют определенную целостность и единство. Это множество обладает новым качеством, не присущим отдельным элементам [7]. Сложная организационно-техническая система является одной из форм реализации сложной системы [8]. Особенность этой системы — включение когнитивных и адаптивных аспектов в ее функционирование. Такую систему сложно проектировать из типовых элементов [9], поэтому поиск ее структуры и ее элементов осуществляется "сверху-вниз". Задачей системного подхода является нахождение этих элементов и связей между ними. Для решения такой задачи применяют различные методы и подходы. В данной работе для исследования структуры сложной организационно-технической системы применяется относительно новый дихотомический метод анализа [10, 11]. Еще одной особенностью сложной организационно-технической системы является потенциальная возможность ее саморазвития. Это можно выявить с применением предлагаемого дихотомического подхода.

1. Сложная организационно-техническая система как этап эволюции сложных систем

Для того чтобы определить место сложной организационно-технической системы среди сложных систем следует проанализировать эволюцию этих систем. Основой системного анализа является понятие сложной системы как некой абстракции и модели. Сложная система трактуется как совокупность частей и элементов, связей и отношений между ними [1, 7]. На практике абстрактных систем не существует и все реальные системы отличаются от абстрактной сложной системы. На рис. 1 дана схема эволюции сложных систем, в которую входит сложная организационно-техническая система. На примере этой схемы можно показать различие между системами, входящими в эту классификацию.

Прикладные системы отличаются большим разнообразием [12]. В простейшем понимании прикладная система — это техническая, инфор-

мационная или автоматизированная система, решающая конкретные задачи в предметной области. Прикладная система является не универсальной, а предметно-ориентированной. Теоретической основой прикладной системы является сложная система, в силу этого прикладная система обладает свойствами целостности, системности и эмерджентности. Частным случаем прикладной системы является техническая система.

Развитием прикладной системы является сложная техническая система (СТС) [13]. Сложная техническая система отличается от прикладной системы характеристикой "сложность". Это в первую очередь структурная сложность. В аспекте интеграции СТС иногда представляет собой интеграцию нескольких прикладных или технических систем, что дает основание рассматривать ее как "систему систем".

Сложная социальная система (ССС) может быть рассмотрена как общество или группы в обществе. Сложная социальная система представляет собой совокупность отдельных социальных групп, организаций, институтов. Ее части соединены между собой различными связями и образуют социальную структуру [14]. В сравнении с СТС эта система более вариабельная и не детерминирована. Социальные системы по Луману это системы коммуникации, а общество — самая обширная социальная система. Будучи социальной системой, сегодняшнее общество является мировым сообществом [14].

Существует синтез двух последних рассмотренных систем, которые называют сложными социотехническими системами. Социотехнические системы в противоположность теориям технологического детерминизма и СТС, включающим одностороннее взаимодействие человека и техники, основываются на идее двухстороннего взаимодействия человека и техники. Некоторые информационные системы являются социотехническими. Другие (односторонняя обработка) являются техническими.

Сложная организационно-техническая система (СОТС) характеризуется включением организационного управления в работу сложной технической системы [8, 9]. Термин "сложная" отражает большой масштаб системы и возможное наличие различных видов сложности (структурной, функциональной, управленческой, потоковой) в срав-



Рис. 1. Эволюция сложных систем

нении с прикладной системой или с технической системой. Сложная организационно-техническая система содержит характеристики социальных систем и поэтому теория социальных систем [15] может быть принята для нее с большей полнотой, чем для СТС.

Сложная организационно-техническая система характеризуется следующими основными признаками: зависимостью ее функционирования от когнитивного фактора; допустимой изменчивостью структуры во времени; изменяющимися во времени целями системы; необходимостью поддержания целостности и развития системы; множеством информационных ситуаций, которые могут не соответствовать запланированным ранее ситуациям. Признак "необходимость по поддержанию целостности и развития системы" требует применения аутопойезиса как механизма, обеспечивающего функционирование и жизнедеятельность системы. СОТС не является чисто интеллектуальной системой, но содержит интеллект. Эта система характеризуется следующими способностями:

- формировать, поддерживать и развивать внутреннюю субъективную информационную модель мира в категориях и понятиях, это свойство мультиагентных систем;
- извлекать знания и производить новые знания на основе кластеризации элементов, это свойство киберфизических систем;
- моделировать на основе правил;
- ресурсностью, т. е. возможностью обучаться на своем и чужом опыте;
- многоцелевым анализом и управлением;
- формировать собственные ресурсы;
- преобразовывать неявное знание в явное;
- выполнять качественный анализ принимаемых решений и качественную оценку последствий действий.

При наличии аутопойезиса [15] каждый элемент СОТС обладает внешним или внутренним интеллектом. То есть им можно управлять извне или на основе кластеризации (объединения внутреннего интеллекта) он создает новые правила принятия решений. Каждый элемент СОТС при наличии аутопойезиса может осуществлять информационное взаимодействие с другими элементами. Сообщество элементов может осуществлять взаимодействие с внешней средой. Каждый элемент системы способен к саморазвитию и кооперации с другими элементами системы.

Информационное взаимодействие между элементами в пределах системы и системы со средой носит неоднозначный характер. Информационное взаимодействие между элементами основано на принципе информационного соответствия [16]. Информационное взаимодействие между элементами и средой основано на принципе редукции [15].

Элементы СОТС с интеллектом и памятью с течением времени накапливают информацион-

ные отличия друг от друга и осуществляют самоорганизацию. При этом могут возникать новые целевые функции как результат саморазвития системы. Саморазвитие системы состоит в том, что первоначальные элементы могут быть через некоторое время заменены другими элементами. Наличие элементов системы с интеллектом и памятью создает механизм аутопойезиса и обеспечивает саморазвитие СОТС. Модификация элементов СОТС изменяет реакцию системы на воздействие внешней среды. Аутопойезис [15] является связующим звеном между теорией систем и теорией эволюции. В связи с этим за рубежом выделяют класс аутопойетических систем, к числу которых относится СОТС.

2. Структурное моделирование сложной организационно-технической системы

Построение структуры сложной системы представляет собой непростую задачу [3]. Наиболее известные технологии формирования структуры — системы IDEF и SADT. Однако они ориентированы на типовые бизнес-процессы. Такие технологии реализуют проектирование "снизу-вверх" и не всегда являются оптимальными.

Представляет интерес проектирование структуры "сверху-вниз" в соответствии с качественными признаками. Такую задачу можно решить методами дихотомического анализа [10, 11]. Рассмотрим для примера сложную систему, структурно содержащую несколько качеств. Несколько качеств — это характерный признак СОТС. Схематически такая система приведена на рис. 2.

Пересекающиеся подмножества соответствуют пересечению качеств или свойств. Построение структуры сводится к построению "чистых", не пересекающихся множеств. Формальная запись исходной системы имеет вид

$$Sys = F(M1, M2, M3). \quad (1)$$

Допустимо, что первичные подмножества $M1$, $M2$, $M3$ пересекаются. Они образуют смешанный уровень деления системы. В качестве методики деления используем оппозиционный подход [17, 18].

Пусть $a1$ однородная, не пересекающаяся с другими множествами часть множества $M1$, $a2$ —

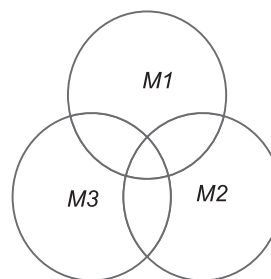


Рис. 2. Схематическое изображение СОТС как совокупности пересекающихся множеств

однородная часть множества $M2$, $a3$ — однородная часть множества $M3$. Дихотомическое деление системы Sys имеет вид

$$\begin{aligned} Sys &= a1 + \text{не } a1; \\ \text{не } a1 &= a2 + (\text{не } a1) \& (\text{не } a2); \\ (\text{не } a1) \& (\text{не } a2) &= a3 + (\text{не } a1) \& (\text{не } a2) \& (\text{не } a3) \end{aligned}$$

и т. д.

Структурно схема деления показана на рис. 3.

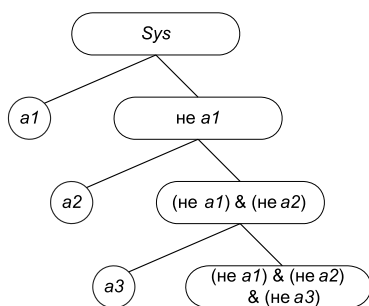


Рис. 3. Дихотомическое оппозиционное деление СОТС

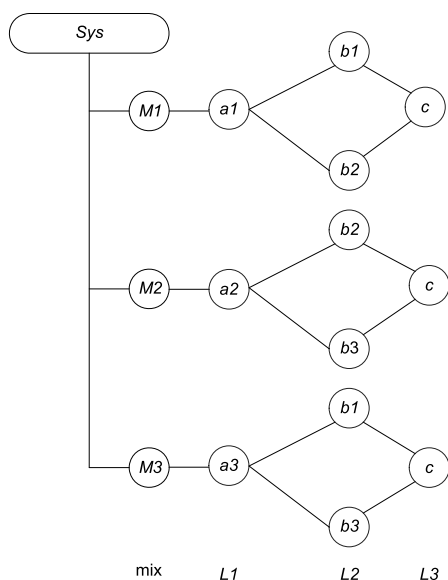


Рис. 4. Структура СОТС, полученная дихотомическим делением из системы на рис. 2

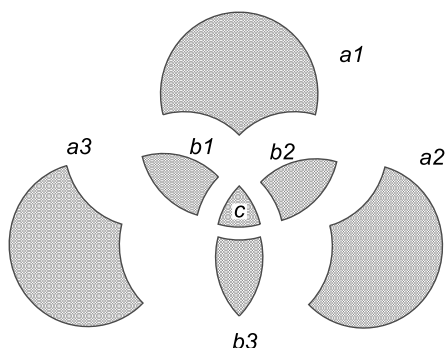


Рис. 5. Результат дихотомического деления системы, показанной на рис. 2, на чистые классы

Особенность выделенных частей системы $a1$ — $a3$ в том, что они однородны и содержат только одно качество. Оставшуюся часть $[(\text{не } a1) \& (\text{не } a2) \& (\text{не } a3)]$ можно продолжать делить подобным алгоритмом. Результат окончательного деления или структура системы представлена на рис. 4. На этом рисунке выделены уровни, которые содержат разное число качеств. Первый уровень $L1$ содержит одно качество, второй $L2$ — два, третий уровень $L3$ — три качества. Формально элементы структуры приведены в виде следующих выражений.

$$\begin{aligned} \text{Первый уровень} \\ a1 \subseteq M1, a2 \subseteq M2, a3 \subseteq M3; \end{aligned} \quad (2)$$

$$\begin{aligned} \text{Второй уровень} \\ b1 = M1 \cap M3; b2 = M1 \cap M2; b3 = M3 \cap M2; \end{aligned} \quad (3)$$

$$\begin{aligned} \text{Третий уровень} \\ c = M1 \cap M2 \cap M3. \end{aligned} \quad (4)$$

Структурной схеме на рис. 4 будет соответствовать результат деления на рис. 5. При этом следует отметить, что элементы этих множеств будут обладать разными свойствами.

Элементы подмножеств первого уровня $L1$ обладают одним свойством или одним качеством: $a1 = f1(m1)$, $a2 = f2(m2)$, $a3 = f3(m3)$. Элементы подмножеств второго уровня $L2$ обладают двумя свойствами или двумя качествами: $b1 = f4(m1, m3)$, $b2 = f5(m1, m2)$, $b3 = f6(m2, m3)$. Элементы подмножества третьего уровня $L3$ обладают тремя свойствами или тремя качествами: $c = f4(m1, m2, m3)$.

3. Построение иерархии дихотомического деления

Построение структуры в виде сетевой модели не является завершением проектирования системы. Важным является построение иерархии в системе. Выше проводилось дихотомическое деление S на фрагменты. Пусть S обозначает граф произвольного вида, элементы которого отличаются какими-либо пометками, называемыми их именами. Граф Ps называется фрагментом графа S ($Ps \subseteq S$), если Ps образован подмножеством элементов графа S . Пусть F — иерархия фрагментов графа S .

Иерархия F представляет собой такое множество фрагментов графа Ps , что при $G \in F$ и для любых двух фрагментов Ps_1 и Ps_2 из F выполняется условие: либо фрагменты Ps_1 и Ps_2 не пересекаются, либо один из них является частью (подфрагментом) другого. Фрагмент S — основной фрагмент иерархии F . Фрагмент $Ps \in F$ — элементарный, если в F нет фрагментов S , являющихся подфрагментами фрагмента Ps . Для любых $Ps_1, Ps_2 \in F$ фрагмент Ps_1 — прямой подфрагмент Ps_2 (или, что то же самое, фрагмент, непосредствен-

но вложенный в Ps_2), если Ps_1 — подфрагмент Ps_2 и не существует такого $Ps_3 \in F$, отличного от Ps_1 и Ps_2 , что $Ps_1 \subseteq Ps_3 \subseteq Ps_2$.

На рис. 4 приведен пример структуры системы S , полученной после дихотомического деления. На рис. 6 приведены фрагменты Ps , входящие в S , полученные после дихотомического деления.

Используя правила, описанные выше, построим иерархию F . На рис. 7 приведена иерархия F системы S , построенная с учетом наличия фрагментов, показанных на рис. 5.

Построение иерархии является завершающим этапом структурного моделирования сложной организационно-технической системы.

Запись структуры СОТС может иметь два формальных описания: иерархическое и классификационное. Классификационное описание констатирует наличие "чистых подмножеств" как базиса системы (см. рис. 5):

$$Sys = Fk(a1, a2, a3, b1, b2, b3, c). \quad (5)$$

Классификационное описание имеет вид, констатирующий иерархическую вложенность и служит основой оценки структурной сложности

$$Sys = Fi(M1[a1, b1(c), b2(c)], M2[a2, b2(c), b3(c)], M3[a3, b1(c), b3(c)]). \quad (6)$$

Дихотомический анализ позволяет не только выявить классификационные признаки объекта СОТС, но и оценить ее сложность. Согласно дихотомии объекты можно разделять на группы "простые — сложные". Простым назовем объект, который имеет одноуровневое описание, например $a1$. Описание простого объекта получается линейным прохождением дерева разбора. Сложным назовем объект, первичные параметры которого включают вторичные параметры, например $b1(c)$. Вложенность параметров определяет уровень сложности. Для примера на рис. 4 и в выражении (6) уровень сложности равен 3. Это обусловлено тем, что подмножество $c = M1 \cap M2 \cap M3$ находится на третьем уровне иерархии.

Заключение

Модель системы на рис. 2 и последующий ее разбор дает основание говорить о такой системе как аутопойетической системе [15]. Это обусловлено наличием обратной связи, которую создает подмножество c . Наличие обратной связи создает механизм самоорганизации. Дихотомическое структурное моделирование СОТС использует принципы, лежащие в основе системного анализа [19]. Особенностью дихотомического структурного анализа является возможность указания в структурной модели множества зависимых и независимых частей и элементов. Это позволяет вы-

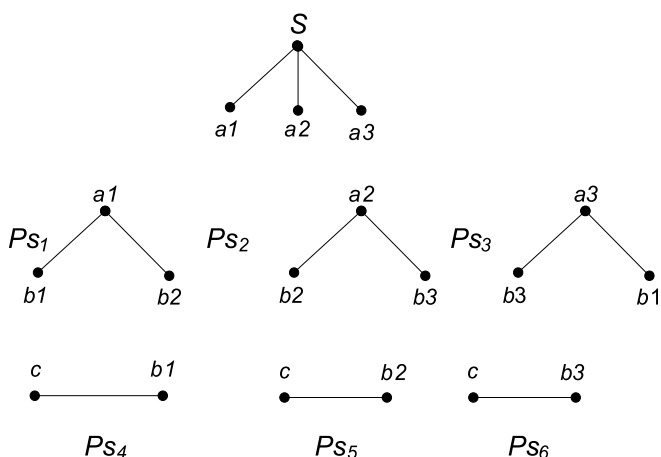


Рис. 6. Фрагменты S , полученные после дихотомического деления

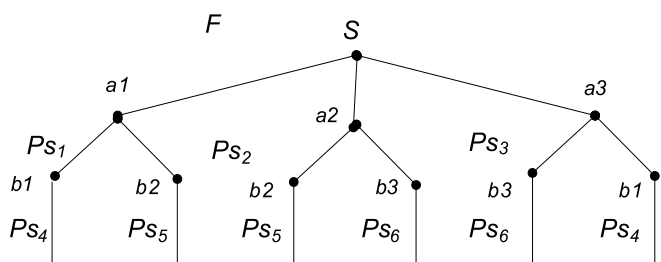


Рис. 7. Иерархия системы S , выраженная через графы фрагментов

являть не только структуру системы (см. рис. 4), но и строить иерархию такой системы (рис. 7). Дихотомический анализ можно рассматривать как направление развития системного анализа. Сложная организационная система является этапом развития сложных систем. Дальнейшим направлением исследования является подключение когнитивных моделей в схемы СОТС.

Список литературы

1. Месарович М., Такахара Н. Общая теория систем: математические основы. М.: Мир, 1978. 311 с.
2. Кудж С. А. Системный подход // Славянский форум. 2014. № 1 (5). С. 252—257.
3. Бутко Е. Я. Системный подход в формировании структуры // Славянский форум. 2017. № 2 (16). С. 25—31.
4. Монахов С. В., Савиных В. П., Цветков В. Я. Методология анализа и проектирования сложных информационных систем. М.: Просвещение, 2005. 264 с.
5. Оболяева Н. М. Системный подход к анализу качества образования // Управление образованием: теория и практика. 2012. № 3. С. 101—105.
6. Ожерельева Т. А. Системный анализ пространственной инновации // Международный журнал прикладных и фундаментальных исследований. 2013. № 12. С. 116—120.
7. Кудж С. А. Многоаспектность рассмотрения сложных систем // Перспективы науки и образования. 2014. № 1. С. 38—43.
8. Тихонов А. Н., Иванников А. Д., Соловьев И. В., Цветков В. Я. Основы управления сложной организационно-технической системой. Информационный аспект. М.: МаксПресс, 2010. 228 с.

9. Корнаков А. Н. Модель сложной организационно-технической системы // Перспективы науки и образования. 2015. № 2. С. 44—50.
10. Цветков В. Я. Дихотомический анализ сложности системы // Перспективы науки и образования. 2014. № 2. С. 15—20.
11. Раев В. К. Дихотомический метод уменьшения информационной неопределенности // Перспективы науки и образования. 2017. № 2 (26). С. 7—11.
12. Цветков В. Я. Прикладные системы // Известия высших учебных заведений. Геодезия и аэрофотосъемка. 2005. № 3. С. 78—85.
13. Воронин А. Н., Зиятдинов Ю. К., Харченко А. В. Сложные технические и эргатические системы: методы исследования. Харьков: Факт, 1997. 240 с.

14. Luhmann N. The World Society as a Social System // International Journal of General Systems. 1982. Vol. 8, no. 3. P. 131—138.
15. Luhmann N. Soziale Systeme: Grundriß einer allgemeinen Theorie, Frankfurt: Suhrkamp, 1984. (English translation: Social Systems, Stanford: Stanford University Press, 1995).
16. Цветков В. Я. Информационное соответствие // Международный журнал прикладных и фундаментальных исследований. 2016. № 1—3. С. 454—455.
17. Tsvetkov V. Ya. Opposition Variables as a Tool of Qualitative Analysis // World Applied Sciences Journal. 2014. N. 30 (11). P. 1703—1706.
18. Савиных В. П. Оппозиционный анализ в информационном поле // Славянский форум, 2016. № 3 (13). С. 243—248.
19. Цветков В. Я. Решение проблем с использованием системного анализа // Перспективы науки и образования. 2015. № 1. С. 50—55.

A. V. Buravtsev, Deputy Director, e-mail: mister_j@mail.ru, Institute of Information Technologies and Computer-Aided Design, Moscow Technological University (MIREA), Moscow, 119454, Russia,
V. Ya. Tsvetkov, D. Sc., Professor, Center for strategic analysis and development, the deputy head, e-mail: cvj2@mail.ru, Research and Design Institute of Design Information, Automation and Communication on Railway Transport, Moscow, Russia

Analysis of the Structure of a Complex Organizational and Technical System

The article analyzes the properties of a complex organizational and technical system. The article offers a method of forming the structure and hierarchy of a complex organizational and technical system. Paper shows the place of a complex organizational and technical system in a number of complex systems. The article reveals the content and features of a complex organizational and technical system. The article notes the inclusion of cognitive factors in the work of a complex organizational and technical system. The article reveals the application of the dichotomous method for constructing the structure of a complex organizational and technical system. The dichotomous method is applied in combination with the opposition method. The combination of the dichotomous and oppositional method allows one to sequentially isolate the qualitative components in the form of a complex system structure. The dichotomous method is supplemented by the method of constructing hierarchies, which uses graph theory.

Ключевые слова: system analysis, complex system, social system, complex organizational and technical system, autopoiesis, modeling, system structure, dichotomous division, hierarchy modeling

References

1. Mesarovic M., Takahara N. *General theory of systems: mathematical foundations*, Moscow, Mir, 1978, 311 p. (in Russian).
2. Kudzh S. A. System approach, *Slavic Forum*, 2014, no. 1 (5), pp. 252—257 (in Russian).
3. Butko E. Ya. System approach in the formation of the structure, *Slavic Forum*, 2017, no. 2 (16), pp. 25—31 (in Russian).
4. Monakhov S. V., Savinykh V. P., Tsvetkov V. Ya. Methodology of analysis and design of complex information systems, Moscow, Enlightenment, 2005, 264 p. (in Russian).
5. Obolyaeva N. M. A Systems Approach to the Analysis of the Quality of Education, *Educational Management: Theory and Practice*, 2012, no. 3, pp. 101—105 (in Russian).
6. Ozherelyeva T. A. System analysis of spatial innovation, *International Journal of Applied and Fundamental Research*, 2013, no. 12, pp. 116—120 (in Russian).
7. Kudzh S. A. Multidimensionality of the consideration of complex systems, *Prospects of science and education*, 2014, no. 1, pp. 38—43 (in Russian).
8. Tikhonov A. N., Ivannikov A. D., Solovyov I. V., Tsvetkov V. Ya. Basics of managing a complex organizational and technical system. Information aspect. Moscow, MaxPress, 2010, 228 p. (in Russian).
9. Kornakov A. N. Model of a complex organizational and technical system, *Perspectives of science and education*, 2015, no. 2, pp. 44—50 (in Russian).
10. Tsvetkov V. Ya. Dichotomous analysis of the complexity of the system, *Perspectives of science and education*, 2014, no. 2, pp. 15—20 (in Russian).
11. Raev V. K. Dichotomous method of reducing information uncertainty, *Perspectives of science and education*, 2017, no. 2 (26), pp. 7—11 (in Russian).
12. Tsvetkov V. Ya. Applied systems, *News of higher educational institutions. Geodesy and aerial photography*, 2005, no. 3, pp. 78—85 (in Russian).
13. Voronin A. N., Ziatdinov Yu. K., Kharchenko A. V. Complex technical and ergatic systems: research methods, Kharkov, Fact, 1997, 240 p. (in Russian).
14. Luhmann N. The World Society as a Social System, *International Journal of General Systems*, 1982, vol. 8, no. 3, pp. 131—138.
15. Luhmann N. *Soziale Systeme: Grundriß einer allgemeinen Theorie*, Frankfurt: Suhrkamp, 1984. (English translation: Social Systems, Stanford: Stanford University Press, 1995).
16. Tsvetkov V. Ya. Information correspondence, *International Journal of Applied and Fundamental Research*, 2016, no. 1—3, pp. 454—455 (in Russian).
17. Tsvetkov V. Ya. Opposition Variables as a Tool of Qualitative Analysis, *World Applied Sciences Journal*, 2014, vol. 30, no. 11, pp. 1703—1706 (in Russian).
18. Savinykh V. P. Opposition analysis in the information field, *Slavic Forum*, 2016, no. 3 (13), pp. 243—248 (in Russian).
19. Tsvetkov V. Ya. Solution of problems with the use of system analysis, *Prospects of science and education*, 2015, no. 1, pp. 50—55 (in Russian).

А. Ф. Резчиков, чл.-кор. РАН, д-р техн. наук, проф., e-mail: rezchikov1939@mail.ru, ФГБУН "Институт проблем точной механики и управления РАН", г. Саратов, 410028,

Н. В. Яндыбаева, канд. техн. наук, e-mail: nat07@inbox.ru,

Балаковский филиал ФГБОУ ВО "Российская академия народного хозяйства и государственной службы при Президенте Российской Федерации", г. Балаково,

В. А. Кушников, д-р. техн. наук, проф., e-mail: kushnikoff@yandex.ru,

В. А. Иващенко, д-р. техн. наук, проф., e-mail: iptmuran@san.ru,

А. С. Богомолов, канд. физ.-мат. наук, доцент, e-mail: alexbogomolov@yandex.ru,

Л. Ю. Филимонюк, канд. техн. наук, e-mail: filimonyukleonid@mail.ru, ФГБУН "Институт проблем точной механики и управления РАН", г. Саратов, 410028

Информационно-советующая система для моделирования и прогнозирования динамики показателей национальной безопасности на основе причинно-следственных комплексов и уравнений Форрестера

Разработано математическое обеспечение для информационно-советующей системы, предназначенной для осуществления имитационного моделирования и прогнозирования основных показателей национальной безопасности РФ. Математической основой системы является комплекс моделей, построенных с использованием аппарата системной динамики, теории причинно-следственных комплексов и регрессионного анализа. Представлен алгоритм расчета прогнозных значений показателей национальной безопасности с использованием разработанного специализированного программного обеспечения.

Ключевые слова: национальная безопасность, системная динамика, дифференциальные уравнения, объектно-ориентированное программирование

Введение

Актуальными сегодня являются исследования в области проблем перехода страны к устойчивому развитию. Концепция устойчивого развития является на сегодняшний день самой распространенной "всемирной моделью будущего цивилизации", которая предполагает повышение качества жизни граждан без ущерба (экономического, социального, экологического) для будущих поколений. Исследованиями в области перехода страны к устойчивому развитию с использованием моделей и методов мировой динамики занимались Малинецкий Г. Г., Акаев А. А., Коротаев А. В., Малков С. Ю., Подлазов А. В., Садовничий В. А. и др. [1, 2]. В целях обеспечения устойчивого и безопасного развития страны 31 декабря 2015 г. Президентом РФ был подписан Указ № 682 "О стратегии национальной безопасности Российской Федерации" [3]. Стратегия национальной безопасности России содержит приоритеты в области внутренней и внешней политики по обеспечению национальной безопасности РФ, а также оценку текущего состояния национальной безопасности РФ на долгосрочный период. В Стратегии приводятся основные показатели национальной безопасности.

Постановка задачи исследования имеет следующий вид: необходимо разработать математические модели и алгоритмы, а также программное обеспечение для имитационного моделирования и прогнозирования основных показателей национальной безопасности РФ.

1. Описание комплекса математических моделей

Для решения поставленной задачи авторами был разработан комплекс моделей, состоящий из модели системной динамики и регрессионных моделей [4–7].

В качестве моделируемых переменных в разработанной на основе системной динамики математической модели приняты показатели национальной безопасности: $X_1(t)$ — удовлетворенность граждан степенью защищенности своих конституционных прав и свобод; $X_2(t)$ — доля современных образцов вооружения, военной и специальной техники в Вооруженных Силах РФ; $X_3(t)$ — ожидаемая продолжительность жизни; $X_4(t)$ — валовой внутренний продукт на душу населения; $X_5(t)$ — децильный коэффициент; $X_6(t)$ — уровень инфляции; $X_7(t)$ — уровень безработицы; $X_8(t)$ — доля расходов в валовом внутреннем про-

дукте на развитие науки, технологий и образования; $X_9(t)$ — доля расходов в валовом внутреннем продукте на культуру; $X_{10}(t)$ — доля территории Российской Федерации, не соответствующая экологическим нормативам. Данный набор переменных не обладает полнотой характеристик изучаемой системы и моделирует некоторую проекцию всей системы на выбранное подпространство его характеристик.

Связи между моделируемыми переменными X_1 — X_{10} представлены с помощью орграфа, разработанного на основе формального аппарата причинно-следственных комплексов (рис. 1).

Представим производные моделируемых переменных X_1 — X_{10} как функции данных переменных. Производные уровней по времени $\frac{dX_i(t)}{dt}$, $i = 1, \dots, n$, модельный смысл которых — скорости изменения уровней в единицу времени, называются потоками. Тогда связи между потоками и моделируемыми переменными (уровнями) можно записать в виде систем дифференциальных уравнений следующего вида:

$$\frac{dX_i(t)}{dt} = F_i(X_1(t), \dots, X_n(t)), i = 1, \dots, n. \quad (1)$$

Особенностью подхода Дж. Форрестера в системной динамике являются упрощение и декомпозиция системы [8]. Представим функции

F_i в виде разложений в ряд по степеням $X_k(t)$, но рассмотрим только первые, линейные члены разложения. Коэффициенты при членах разложения определяются экспериментально. Система в этом случае имеет вид:

$$\frac{dX_i(t)}{dt} = \alpha_{i,0} + \alpha_{i,1}X_1(t) + \dots + \alpha_{i,n}X_n(t), \quad (2)$$

$$i = 1, \dots, n.$$

Произведения $\alpha_{i,k}X_k(t)$, $i = 1, \dots, n$, — это темпы i -го потока, t — время.

В силу сложности анализируемой системы вводится нелинейность — пусть темпы зависят от уровней:

$$\alpha_{i,k} = \alpha_{i,k}(X_1(t), \dots, X_n(t)), k = 1, \dots, n.$$

Данная зависимость имеет мультипликативный вид:

$$\alpha_{i,k}(X_1(t), \dots, X_n(t)) = \alpha_{i,k} f_{i,k,1}(X_1(t)) \dots f_{i,k,n}(X_n(t)), \quad (3)$$

$$k = 1, \dots, n.$$

Здесь $\alpha_{i,k}$ — константы из уравнения (2), каждый из множителей $f_{i,k,l}$, $l = 1, \dots, n$, зависит только от одного аргумента — "своего" уровня $X_l(t)$, при этом "базовым" значением множителей $f_{i,k,l}$

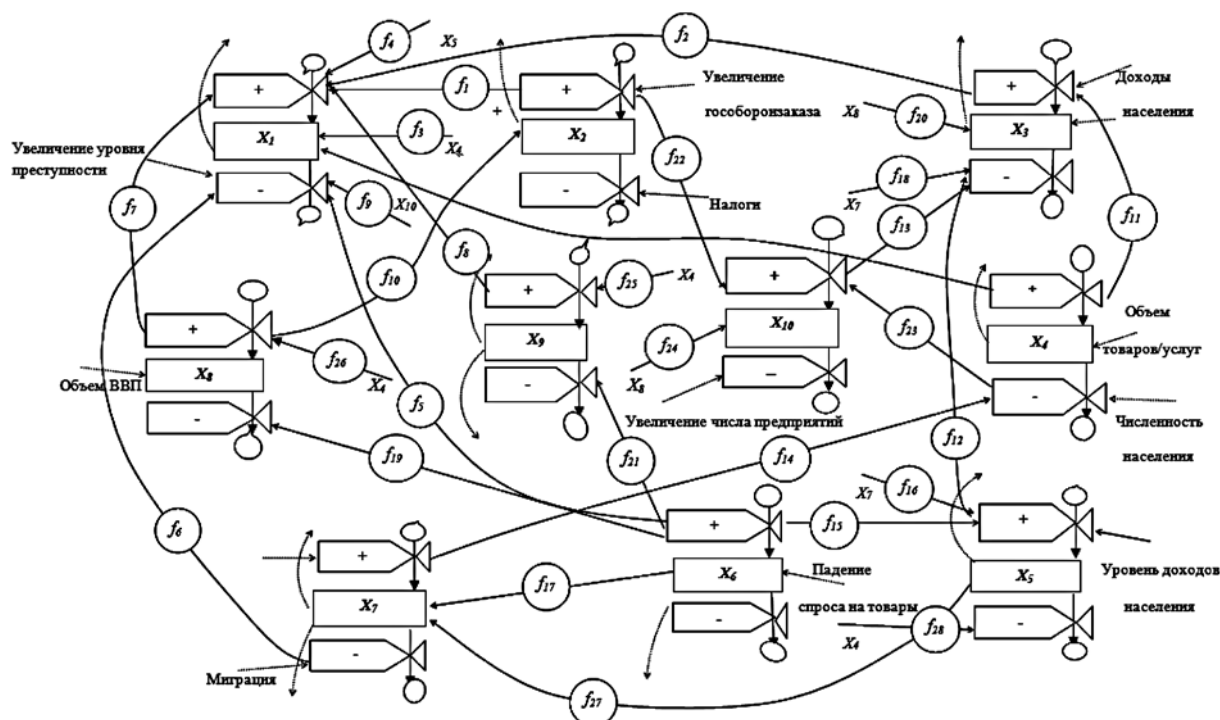


Рис. 1. Орграф причинно-следственных связей между моделируемыми переменными X_1 — X_{10}

считается 1, от которого под действием аргументов они могут отклоняться в ту или иную сторону. Окончательно модель имеет вид:

$$\frac{dX_i(t)}{dt} = \alpha_{i,0} + \sum_{k=1}^n \alpha_{i,k} \prod_{l=1}^n f_{i,k,l}(X_l(t)), X_k(t), \quad (4)$$

$$i = 1, \dots, n.$$

Система уравнений (4) дает декомпозицию исходной системы общего вида (1) [9].

Разработанная математическая модель представляет собой систему дифференциальных уравнений, описывающих состояние анализируемой системы:

$$\begin{cases} \frac{dX_1(t)}{dt} = (P_l(t) + P_{se}(t))f_1(X_2)f_2(X_3)f_3(X_4)f_7(X_8) \times \\ \times f_8(X_9) - Sh_{pr}(t)f_9(X_{10})f_4(X_5)f_5(X_6)f_6(X_7); \\ \frac{dX_2(t)}{dt} = (V(t) + SO(t))f_{10}(X_8) - I(t); \\ \frac{dX_3(t)}{dt} = (E(t) + Zd(t) + X_8(t))f_{11}(X_4)f_{20}(X_8) - \\ - (BN(t) + I(t) + U(t))f_{12}(X_5)f_{13}(X_{10})f_{18}(X_7); \\ \frac{dX_4(t)}{dt} = \left(\frac{V(t)}{P(t)} + D(t) \right) - (I(t) + U(t))f_{14}(X_7); \\ \frac{dX_5(t)}{dt} = (P(t) + Sc(t) + U(t))f_{16}(X_7) - I(t)f_{15}(X_6); \\ \frac{dX_6(t)}{dt} = (De(t) + D(t) + E(t)) - (V(t) + W(t)); \\ \frac{dX_7(t)}{dt} = (I(t) + Sc(t) + D(t))f_{15}(X_6) - \\ - (W(t) + T(t) + V(t))f_{27}(X_5); \\ \frac{dX_8(t)}{dt} = (V(t) + H(t) + Tch(t) + Sr(t))f_{26}(X_4) - \\ - (I(t) + D(t) + M(t) + P(t) + Sc(t))f_{19}(X_6); \\ \frac{dX_9(t)}{dt} = (V(t) + D(t) + Sc(t))f_{25}(X_4) - \\ - (T(t) + P(t))f_{21}(X_6); \\ \frac{dX_{10}(t)}{dt} = (PZ(t) + VZ(t))f_{22}(X_2) - \\ - Ze(t)f_{23}(X_4)f_{24}(X_8). \end{cases} \quad (5)$$

В уравнениях (5) приняты следующие обозначения: $X_1(t) \dots X_{10}(t)$ — текущие уровни значений моделируемых переменных $X_1 \dots X_{10}$; $P_l(t)$ — степень реализованности личных прав граждан (права на жизнь, права на свободу и личную неприкосновенность, права на свободу мысли и слова, права на свободу от вмешательства в личную жизнь); $P_{se}(t)$ — степень реализованности социально-экономических прав (права на жилище, права на защиту прав и свобод, права на охрану здоровья, права на труд, отдых, социальное обеспечение,

права избирать и быть избранным, права частной собственности и свободы предпринимательства, права на участие в общественной, политической жизни); $Sh_{pr}(t)$ — число зарегистрированных преступлений; $P(t)$ — численность населения; $SO(t)$ — объем гособоронзаказа; $E(t)$ — средняя заработная плата; $Zd(t)$ — доля расходов ВВП на здравоохранение; $BN(t)$ — заболеваемость населения; $U(t)$ — уровень безработицы; $De(t)$ — денежная эмиссия; $I(t)$ — инфляция; $Sc(t)$ — доходы населения; $V(t)$ — валовой внутренний продукт страны; $D(t)$ — демографические факторы (численность экономически активного населения); $W(t)$ — предложение труда; $T(t)$ — налоги; $H(t)$ — число учебных заведений; $Tch(t)$ — численность профессорско-преподавательского состава с учеными степенями и званиями; $Sr(t)$ — среднегодовой объем финансирования научных исследований; $M(t)$ — миграция; $PZ(t)$ — число промышленных предприятий; $VZ(t)$ — объем выбросов загрязняющих веществ в воду, почву, воздух; $Ze(t)$ — затраты на охрану окружающей среды [10–12].

Рассмотрим процедуру построения дифференциального уравнения, описывающего изменения переменной $X_4(t)$ — валового внутреннего продукта на душу населения:

$$\begin{aligned} \frac{dX_4(t)}{dt} &= X_4(t) \left(\left(\frac{V(t)/P(t) + D(t)}{X_4(t)} \right) - \right. \\ &- \left. \left(\frac{I(t) + U(t)}{X_4(t)} \right) f_{14}(X_7) \right) = \left(\frac{V(t)}{P(t)} + D(t) \right) - \\ &- (I(t) + U(t))f_{14}(X_7). \end{aligned} \quad (6)$$

Здесь $X_4(t)$ — текущий уровень валового внутреннего продукта на душу населения; $V(t)$ — валовой внутренний продукт; $P(t)$ — численность населения; $D(t)$ — численность экономически активного населения; $I(t)$ — инфляция; $U(t)$ — безработица. На рис. 2 приведен подграф моделируемой переменной X_4 .

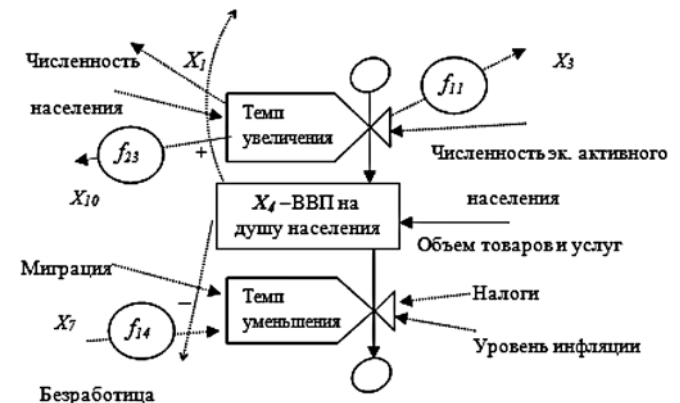


Рис. 2. Подграф переменной X_4

Величина $f_{14}(X_7)$ — функциональная зависимость валового внутреннего продукта на душу населения от уровня безработицы, используемая в уравнении (6), задается экспертами и служит для учета взаимовлияния моделируемых переменных.

В некоторых случаях описания X_4 с помощью системы уравнений (5) недостаточно ввиду невысокой точности вычислений, поэтому детализируем X_4 , описав все функции, входящие в данную переменную. Для этого выполним декомпозицию моделируемой переменной X_4 — запишем переменные, входящие в X_4 в виде следующих дифференциальных уравнений:

$$\begin{cases} \frac{dV(t)}{dt} = (Ex(t) + Im(t) + OP(t) + T(t))f_2(X_3^4)f_1(X_2^4) - (I(t) + U(t))f_3(X_5^4)f_4(X_6^4); \\ \frac{dP(t)}{dt} = R(t) - (S(t) + M(t))f_5(X_3^4); \\ \frac{dD(t)}{dt} = (P(t) + F(t)) - (VT(t) + U(t)); \\ \frac{dU(t)}{dt} = (M(t) + D(t) + Sc(t) + I(t))f_6(X_3^4)f_7(X_6^4) - (W(t) + T(t) + V(t)); \\ \frac{dI(t)}{dt} = (De(t) + SD(t)) - U(t). \end{cases} \quad (7)$$

Дифференциальные уравнения построены на основе орграфа моделируемой переменной, приведенного на рис. 3.

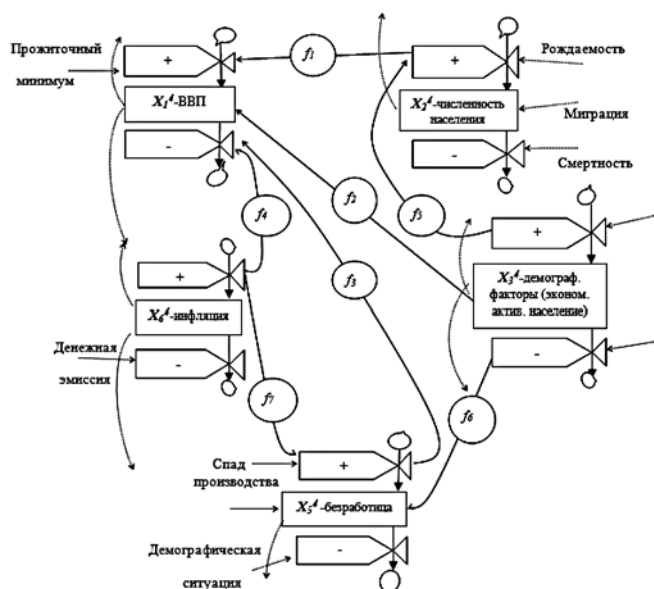


Рис. 3. Орграф моделируемой переменной X_4

В уравнении (7) приняты следующие обозначения: $Ex(t)$ — экспорт; $Im(t)$ — импорт; $T(t)$ — налоги; $OP(t)$ — объем производства товаров и услуг; $R(t)$ — рождаемость; $S(t)$ — смертность; $M(t)$ — миграция; $F(t)$ — фонд рабочего времени; $VT(t)$ — спрос на труд; $SD(t)$ — государственный долг; $Sc(t)$ — доходы населения; $W(t)$ — предложение труда; $De(t)$ — денежная эмиссия.

Для моделируемой переменной X_4 — ВВП на душу населения — рассчитаем значения показателей $V(t)$, $P(t)$, $D(t)$, $U(t)$, $I(t)$ за 2016 г. по разработанной модели (7) и сравним с расчетными значениями по регрессионной модели и ретроспективными (фактическими) значениями показателей за тот же период. Так как рассматриваемые переменные имеют разные размерность и числовой масштаб, то при составлении зависимости и систем уравнений каждое их значение указывается в отношении к аналогичному значению базового 2000 г. По разработанной модели на основе системной динамики имеем

$$\begin{aligned} \frac{dV(t)}{dt} &= (Ex(t) + Im(t) + OP(t) + T(t))f_2(X_3^4)f_1(X_2^4) - (I(t) + U(t))f_3(X_5^4)f_4(X_6^4) = \\ &= (8,04 + 10,03 + 12,7 + 7,32) \cdot 0,8 \cdot 0,4 - \\ &- (0,64 + 0,53) \cdot 0,7 \cdot 0,6 = 11,7. \end{aligned}$$

Значения остальных показателей $P(t)$, $D(t)$, $U(t)$, $I(t)$ в модели (7) вычисляются аналогично. В модели (7) используются следующие функциональные зависимости, вид которых определяется с помощью регрессионного анализа:

$$\begin{aligned} f_1(X_1^4) &= -1,4(X_1^4)^3 + 2,2(X_1^4)^2 - 0,005(X_1^4) + 0,07; \\ f_2(X_3^4) &= -0,62(X_3^4)^3 + 1,1(X_3^4)^2 + 0,64(X_3^4) - 0,088; \\ f_3(X_5^4) &= 2,1(X_5^4)^3 + 3,7(X_5^4)^2 - 2,8(X_5^4) + 1,3; \\ f_4(X_6^4) &= 1,6(X_6^4)^3 - 2,4(X_6^4)^2 + 0,004(X_6^4) + 0,96; \\ f_5(X_3^4) &= 1(X_3^4)^3 - 2,1(X_3^4)^2 + 1,5(X_3^4) - 0,14; \\ f_6(X_3^4) &= -0,1(X_3^4)^3 + 0,47(X_3^4)^2 + 0,5(X_3^4) + 0,03; \\ f_7(X_6^4) &= 1(X_6^4)^3 - 2,3(X_6^4)^2 + 2,6(X_6^4) - 0,37. \end{aligned}$$

Аналогично записываются дифференциальные уравнения для моделируемых переменных X_1 – X_3 и X_5 – X_{10} . Проверка адекватности разработанной математической модели (7) осуществляется с помощью уравнений регрессии, построенных по статистическим данным за 2000–2015 гг. [13]:

$$\begin{aligned} V(t) &= -0,0001t^4 + 0,0028t^3 + 0,0158t^2 + \\ &+ 0,2008t + 0,7244; \end{aligned}$$

Проверка адекватности разработанных моделей (7) и (8)

Моделируемые переменные	Фактические значения переменных		Прогнозные значения переменных			
			вычисленные по регрессионной модели		вычисленные по модели системной динамики	
	2015 г.	2016 г.	2016 г.	Абсолют. погреш., %	2016 г.	Абсолют. погреш., %
$V(t)$ — ВВП	11,4	11,78	12,7	7,8	11,7	0,7
$P(t)$ — численность населения	0,998	1,0	0,999	1	1,01	1
$D(t)$ — численность эк. актив. населения	1,03	1,05	1,06	0,9	1,03	1,9
$U(t)$ — уровень безработицы	0,53	0,51	0,535	4,9	0,53	3,9
$I(t)$ — уровень инфляции	0,64	0,27	0,29	7,4	0,28	3,7

$$\begin{aligned}
 P(t) &= -0,00002t^3 + 0,001t^2 - 0,0114t + 1,0096; \\
 D(t) &= -0,00002t^4 - 0,0008t^3 + 0,0099t^2 - \\
 &\quad - 0,0367t + 1,0259; \\
 I(t) &= 0,0001t^4 - 0,0048t^3 + 0,0628t^2 - \\
 &\quad - 0,3648t + 1,3706; \\
 U(t) &= -0,0000005t^4 - 0,0029t^3 + 0,0394t^2 - \\
 &\quad - 0,2298t + 1,1873.
 \end{aligned}
 \tag{8}$$

В таблице приведены прогнозные значения моделируемых переменных, вычисленных по модели системной динамики (7), регрессионной модели (8) и проведено сравнение с фактическими значениями, а также рассчитана абсолютная погрешность.

Абсолютная погрешность вычисленных по разным моделям значений моделируемых переменных при сравнении с фактическими за 2016 год не превышают 8 %, что говорит об адекватности разработанного комплекса моделей.

2. Алгоритмы и программное решение для расчета прогнозных значений показателей национальной безопасности

Для расчета прогнозных значений показателей национальной безопасности разработан следующий алгоритм.

1. Осуществляется выбор моделируемой переменной $X_1 \dots X_{10}$.
2. Выбираются значения функциональных зависимостей $f_i(X_i)$ в моделируемой переменной.
3. Задаются границы временного интервала, на котором будет осуществляться прогнозирование.
4. Выполняется расчет прогнозных значений социально-экономических показателей $X_1 \dots X_7$.
5. Рассчитываются прогнозные значения для всей модели (5) в целом — для моделируемых переменных $X_1 \dots X_{10}$ решается задача Коши методом Рунге-Кутты 4-го порядка.

6. Выполняется анализ адекватности построенной модели по регрессионной модели, ретроспективным данным и модель при необходимости корректируется.

Блок-схема алгоритма расчета прогнозных значений показателей национальной безопасности приведена на рис. 4.

Для реализации разработанного математического обеспечения была создана "Программа для моделирования и прогнозирования основных показателей национальной безопасности РФ" (сви-

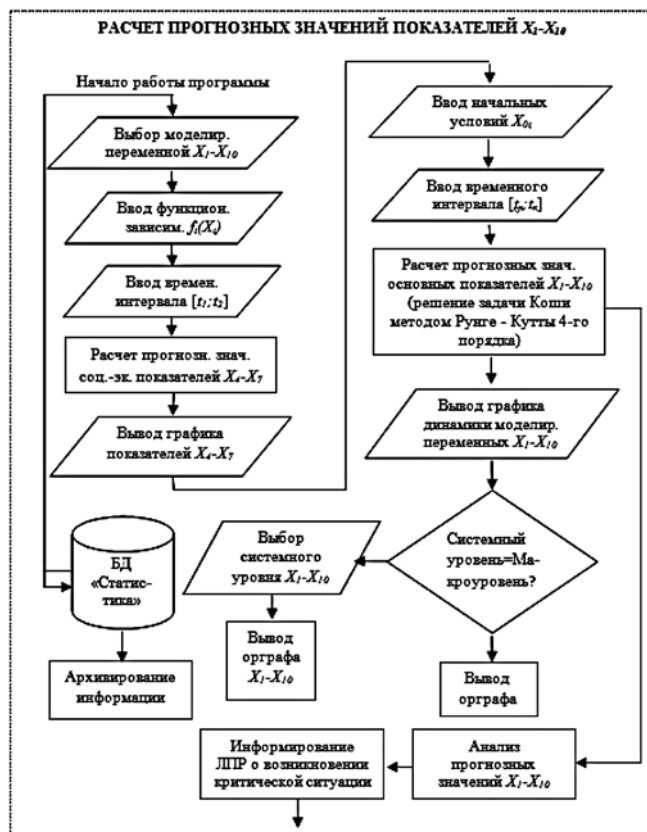


Рис. 4. Блок-схема алгоритма расчета прогнозных значений показателей национальной безопасности

детельство о государственной регистрации программы для ЭВМ № 2016661727, дата регистрации в Реестре программ для ЭВМ 19.10.2016) [14].

На рис. 5 (см. вторую сторону обложки) приведен интерфейс программы при вычислении прогнозных значений социально-экономических показателей X_4 — X_7 . Расчет прогнозных значений показателей национальной безопасности по разработанной модели (5) на временном интервале $[0; 10]$ лет приведен на рис. 6 (см. вторую сторону обложки).

Начальные условия X_{0i} — нормированные значения показателей национальной безопасности РФ в 2015 г.:

$$X_{0i} = [1 \ 3,13 \ 1,09 \ 14,18 \ 1,19 \ 0,64 \ 0,5 \ 1,45 \ 1,28 \ 1].$$

3. Информационно-советующая система для расчета прогнозных значений показателей национальной безопасности

Расчетные значения показателей национальной безопасности, определенные по модели (5), являются основой для разработки прогноза социально-экономического развития страны, который формируется на основе соответствующих прогнозов регионов [15]. Известно, что точность региональных прогнозов в силу ряда причин сегодня не превышает 60 %. Для повышения опе-

ративности принятия управленческих решений и точности прогнозирования была разработана автоматизированная информационно-советующая система, фрагмент которой приведен на рис. 7.

Информационная система позволяет автоматизировать процесс аккумулирования и обработки статистической информации, разработки прогнозных моделей и верификации полученных результатов прогнозирования.

Статистическая информация (демографические показатели, уровень цен, доходы населения, занятость, объемы промышленного и сельскохозяйственного производства и др.), необходимая для построения модели (5), поступает из территориальных отделов статистики в базу данных "Статистика", расположенную в отделе экономического анализа и прогнозирования муниципального образования. Далее с использованием "Программы для моделирования и прогнозирования основных показателей национальной безопасности РФ" строится прогнозная модель, проводится проверка ее адекватности.

Из администрации муниципального образования сформированный прогноз, являющийся основой для разработки прогноза социально-экономического развития страны, поступает в Министерство экономического развития области. Среднесрочный прогноз разрабатывается ежегодно на очередной финансовый год и плановый период Министерством экономического развития РФ на основе анализа внешних и внутренних условий социально-экономического развития РФ с учетом основных направлений бюджетной, налоговой и таможенно-тарифной политики, а также на основе данных, представляемых федеральными органами исполнительной власти, органами исполнительной власти субъектов Российской Федерации и другими участниками стратегического планирования¹.

Далее Министерство экономического развития РФ на основе сценарных условий и основных параметров среднесрочного прогноза, одобренных Правительством РФ, разрабатывает среднесрочный прогноз и направляет его на рассмотрение и согласование в Министерство финансов РФ, заинтересованные федеральные органы исполнительной власти и Центральный банк РФ, а также представляет его в Правительственную комиссию по бюджетным проектам.

В свою очередь среднесрочный прогноз, рассмотренный и одобренный Правительством РФ, Министерство экономического развития РФ до-

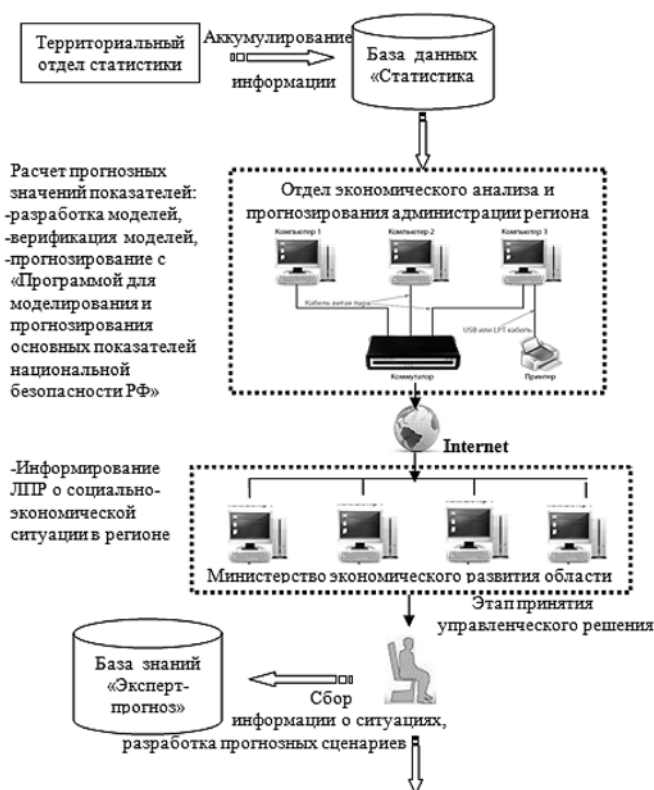


Рис. 7. Фрагмент информационно-советующей системы

¹ Постановление Правительства РФ от 14 ноября 2015 г. № 1234 "О порядке разработки, корректировки, осуществления мониторинга и контроля реализации прогноза социально-экономического развития Российской Федерации на среднесрочный период и признании утратившими силу некоторых актов Правительства Российской Федерации".

водит до сведения федеральных органов исполнительной власти, органов управления Пенсионного фонда РФ, Фонда социального страхования РФ, Федерального фонда обязательного медицинского страхования, Центрального банка РФ, органов исполнительной власти субъектов РФ и других участников стратегического планирования.

Заключение

В работе приведено описание разработанного математического обеспечения для информационно-советующей системы, предназначенной для имитационного моделирования и прогнозирования основных показателей национальной безопасности РФ, определенных в Указе Президента РФ "О стратегии национальной безопасности Российской Федерации". Построен граф причинно-следственных связей между основными показателями национальной безопасности РФ и на его основе записана система дифференциальных уравнений с использованием принципа системной динамики.

Приведен алгоритм расчета прогнозных значений показателей национальной безопасности, реализованный с помощью специализированной "Программы для моделирования и прогнозирования основных показателей национальной безопасности РФ".

Представлена структура прототипа автоматизированной информационно-советующей системы поддержки принятия управленческих решений, предназначенной для имитационного моделирования и прогнозирования показателей национальной безопасности РФ.

Список литературы

1. Акаев А. А., Садовничий В. А. Математическое моделирование глобальной, региональной и национальной динамики с учетом воздействия циклических колебаний // Социальный анализ и моделирование: сайт. URL: <http://socmodel.com/Stat'i/20113A>. (дата обращения 01.02.2017).
2. Геловани В. А., Бритков В. Б., Дубовский С. В. Россия в мировой системе (1990—2022) // Прогноз и моделирование кризисов и мировой динамики / Отв. ред. А. А. Акаев, А. В. Коротаев, Г. Г. Малинецкий. М.: ЛИБРОКОМ, 2009. С. 172—188.
3. О стратегии национальной безопасности Российской Федерации: Указ Президента РФ от 31 декабря 2015 г. № 683 // Российская газета. № 6871 от 13.01.2016.
4. Ключев В. В., Байбурун В. Б., Резчиков А. Ф., Кушников В. А., Богомолов А. С., Филимонок Л. Ю. Модели и алго-

ритмы мониторинга глобальной безопасности на основе деревьев событий // Контроль. Диагностика. 2015. № 8. С. 70—74.

5. Новожилов Г. В., Резчиков А. Ф., Неймарк М. С., Цесарский Л. Г., Кушников В. А., Богомолов А. С., Филимонок Л. Ю., Шоломов К. И. Управление авиационно-транспортными системами на основе причинно-следственных деревьев событий // Полет. 2015. № 6 (6). С. 13—17.

6. Яндыбаева Н. В., Кушников В. А. Математические модели, алгоритмы и комплексы программ для мониторинга эффективности образовательной деятельности вуза // Проблемы управления. 2015. № 1. С. 53—63.

7. Кушников В. А., Яндыбаева Н. В. Управление образовательным процессом вуза на основе модели Дж. Форрестера // Вестник Саратовского государственного технического университета. 2011. № 2 (55). С. 172—176.

8. Форрестер Дж. Мировая динамика. М.: АСТ; СПб.: Terra Fantastica, 2003. 379 с.

9. Бродский Ю. И. Лекции по математическому и имитационному моделированию. М. — Берлин: Директ-Медиа, 2015. 240 с.

10. Цвиркун А. Д., Резчиков А. Ф., Яндыбаева Н. В., Кушников В. А. Имитационное моделирование показателей национальной безопасности РФ // Управление развитием крупномасштабных систем (MLSD'2014): Сб. научных трудов / под общ. ред. С. Н. Васильева, А. Д. Цвиркуна; Ин-т проблем управления им. В. А. Трапезникова Рос. акад. наук. М.: ИПУ РАН, 2014. С. 155—163.

11. Резчиков А. Ф., Цвиркун А. Д., Кушников В. А., Яндыбаева Н. В., Иващенко В. А. Методы прогнозной оценки социально-экономических показателей национальной безопасности // Проблемы управления. 2015. № 5. С. 37—44.

12. Яндыбаева Н. В., Кушников В. А. Математическая модель для прогнозирования изменений показателей национальной безопасности России // Научное обозрение. 2015. № 10-1. С. 115—121.

13. Ключев В. В., Резчиков А. Ф., Кушников В. А., Иващенко В. А., Богомолов А. С., Филимонок Л. Ю., Яндыбаева Н. В. Математические модели для контроля, диагностики и прогнозирования состояния национальной безопасности России // Контроль. Диагностика. 2016. № 3. С. 43—51.

14. Яндыбаева Н. В. Программный продукт для моделирования и прогнозирования основных показателей национальной безопасности Российской Федерации // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. 2016. № 3. С. 94—100.

15. Яндыбаева Н. В. Модели и методы прогнозирования социально-экономического развития региона // Проблемы развития устойчивых отношений между государством, гражданским обществом и бизнесом: вызовы времени. Сборник статей по материалам конференции, посвященной Дню российской науки. Саратов: Балаковский филиал ФГБОУ ВО РАНХиГС. 2016. С. 165—173.

A. F. Rezchikov, D. Sc., Professor, Corresponding Member of the Russian Academy of Sciences, e-mail: rezchikov1939@mail.ru, Institute of Precision Mechanics and Control, Russian Academy of Sciences, Saratov,
N. V. Yandybaeva, Ph. D., e-mail: nat07@inbox.ru, Balakovsky Branch of the Russian Presidential Academy of National Economy and Public Administration, Balakovo,
V. A. Kushnikov, D. Sc., Professor, e-mail: kushnikoff@yandex.ru,
V. A. Ivashchenko, D. Sc., Professor, e-mail: iptmuran@san.ru,
A. S. Bogomolov, Ph. D., Associate Professor, e-mail: alexbogomolov@yandex.ru,
L. Y. Filimonyuk, Ph. D., e-mail: filimonyukleonid@mail.ru,
 Institute of Precision Mechanics and Control, Russian Academy of Sciences, Saratov

Information and Advisory System for Modeling and Forecasting the Dinamik of National Security Indicators on the Basis of Cause-Effect Complexes and Forrester Equations

The article presents the structure of the information and advisory system intended for imitation modeling and forecasting of the main indicators of the national security of the Russian Federation. Mathematical basis of the system is a complex of mathematical models, consisting of a model of system dynamics and regression models.

Mathematical model developed based on system dynamics models, allows formalizing the complex casual relationships between system variables. System dynamics model consists of the following elements: the system levels, which represent the accumulation of (accumulation) in feedback circuits; streams conveying the content of one level to another; making procedures that govern the rate of flow between the levels; information channels connecting procedures decisions levels. To describe the model developed using the apparatus of ordinary differential equations. As systemic levels presented the indicators of national security. To illustrate the casual relationships between system-level mathematical model developed using the graph model. Regression models are used to verify the adequacy of system dynamics models. Forecast values of national security indicators are determined from the solution of a system of nonlinear differential equations.

A heuristic numerical algorithm is developed to calculate the forecast values of national security indicators based on the developed mathematical software. The block diagram of the calculated algorithm is given. To increase the efficiency and accuracy of management decisions, a "Program for simulation modeling and forecasting of main indicators of national security" was developed. A fragment of the information and advisory system designed to support the adoption of managerial decisions is described. The information system can be used by specialists of the Department of Economic Analysis and Forecasting of Administrations of Regional and Federal Levels. She is allows to automate the process of accumulation and processing of statistical information, development of forecast models and verification of the results of forecasting.

Keywords: national security, system dynamics, differential equations, object-oriented programming

References

1. Akaev A. A., Sadovnichij V. A. *Matematicheskoe modelirovanie global'noj, regional'noj i nacional'noj dinamiki s uchetoм vozdeystviya ciklicheskih kolebanij*, Social'nyj analiz i modelirovanie: sajt. URL: <http://socmodel.com/Stat'i/20113A>. (access 01.02.2017) (in Russian).
2. Gelovani V. A., Britkov V. B., Dubovskij S. V. *Rossija v mirovoj sisteme (1990–2022). Prognoz i modelirovanie krizisov i mirovoj dinamiki*, ed. A. A. Akaev, A. V. Korotaev, G. G. Malineckij, Moscow, LIBROKOM, 2009, pp. 172–188 (in Russian).
3. *O strategii natsional'noi bezopasnosti Rossijskoi Federatsii* (utv. Ukazom Prezidenta RF ot 31 dekabrya 2015 g. № 683), Rossiiskaya gazeta. no. 6871 ot 13.01.2016 (in Russian).
4. Kljuev V. V., Bajburin V. B., Rezchikov A. F., Kushnikov V. A., Bogomolov A. S., Filimonjuk L. Ju. *Modeli i algoritmy monitoringa global'noj bezopasnosti na osnove derev'ev sobytij, Kontrol'*. Diagnostika, 2015, no. 8, pp. 70–74 (in Russian).
5. Novozhilov G. V., Rezchikov A. F., Nejmark M. S., Cesar'skij L. G., Kushnikov V. A., Bogomolov A. S., Filimonjuk L. Ju., Sholomov K. I. *Upravlenie aviacionno-transportnymi sistemami na osnove prichinno-sledstvennyh derev'ev sobytij, Polet*, 2015, no. 6 (6), pp. 13–17 (in Russian).
6. Jandybaeva N. V., Kushnikov V. A. *Matematicheskie modeli, algoritmy i kompleksy programm dlja monitoringa jeffektivnosti obrazovatel'noj dejatel'nosti vuza, Problemy upravlenija*, 2015, no. 1, pp. 53–63 (in Russian).
7. Kushnikov V. A., Jandybaeva N. V. *Upravlenie obrazovatel'nym processom vuza na osnove modeli Dzh. Forrester, Vestnik Saratovskogo gosudarstvennogo tehničeskogo universiteta*, 2011, no. 2 (55), pp. 172–176 (in Russian).
8. Forrester Dzh. *Mirovaja dinamika*. Moscow, AST; Saint-Peterburg, Terra Fantastica, 2003. 379 p. (in Russian).
9. Brodskij Ju. I. *Lekcii po matematicheskomu i imitacionnomu modelirovaniju*, Moscow–Berlin: Direkt-Media, 2015, 240 p. (in Russian).
10. Tcvirkun A. D., Rezchikov A. F., Jandybaeva N. V., Kushnikov V. A. *Imitacionnoe modelirovanie pokazatelej nacional'noj bezopasnosti RF, Upravlenie razvitiem krupnomasshtabnyh sistem (MLSD'2014), Sbornik nauchnyh trudov*, eds. S. N. Vasil'eva, A. D. Cvirikuna; In-t problem upravlenija im. V. A. Trapeznikova Ros. akad. nauk, Moscow, IPU RAN, 2014, pp. 155–163 (in Russian).
11. Rezchikov A. F., Tcvirkun A. D., Kushnikov V. A., Jandybaeva N. V., Ivashchenko V. A. *Metody prognoznoi otsenki sotsial'no-ekonomicheskikh pokazatelej natsional'noi bezopasnosti. Problemy upravleniya*. 2015, no. 5, pp. 37–44 (in Russian).
12. Jandybaeva N. V., Kushnikov V. A. *Matematicheskaya model' dlja prognozirovaniya izmenenij pokazatelej natsional'noi bezopasnosti Rossii. Nauchnoe obozrenie*. 2015, no. 10-1, pp. 115–121 (in Russian).
13. Kljuev V. V., Rezchikov A. F., Kushnikov V. A., Ivashchenko V. A., Bogomolov A. S., Filimonyuk L. Yu., Jandybaeva N. V. *Matematicheskie modeli dlja kontrolya, diagnostiki i prognozirovaniya sostoyaniya natsional'noi bezopasnosti Rossii, Kontrol'*. Diagnostika, 2016, no. 3, pp. 43–51 (in Russian).
14. Jandybaeva N. V. *Programmyj produkt dlja modelirovaniya i prognozirovaniya osnovnyh pokazatelej nacional'noj bezopasnosti Rossijskoj Federacii, Vestnik Astrahanskogo gosudarstvennogo tehničeskogo universiteta. Serija: Upravlenie, vychislitel'naja tehnika i informatika*, 2016, no. 3, pp. 94–100 (in Russian).
15. Jandybaeva N. V. *Modeli i metody prognozirovaniya social'no-ekonomicheskogo razvitiya regiona, Problemy razvitiya ustojchivyh otnoshenij mezhdru gosudarstvom, grazhdanskim obshhestvom i biznesom: vyzovy vremeni, Sbornik statej po materialam konferencii, posvjashhjonnoj Dnju rossijskoj nauki*, Balakovskij filial FGBOU VO RANHiGS, Saratov, 2016, pp. 165–173 (in Russian).

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

SOFTWARE ENGINEERING

УДК 007.51 + 004.055

А. И. Разумовский, канд. техн. наук, вед. инженер, e-mail: razumowsky@yandex.ru,
Институт проблем управления РАН, г. Москва

Информационная избыточность в разработке программных систем

Рассмотрена проблема разработки программных систем с точки зрения создания внешнего творческого контроля над информационным содержанием процесса разработки. Сделана попытка подчинить процесс создания программных систем избыточности информационного содержания посредством объединения управления как над алгоритмическими и структурными элементами, так и над любым возможным неформальным представлением данных, что способно обеспечить целостность разработки систем. Связующая роль между элементами проектирования при творческом контроле отведена эвристике доступности, которая действует как качественная оценка частоты или вероятности выбора за счет легкости воспоминания или ассоциации.

Ключевые слова: проектирование программных систем, алгоритм, структуризация, эвристика доступности, творческая активность, информационная избыточность, творческий человеческий фактор

Введение

При создании любой программной системы (ПС) определяющую роль играет комплексный мотив связности автоматической и неформальной составляющих среды разработки. В них системообразующее ядро всегда отведено неформально действующему "человеческому фактору". Известно, что использование автоматизированных систем позволяет освободить разработчика от бремени согласования деталей, позволяя ему специально сосредоточиться на творческих аспектах процесса проектирования [1]. Одновременно с таким отношением существует отчетливо ясное представление автоматизированных систем как систем "накопления вычислительного знания о машинных методах теории регулирования и управления" [2]. То есть исходя из самого понятия автоматизации можно предположить, что универсум жизненного цикла автоматизированной системы обязан включать в себя элементы творчески значимого контроля. Более того, подтверждением этому является единственная возможность обеспечить ответственность за качество результатов и их оценочно конкретную достоверность. Как однозначно подчеркивается в работе [3], "проблема достоверности результатов решения задач, в значительной мере, лежит между реальным миром и теоретическими знаниями, и посредником между ними является человек". Действительно, противоречие между формальным и качественным представлением

реальной задачи ведет к снижению содержательных характеристик программного продукта, что, в свою очередь, умеряет его надежность функционирования, а также комфорт использования.

Проблемы, возникающие при разработке и использовании ПС, так или иначе связаны с недостаточной поддержкой творческой активности человека, и в срезе имеющихся технологий проектирования ПС могут быть выражены следующим образом:

- неполнота информации в процессе проектирования и алгоритмизации;
- узость туннеля видимости результатов алгоритма;
- унификация и деперсонификация данных проекта.

Эти проблемы имеют две основные причины:

- рационализация процесса разработки ПС;
- унификация средств разработки ПС.

В работе [4] авторы сетуют: "Мы никогда не отыщем процесс, который дал бы нам возможность проектировать программы строго рациональным образом". Однако далее они отмечают: "Если мы держим в голове идеальный процесс, становится легче измерять успехи проекта", иначе говоря, творческие возможности проектировщика, считают авторы, должны быть подчинены определенным правилам процесса разработки ПС, что позволит обеспечить его управляемость. Таким образом, главным вопросом проектирования ПС остается проблема соотношения творческого и рационального в поиске, выработке

и принятии решений. Сужение области творчества разработчиков приводит к слабому профессиональному росту, безынициативности, небрежности [5]. Более того, в книге Р. Акоффа [6] выражается следующее мнение: "Ограничения, накладываемые общей логикой и абстрактной математикой на процедуры принятия решений, практически исключают возможность изучения *истинно творческих решений* и сводят науку о решениях к совокупности механических, а потому скучных и однообразных примеров принятия решений". Поскольку творческий процесс является спонтанным, необходимо учесть, что рожденная идея также может и не быть напрямую связана с решаемой проблемой [7]. Последнее означает, что при проектировании ПС необходимо заручиться возможностью зафиксировать все приходящие и имеющие информационно-техническое представление идеи. Фиксация рожденных идей, ассоциаций и оценок необходима также еще и из-за такой специфической особенности когнитивно-творческого аппарата человека, как ограниченный объем его кратковременной памяти — число Миллера: 7 ± 2 [8]. В книге акад. Ларичева О. И. [9] делается предположение о двух основных путях решения человеком задач: "а) объединение отдельных единиц информации в блоки с целью одновременного восприятия этих блоков, б) использование специальных эвристик, приспособляющих задачу к возможностям системы обработки информации человеком".

Математических моделей, адекватно конструктивно описывающих творческий человеческий фактор (ТЧФ), не существует. Конструктивно — значит, что математической формулой можно заменить определенную интеллектуальную функцию ТЧФ, или точнее, — его творческий импульс. Нет никакой надобности создавать, искать, отлаживать математические выражения, описывающие нечто, не являющееся ТЧФ или его деятельностью. Все подобные модели в ту или иную меру используют логико-вероятностный подход, основанный на субъективной экспертной оценке и, следовательно, далеко не вполне соответствуют действительности. Нами предлагается обустраивать ТЧФ индивидуально — тогда субъективность представления не только не будет помехой, но, наоборот, послужит важнейшей качественной опорой процесса участия ТЧФ в разработке. Как это организовать?

Построение избыточности информационного пространства

Общий порядок отдельного действия ТЧФ при разработке программных систем (ПС) таков: ТЧФ субъективно оценивает локальную задачу и творчески вырабатывает решение. Имеются два класса

информации, обуславливающих выработку решений, — доступная информация и воображаемая информация, или перцептивное воображение [10]. Первая характеризуется всеми видимыми элементами процесса проектирования ПС: языковым и структурным представлением алгоритма, комментариями, отладочными данными, интерфейсными элементами среды разработки. Воображаемая информация — это то, что скрыто от глаз и рук проектировщика, но ассоциативно и интуитивно через множество инсайтов прокладывает дальнейший путь развития проекта ПС. С ценностной точки зрения воображаемая информация имеет перед видимой безусловное преимущество, однако именно воображаемая информация и испытывает наибольшие потери, поскольку ее невозможно впоследствии полностью воспроизвести, удастся лишь частично припомнить антураж творческого акта выработки решения.

Мы предполагаем, что воображаемую информацию можно частично зафиксировать в целях ее многократного повторного использования. Однако для реального использования такой информации придется программно регистрировать весь спектр возникающих в сознании данных, многие из которых совпадают по форме, отличаясь по содержанию. Это означает создание избыточного множества проектных элементов (ИМПЭ).

Итак, ИМПЭ — это зафиксированные и зарегистрированные факты и фрагменты процесса проектирования. Они включают в себя сопроводительные данные — примечания, фотографии, спецификации, требования, а также комплексные данные — произвольно (индивидуально значимо) объединенные в общем графическом контексте (на экране компьютера) информационные элементы проекта в сочетании с сопроводительными данными.

ИМПЭ способны обеспечить:

- совмещение в едином контексте проектной, аналитической и результатной информации;
- представление информации в реальном времени;
- реализацию в графических 2D- или 3D-средах.

Основное предназначение ИМПЭ — выработка и принятие проектных решений посредством опоры на эвристику доступности, которая действует как оценка частоты или вероятности выбора за счет легкости воспоминания или ассоциации. В качестве воспоминаний и ассоциаций выступают ИМПЭ, которые фиксируются и могут быть воспроизведены в последующий момент времени.

Аккумулированные вместе ИМПЭ позиционируются в едином плоскостном или пространственном контексте, который мы назовем креативно-контекстной формой (ККФ), поскольку определяющим фактором ее значимости будет концентрация и фиксация всех возможных ин-

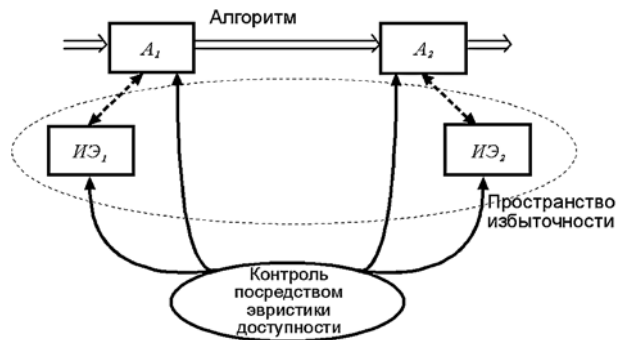


Рис. 1. Информационная модель восприятия и контроля информационной избыточности

формационных элементов в момент творческого озарения. Это своего рода фиксированные следы памяти — временные связи в коре мозга, служащие физиологической основой запоминания и воспроизведения. Кроме того, аккумуляция и фиксация ИМПЭ также соотнобразуются с теорией двойного кодирования А. Пайвио [11], согласно которой познание включает в себя деятельность двух отдельных подсистем: вербальной — обрабатывающей языковую информацию, и невербальной (образной), предназначенной для неязыковых объектов и событий.

Для выработки и принятия решения важно иметь в поле восприятия — на экране — входящие данные, аналитические данные (сравнения, уточнения, ошибки), динамику изменений определенных параметров, а также результаты, в любой возможной форме.

Рис. 1 демонстрирует информационную модель творческого восприятия ИМПЭ. У каждого алгоритмического элемента A_i существует избыточное множество образов (элементов) ИЭ, связанных с ментальным, формальным, ассоциативным представлением разработчика о нем. Кроме того, в это множество могут входить имеющие отношения к нему дополнительные вербальные или образные описания, а также данные промежуточных его состояний, например отладочная информация. При этом контроль связей внутри структуры алгоритма, соответствующий традиционному подходу алгоритмизации, сохраняется также.

Планирование и реализацию локального алгоритма намного проще проводить среди избыточного множества данных, где таковые всегда под рукой и всегда понятны разработчику, а следовательно, и комфортны для использования. Среди избыточных данных может быть теряемая информация, например, примечательная, иллюстративная или контекстно-отладочная, повторное восстановление которой затруднено, что, в свою очередь, тесно связано с понижением производительности творческого труда разработчика, а также с качеством его результата.

Пример реализации алгоритма на базе избыточности данных

Рассмотрим на предельно простом для наглядности примере влияние ИМПЭ на формирование тела алгоритма. Пример основывается на введении в состав упорядочиваемого множества элементов алгоритма определенно важных для разработчика дополнений, в общем случае создающих избыточное множество. Пример призван показать преимущества произвольного соотношения ИМПЭ, что позволит при рассмотрении всего спектра элементов информационного пространства проектирования действовать индивидуально комфортно и вырабатывать решения творчески более активно, без необходимости создания специальных моделей или следования определенной технологии разработки.

Итак, пусть требуется разработать алгоритм определения свойств треугольника, если известны длины трех его сторон. Требуется найти, к какому виду треугольников — равносоставленному, равнобедренному или произвольному — будет относиться исследуемый треугольник, заданный тремя целыми числами. Для простоты положим: дано пять отладочных троек длин сторон треугольников: (1,1,2), (2,1,3), (3,5,3), (2,2,2), (2,5,1).

Разобьем процесс разработки алгоритма на этапы, каждому из которых будет соответствовать добавление к программному коду хотя бы одного нового информационного элемента, в частности, оператора или поля данных. Первоначально нам нужно создать минимальный по содержанию алгоритм, который при этом позволит получить итоговый результат его функционирования. Этот результат, скорее всего, будет ошибочным, как и сам алгоритм, однако позволит поместить эти данные в контекст разработки на ККФ и осуществлять с помощью эвристики доступности наблюдение и выработку решений о развитии алгоритма.

Поэтапная разработка алгоритма показана на рис. 2.

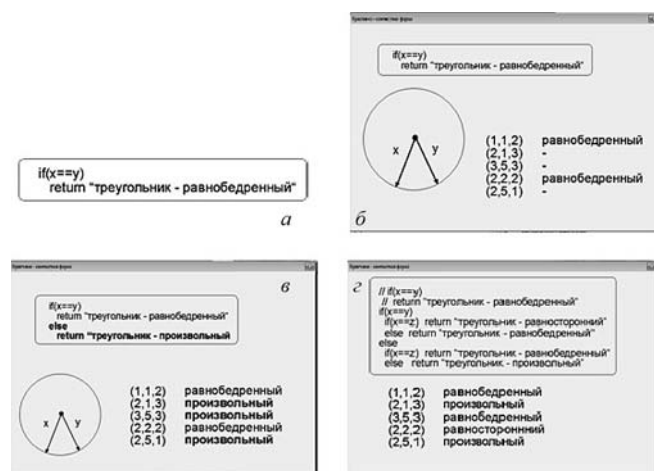


Рис. 2. ККФ последовательной разработки алгоритма

Первый этап создания алгоритма завершается реализацией всего одного оператора сравнения двух из трех входных величин (рис. 2, а). По результатам этого сравнения делается заключительный вывод об искомом свойстве треугольника. Экранная ККФ второго этапа заполняется программным кодом первого этапа, а также, для наглядности, рисунком, иллюстрирующим понятие "равнобедренный треугольник". Кроме того, осуществляется запуск скомпилированного алгоритма, и его результаты также располагаются на ККФ рядом с исходными данными и рисунком (рис. 2, б). Согласно содержанию ККФ, правильный результат получен лишь для 1-й тройки — (1, 1, 2).

Следующий этап создания алгоритма стимулируется необходимостью получения правильных результатов для остальных четырех отладочных троек. Это означает необходимость коррекции оператора, записанного на этапе 2, а. Добавим новый вывод при отрицательном ответе в операторе сравнения и поместим полученные после запуска скорректированной программы ее результаты на ККФ (рис. 2, в).

На очередном этапе разработки проводятся аналогичные действия. Кроме того, условие сравнения для третьего параметра при этом не удаляется, а комментируется неправильный код предыдущего этапа. Здесь следует обратить внимание, что в обстановке избыточности нет необходимости удалять лишние элементы — их всегда можно поместить на ККФ.

После запуска измененного алгоритма полученные результаты помещаем на очередную ККФ (рис. 2, г). Поскольку результаты удовлетворительны, можно считать процесс разработки завершенным.

Обращает на себя внимание тот факт, что накапливаемое множество избыточных информационных элементов не влияет на структуру и качество алгоритма непосредственно, поскольку вся необходимая информация в основном имеется в специализированных средствах отладки программ. Наличие ИМПЭ определяют возможность сохранения и последующего воспроизведения тех фактов, того индивидуального знания об объекте разработки, части которого стали результатом творческой выработки локальных решений и связаны между собой через эвристику доступности разработчика. Показанная на примере последовательность индивидуальных элементарных действий всегда присутствует в процессе создания ПС, однако почти вся информация, что участвует при выработке очередного локального решения, развивающего алгоритм, — отладочная, иллюстративная и т. п., — теряется, за ничтожным исключением. При обращении к структуре или коду алгоритма эти данные часто весьма трудно восстановить, а значит, для коррекции алгоритма будет необходимо вновь повто-

рить некоторые шаги разработки, сличая результаты выполнения программы с ожидаемыми. На сохранение и последующее восстановление всей возможной информации о проекте, в том числе вербальной или образной, связанной между собой посредством эвристики доступности, и ориентирован настоящий способ разработки.

Обсуждение

В описаниях методологий и техник разработки ПС сегодня используются два подхода, которым условно можно дать именованья "русский" и "западный". Для "русского" взгляда на процесс разработки ПС характерно стремление к формально-логическому описанию предмета. Для "западного", напротив, важно максимально приблизить комфорт разработки к пользователю, особенно это проявляется посредством эксплуатации принципа повторного использования программного кода, поскольку это "приводит к существенной экономии выразительных средств" [1, 12]. В "русском" подходе изъясном является упор на логику разработки, которую между тем часто нельзя сходу отследить и формализовать, а также на стремлении математически описать и обосновать личностные, нерациональные, случайные моменты, в том числе и возникающие в динамике процесса разработки ПС. Поэтому чтобы детально разобраться с причинами неудач, или наоборот, успехов проектов ПС, важно рассмотреть взаимосвязи разработчика, используемых им технологий и получаемого результата с точки зрения продуктивности и издержек.

На рис. 3 представлены пять уровней соответствия технологии и результата в разработке ПС. Нижний уровень характеризует готовую к расчету исходной задачи программу, результатом выполнения которой является целевое значение. Предшествующий исполнению ПС — уровень программирования, определяющий формально описанный на языке программирования алгоритм ПС. В рамках технологической поддержки решений на этом уровне выступают разнообразные среды разработки программного обеспечения, такие, например, как Microsoft Visual Studio.



Рис. 3. Пять информационных уровня разработки ПС

Технологический уровень проектирования представляет собой комплекс мероприятий по поддержке целостности проекта в рамках его жизненного цикла. Среди средств обеспечения такой поддержки наиболее ярко выступают, во-первых, парадигмы проектирования, типа объектно-ориентированного проектирования (ООП), а во-вторых, технологии инструментальных средств разработки, например, система Rational Rose с использованием языка UML, или стандарты проектирования SADT/DFD. Кроме того, широко применяются также техники объединения проектных с программными решениями в виде различных шаблонов проектирования.

Организационный уровень ориентирован на человеческий фактор в плане результативности и качества производимых им действий и подкрепляется методологически созданием специальных условий для получения лучшего результата разработки. Здесь могут применяться разнообразные схемы и техники взаимодействия в социально-корпоративной среде — от экстремального программирования в смысле Agile [13] до парного программирования [14].

Наконец, верхний уровень — индивидуального восприятия — тот уровень, продуктивному исследованию которого посвящен этот текст. В литературе очень немного сказано о деятельности на этом уровне и в основном это общие слова, сетование на слишком малое внимание, обращаемое на человеческий фактор вообще и, в частности, в плане его восприятия и поведенческого, творческого комфорта. Можно сделать предположение, что причиной тому — объективная сложность, которая стоит за необходимостью учета всех нюансов деятельности человека как творчески активного индивида, вместо того чтобы рассматривать его как элемент рабочей схемы или методологии.

Таким образом, с уверенностью предполагая невозможность или неэффективность использования формально-математического или формально-информационного моделирования для исследования индивидуальных способностей создания и реализации качественных ПС, мы оказываемся в ситуации, когда лишь неформальный подход позволит выработать верное решение. Насколько невозможна точная математическая модель создания и реализации ПС, настолько невозможен численный эксперимент с точными проверяемыми результатами.

Человеческому фактору в разработке ПС уделяется чрезвычайно мало внимания, хотя, казалось бы, его ведущую роль и вклад в разработку ПС невозможно отрицать. Исключениями являются, например, работы [15, 16]. Так, в работе [16] автор, известный американский методолог и консультант разработки ПС А. Коуберн, выделяет несколько особенностей человеческой природы, связанных с качеством разработки ПС: временная

синхронизация в коммуникации между людьми; непостоянство людей (в нашем случае это переключается с важностью индивидуальной выработки решения); инициативность, которая прямо влияет на успех проекта (в нашем случае — это ответственность за конкретный результат).

Надо при этом отметить, что ни в одной из книг и статей о влиянии человеческого фактора на разработку ПС не указывается творческий процесс, а лишь говорится о внешних его проявлениях, без соотнесения их в общем плане творческой инкубации и инсайта решения. Кроме того, внимание специалистов направлено больше на организацию коммуникации и внешней среды труда разработчиков, нежели на отдельные индивидуально значимые компоненты выработки решений.

На рис. 4 приведена упрощенная алгоритмическая схема операций при неформальном подходе к разработке ПС. Каждый элемент итерации выработки очередного решения начинается с его инкубации на основании представленных на ККФ ИМПЭ. После того как новое решение выработано и принято, оно оформляется в алгоритме на проектном или программном уровне и затем заносится и позиционируется на ККФ. В качестве данных, заполняющих ККФ, выступают наиболее значимые для разработчика элементы или их фрагменты. Это могут быть детали программного кода, комментарии, отладочные фрагменты, сопроводительные схемы и рисунки результата выполнения программы. Их ценность состоит не только в сохранении вообще, но и в соотнесении друг с другом определенным, важным для разработчика образом. Правильная совмещенность в избыточном контексте обеспе-

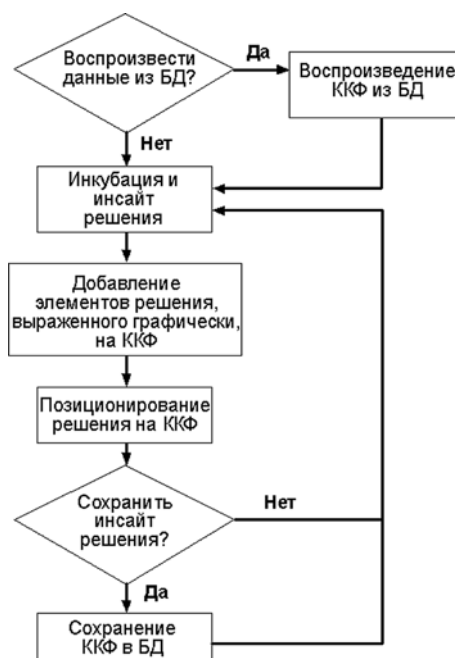


Рис. 4. Схема алгоритма действий творческого "контроллера"

чивает комфорт использования разработчиком эвристики доступности и, соответственно, инкубацию очередного решения.

Каждое решение может быть немедленно сохранено в базе данных ККФ или, при не полной заполненности ККФ, такое сохранение может быть отложено. Позиционирование решений на ККФ и их сохранение в базе данных позволит не только осуществить лучшую инкубацию последующего решения через эвристику доступности, но и воспроизвести воспоминание решения с сопутствующим антуражем. Это особенно важно при необходимости повторных возвратов к предыдущим итерациям разработки и выполняет отчасти роль расширенной кратковременной памяти, что непосредственно способствует лучшему обзору и использованию данных ККФ для новой инкубации и инсайта.

Практический образец разработки алгоритма

Вполне представить преимущества способа индивидуальной избыточности в разработке ПС, по-видимому, возможно лишь на реальном примере создания конкретного алгоритма.

Рассмотрим последовательность разработки алгоритма нахождения эквидистанты плоского контура (ЭПК). Особенность данного алгоритма заключается в необходимости хранения и использования всей возможной полноты информации о структуре исходного контура и деталях последовательного изменения алгоритма, в том числе такие данные, которые являются результатами локальных итераций проекта, а также разнообразные промежуточные расчеты. Необходимая целостность знаний об алгоритме может быть обеспечена только за счет создания такой информационно-функциональной среды, которая бы определяющим образом имела возможность накапливать, представлять и сохранять весь спектр информации.

Содержательная суть алгоритма расчета ЭПК заключается в последовательном нахождении значимых биссектрис углов между содержащими отрезки контура прямыми, а затем пересечении полученной сетки биссектрис эквидистантными к каждому из отрезков контура линиями. Полученные при последнем пересечении точки будут последовательно образовывать целевые эквидистантные контуры (рис. 5).

Проведем качественные оценки множеств информационных элементов, потребных для выработки очередного локального решения. Для хранения данных о каждом ребре исходного контура используется структура, состоящая из девяти полей, причем каждое из них применяется не только непосредственно для очередных расчетов, но и для контроля промежуточных результатов в текстовом и графическом виде.

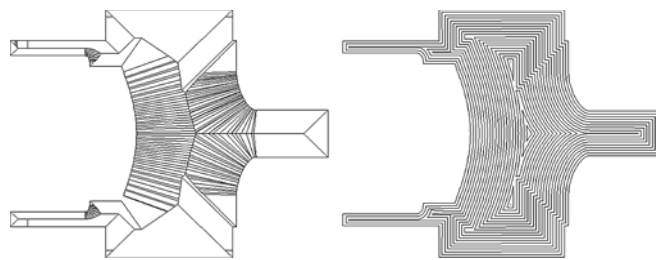


Рис. 5. Результаты выполнения алгоритма построения ЭПК

Сравним два случая сложности: в начале работы алгоритма и в середине. Ситуация начала работы алгоритма характеризуется в значительной мере лишь исходными данными, каждый из массивов полибиссектрисы (ПБ) содержит всего одну точку, исходную, поэтому наблюдение над данными для выработки решения представляет незначительные трудности. Второй случай отличается от первого насыщенными массивами ПБ, а также более сложным пониманием текущих результатов и их интерпретацией, без которой нельзя выработать и принять решение о возможности дальнейшей разработки алгоритма.

Во втором случае число необходимых для наблюдения информационных элементов возрастает примерно от 1,5 до 7 раз для контуров с числом точек не более 500. Оценим, каково приблизительно число таких элементов для формирования условий инкубации очередного решения. Для первого случая число необходимых информационных элементов составляет примерно от 10 до 20. Для второго случая таких элементов требуется, соответственно, от 15 до 140. Увеличение в 1,5...7 раз требуемых для наблюдения элементов происходит из-за того, что число точек ПБ, координаты которых должны быть учтены, значительно возрастает от нуля в начале работы алгоритма. В простейшем случае имеем минимум 7 элементов — в начале работы алгоритма, и минимум 11 — в середине его работы, так как добавляются также еще две координаты и индекс отрезка пересечения. Но в среднем важных для наблюдения элементов увеличивается сразу на 4...5 точек, по паре координат с каждой стороны сегмента контура, что означает резкое возрастание элементов наблюдения и выработки решений примерно в 5 раз.

Естественно, такие оценки делаются исходя из индивидуальных представлений о важности информационного элемента в контексте прочих. Осуществляя наблюдение, поиск и принятие решения на базе эвристики доступности, разработчик выступает как эксперт, и сделанный им выбор опирается на личную эвристическую достоверность. В этом и состоит ценность творческого подхода, в противоположность компьютерному синтезу решений, где существует жесткая система правил выбора и предпринимаемых мер.

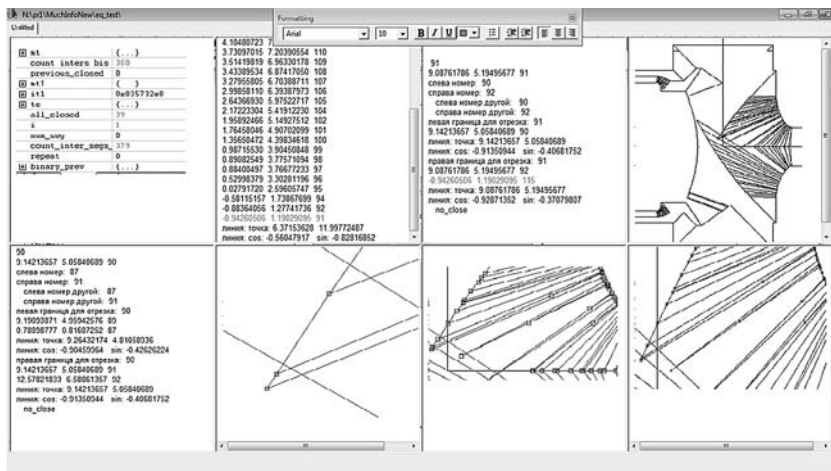


Рис. 6. ККФ разработки алгоритма ЭПК

Напротив, в условиях свободы творчества выработку решений и их оценку проводит сам разработчик на основе собственного опыта, знаний и способностей, применяемых вне технологических или аксиоматических препятствий.

Необходимость совмещения в едином контексте наблюдения наиболее важную с точки зрения разработчика информацию о некоторой проблемной ситуации, решение которой необходимо осуществить, иллюстрирует рис. 6. На этом рисунке изображена одна ККФ из полученных при проектировании алгоритма нахождения ЭПК в ситуации возникновения неожиданной ошибки на 25-м шаге итерации функционирования алгоритма.

Настоящая ККФ включает восемь слайдов для выбранных информационных элементов, каждый из которых содержит важную информацию для творческой выработки решений по исправлению возникшей ошибки функционирования. Данные четыре слайдов отображают ситуацию с графической точки зрения в разном масштабе. Еще три слайда содержат текстовую отладочную и log-информацию, относящуюся к текущим данным, связанную с пересечением ПБ. Последний слайд отображает соотнесенные разработчиком важными отладочные данные, вырезанные как часть экрана в момент возникшей ошибки из среды MS Visual Studio.

Заключение

Подводя итог, перечислим главные цели, достигаемые использованием метода информационной избыточности:

- возможность непосредственно увидеть с помощью данных на ККФ новое расширение алгоритма;
- возможность в деталях восстановить информационное содержание любого этапа разработки;
- возможность вести одновременное наблюдение над несколькими вариантами алгоритма,

и, как следствие, обеспечить качественное неформальное принятие лучших среди них решений;

- возможность наблюдать и искать решение в индивидуально значимом, конкретном контексте избыточных, совместно позиционируемых, данных.

Представленный в статье порядок организации накопления и воспроизведения ИМПЭ позволит соединить и зафиксировать в едином контексте любые графически выразимые информационные элементы разработки алгоритма ПС. Располагаемая на ККФ информация в своей совокупности содержит тот строй мыслей, который соответствует найденному

решению в момент творческой инкубации и инсайта, свидетельствующему о верности решения.

Вообразим на мгновение, что при разработке алгоритма ИМПЭ не применяются. В таком случае результатом станут многочисленные потери нюансов и атрибутов творческого инсайта, потери локальных данных проектирования и программирования, например, отладочных данных, издержки отсутствия многомерности представления информации и, наконец, потери от несогласования разноплановых данных между собой, т. е. целостности представления, что обеспечивается позиционированием разнообразных и избыточных элементов на ККФ. Все это вместе взятое потребует гораздо большего интеллектуально-творческого усилия, большего времени для нового "погружения" в содержательную суть проекта и деталей алгоритма. Это приведет к выработке и принятию менее надежных и качественных решений, а в некоторых сложных случаях, когда для их инкубации важно одновременно учесть достаточно много данных, количественно превышающих "магическое число" [8], найти правильный ответ станет и вовсе затруднено.

Представления о возможных преимуществах использования избыточности для построения качественных алгоритмов и систем своим истоком имеют книгу И. Пригожина и И. Стенгерс "Порядок из хаоса" [17]. В ней приведено множество аргументов в пользу избрания для разработки и поддержки работоспособности систем недетерминистского подхода, в частности, отсутствие возможности точного описания предмета. Приводимые в книге слова Л. Розенфельда как нельзя кстати подходят для подведения окончательной черты: "Включение спецификации условий наблюдения в описание явлений не произвольное решение, а необходимость, диктуемая самими законами протекания явлений и механизмом их наблюдения, что делает эти условия неотъемлемой частью объективного описания явлений" [17].

Список литературы

1. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++. М.: Бином, 2001, 560 с.
2. Васильев С. Н., Опарин Г. А., Феоктистов А. Г. Интеллектуальный подход к автоматизации моделирования сложных управляемых систем // Современные проблемы прикладной математики и механики: Труды межд. конф. RDAMM-2001. Новосибирск: ИВТ СО РАН, 2001. Т. 6. Ч. 2. С. 159—168.
3. Абрамова Н. А. Логический подход к анализу достоверности идентификации // Проблемы управления, 2005, № 5. С. 77—82.
4. Parnas D. and Clements P. A Rational Design Process: How and Why to Fake It. IEEE Transactions on Software Engineering, 1986. Vol. 12, N. 2.
5. Страуструп Б. Язык программирования C++. М.: Бином, 2017. 1136 с.
6. Акофф Р., Эмери Ф. О целеустремленных системах. М.: ЛКИ, 2008. 272 с.
7. Богоявленская Д. Б. Психология творческих способностей. М.: Академия, 2002. 320 с.
8. Miller G. The Magical Number Seven, Plus or Minus Two: Some Limits on or Capacity for Processing Information, Psychological Review, USA, 1956, N 63. P. 81—97.
9. Ларичев О. И. Объективные модели и субъективные решения. М.: Наука, 1987. 144 с.
10. Андерсон Дж. Р. Когнитивная психология. СПб.: Питер, 2006. 589 с.
11. Paivio A. Imagery and verbal processes. New York: Holt, Rinehart, and Winston. 1971.
12. Бек К. Экстремальное программирование: разработка через тестирование. СПб.: Питер, 2017. 224 с.
13. Cockburn Alistair. Agile Software Development, 1st edition, Addison-Wesley Professional, December 2001.
14. Cockburn A., Laurie W. The Costs and Benefits of Pair Programming // The University of Utah, 2001. 11 p. URL: <http://www.cs.utah.edu/~lwilliam/Papers/XPSardinia.PDF>.
15. Демарко Т., Листер Т. Человеческий фактор: успешные проекты и команды. СПб.: Символ-Плюс, 2005. 249 с.
16. Cockburn A. Characterizing people as non-linear, first-order components in software development // Humans and Technology, October 1999.
17. Пригожин И., Стенгерс И. Порядок из хаоса: Новый диалог человека с природой. М.: Прогресс, 1986. 432 с.

A. I. Razumowsky, Ph. D., Leading Engineer, e-mail: razumowsky@yandex.ru,
Institute of Control Sciences RAS, Moscow, Russia

Information Redundancy in the Development of Software Systems

The problem of developing software systems from the point of view of organizing external creative control over the information content of the development process is considered. An attempt was made to subordinate the process of creating software systems for redundancy of information content, by combining control over both algorithmic and structural elements, and over any possible informal data representation, which will ensure the integrity of the development of software systems. The connectivity between design elements under creative control is the accessibility heuristic that acts as a qualitative estimate of the frequency or probability of choice due to the ease of recalling or association. The study of methodological levels of software system development is carried out. It was found that the individual level of creative activity of a person is almost not studied, therefore, the completeness and integrity of the development of software systems will not be achieved. The simplified scheme of actions of the developer is given at the informal approach with use of an excessive set of design elements. For positioning in a single visual context of redundant data, a creative-contextual form is proposed that serves to concentrate and fix all possible information elements at the time of creative insight. An example of the implementation of a geometric algorithm with the help of creative-contextual forms is considered, as well as estimates of the sets of information elements necessary for the creative development of the next local solution.

Keywords: designing of software system, algorithm, structuring, accessibility heuristic, creative activity, information redundancy, creative human factor

References

1. Buch. G. *Ob'yektno-oriyentirovanny analiz i proyektirovaniye s primerami prilozheniy na C++*, Moscow, Binom, 2001, 560 p. (in Russian).
2. Vasil'yev S. N., Oparin G. A., Feoktistov A. G. Intel'ektnyy podkhod k avtomatizatsii modelirovaniya slozhnykh upravlyayemykh sistem, *Sovremennyye problemy prikladnoy matematiki i mekhaniki: Trudy mezhd. konf. RDAMM-2001*. Novosibirsk: IVT SO RAN, 2001, vol. 6, path. 2, pp. 159—168 (in Russian).
3. Abramova N. A. Logicheskiy podkhod k analizu dostovernosti identifikatsii, *Problemy upravleniya*, 2005, vol. 5, pp. 77—82 (in Russian).
4. Parnas D. and Clements P. 1986. A Rational Design Process: How and Why to Fake It. *IEEE Transactions on Software Engineering*, 1986, vol. 12, no. 2.
5. Strastrup B. *Yazyk programmirovaniya S++*, Moscow, Binom, 2017, 1136 p. (in Russian).
6. Akoff R., Emeri F. *O tselestremennyykh sistemakh*. Moscow, LKI, 2008. 272 p. (in Russian).
7. Bogoyavlenskaya D. B. *Psikhologiya tvorcheskikh sposobnostey*. Moscow, Akademiya, 2002. 320 p. (in Russian).
8. Miller G. *The Magical Number Seven, Plus or Minus Two: Some Limits on or Capacity for Processing Information*, Psychological Review, USA, 1956, no. 63, pp. 81—97.
9. Larichev O. I. *Ob'yektivnyye modeli i sub'yektivnyye resheniya*. Moscow, Nauka, 1987. 144 p. (in Russian).
10. Anderson Dzh. R. *Kognitivnaya psikhologiya*. SPb.: Piter, 2006. 589 p. (in Russian).
11. Paivio A. *Imagery and verbal processes*. New York: Holt, Rinehart, and Winston. 1971.
12. Bek K. *Ekstremal'noye programmirovaniye: razrabotka cherez testirovaniye*. SPb.: Piter, 2017. 224 p. (in Russian).
13. Cockburn Alistair, *Agile Software Development*, 1st edition, December, Addison-Wesley Professional, December 2001.
14. Cockburn A., Laurie W. The Costs and Benefits of Pair Programming, *The University of Utah*, 2001. 11 p. URL: <http://www.cs.utah.edu/~lwilliam/Papers/XPSardinia.PDF>.
15. Demarko T., Lister T. *Chelovecheskiy faktor: uspeshnyye proyekty i komandy*. SPb.: Simvol-Plyus, 2005. 249 p. (in Russian).
16. Cockburn A. Characterizing people as non-linear, first-order components in software development, *Humans and Technology*, October 1999.
17. Prigozhin I., Stengers I. *Poryadok iz khaosa: Novyy dialog cheloveka s prirodoy*. Moscow, Progress, 1986, 432 p. (in Russian).

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

INFORMATION TECHNOLOGIES IN EDUCATION

УДК 004.94

Р. Р. Загидуллин, д-р техн. наук, проф. каф. polysoft@list.ru,
Уфимский государственный авиационный технический университет,
Д. С. Занин, канд. пед. наук, исполнительный директор,
First Business School, Уфа

Определение интегральной оценки профессиональной квалификации студента вуза на различных этапах обучения

Представлена численная методика определения интегрального показателя готовности специалиста (бакалавра, магистра) с точки зрения требований профессионального стандарта. Данная методика может быть использована в качестве инструмента сравнения подготовки студентов различных вузов и текущего показателя, востребованного в процессе обучения, как для преподавателей, руководства вузов, так и для самих студентов. Эта оценка может быть реализована на сервере университета, а также с помощью облачных технологий и базируется на концепции цифрового пространства РФ.

Ключевые слова: нормирование оценок, ранжирование дисциплин, математическая модель, весовые коэффициенты, значимость трудовой функции, коэффициент значимости дисциплины, интегральная оценка специалиста, интегральная оценка профессиональной квалификации

Введение

В процессе обучения часто возникает задача определения как итогового, так и промежуточного показателя уровня подготовленности студента с точки зрения требований реального сектора экономики (работодателей). Требования по подготовке специалистов лишь описательно представлены в государственных образовательных стандартах (ГОС) и отражены в стандартных показателях уровней квалификации [1–3], которые, в конечном итоге складываются из того множества дисциплин, которые освоил студент в процессе обучения. В то же время как сами уровни, так и их содержание (компетенции), носят описательный характер, требующий численной оценки. Эти численные данные в дальнейшем могут быть полезны в различных случаях как при сравнении успеваемости отдельных студентов (внутри группы, факультета), так и при сравнении между вузами.

1. Постановка задачи

В течение L лет любой i -й студент (бакалавр, магистр) из всего множества студентов вуза $S\{s = 1, \dots, n\}$ изучает множество неповторяющихся дисциплин $D\{D_k, k = 1, \dots, m\}$. В этом множе-

стве дисциплин присутствуют все дисциплины, по которым в итоге выставляются экзаменационные и зачетные оценки (дифференцированные и недифференцированные зачеты).

По каждой из дисциплин учащийся в конце курса или — j -го года обучения ($j \in L(1, \dots, 4) \vee \vee L(1, \dots, 6)$), вне зависимости от числа семестров, отводимых на освоение дисциплины, получает оценку $b(D_k)$ — отлично, хорошо, удовлетворительно или неудовлетворительно. При этом использование множества $L(1, \dots, 4)$ означает, что оценка ищется для бакалаврской ступени обучения, а использование множества $L(1, \dots, 6)$ означает, что оценка ищется для полного цикла обучения (бакалаврского + магистратура).

Каждая k -я дисциплина из множества $D\{D_k, k = 1, \dots, m\}$ в какой-то мере уникальна и отличается от любой другой дисциплины приоритетом (важностью) с точки зрения требований к специалисту согласно ГОС — $\alpha(D_k)$.

Таким образом, любой i -й студент в любой момент времени всего процесса обучения характеризуется неким вектором

$$F(i, j, b(D_k, k = 1, \dots, m)). \quad (1)$$

Необходимо на базе вектора (1), представленного в общем виде, составить интегральную

оценку подготовки специалиста в численном виде, которая могла бы быть вычислена на любой момент времени процесса обучения.

Введем следующее тождество для численного значения нашего вектора F :

$$F(100\%) \equiv F(1), \quad (2)$$

из которого следует, что значение F мы будем вычислять для вещественного интервала $[0...1]$, которое в дальнейшем всегда можно перевести в процентное отношение.

Рассмотрим предлагаемое решение задачи, которое состоит из нескольких процедур.

2. Процедура нормирования оценок

Положим, что любая оценка из ранее озвученного кортежа {отлично, хорошо, удовлетворительно, неудовлетворительно} для каждой дисциплины из D может быть представлена в виде следующих вещественных чисел:

$$\begin{aligned} b_k(\text{отлично} &= \frac{5}{5}); \\ b_k(\text{хорошо} &= \frac{4}{5}); \\ b_k(\text{удовлетворительно} &= \frac{3}{5}); \\ b_k(\text{неудовлетворительно} &= \frac{0}{5}). \end{aligned} \quad (3)$$

Значение оценки "неудовлетворительно", представленное в виде $\frac{0}{5}$, обусловлено тем, что данная оценка, по сути, означает, что k -ю дисциплину из D студент не знает и, естественно, знания по ней отсутствуют, а значит, и учитывать эту оценку не имеет смысла до тех пор, пока студент не пересдаст данную дисциплину на лучшую оценку.

Те же значения, которые представлены в (3), касаются и дифференцированных зачетов. Если же зачет не является дифференцированным, то его оценка может приобретать только два значения — "1" в случае успешной сдачи и "0" в случае неуспешной сдачи.

Таким образом, на любой момент времени обучения любой i -й студент по любой k -й дисциплине может иметь оценку от 0 до 1. При этом важно отметить, что значение "0" не всегда означает, что он получил по данной дисциплине оценку "неудовлетворительно". Например, если в учебном году изучается десять дисциплин из 60 по полному курсу (для множества $L(1, ..., 6)$), то студент, благополучно сдавший все экзамены за первый курс, имеет по оставшимся 50 дисциплинам оцен-

ку, равную "0", так как он еще не приступал к их изучению. В то же время, если учебное заведение поддерживает систему подготовки дисциплин и сдачи по ней экзаменов и зачетов экстерном, то студент может иметь оценку по дисциплине, читаемой на старших курсах, отличную от "0".

Таким образом, любой i -й студент из всего множества S в процессе обучения может характеризоваться множеством оценок по всем дисциплинам:

$$S_i(b_k, k = 1, ..., m). \quad (4)$$

3. Ранжирование дисциплин

Для ранжирования дисциплин будем использовать метод, который применяется для ранжирования частных критериев в функционале [4], который представим с учетом весовых оценок α_k при каждой дисциплине, которые отражают важность данной дисциплины с точки зрения компетенций специалиста:

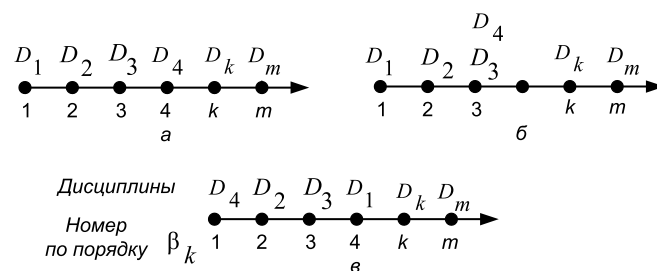
$$D = (D_1\alpha_1, D_2\alpha_2, ..., D_m\alpha_m), \quad (5)$$

$$\alpha_1 + \alpha_2 + ... + \alpha_m = 1, \quad (6)$$

где (6) — уравнение нормировки.

Для этого все дисциплины множества $D(D_k, k = 1, ..., m)$ в целях определения весовой оценки α_k сравниваются между собой попарно, что существенно повышает достоверность назначения α_k по сравнению со случаем, когда α_k изначально определяется как среднее арифметическое от мнения группы экспертов (экспертная оценка). Графическая интерпретация попарного сравнения дисциплин представлена на рисунке.

Представим натуральный ряд чисел от единицы до m , на котором расположим наши дисциплины D_k . Положим, что чем большую важность имеет та или иная дисциплина D_k в множестве $D(D_k, k = 1, ..., m)$, тем большее значение имеет ее местоположение на этой оси и тем большее значение имеет коэффициент α_k . При этом могут возникнуть самые различные случаи упорядочения. Например, если коэффициент α_k каждой последующей дисциплины больше коэффициента предыдущей дисциплины, то номера дисци-



Графическая интерпретация попарного сравнения дисциплин

плин будут совпадать с их номером по порядку расположения на оси (рисунок а).

Возможен случай, когда некоторые коэффициенты ряда дисциплин одинаковы (рисунок б). Отметим, что номер дисциплины не имеет никакого отношения к ее номеру по порядку расположения на оси, который обозначим через β_k (рисунок в). При этом номер β_k будет выполнять еще и роль веса дисциплины, так как изначально известно, что чем большую важность имеет та или иная дисциплина D_k , тем большее значение имеет ее местоположение на числовой оси.

Таким образом, вводим возможность экспертного попарного сравнения коэффициентов дисциплин по принципу "больше или равно", например, $(\alpha_k > \alpha_l)$, $(\alpha_k = \alpha_l)$; $k, l \in [1, \dots, m]$, т. е. с минимальной долей неопределенности.

Такой вариант назначения оценок наиболее характерен для случаев с большим числом различных по своей природе дисциплин. В данном случае эксперты могут просто расположить дисциплины D_k на оси чисел так, как это выполнено на рисунке. Такое расположение дисциплин можно формально отразить заполнением следующей матрицы оценок предпочтения между дисциплинами

$$B = \left\{ \beta_k > \beta_l \right\}; \beta_k \in D_k, \beta_l \in D_l; k, l \in [1, \dots, m], \quad (7)$$

т. е. необходимо указать, какие дисциплины предпочтительнее других или равны им по значимости. Если $\beta_k > \beta_l$, то дисциплина D_k имеет вес больший, чем D_l , если $\beta_k = \beta_l$, то значения дисциплин D_k и D_l одинаковы.

Рассмотрим далее метод получения весовых коэффициентов α_k с учетом матрицы (7) для интересующего нас случая целочисленного ранжирования.

Необходимо выстроить все частные дисциплины из множества

$$D = (D_1, D_2, \dots, D_m) \quad (8)$$

по принципу перечисления, и каждая частная дисциплина будет иметь какую-либо целочисленную оценку, равную значимости в силу отношений (7). При этом какая-либо минимальная оценка $x_k = 1$ соответствует дисциплине с минимальной значимостью, и некоторые дисциплины могут иметь равные оценки в случае равенства их значимости в (7).

При требовании целочисленности к оценкам x_k получим следующую модель в виде задачи целочисленного линейного программирования (ЗЦЛП) с неизвестными значениями x_k :

$$W = \sum_{k=1}^m x_k \rightarrow \min; \quad (9)$$

$$\begin{cases} x_k - x_l \geq 1, & \beta_k > \beta_l; k, l \in [1, \dots, m]; \\ x_k - x_l = 0, & \beta_k = \beta_l; k, l \in [1, \dots, m]; \end{cases} \quad (10)$$

$$x_k \geq 1; k = [1, \dots, m]. \quad (11)$$

После нахождения значений x_k вычисляются искомые значения весовых коэффициентов

$$\alpha_k = \frac{x_k}{\sum_{k=1}^m x_k} \quad (12)$$

при выполнении условия нормировки (6).

Данный метод дает однозначное распределение весовых коэффициентов согласно "баллам" — оценкам x_k , которые получили те или иные частные дисциплины в процессе ранжирования, и предназначен для тех случаев, когда удастся лишь попарно сравнить значимость дисциплин, входящих в (8).

Разберем модель (9)—(12) более подробно.

Целевая функция W , по сути, выполняет роль ограничения сверху, так как при ее отсутствии получим бесконечное число решений. Выражение (10) в общем случае представляет собой не что иное, как систему линейных алгебраических неравенств вида

$$a_k x_k - a_l x_l \geq b_r, \quad (13)$$

где значения a_k равны либо 1, либо -1 , а значения b_r равны 1. Согласно (7) могут встречаться и равенства типа $a_k x_k - a_l x_l = 0$, но акцентируем внимание именно на том, что основное в (10) — это неравенства. Тем более что система (10) должна содержать хотя бы одно неравенство, в противном случае задача нахождения коэффициентов α_k сводится к весьма тривиальному решению $\alpha_k = 1/m$ для всех k . Число всех уравнений (неравенств и равенств) в (10) определится как

$$u = \sum_{k=1}^m (m - k), \quad (14)$$

где m — число дисциплин.

Таким образом, значение коэффициента α_k в (12) вычисляется по аналогии с критерием оптимальности Шепли [5], когда выигрыш каждого игрока равен его среднему вкладу в благосостояние некой общей коалиции, по сути — математическому ожиданию. По той же аналогии с распределением выигрышей в (9)—(12) оговаривается, что все значения переменных x_k должны быть не меньше единицы, тем более, что изначально мы воспользовались числовой осью, где единица — минимальное знакоместо для частных дисциплин (см. рисунок).

Задача (9)—(12) легко решается с помощью различных математических пакетов, например, с помощью пакета символьной математики Maple [6], что показано в работе [4].

Для облегчения сравнения пар дисциплин (7) удобнее всего использовать вспомогательный метод, позволяющий сравнивать данные пары с точки зрения выполняемых трудовых функций.

Допустим, что для той или иной профессии имеется некое конечное число трудовых функций $V(1, \dots, v)$.

Для каждой трудовой функции введем табл. 1 с перечислением критериев и коэффициентов значимости трудовой функции профессии.

Для каждого критерия имеется свой коэффициент значимости t_3 , например, для l -й трудовой функции — t_{3l} .

Коэффициенты будут иметь свои веса (от 16 до 1), определяющие важность критерия. При анализе дисциплины с точки зрения трудовой функции коэффициент может быть как натуральным числом из последовательности {16, 8, 4, 2, 1}, если дисциплина отвечает критерию в полной мере,

так и значением "0", если дисциплина не отвечает данному критерию. Представленное распределение весов построено по следующему принципу.

Наиболее важным критерием с точки зрения трудовой функции является критерий t_{31} , наименее важным — критерий t_{35} . При этом критерий t_{31} , по сути, является поглощающим к остальным критериям, т. е. если k -я дисциплина по отношению к l -й трудовой функции имеет коэффициент $t_{3lk1} = 16$, то априори считается, что она удовлетворяет всем остальным критериям. Но если по результатам анализа окажется, что $t_{3lk1} = 0$, то сумма всех остальных коэффициентов критериев не должна превышать значения этого коэффициента. То есть должно выполняться следующее условие по подмножествам нашей последовательности:

$$\begin{aligned} t_{3lk1} &> (t_{3lk2} + t_{3lk3} + t_{3lk4} + t_{3lk5}), \\ t_{3lk2} &> (t_{3lk3} + t_{3lk4} + t_{3lk5}), \\ t_{3lk3} &> (t_{3lk4} + t_{3lk5}), t_{3lk4} > t_{3lk5}. \end{aligned} \quad (15)$$

Для каждой k -й дисциплины будет множество таких таблиц (см. табл. 1), соответствующих множеству трудовых функций V . По каждой k -й дисциплине определяется общая сумма коэффициентов l -й трудовой функции $\sum_{z=1}^5 t_{3lz}$, а для всего множества трудовых функций V коэффициент значимости k -й дисциплины t_{3k} определится как сумма

$$t_{3k} = \sum_{l=1}^v \sum_{z=1}^5 t_{3lzk}, k = 1, \dots, m, \quad (16)$$

где v — число трудовых функций.

Следовательно, по каждой k -й дисциплине мы получаем суммарную оценку, которая была получена при сравнительном анализе дисциплины с основными критериями (см. табл. 1), множества трудовых функций, определяющих профессиональную подготовленность.

Тогда выражение (7) можно переписать следующим образом:

$$\begin{aligned} B = \left\{ \beta_k > \beta_l, t_{3k} > t_{3l} \right\}; \\ \beta_k \in D_k, \beta_l \in D_l; k, l \in [1, \dots, m]. \end{aligned} \quad (17)$$

Таким образом, благодаря анализу, представленному выше (см. табл. 1), и значения (16) становится не только легче сравнивать дисциплины по важности в (7), но и оценивать это сравнение с точки зрения множества трудовых функций V для любой профессии, для которой можно построить это множество. При этом существенно повышается достоверность сравнения дисциплин путем замены в (7) оценки экспертной природы на подробный анализ дисциплин с точки зрения критериев значимости трудовой функции профессии.

Таблица 1

Таблица критериев значимости
трудовой функции профессии

№ дисциплины	Критерии значимости трудовой функции l ($l \in V$)	Коэффициент значимости
1	Дисциплина полностью осваивает трудовую функцию с получением конкретного продукта и предусматривает освоение смежных направлений и производство инновационных продуктов	$t_{31l} \in \{16, 0\}$
2	Дисциплина предусматривает практическое освоение трудовой функции и реализуется на практике с учетом полученного производственного продукта	$t_{32l} \in \{8, 0\}$
3	Дисциплина предусматривает практическое и лабораторное освоение по данной трудовой функции	$t_{33l} \in \{4, 0\}$
4	Дисциплина предусматривает теоретическое освоение (знания) по данной трудовой функции	$t_{34l} \in \{2, 0\}$
5	Дисциплина не предусматривает освоение данной трудовой функции (соответственно слабо связана с получением данной квалификации)	$t_{35l} \in \{1, 0\}$
	Сумма по дисциплине	$\sum_{z=1}^5 t_{3lz}$

При создании системы неравенств (7) и (17) могут возникнуть определенные проблемы. Дело в том, что ввиду большого числа дисциплин появляется большое число неравенств, а значит, велика вероятность получения хотя бы в одном неравенстве случайной ошибки, что может привести к появлению несовместной системы неравенств, т. е. система не будет решаться. Для этого случая необходимо использовать алгоритм, представленный в работе [4]. Данный алгоритм снимает проблему совместности линейных алгебраических неравенств и имеет весьма невысокую вычислительную сложность.

4. Интегральная оценка готовности специалиста

Определив на предыдущих этапах (см. пп. 2 и 3) экзаменационные оценки и получив коэффициенты значимости α_k для всех дисциплин, определим на основе множества (4) и оценки α_k интегральную оценку готовности i -го студента как специалиста на основе следующей зависимости:

$$F_i = \sum_{k=1}^m b_k \alpha_k. \quad (18)$$

Таким образом, на основе общего описательного представления (4) мы получили численный вид интегральной оценки, которая может быть вычислена для любого студента. Данная оценка, согласно (3) и (6), имеет максимальное значение, равное "1", и поэтому, согласно коммутативности выражения (2), может быть переведена в процентное отношение, например,

$$F_i = 0,95 \equiv F_i = 95 \ \%.$$

Необходимо дополнить, что матрица (7), а также соответствующий ей анализ с помощью табл. 1 составляются выпускающей кафедрой на основе анализа той или иной дисциплины D_k из всего множества дисциплин D , множества компетенций, представленных в ГОС и требований соответствующего уровня квалификации [1], а также анализа дисциплины по отношению к трудовым функциям (см. табл. 1). Результирующие интегральные оценки F_i вычисляются деканатами факультетов и могут быть размещены на общевузовском сервере, доступном не только для преподавателей и студентов, но также для работодателей.

Интегральная оценка готовности специалиста, полученная с помощью предложенной методики, резко отличается от традиционного способа определения итоговой оценки как средней арифметической величины от суммы оценок всех дисциплин, поскольку за счет коэффициентов α повышает значения баллов тех дисциплин, которые являются более важными и определяют основной

базис той или иной специальности или направления. Кроме того, необходимо отметить, что процедуры попарного сравнения дисциплин по важности, что представлено выражениями (7) и (17), должны осуществлять ведущие специалисты кафедры, отвечающие за выпуск специалистов по тому или иному направлению или специальности. В этом случае они выполняют роль экспертов.

5. Интегральная оценка профессиональной квалификации

В рамках предложенного системного подхода можно попытаться оценить таким же образом профессиональную квалификацию студента на различных этапах обучения.

В профессиональном стандарте есть такое понятие, как квалификация (необходимый набор знаний, умений, опыта для выполнения трудовой функции), являющаяся некой единицей стандарта и имеющая девять уровней. Предполагается, что в рамках обучения по максимальной программе (магистратура) студент, успешно закончивший магистратуру, квалифицируется шестым уровнем. Каждый уровень определяется конечным множеством таких понятий, как:

$$Q: \{ \text{Полномочия и ответственность, Характер умений, Характер знаний} \}. \quad (19)$$

В данном случае, в отличие от множества критериев, приведенных в табл. 1, элементы множества (19) являются ситуационными по отношению к занимаемой должности и конкретным требованиям работодателя. Для руководящего звена важнее "Полномочия и ответственность", а для остальных элементов множества характер знаний или умений определяется содержанием уровня "б" национальной рамки квалификации [1]. По сути дела мы должны дифференцировать значение F_i , полученное в (18) на три следующих составляющих

$$F_i = F_i^1 + F_i^2 + F_i^3. \quad (20)$$

В принципе, в (20) нас вполне устроят процентные значения для каждого элемента множества $(F_i^1 + F_i^2 + F_i^3)$, поскольку именно эти типы данных чаще всего важны работодателям. Рассмотрим методику их получения.

Любая дисциплина в ГОС определена множеством компетенций $G(1, \dots, g)$, в которых указаны в той или иной мере приведенные выше элементы множества (19). Составим табл. 2, где будем сравнивать компетенции той или иной дисциплины с элементами множества (19).

Если на пересечении той или иной дисциплины и столбца с соответствующим значением (16)

Таблица 2

Анализ компетенций

№	Дисциплины	Компетенции	Полномочия и ответственность	Характер умений	Характер знаний
1	Дисциплина 1	Компетенция 1 ₁	{0, 1}	{0, 1}	{0, 1}
		...	{0, 1}	{0, 1}	{0, 1}
		Компетенция 1 _g	{0, 1}	{0, 1}	{0, 1}
...	...	...	...	...	...
2	Дисциплина 2	Компетенция 1 ₁	{0, 1}	{0, 1}	{0, 1}
		...	{0, 1}	{0, 1}	{0, 1}
		Компетенция 1 _g	{0, 1}	{0, 1}	{0, 1}
...	...	...	...	...	...
k	Дисциплина k	Компетенция 1 ₁	{0, 1}	{0, 1}	{0, 1}
		...	{0, 1}	{0, 1}	{0, 1}
		Компетенция 1 _g	{0, 1}	{0, 1}	{0, 1}
...	...	...	...	...	...
			Q_1	Q_2	Q_3

рассматриваемая компетенция дополняет этот элемент, т. е. является важной, то ставится оценка "1", в противном случае — "0". В итоге для каждого элемента множества (19) мы имеем суммы компетенций Q_i ($i = 1, 2, 3$):

$$Q_i = \sum_{q=1}^1 \sum_{j=1}^k Q_{jq}, i \in \{1, 2, 3\} \quad (21)$$

с общей суммой

$$Q = \sum_{i=1}^3 Q_i. \quad (22)$$

Зная частные значения (21) и общую сумму (22) для каждого значения Q_i можно определить процентное отношение относительно (22). Например, если $Q_i = 120$, а частные значения образуют множество {45, 35, 40}, то получим множество

$$Q\{Q_1 = 37,5 \%, Q_2 = 29,1 \%, Q_3 = 33,3 \%\}.$$

В последующем, относительно нашего раннего примера (см. п. 4) — $F_i = 0,95 \equiv F_i = 95 \%$, можно использовать следующую запись, содержащую по каждому i -му студенту информацию как по интегральной оценке, так и по ее процентной дифференциации относительно уровня квалификации Q , например

$$F_i(0,95) = F_i^1 + F_i^2 + F_i^3 = \{F_i^1(Q_1 = 37,5\%), F_i^2(Q_2 = 29,1\%), F_i^3(Q_3 = 33,3\%)\}. \quad (23)$$

В выражении (23) в дальнейшем можно учесть в виде параметра F_i^4 также такие личностные особенности психологии специалиста, как лидерство и способность к руководству коллективом. Это очень важный вопрос с точки зрения, прежде всего, работодателя, поскольку не каждый высококлассный специалист может быть еще и талантливым руководителем. При этом F_i^4 необходимо определять по методикам психологов. Главное при этом — выйти на численные показатели и не пытаться интегрировать все в один показатель (необходимо оставить выражение (23) как множество ввиду разнородности показателей).

Выводы

В данной работе впервые предложен метод, когда оценка готовности специалиста, представленная лишь описательно, за счет нормирования множества экзаменационных и зачетных оценок и за счет предложенной математической модели класса ЗЦЛП, позволяющей вычислить весовые оценки важности на множестве дисциплин, с учетом множества компетенций, приобретает численный вид в рамках концепции цифрового представления информации в РФ. При этом учитываются успеваемость студента и количественный вклад дисциплины в квалификацию студента с точки зрения освоения всех трудовых функций, предусмотренных профессиональными стандартами, что может быть использовано в дальнейшем для сравнения, последующего численного анализа подготовки специалистов и анализа работодателем при приеме на работу. Эта информация может быть доступна в режиме on-line как на сервере университета, так и с помощью облачных технологий как студентам и преподавателям, так и потенциальным работодателям.

Список литературы

1. Ливанов Д. А., Топилин М. А., Шохин А. Н. Национальная рамка квалификаций в Российской Федерации. 2012. URL: www.misis.ru/Portals/0/UMO/Proekt%20NRK.docx
2. Федеральный закон № 122-ФЗ от 02.05.2015 "О внесении изменений в Трудовой кодекс Российской Федерации и статьи 11 и 73 Федерального закона "Об образовании в Российской Федерации".
3. Федеральный закон № 238-ФЗ от 03.07.2016. "О независимой оценке квалификации".
4. Загидуллин Р. Р. Планирование машиностроительного производства. Старый Оскол. Изд-во ТНТ, 2013. 392 с.
5. Оуэн Г. Теория игр. М.: Мир, 1971. 230 с.
6. Говорухин В. Н., Цибулин В. Г. Введение в Maple. Математический пакет для всех. М.: Мир, 1997. 208 с.

Determination of Integral Assessment of Professional Qualifications University Student at Different Stages Training

The article presents a numerical method for determination of the integral index of professional readiness (BA, MA) with respect to a professional standard. This technique can be used both as a tool for comparing the training of students from various universities and the current index in demand in the learning process, both for teachers, university management, and for the students themselves. This assessment can be accessed both on the university server and using cloud technologies. Is based on the concept of the digital space of the Russian Federation.

Keywords: rationing estimates ranging disciplines, mathematical model, the weights, the importance of labor function, the significance of the coefficient of discipline, integral evaluation specialist, an integrated assessment of professional qualifications

References

1. **Livanov D. A., Topilin M. A., Shohin A. N.** Nacional'naja ramka kvalifikacij v Rossijskoj Federacii. Moscow, 2012. URL: www.misis.ru/Portals/0/UMO/Proekt%20NRK.docx (in Russian).
2. **Federal'nyj zakon.** Federal'nyj zakon № 122-FZ ot 02.05.2015 "O vnesenii izmenenij v Trudovoj kodeks Rossijskoj

Federacii i stat'i 11 i 73 Federal'nogo zakona "Ob obrazovanii v Rossijskoj Federacii" (in Russian).

3. **Federal'nyj zakon № 238-FZ** ot 03.07.2016. "O nezavisimoj ocenke kvalifikacii". (in Russian).

4. **Zagidullin R. R.** Planirovanie mashinostroitel'nogo proizvodstva. Staryj Oskol. Izd-vo TNT, 2013. 392 p. (in Russian).

5. **Oujen G.** Teorija igr. Moscow, Mir, 1971. 230 p. (in Russian).

6. **Govoruhin V. N., Cibulin V. G.** Vvedenie v Maple. Matematiceskij paket dlja vseh. Moscow, Mir, 1997. 208 p. (in Russian).

УДК 519.876.5

А. С. Чирцов, д-р техн. наук, проф., e-mail: alex_chirtsv@mail.ru,
СПбГЭУ "ЛЭТИ" им. В. И. Ульянова (Ленина), РПГУ им. Герцена,

Д. А. Козуненко, Заместитель Генерального директора по информационным технологиям,
e-mail: kozunenko@svega-computer.ru, ООО "СВЕГА-Компьютер"

Использование технологий создания стереоскопических изображений при численном моделировании сложных физических систем для лекционных демонстраций и учебных исследований

Рассматриваются варианты эффективного использования простой и широкодоступной для системы образования технологии создания стереоскопических изображений для визуализации результатов интерактивного компьютерного моделирования сложных систем и происходящих в них процессов, основанного на принципах физического объектно-ориентированного моделирования. Анализируются возможности использования подхода для организации массового индивидуализированного образования в области точных наук, включающего привлечение наиболее подготовленных и мотивированных обучаемых к активным формам творческого изучения учебного материала с элементами самостоятельного научного исследования. Описанный подход иллюстрируется на примере использования рассматриваемых технологий для сопровождения курса релятивистской электродинамики.

Ключевые слова: 3D-моделирование, стереоскопическая визуализация, объектно-ориентированное моделирование, популяризация современной физики, релятивистская электродинамика, пространство Минковского, 4D-визуализации, моделирование сложных систем

Введение

Исторические, культурные и общечеловеческие последствия беспрецедентного скачка в развитии информационных, мультимедийных

и телекоммуникационных технологий, произошедшего на рубеже XX и XXI тысячелетий, пока не поддаются адекватной оценке. Компьютерные технологии проникают во все области жизни и деятельности нашей цивилизации, открывая

кажущиеся сегодня неограниченными и далеко неоднозначными перспективы в таких областях, как автоматизация интеллектуальной деятельности и генерация виртуальной реальности. Описанные процессы в полной мере коснулись системы образования [1].

Еще на первых этапах ее информатизации авторами был сформулирован принцип целесообразности использования — ИТ и мультимедийных технологий в учебной сфере: применение сложных технических решений в образовании целесообразно лишь в тех случаях, когда они открывают новые (по сравнению с традиционными подходами) возможности в обучении [2]. На сегодняшний день такими прорывными для образовательной сферы областями являются компьютерное моделирование, организация массового индивидуализированного обучения и использование технологий виртуальной реальности для наиболее полной реализации педагогических идей по двум первым направлениям [3].

В настоящее время основное внимание в области 3D-визуализаций посвящено сложным и дорогостоящим системам виртуальной реальности и различным вариантам голографических технологий, преимущественно ориентированным на использование в индустрии развлечений. Развивающиеся подходы мало приемлемы для сферы образования не только по экономическим причинам, но и в связи с понятной настороженностью преподавателей-практиков (потенциальных разработчиков качественного учебного контента) по отношению к техническим средствам, используемым в областях, в известной степени противостоящих решению задач качественного фундаментального обучения. В связи с этим оказывается актуальной задача создания относительно простого, недорогого, надежного и безопасного оборудования для стереоскопических визуализаций электронного образовательного контента, доступность которого позволит начать широкомасштабную деятельность по созданию оригинальных электронных учебных 3D-ресурсов и их апробацию в реальном учебном процессе в целях внедрения новых форм активного обучения, сопровождающихся максимально глубоким погружением в виртуальный мир учебной среды.

1. Постановка задачи

Интенсивное развитие численного моделирования в современной учебно-научной сфере обусловлено не только его существенно большими по сравнению с аналитическими методами возможностями расчетов структур и эволюций сложных систем, но и большим удобством в представлении результатов этих расчетов. Решение актуальных сегодня задач современной физики, квантовой химии, астрономии, теории сложных систем и

других сильно математизированных дисциплин даже в немногочисленных случаях, когда оказываются возможными аналитические решения или их асимптотики, как правило, оказываются весьма громоздкими и трудными для восприятия, а их традиционное представление путем записи на доске требует временных затрат, плохо совместимых с современными тенденциями сокращения аудиторных часов, отводимых на изучение дисциплин или представление научных докладов. Численное моделирование, особенно в случае использования интерактивных вариантов его реализации, наряду с существенной экономией времени на выполнение расчетов, позволяет существенно повысить удобство представления его результатов. Речь идет не только о наглядных демонстрациях виртуальных анимированных моделей моделируемых систем, но и о существенно более абстрактных вариантах визуализации результатов: многомерные графики, их дополнения динамически изменяющимися во времени зависимостями отображаемых на них соотношений, демонстрации особенностей протекания процессов для наблюдателей из других систем отсчета или визуализация в абстрактных пространствах, отличающихся от 3D-координатного (импульсном, фазовом, Фурье-образов и т.д.).

Следует учитывать, что численное моделирование с приближенной к реальным зрительным образам визуализацией результатов не должно рассматриваться как попытка подмены реальных экспериментальных демонстраций и, тем более, экспериментальных исследований виртуальными компьютерными симуляциями. Речь идет об альтернативе, существенным образом дополняющей и частично облегчающей традиционное для теоретического рассмотрения аналитическое описание.

В практике создания учебных и научных демонстраций доминирующая роль до сих пор принадлежит использованию 2D- и квази3D-графиков (последнее подразумевает создание различных аксонометрических проекций виртуальных, генерируемых лишь в числовой форме 3D-сцен на двумерную поверхность экрана). В случае сложных систем такие проекции не всегда дают полное и адекватное представление реальной системы. Проблема частично снимается в случае использования программных средств, обеспечивающих возможность трехмерных вращений сцен квази3D-графики, но и это решение не всегда оказывается приемлемым, например, в случае динамичной временной эволюции моделируемой системы. Что же касается использования аксонометрических 2D-проекции систем с большей размерностью (например, четырехмерного псевдоевклидова пространства — времени Минковского), то восприятие подобных визуализаций, как правило, проигрывает по простоте и формам представления информации.

Введем следующие понятия и определения.

Под *физической моделью системы* будем понимать упрощенную совокупность элементов системы и их взаимодействий друг с другом и внешними объектами, сохраняющую совокупность характерных черт и свойств, подлежащих изучению или обсуждению.

Под *численной моделью* будем подразумевать компьютерную программу (или комплекс программ), реализующую представление объекта, системы или понятия в форме, отличной от реальной, но приближенной к алгоритмическому описанию, включающей и набор данных, характеризующих свойства системы, и динамику их изменения со временем [4].

Под *объектно-ориентированным моделированием* (OOM) в соответствии с принятой терминологией [5] будем понимать модификацию объектно-ориентированной технологии применительно к моделированию сложных многокомпонентных динамических систем, представленных языком моделирования Modelica и реализованных в явном виде в пакетах AnyLogic, MvStadium, Ptolemy II, неявно — в Simulink, Stateflow и других компонентах пакета MATLAB.

Термин *физическое объектно-ориентированное моделирование* (PhOOM) был предложен в связи с разработкой электронных конструкторов — высокоинтерактивных программ, позволяющих пользователю самостоятельно создавать оригинальные численные модели сложных физических систем, собирая их непосредственно на экране компьютера из "физических примитивов" — программных объектов, соответствующих хорошо разработанной в физике иерархии организации элементов физических систем [6].

Простота и физическая наглядность, положенные в основу PhOOM-подхода к моделированию достаточно сложных и разнообразных физических систем, позволили поставить задачу обеспечения массовой доступности электронных конструкторов для создания оригинальных разработок новых моделей пользователями, не являющимися специалистами не только в OOM, но и в области аналитического описания физических систем. Эта задача требует создания не только относительно простого, дружелюбного и легко осваиваемого интерфейса электронных конструкторов, но аналогично простых и наглядных средств отображения результатов численного моделирования. В связи с этим целесообразно определить терминологию для наиболее важных для физики способах отображения результатов моделирования.

Двумерным изображением (2D-графикой) будем называть возникающие в результате оцифровки фото- или видеоизображений или численного моделирования существенно плоские растровые или векторные визуализации сцен, представимые в виде двумерных массивов и до-

пускающие вращения только вокруг осей, перпендикулярных плоскости монитора или экрана. Типичными примерами двумерных изображений являются графики скалярных функций одной переменной или проекции сцен реального пространства на плоскость.

Квазитрехмерным изображением (3D-графикой) назовем плоские аксонометрические визуализации сцен, представимые в виде трехмерных массивов и допускающие вращения вокруг любой из трех координатных осей, сопровождающихся взаимными перемещениями составляющих сцену объектов.

Стереоскопическим изображением (3D +-графикой) будем называть изображения, создающие реальное ощущение объема в результате использования различных технологий вне зависимости от вида применяемых технологий (использование стереопар, голографических методов и т. д.).

В статье рассматривается новая сравнительно простая и отличающаяся низкой ресурсоемкостью технология создания компьютерных стереоскопических изображений и обсуждаются варианты ее использования в современном образовании, включая перспективы совмещения с технологиями автоматизированной разработки интерактивных физических моделей на базе PhOOM-подхода.

Математические модели, используемые при создании стереоскопических визуализаций результатов PhOOM

Численное моделирование конфигураций и временных эволюций физических систем в рамках PhOOM осуществляется в результате расчетов конфигураций гравитационных ($\mathbf{g}$), электромагнитных ($\mathbf{E}$) и магнитных ($\mathbf{B}$) полей (и/или их потенциалов) от источников — распределенных в пространстве масс ρ_M , плотностей электрических зарядов ρ_q и плотностей токов $\mathbf{j}$, задаваемых в виде совокупностей графических примитивов:

$$\begin{cases} \mathbf{g}(\mathbf{R}) = -G \int_V \rho_M(\mathbf{r}) \frac{\mathbf{R} - \mathbf{r}}{|\mathbf{R} - \mathbf{r}|^3} dV(\mathbf{r}) - G \sum_j M_j \frac{\mathbf{R} - \mathbf{r}_j}{|\mathbf{R} - \mathbf{r}_j|^3}; \\ \mathbf{E}(\mathbf{R}) = \int_V \rho_q(\mathbf{r}) \frac{\mathbf{R} - \mathbf{r}}{|\mathbf{R} - \mathbf{r}|^3} dV(\mathbf{r}) + \sum_j q_j \frac{\mathbf{R} - \mathbf{r}_j}{|\mathbf{R} - \mathbf{r}_j|^3}; \\ \mathbf{B}(\mathbf{R}) = \int_V \left[\frac{\mathbf{j}(\mathbf{r})}{c}, \frac{\mathbf{R} - \mathbf{r}}{|\mathbf{R} - \mathbf{r}|^3} \right] dV(\mathbf{r}) + \sum_j \frac{I_j}{c} \int_l \left[d\mathbf{l}_j, \frac{\mathbf{R} - \mathbf{r}_j}{|\mathbf{R} - \mathbf{r}_j|^3} \right], \end{cases} \quad (1)$$

где $\mathbf{R}$ — радиус-вектор точки наблюдения, для которой проводится расчет полей; $\mathbf{r}$ — текущие точки в объемах, занимаемых источниками полей, по которым осуществляется интегрирование в случае непрерывных распределений источников или суммирование для дискретных распределений.

Движение составляющих систему частиц рассчитывалось в соответствии с известными уравнениями движения для релятивистской некантовой частицы с массой покоя m , импульсом p , движущейся со скоростью v :

$$\begin{cases} \frac{d\mathbf{p}}{dt} = \mathbf{F}(\mathbf{r}, \mathbf{v}, t), \mathbf{v} = \mathbf{p}(m^2 + \alpha^2 \mathbf{p}^2)^{-1/2}, \frac{d\mathbf{r}}{dt} = \mathbf{v}; \\ \mathbf{F}(\mathbf{r}, \mathbf{v}, t) = m\mathbf{g}(\mathbf{r}, t) + q\mathbf{E}(\mathbf{r}, t) + \\ + \alpha[\mathbf{v}, \mathbf{B}(\mathbf{r}, t)] - \eta(\mathbf{r}, \mathbf{v}, t)\mathbf{v} + \mathbf{F}_{rad} \end{cases} \quad (2)$$

или их классическими аналогами, получаемыми в нерелятивистском пределе $\alpha = 1/c \rightarrow 0$. В уравнении (2) помимо носящих фундаментальный характер гравитационных и электромагнитных сил учтены диссипативные силы вязкого ($-\eta\mathbf{v}$) и радиационного ($\mathbf{F}_{rad}$) трения. Для последней предусмотрен выбор одного из двух приближений (Лоренца и вязкого трения).

Получаемые в результате численного моделирования конфигурации систем использовались для формирования стереопар. Последние представляли собой пару стандартных квазитрехмерных визуализаций сцены, развернутых вокруг вертикальной оси на углы, подбираемые при настройке для получения максимального стереоскопического эффекта. Вращения сцены вокруг координатных осей осуществлялись с помощью стандартных матриц поворота:

$$\hat{R}(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{pmatrix}.$$

2. Методы решения комплексной задачи ГПР

Для решения комплексной задачи создания простого в использовании и доступного для широкой номенклатуры образовательных учреждений компьютерного комплекса автоматизированной подготовки интерактивных учебных моделей сложных физических систем с их стереоскопической визуализацией оправдано использование двух технологий: физического объектно-ориентированного моделирования (PhOOM) и многополосного оптического фильтра (МОФ).

В рамках первой из упомянутых технологий осуществляется построение, настройка и редактирование моделируемой системы, численное моделирование ее эволюции в режиме реального времени демонстрации и формирование последовательностей двух изображений для создания и воспроизведения стереопар.

Формирование стереопар и их воспроизведение удобно осуществлять в

рамках технологии МОФ, позволяющей создавать качественные стереоизображения без использования дорогостоящего оборудования.

Представляется оправданным кратко остановиться на каждой из технологий отдельно.

2.1. Автоматизированная разработка виртуальных моделей физических систем в рамках PhOOM-технологии

В основе идеи электронных конструкторов лежит общая логика построения физики, стремящаяся к максимально упрощенному описанию многообразной действительности путем ее представления в виде совокупности элементов, число базовых типов которых невелико. Практически реализуемым приближенным подходом является представление тел в виде совокупностей материальных точек или их систем, поведение которых описывается классическими уравнениями динамики (или их релятивистскими аналогами). Взаимодействия между материальными точками в рамках такого подхода могут описываться с помощью силовых полей, корректируемых в зависимости от конфигурации моделируемой системы. Именно в рамках такого подхода открывается принципиальная возможность автоматизации разработки широкого класса моделей классической физики.

Сама идеология электронного конструктора, подразумевающего возможность изменения числа и типов объектов, включаемых в моделируемую систему, хорошо коррелирует с принципами объектно-ориентированного программирования, на базе которого осуществлялась разработка [7].

Для реализации этой идеи было признано целесообразным создание двух основных базовых семейств классов — "частицы" и "поля" (*_Particle*, *_Field*). Третий базовый класс (*_Window*) разрабатывался для обеспечения диалогов с пользователем и визуализации результатов моделирования в запрашиваемой им форме (рис. 1). Современная версия программы содержит более ста до-

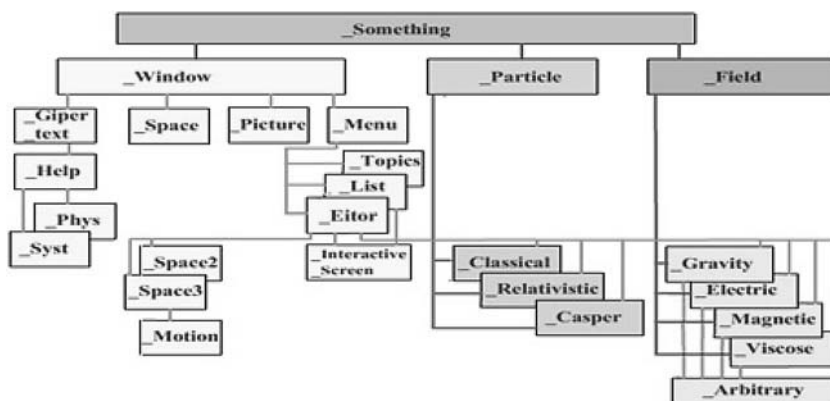


Рис. 1. Основные классы программы-конструктора "Движение заряженных частиц в силовых полях: иерархия классов первой версии"

черных классов объектов, принадлежащих этим трем базовым семействам.

Класс *_Particle* является основой для построения электронных моделей частиц с редактируемым набором исходных свойств и меняющимися кинематическими характеристиками. В методах класса предусмотрена возможность получения объектами данных от полей (объектов класса *_Field*) о состоянии последних и передачи окнам анимации и построения графиков по их запросам собственных кинематических характеристик вместе с графическими образами, устанавливаемыми пользователем. Дочерние классы *_Classic* и *_Relativistic* обеспечивают моделирование поведения макрочастиц в приближениях классической и релятивистской физики. Для этого на каждом шаге эволюции виртуальной системы осуществляется автоматическая генерация собственных уравнений движения каждого объекта и их численное интегрирование по самонастраиваемым алгоритмам, содержащимся во внутренних методах классов.

Объекты семейства *_Field* передают "частицам" информацию о внешних силовых полях, определяемых при построении моделей, осуществляют опрос "частиц" об их состояниях и на основе полученных данных корректируют свои состояния в соответствии с уравнениями (1). Описанный протокол обмена данными осуществляет "физический" вариант учета взаимодействий. Помимо него была предусмотрена возможность введения "нефизических взаимодействий" (парных и однонаправленных) путем организации прямых диалогов внутри пар "частиц", указываемых при составлении модели. Последнее существенно расширяет функции конструктора (в частности, обеспечивая возможность создания серий поэтапно усложняющихся демонстраций), иллюстрирующие методически важный метод уточнения физических моделей путем включения в нее все более слабых взаимодействий. Возникающие при втором подходе "дополнительные степени свободы" также полезны при создании демонстраций явлений, описание которых выходит за рамки классической физики (например, эффекты запаздывания взаимодействий).

В ходе моделирования фокус активности поочередно передается каждому из объектов сгенерированной системы. Активный объект опрашивает все принципиально способные повлиять на его эволюцию объекты об их состояниях, после чего самостоятельно вырабатывает "стратегию своего поведения" на очередном шаге

интегрирования в зависимости от собственной настройки и методов его класса. В результате осуществляется численное интегрирование системы уравнений движения типа (2).

Конструктор допускает моделирование движения в гравитационных, электрических и магнитных полях, изменяющихся в пространстве и времени по определяемым пользователем законам. Допускается учет действия диссипативные силы, включая радиационное трение.

Интегрирование уравнения движения (1) методом Рунге — Кутты четвертого порядка осуществляется в результате "взаимодействий" всей совокупности составляющих виртуальную систему объектов (рис. 2).

Использованный подход к моделированию не требует составления систем дифференциальных уравнений, описывающих систему в целом, или разработки каких-либо глобальных алгоритмов для численного моделирования каждой конкретной созданной системы. Любое редактирование модели практически не изменяет ее описания: уничтожение, добавление или изменение любого элемента системы учитывается автоматически через изменение потоков данных, которыми обмениваются объекты виртуальной системы в ходе диалогов, имитирующих взаимодействия. Описанные особенности являются отличительной чертой объектно-ориентированного моделирования.

Отдельного обсуждения требует вопрос о соответствии результатов численного моделирования,

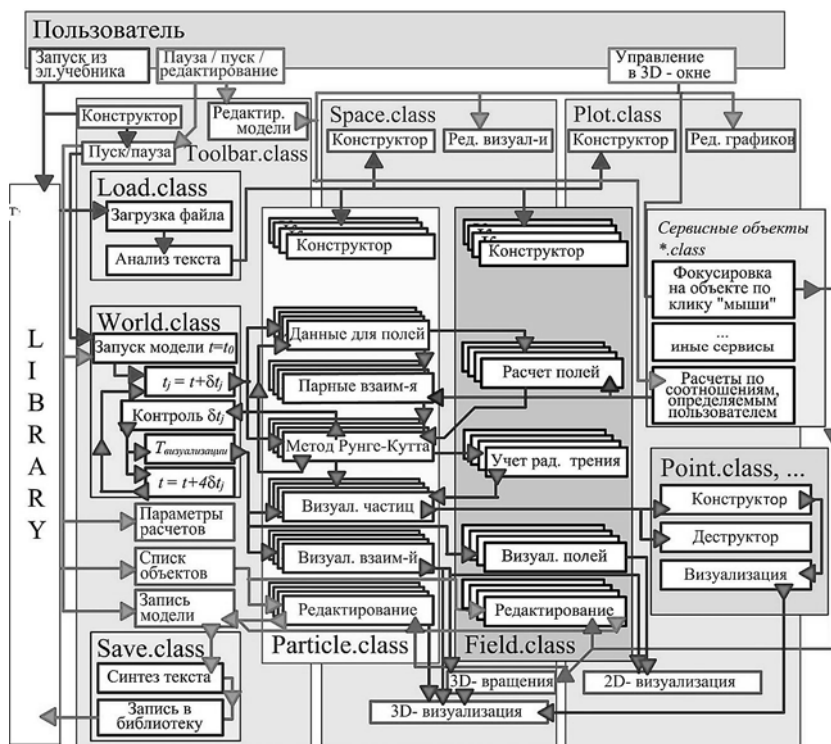


Рис. 2. Упрощенная схема алгоритма моделирования эволюции виртуальной физической системы программы-конструктора "Частицы в силовых полях"

осуществляемого с помощью конструируемых пользователем виртуальных электронных аналогов физических систем, их реальным свойствам и характеристикам. Сформулированная проблема распадается на две: обеспечение корректности и необходимой точности вычислений в ходе численного моделирования и проблема принятия решения о физической правильности и соответствии реальной физической системе самой виртуальной модели и алгоритмов ее функционирования.

Что касается первой проблемы, то точность и корректность вычислений обеспечивается использованием апробированных алгоритмов интегрирования уравнений движения и автоматизированным выбором шага интегрирования. Эффективным методом контроля качества моделирования являются многочисленные сопоставления его результатов с аналитическими решениями в тех, как правило, предельных случаях, когда последние оказываются возможными. Для реализации такой возможности в конструкторы введены специальные объекты, поведение которых определяется задаваемым пользователем алгоритмом.

Проблема контроля правильности построения самой модели реальной физической системы является принципиально сложной. Предоставляемая пользователю практически неограниченная свобода в построении виртуальных физических систем допускает возможность создания демонстраций, заведомо не соответствующих физической реальности. Проблема не может быть решена путем включения в систему программы-конструктора соответствующего модуля внутреннего контроля правильности построения физической модели. Во-первых, сама идея создания проверяющей программы подразумевает наличие исчерпывающего знания о законченной теоретической картине окружающего мира, что невозможно. Во-вторых, наличие такой системы сделало бы невозможными весьма полезные с методической точки зрения демонстрации систем, содержащих гипотетические физические объекты (магнитные монополи, кварки и т.д.). Последнее лишило бы компьютерное моделирование одного из основных его преимуществ перед реальным экспериментом.

Визуализация результатов моделирования осуществляется с помощью окон, число и настройка которых определяется автором модели, но может быть скорректирована пользователем. В существующей версии предусмотрена возможность открытия лишь одного окна с квазитрехмерной анимацией и возможностью произвольного 3D-вращения сцены. Указанное ограничение легко снимается в рамках идеологии программы. Взаимная независимость объектов позволяет настраивать геометрию сцен в каждом окне в отдельности и, следовательно, обеспечивать параллельный поток информации для каждого из кадров стереопар.

2.2. Использование МОФ-технологии для формирования стереоскопического образа PhOOM-моделей

Эффект настоящего объемного изображения можно получить с использованием стереопары. Стереопара — это пара плоских квазитрехмерных изображений одного и того же объекта, имеющая различие между изображениями, за счет которых и создается эффект объема. Современный кинематограф немыслим без 3D-стереофильмов. Сегодня почти невозможно представить себе высокобюджетный, насыщенный спецэффектами блокбастер, который бы не являлся 3D-стереофильмом. На протяжении почти ста лет для производства стереофильмов используется съемка стереопары двумя объективами на одну или две киноплёнки.

В настоящее время существует достаточно обширный набор технологий визуализаций трехмерных стереоскопических изображений, которые рассмотрены ниже:

2.2.1. Затворный метод. Данная технология заключается в попеременной демонстрации на экране изображений, предназначенных для левого и правого глаза. При этом стекла очков поочередно затемняются, так что каждый глаз поочередно видит предназначенное только ему изображение. К недостаткам данного метода можно отнести значительное ослабление светового потока, возникающий эффект раздвоения быстро движущихся объектов, повышенная утомляемость глаз за счет нагрузки на переносицу из-за повышенной массы очков. Очки имеют электронную "начинку", вследствие чего очень плохо поддаются санитарной обработке.

2.2.2. Метод линейной поляризации. В случае линейной поляризации два изображения накладываются друг на друга на один и тот же экран через фильтры в проекторах, расположенные под углом 90° друг к другу. К недостаткам данного метода можно отнести то, что зрителю приходится держать голову на одном уровне, не наклоняя ее, иначе эффект объема пропадет. Понятно, что при использовании такого метода может возникнуть усталость и боль в шее.

2.2.3. Метод круговой поляризации. При использовании круговой поляризации два изображения также накладываются друг на друга через фильтры с противоположно направленной поляризацией. При этом в отличие от линейной поляризации, если зритель наклоняет голову, видимая объемность стереоизображения не пропадает. Однако качество экрана по-прежнему является важным.

2.2.4. Использование анаглифов [8]. В старых методах с использованием съемочных световых фильтров и пленки изображения для правого и левого глаза печатались или воспроизводились в качестве проекции как единое изображение, или же два изображения через специальные све-

тофильтры попадали на одну пленку. Одно пропусклось через красный фильтр, а другое через фильтр контрастирующего цвета — синего, зеленого или бирюзового. Недостатком метода является трудность воспроизведения цветного изображения и большая психологическая нагрузка на зрителя, связанная с восприятием правым и левым глазом изображений в неперекрывающихся спектральных диапазонах.

2.2.5. Метод, использующий многополосный оптический фильтр. Этот метод, так же как и поляризационные, использует пассивные очки. Однако в отличие от поляризационных методов можно использовать стандартный, а значит, более дешевый экран, к поверхности которого не предъявляются требования сохранения поляризации в отраженном свете. При этом МОФ-технология позволяет получать ощущение практически естественного по цветовой гамме изображения и свободна от вредного для глаз низкочастотного мигания, присущего затворному методу.

3. Эксперименты по апробации технологий

К настоящему времени каждая из объединяемых технологий прошла апробации. PhOOM-симуляции успешно используются в учебном процессе. На базе более 250 созданных оригинальных виртуальных моделей физических систем разной сложности создано семь электронных сборников мультимедийных учебных ресурсов для учащихся старших классов средних учебных заведений и студентов университетов [9—11], широко используется в on-line курсах, создаваемых в МООС-форматах [12].

Возможность использования моделей, генерируемых PhOOM-конструкторами для создания стереопар, тестировалась на простейших двухцветных анаглифах [13]. Из коллекции интерактивных 3D-моделей физических систем, созданных на базе оригинальных программ виртуальных физических конструкторов, были выбраны наиболее сложные для зрительного восприятия (рис. 3, см. третью сторону обложки).

На рис. 3 приведены 3D-варианты следующих симуляций, порожденных электронными конструкторами: "Движение электрона в поле фиксированного электрически заряженного монополя" (а), "Общий случай движения заряженной частицы в постоянных однородных электрическом и магнитном полях", "Магнитный диполь вблизи ферромагнетика" (б), "Движение гироскопа на вращающейся опоре", "Система нуклонов, взаимодействующих с силами, соответствующими потенциалу Юкавы",

"Возникновение внутренних каустик в слабосогласованном резонаторе", "Искривление солнечных лучей в атмосфере планеты", "Векторная модель атома с учетом эффектов тонкого и сверхтонкого расщепления" (в), "Трехмерная модель радуги" (г), объектив "Гелиос" (д).

Апробация МОФ-технологии стереоскопических визуализаций осуществлена компанией "СВЕГА-Компьютер" (Россия, г. Санкт-Петербург). Действующий макет проекционной стереоустановки был создан в двух вариантах (стационарном и мобильном), предназначенных для применения в сфере образования. Работоспособность подхода и высокое качество объемного изображения были успешно продемонстрированы 3 апреля 2017 г. на заседании Координационного комитета кафедр ЮНЕСКО в г. Белгород. Используемые в данных изделиях технологическое и контентное обеспечение наглядно продемонстрировали возможности применения многомерных мультимедийных электронных образовательных ресурсов в начальных, средних и высших учебных заведениях. В настоящее время данные изделия внедрены и успешно используются в ряде школ Российской Федерации в рамках реализации пилотного проекта ИИТО ЮНЕСКО и ассоциированных школ ЮНЕСКО "LFF — Learning For the Future / Обучение для будущего".

Совместимость PhOOM- и МОФ-технологий проверялась на тестовом учебном фильме по стереометрии четырехмерного пространства — времени Минковского, созданного по мотивам одноименной лекции, прочитанной на летней школе МФТИ [14]. Поставленный эксперимент интересен не только с технической точки зрения, но и как первый эксперимент по новому варианту визуализаций сцен псевдоевклидового пространства путем их проектирования из 4D- в 3D-подпространство. Простейшим объектом такого пространства является световой конус. Его традиционное изображение на плоскости и варианты анаглифической визуализации проекции четырехмерного объекта в трехмерное пространство приведены на рис. 4.

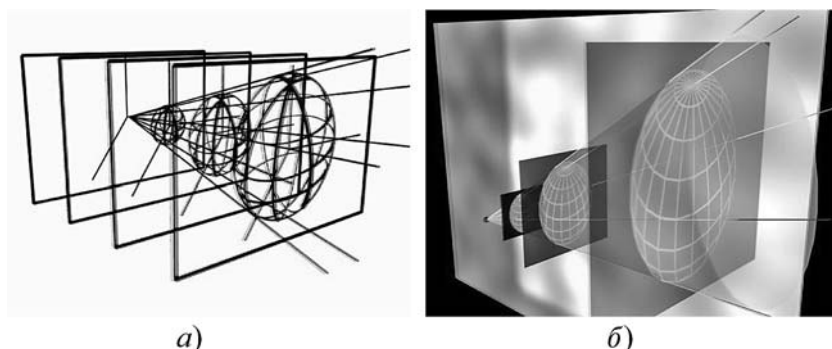


Рис. 4. Варианты изображения светового конуса в четырехмерном пространстве — времени

Псевдоевклидовость метрики пространства Минковского проявляется при его четырехмерных поворотах, соответствующих переходам в движущиеся (относительно исходной) инерциальные системы отсчета. Соответствующие демонстрации составляют основу сюжетов готовящихся стереоскопических видеоклипов, иллюстрирующих четырехмерные геометрические образы расширяющейся Вселенной для открытой и закрытой моделей, а также — связь между эффектами гравитационного взаимодействия и изменением геометрических свойств пространства — времени в окрестностях гравитирующих тел.

С точки зрения организации стереоскопических 3D-визуализаций представляют интерес демонстрации четырехмерных поверхностей, соответствующих релятивистским инвариантам типа квадратов четырех векторов $c^2t^2 - x^2 - y^2 - z^2 = \text{const}$ (рис. 5, см. третью сторону обложки).

Заключение

Первый опыт создания и использования стереоскопических демонстраций в интересах преподавания физики указывает на существование дополнительных новых форм эффективной и информационно емкой подачи учебно-научного материала, использование которых наиболее оправдано в случае визуализации результатов численного моделирования сложных физических систем.

Помимо физики трехмерные стереоскопические визуализации в сочетании с интерактивным учебным моделированием могут и должны найти широкое применение в широком спектре предметных областей. Без использования 3D-моделирования в наши дни немыслимы никакое промышленное производство, строительство и архитектура, наука и медицина. Прежде чем найти свое воплощение в материале, практически любое промышленное изделие проходит стадию виртуального 3D-моделирования. Эргономика современных автомобилей, бытовой техники, мебели и других знакомых нам вещей проходит первоначальную обкатку в виде виртуальных графических моделей. Использование технологий 3D-визуализации в медицине позволяет свести к минимуму возможность ошибки при диагностике, операционном планировании, протезировании и в решении множества других специализированных задач. Архитектурное 3D-моделирование позволяет наглядно увидеть будущее строение во всех деталях и оценить его взаимодействие с окружающим ландшафтом.

На этом пути существуют возможности не только повышения качества и приближения к реальности традиционных анимированных изображений, но и возникают варианты принципи-

ально новых видов визуального сопровождения учебной информации.

Список литературы

1. Бобович А. В., Космачев В. М., Чирцов А. С. Интеграция информационных технологий в образование // Компьютерные учебные программы и инновации. 2001. № 3. С. 39—56.
2. Бутиков Е. И., Чирцов А. С. Законы движения макроскопических тел. Пакет обучающих и демонстрационных программ по курсу общей физики // Труды III Международной конф. Model-oriented Data Analysis. Секция "Моделирование, оптимизация и обработка данных". 28—30 мая 1992, Санкт-Петербург, Ч. 2. 27 с.
3. Чирцов А. С. Серия электронных сборников мультимедийных материалов по курсу общей физики: новые подходы к созданию электронных конструкторов виртуальных физических моделей с простым удаленным доступом // Компьютерные инструменты в образовании. 2010. № 6. С. 42—56.
4. Ножнов В. А. Модель учебного курса для дистанционного образования. Конгресс конференций информационные технологии в образовании [сайт]. URL <http://ito.edu.ru/2009/Tomsk/VI/VI-0-134.html> (дата обращения: 11.06.2017).
5. Колесов Ю. Б., Сениченков Ю. Б. Моделирование систем. Объектно-ориентированный подход: учеб. пособие. СПб.: БХВ-Петербург, 2012. 192 с.
6. Чирцов А. С. Физическое объектно-ориентированное моделирование в курсах механики. LAP LAMBERT Academic Publishing. 2015. 148 с.
7. Matt Weisfeld. The Object-Oriented Thought Process. Fourth Edition. Addison-Wesley Professional, 2013. 336 с.
8. Рожков С. Н., Овсянникова Н. А. Стереоскопия в кино-, фото-, видеотехнике. М.: Парадиз, 2003.
9. Абутиин М. В., Марек В. П., Микушев В. М., Чирцов А. С. Новые варианты использования информационных и мультимедийных технологий для реализации непрерывного высшего профессионального образования // Физическое образование в вузах. 2012. Т. 18, № 1. С. 109—126.
10. Чирцов А. С., Nicljsky D., Брильянов В. А., Ваникович И. В. Использование физического объектно-ориентированного моделирования для развития и индивидуализированного обучения и организации мини-исследований в курсах механики // Научно-технический вестник информационных технологий, механики и оптики. 2017. Т. 17, № 2 (108). С. 159—171.
11. Chirtsov A. S., Sechov S. V. Automation of Creation of Educational Content for a Courses of Physics and Chemistry for Mass Individualised Education // Science and Society. 2016. № 2. P. 34—47.
12. Чирцов А. С. Гравитация: развитие взглядов от Ньютона до Эйнштейна // Прогресс-школа. Научно-образовательный проект. 2016. URL: <https://www.youtube.com/channel/UCnJpbR9H110msdbkwHyokWQ>.
13. Марек В. П., Микушев С. В., Смирнов А. Г., Чирцов А. С. Возможности использования технологий стереоскопических 3D-визуализаций в компьютерных моделях для сопровождения преподавания курсов физики // Компьютерные инструменты в образовании. 2011. Вып. 2. С. 39—57.
14. Иванов М. Г. Геометрический взгляд на специальную теорию относительности // Интернет университет информационных технологий. URL: <http://www.intuit.ru/departament/physics/geomstr/> (дата обращения: 25.04.2011).

The Use of Technology of Stereoscopic Images Creation in Numerical Modeling of Complex Physical Systems for Lecture Demonstrations and Educational Researches

This article discusses the options for effective use of a simple and widely available in the sphere of education technology to create stereoscopic images to visualize the results of interactive computer simulation of complex systems and their processes, based on the principles of physical object-oriented modeling. The possibilities of using the approach for the organization of the mass individualized education in the field of the exact sciences are analyzed, including the attraction of the most qualified and motivated trainees to active forms of creative learning of educational material with elements of independent scientific research. The described approach is illustrated by the example of using the considered technologies to support the course of relativistic electrodynamics.

Keywords: 3D-modeling, stereoscopic visualization, object-oriented modeling, popularization of modern physics, relativistic electrodynamics, Minkowski space, 4D-visualizations, modeling of complex systems

References

1. Bobovich A. V., Kosmachev V. M., Chircov A. S. Integracija informacionnyh tehnologij v obrazovanie (Integration of Information Technologies in Education), *Komp'yutnyye uchebnye programmy i innovacii*. 2001, no. 3, Moscow, pp. 39–56 (in Russian).
2. Butikov E. I., Chircov A. S. Zakony dvizheniya makroskopicheskikh tel. Paket obuchajushchih i demonstracionnyh programm po kursu obshhej fiziki (Laws of macroscopic bodies motion. A package of training and demonstration programs in the course of general physics), *V sb. Tr. III Mezhdunarodnoj konf. Model-oriented Data Analysis. Sekc. Modelirovanie, optimizacija i obrabotka dannyh*. 28–30 maja 1992, S. Peterburg. Ch. 2. 27 p. (in Russian).
3. Chircov A. S. Serija jelektronnyh sbornikov mul'timedijnyh materialov po kursu obshhej fiziki: novye podhody k sozdaniyu jelektronnyh konstruktorov virtual'nyh fizicheskikh modelej s prostym udalennym dostupom (A series of electronic collections of multimedia materials on the course of general physics: new approaches to the creation of electronic designers of virtual physical models with simple remote access), *Komp'yuternye instrumenty v obrazovanii*. 2010, no. 6, pp. 42–56 (in Russian).
4. Nozhnov V. A. Model' uchebnogo kursa dlja distancionnogo obrazovanija (Model of the training course for distance education), *Kongress konferencij informacionnye tehnologii v obrazovanii [web site]*. URL <http://ito.edu.ru/2009/Tomsk/VI/VI-0-134.html> (date of access: 11.06.2017) (in Russian).
5. Kolesov Ju. B., Senichenkov Ju. B. *Modelirovanie sistem. Ob'ektno-orientirovannyj podhod* (Modeling of systems. Object-oriented approach). Ucheb. posobie. SPb.: BHV-Peterburg, 2012, 192 p. (in Russian).
6. Chircov A. S. Fizicheskoe ob'ektno-orientirovannoe modelirovanie v kursah mehaniki (Physical object-oriented modeling in mechanics courses). *LAP LAMBERT Academic Publishing*, 2015, 148 p. (in Russian).
7. Matt Weisfeld. *The Object-Oriented Thought Process*. Fourth Edition. Addison-Wesley Professional, 2013. 336 p.
8. Rozhkov S. N., Ovsjannikova N. A. *Stereoskopija v kino-, foto-, videotekhnike* (Stereoscopy in film, photo, video equipment). Moscow, Paradiz, 2003. (in Russian).
9. Abutin M. V., Marek V. P., Mikushev V. M., Chircov A. S. Novye varianty ispol'zovanija informacionnyh i mul'timedijnyh tehnologij dlja realizacii nepreryvnogo vysshego professional'nogo obrazovanija (New options for using information and multimedia technologies for the implementation of continuous higher professional education, *Fizicheskoe obrazovanie v VUZah*, 2012, vol. 18, no. 1, pp. 109–126. (in Russian).
10. Chircov A. S., Nicjlsky D., Bril'janov V. A., Vanikovich I. V. Ispol'zovanie fizicheskogo ob'ektno-orientirovannogo modelirovanija dlja razvitiya i individualizirovannogo obuchenija i organizacii mini-issledovanij v kursah mehaniki (The using of physical object-oriented modeling for development and individualized learning and organizing of mini-studies in mechanics courses), *Nauchno-tehnicheskij vestnik informacionnyh tehnologij, mehaniki i optiki* 2017, vol. 17, no. 2 (108), pp. 159–171 (in Russian).
11. Chirtsov A. S., Sechov S. V. Automation of Creation of Educational Content for a Courses of Physics and Chemistry for Mass Individualised Education. *Science and Society*, 2016, no. 2, pp. 34–47.
12. Chircov A. S. Gravitacija: razvitie vzgljadov ot N'jutona do Jejnshtejna (Gravitation: development of views from Newton to Einstein). [Available]. *Progress-shkola. Nauchno-obrazovatel'nyj projekt. [web site]*, 2016. URL: <https://www.youtube.com/channel/UCnJpbR9H110msdbkwhyokWQ> (in Russian).
13. Marek V. P., Mikushev S. V., Smirnov A. G., Chircov A. S. Vozmozhnosti ispol'zovanija tehnologij stereoskopicheskikh 3D-vizualizacij v komp'yuternyh modeljah dlja soprovozhdenija prepodavanija kursov fiziki (Possibilities of using stereoscopic 3D-visualization technologies in computer models for accompanying the teaching of physics courses), *Komp'yuternye instrumenty v obrazovanii*, 2011, vyp. 2, pp. 39–57 (in Russian).
14. Ivanov M. G. Geometricheskij vzgljad na special'nuju teoriju odnositel'nosti (Geometrical view on the special relativity theory). *Internet universitet informacionnyh tehnologij [web site]*. URL: <http://www.intuit.ru/departament/physics/geomstr/> (date of access: 25.04.2011) (in Russian).

УДК 004.075

Ле Тхань Тунг Нгуен, магистрант, e-mail: juno249@gmail.com,
А. Г. Кравец, д-р техн. наук, проф., e-mail: agk@gde.ru,
Нгок Зыонг Буй, аспирант, e-mail: ramsetii@gmail.com,
Волгоградский государственный технический университет

Анализ средств и моделей взаимодействия между компонентами в системе управления корпоративной мобильностью

Коммуникация M2M (Machine-2-Machine) является новой технологией, позволяющей осуществлять "бесшовное" взаимодействие большого числа устройств в различных сетях без необходимости вмешательства человека в компоненты распределенных компьютерных систем, в частности, систем управления корпоративной мобильностью (УКМ). В настоящее время появляется ряд специализированных энергоэффективных протоколов M2M на прикладном уровне с низким энергопотреблением и потерями (Low power and Lossy Networks, LLNs), среди которых наиболее популярны HTTP, MQTT, XMPP, CoAP. В данной работе анализируются эти протоколы, они обеспечивают взаимодействие между сервером и устройствами в системе УКМ. Предлагаются формальные средства и модели обмена управляющими сообщениями на основе технологии асинхронной передачи. Проведены эксперименты, направленные на тестирование латентности и джиттера при приеме-отправке сообщений между мобильным устройством и сервером УКМ на базе этих протоколов.

Ключевые слова: управление корпоративной мобильностью, ресурс-ориентированная архитектура, службы Push-уведомлений, архитектура REST, модель публикация/подписка, модель запрос/ответ, обмен сообщениями, асинхронная передача, MQTT, CoAP, XMPP, HTTP, брокер сообщений

Введение

В настоящее время существует несколько компаний-разработчиков решений для управления корпоративной мобильностью (УКМ), которые предоставляют новые инструменты для взаимодействия с корпоративными информационными ресурсами с помощью смартфонов и планшетов вне зависимости от времени или места, что позволяет пользователям работать более продуктивно и обеспечивает оперативность, гибкость и контроль затрат для корпорации [1–3]. При этом существует проблема конфиденциальности данных и информационной безопасности компании, особенно в процессах передачи информации между сервером и мобильными устройствами, имеющими доступ к ресурсам предприятия [4–7].

В системе УКМ существуют следующие процессы [8]: регистрация мобильного устройства; централизованное управление мобильными устройствами, приложениями и контентом; защита мобильного устройства; распределение ресурсов корпорации на устройствах; мониторинг мобильных устройств. На рис. 1 (см. четвертую

сторону обложки) показана контекстная среда работы системы УКМ с участием служб Push-уведомлений — важных критических компонентов мобильных устройств, требующих обновления контекста пользователя [9, 10].

Для описания задач и результатов операций — обмена данными функциональных сервисов на backend-сервере, обмена управляющими сообщениями между сервером и устройствами — используются модели взаимодействия и современные протоколы, которые гарантируют повышение эффективности коммуникации в системе УКМ по таким параметрам, как масштабируемость, работоспособность, снижение потерь пакетов данных при передаче, экономия расхода заряда батареи, пропускной способности, обеспечение безопасности.

Таким образом, целью данной работы является анализ существующих протоколов для управления взаимодействием между шлюзами, сетью Интернет и конечными приложениями/устройствами, а также исследование моделей обмена сообщениями в системе УКМ.

1. Коммуникационные протоколы взаимодействия в корпоративной сети

Для коммуникации между системными компонентами в корпоративной сети на прикладном уровне используются ряд специализированных энергоэффективных протоколов M2M (Machine-2-Machine) с низким энергопотреблением и потерями (Low power and Lossy Networks, LLNs) [11]. Среди них наиболее популярны HTTP (HyperText Transfer Protocol), MQTT (Message Queuing Telemetry Transport), XMPP (The Extensible Messaging and Presence Protocol), CoAP (Constrained Application Protocol) и др. Протокол CoAP представляет собой модель взаимодействия запрос/ответ (клиент/сервер) для манипулирования ресурсами и передачи данных. Протоколы MQTT, XMPP основаны на брокере для обмена сообщениями на основе архитектуры публикация/подписка. У каждого из них есть свои преимущества и недостатки, приведенные ниже.

Протокол HTTP. HTTP (HyperText Transfer Protocol) — протокол передачи гипертекста) был разработан Тимом Бернерсом в марте 1991 г. [12]. В настоящее время HTTP используется в сети Интернет для передачи и получения различных данных веб-приложений от удаленного сервера к клиентскому компьютеру. Протокол HTTP основан на модели взаимодействия запрос/ответ для информационных систем клиент/сервер. Одной из основных проблем, связанных с этой архитектурой, является то, что существует некоторая задержка: во-первых, при отправке клиентом своих данных на сервер, т. е. клиентам после отправки необходимо ожидать результата получения и обработки запроса сервером; во-вторых, при опросе клиентом новых доступных данных. Это повышает нагрузку на сеть и требования к пропускной способности, кроме того, протокол HTTP затрачивает больше энергии батареи.

Новая крупная версия сетевого протокола HTTP/2 создана для решения этих проблем [13]. Протокол HTTP/2 является бинарным, т. е. по сравнению с предыдущим стандартом поток данных разделяется по фреймам с метаданными, имеющими фиксированную структуру и размер. В отличие от протокола HTTP, в протоколе HTTP/2 применяется эффективная технология сжатия данных HPACK, присваивающая имена заголовков и значения записей в таблицах и использующая только необходимый номер записи. Эта технология позволяет уменьшить размер заголовка протокола, в связи с чем снижается время обработки запроса.

Эффективность сетевых ресурсов повышается за счет применения мультиплексирования, которое позволяет клиентам использовать одно со-

единение TCP (Transmission Control Protocol) для множества одновременных потоков запросов от любой стороны — клиента или сервера. Каждый поток имеет приоритет, указанный клиентом самостоятельно на основе важности и зависимости одного потока от другого.

В протокол HTTP/2 внедрили Server Push, позволяющий серверу сразу же, не дожидаясь ответа клиента, добавить нужные файлы в кэш для быстрой выдачи. Более того, HTTP/2 поддерживает весь набор методов доступа протокола HTTP (GET, PUT, POST и т. п.), URI (Uniform Resource Identifier), при этом большее число заголовков. Для повышения безопасности соединения и защиты данных применяется TLS. Данные изменения позволили повысить производительность сервисов и мобильных устройств, оптимизировать трафик и уменьшить задержку доступа и нагрузку на сервер и клиента.

Протокол MQTT. Протокол MQTT (Message Queuing Telemetry Transport) [14] — простой, быстроедействующий протокол, разработанный компанией IBM на основе модели взаимодействия публикации/подписки для Интернета вещей, он также может быть использован для надежной корпоративной системы передачи сообщений на мобильные устройства и между устройствами.

Характеристики протокола:

- он работает по принципу публикация/подписка, которая обеспечивает распределение сообщений и развязки приложений "один ко многим";
- транспорт обмена сообщениями не зависит от содержания данных;
- небольшие накладные расходы на транспортном уровне (фиксированный размер заголовка длиной 2 байт), протокол обмена также сведен к минимуму числа потоков данных для уменьшения сетевого трафика и контроля потерь во время передачи сообщений;
- поддержка уровней качества сервиса (Quality of Service, QoS) [15] для доставки сообщений;
- для внедрения не требуется устанавливать большое количество программного обеспечения.

Протокол MQTT использует брокеры сообщений, через которые клиенты могут зарегистрировать подписки и публиковать свои данные. Брокеры обеспечивают маршрутизацию сообщений от публикаторов к подписчикам. Сообщения по определенной теме доставляются подписанным клиентам в зависимости от очереди маршрутизации.

Сообщения в протоколе MQTT доставляются с гарантией на основе QoS. Существуют три уровня QoS:

- *At most once* (QoS = 0, максимум однократная доставка) означает, что публикатор выполняет однократную отправку сообщения, но не контролирует доставку;

— *At least once* (QoS = 1, минимум однократная доставка): доставка сообщения контролируется, однако разрешается доставлять более одного раза;

— *Exactly once* (QoS = 2, однократная доставка): доставка сообщения гарантируется лишь один раз.

Для обеспечения безопасности передачи сообщений протокол MQTT использует TLS/SSL (Transport Layer Security/Secure Sockets Layer).

Протокол XMPP. XMPP (The Extensible Messaging and Presence Protocol) [16] — расширяемый протокол, стандартизирован IETF (Internet Engineering Task Force) для системы мгновенного обмена сообщениями и информацией о присутствии.

XMPP работает по TCP и обеспечивает модели взаимодействия публикация/подписка (асинхронная) и запрос/ответ (синхронная). Он предназначен для близких коммуникаций в реальном времени и таким образом поддерживает небольшой "след" сообщения (*message footprint*) и низкую задержку обмена сообщениями. Протокол XMPP обеспечивает простой метод адресации устройств вида адреса электронной почты (name@domain.com) для идентификации пользователей, по которому данные передаются в независимости от точек доступа. Схема адресации состоит из набора элементов, образующих доменный идентификатор и идентификатор ресурса.

XMPP обеспечивает безопасность связи с помощью криптографических протоколов TLS/SSL, встроенных в ядро спецификации протокола. Но он не использует сервис качества QoS, который делает его непрактичным для коммуникации, и только наследственные механизмы протокола TCP гарантируют надежность. В XMPP используется специализированный язык разметки XML (eXtensible Markup Language), который создает дополнительные накладные расходы на передачу данных вследствие ненужных тегов XML и требует дополнительных вычислительных мощностей для XML-парсинга, что увеличивает потребление энергии устройства.

Протокол CoAP. Протокол CoAP (Constrained Application Protocol) [17] был разработан IETF (Internet Engineering Task Force) с использованием расширенных методов HTTP для встроенных устройств. CoAP определяет заголовок сообщения, коды запроса/ответа, параметры сообщения, а также механизмы повторной передачи. В отличие от текстового протокола HTTP, CoAP транспортирует сообщения асинхронно в виде бинарных данных через UDP (User Datagram Protocol) и передает меньше данных в заголовках запроса, что уменьшает размер служебных данных, снижает требования к пропускной способности и повышает гибкость в моделях взаимодействия. Он создает альтернативу протокола HTTP

для RESTful API (Representational State Transfer Application Programming Interface) на устройствах с ограниченными ресурсами и использует подмножество основных методов HTTP (GET, POST, PUT и DELETE) для представления ресурсов, ориентированных на взаимодействие в архитектуре клиент/сервер. Протокол CoAP также обеспечивает возможность многоадресной рассылки сообщений и обработку единого обмена сообщениями, выполняемого между конечными точками. Сообщения могут быть следующих видов:

- *Confirmable* — сообщения требуют подтверждения. Когда пакеты не теряются, каждое сообщение вызывает обратное сообщение типа *Acknowledgement* или *Reset*;
- *Non-confirmable* — сообщения не требуют подтверждения;
- *Acknowledgement* — подтверждает получение *Confirmable*-сообщения, но не указывает на успех или неудачу;
- *Reset* — указывает на то, что конкретное сообщение (*Confirmable* или *Non-confirmable*) было получено, но контекст, подлежащий обработке, отсутствует.

CoAP представляет собой модель взаимодействия запрос/ответ, который использует как синхронную, так и асинхронную технологии. Протокол CoAP поддерживает унифицированный идентификатор ресурса (URI) в запросах для идентификации интерфейса, экземпляра объекта или ресурса.

В протоколе CoAP использован DTLS (Datagram Transport Layer Security) для обеспечения высокого уровня безопасности связи и передачи данных по протоколу UDP. Режимы безопасности DTLS включают в себя как предварительный, так и открытый ключи для поддержки встраиваемых устройств.

Сравнительный анализ протоколов для обмена сообщениями. На основании проведенного анализа можно сделать следующие выводы:

- большинство рассмотренных протоколов используют транспорт TCP, что обеспечивает необходимый уровень надежности;
- протокол MQTT доставляет сообщения между устройствами по принципу взаимодействия публикация/подписка через центральный брокер сообщений с использованием QoS для проверки обмена;
- протокол XMPP обеспечивает быстрый асинхронный обмен в качестве метода адресации и поддерживает огромное число пользователей и разработчиков сети Интернет;
- протокол CoAP работает поверх UDP, что уменьшает объем служебных данных в передаваемом пакете для сетей с низким энергопотреблением;

Сравнительный анализ протоколов для обмена сообщениями в системе УКМ

Критерии	HTTP	XMPP	CoAP	MQTT
Модель	Запрос/ответ	Публикация/подписка, Запрос/ответ	Запрос/ответ	Публикация/подписка, Запрос/ответ
Качество сервиса	1 вид	1 вид	4 вида	3 вида
Тип распределения данных	Один к одному	Один ко многим	Один ко многим	Один ко многим
Обеспечение безопасности	SSL/TLS	SASL/TLS	DTLS	SSL/TLS
Формат сообщения	JSON/XML	XML	JSON/XML/CBOR	JSON/XML
Объем сообщения	Большой	Маленький	Маленький	Маленький
Низкие накладные расходы протокола	Нет	Да	Да	Да
Низкое энергопотребление	Нет	Да	Да	Да
Миллионы подключенных клиентов	Нет	Да	Да	Да
Разнообразие клиентских платформ	Да	Да	Да	Да
Push-сообщения	Да	Да	Да	Да

- протокол HTTP не оптимизирован для низкого энергопотребления или сведения к минимуму числа потоков данных [18].

Критерии и результаты сравнительного анализа протоколов для обмена сообщениями между мобильными устройствами и серверами представлены в таблице.

Протокол MQTT использует QoS, которое позволяет увеличить надежность работы системы и снизить вероятность потерь данных при передаче между компонентами системы, но повышает время задержек сообщений. Вместе с тем MQTT имеет короткую длину заголовка сообщения, которая влияет на снижение затрат энергии батарей и пропускной способности (*bandwidth*).

В рамках анализа указанных выше преимуществ протоколов и требований к решению определенных проблем организации обмена сообщениями между сервером и мобильными устройствами, для реализации процесса обмена сообщениями в системе УКМ выбран протокол MQTT.

2. Модель взаимодействия для управления мобильностью в корпоративной сети

В существующих MEAP (Mobile Enterprise Application Platform — платформа для корпоративных мобильных приложений) решениях [1] устройства и приложения имеют возможность обращаться к ресурсам вычислительных серверов, которые связываются с этими приложениями/устройствами через IP-адрес сервера и управляют доступом к ресурсам. Подход к построению коммуникационной модели с использованием рассмотренных эффективных протоколов для

обеспечения взаимодействия и распространения данных является необходимостью [4, 19]. В системе УКМ агент на мобильном устройстве взаимодействует с различными клиент-сервисными компонентами.

Существуют следующие виды взаимодействия:

— синхронное (запрос/ответ) — клиент делает запрос на сервер и поток, делающий запрос может быть заблокирован во время ожидания своего временного ответа;

— асинхронное (уведомление, публикация/подписка) — клиент отправляет запрос на сервис, отвечающий асинхронно. Клиент не блокируется во время ожидания и работает с предположением о том, что ответ не может быть получен в течение некоторого времени.

Модель взаимодействия запрос/ответ. Эта модель является основным способом связи между компонентами в системе УКМ. При использовании модели запрос/ответ клиент отправляет запрос на сервис, в котором запрос обрабатывается и ответ отправляется обратно. У многих клиентов поток блокирует запросы во время ожидания ответа. Другие клиенты для управления событиями могут использовать асинхронный клиентский код, который может инкапсулироваться. Однако в отличие от использования обмена сообщениями клиент предполагает, что ответ поступит своевременно.

Существуют многочисленные архитектуры для реализации обмена сообщениями. На ресурс-ориентированной архитектуре RESTful [20] основан один из наиболее популярных протоколов.

В этой архитектуре уникальный URI-стандартный адрес используется для описания

ресурсных данных каждого устройства и/или каждой операции.

В качестве примера приведем принцип конкретной операции $\{operationID\}$, управляемой администратором, для конкретного устройства $\{deviceID\}$ с помощью командной строки cURL (по типу HTTP-взаимодействия):

```
curl -X GET -H "Content-Type: application/json"
-H "Authorization: Bearer <TOKEN>"
-k -v baseIP/{deviceID}/{operationID}
```

URI-стандарт включает в себя части *host* (полное имя домена в системе) и *path* (иерархический путь к каталогу). В представленной выше команде адрес $baseIP/{deviceID}/{operationID}$ также разделяется на части *host* $baseIP$ и *part* $\{deviceID\}/{operationID}$.

Масштабируемая модель системы УКМ. Операции в системе УКМ отличаются и поэтому реализуются различными сервисами с конструкцией использования $\{deviceID\}$ и $\{operationID\}$ для адресуемых устройств и операций, на основе следующих предположений:

- есть m типов сервисов, выполняющих m типов операций $\{operationID_j\}$;
- есть n устройств, имеющих n идентификаторов $\{deviceID_j\}$.

Моделирование операции управления мобильностью происходит в координатной плоскости OYZ масштабируемого куба АКФ (The Scale Cube, авторы М. Abbott, Т. Keeven и М. Fisher) [21] по правилу "трех" (концепция Gartner) [22] (рис. 2).

Адресные ресурсы операций на сервере описываются короткем:

$$resource_data = \{interaction_type, deviceID, operationID\}$$

где $resource_data$ — ресурсные данные операции (задачи и/или результаты), описываемые ресурсной моделью;

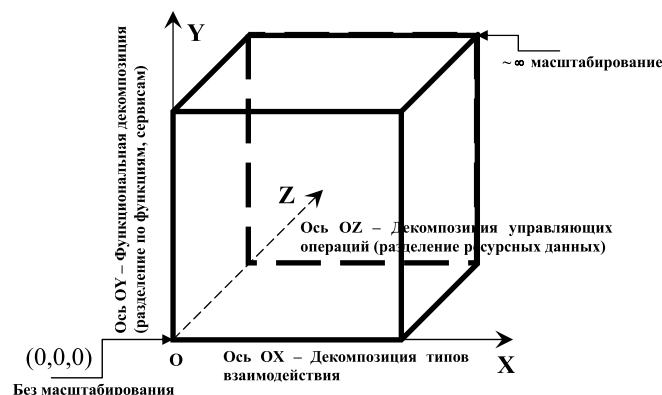


Рис. 2. Масштабируемая модель системы УКМ на основе шкалы куба АКФ

interaction_type — тип взаимодействия при обработке операции;

deviceID — идентификатор устройства — параметр, использующийся для идентификации применения управляющих операций на конкретном устройстве;

operationID — идентификатор процедуры на агенте, обрабатывающей операцию.

Например, операция УКМ с использованием ресурс-ориентированной архитектуры, в которой управляющая операция выполняется методами HTTP в архитектурном стиле REST, описывается следующим образом: $GET/PUT/POST/DELETE baseIP/a590g62f8b5c/changelockcode$.

Таким образом, разработанная масштабируемая модель системы УКМ на основе декомпозиции управляющего пространства мобильности позволяет уточнить измерения куба АКФ, специфичные для данного класса систем, а именно: тип взаимодействия, функция/сервис и операция управления.

Модель взаимодействия публикация/подписка.

В отличие от традиционной модели взаимодействия запрос/ответ новая асинхронная технология, реализующая механизм доставки данных с гарантией в виде сообщения, использует модель публикация/подписка для коммуникации между компонентами [24]. В модели публикация/подписка подписчики обычно получают лишь часть общего числа опубликованных сообщений. Публикаторы отправляют сообщения на промежуточный брокер сообщений или сервис событий, которые действуют как посредники и обеспечивают доставку сообщений между системными компонентами [25]. На этих брокерах клиенты могут зарегистрировать подписки и опубликовать свои данные. Брокеры обеспечивают маршрутизацию сообщений от публикаторов к подписчикам и выполняют определение очереди сообщений. Отправителям не нужно указывать получателя сообщений, и получатели могут не беспокоиться о том, кто присылает им сообщения. Сообщения доставляются подписанным клиентам в зависимости от очереди маршрутизации по определенной теме.

Архитектура публикация/подписка включает в себя масштабируемые свойства — временно-пространственная развязка (*time-space decoupling*) и синхронизационная развязка (*synchronization decoupling*), и предоставляет асинхронное распространение информации от публикаторов к подписчикам без определения конечных адресов получателей [23, 24]. Эти свойства позволяют системам публикация/подписка стать коммуникационным компонентом между системными сущностями в крупномасштабных динамических

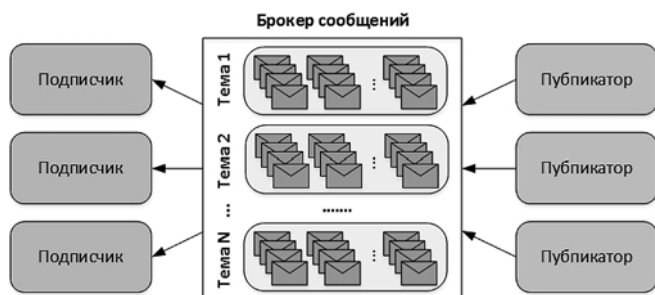


Рис. 3. Концепция коммуникационной модели публикации/подписки

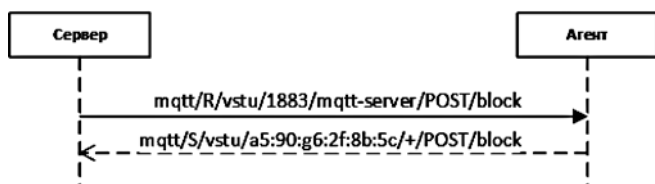


Рис. 4. Пример выполнения операции "изменение кода блокировки экрана устройства"

сетях. На рис. 3 показана концепция коммуникационной модели публикации/подписки.

В рамках работы УКМ по модели публикация/подписка клиенты и сервер должны подписаться на тему для обеспечения доставки сообщений следующего вида:

mqtt/ + /<название-предприятия>/<ид-устройства>/<ид-приложения>/<ид-операции>/#,

где + тема для отправки запроса (Request): *mqtt/R/... /...*,

тема для отправки ответа (Response): *mqtt/S/... /...*

Пример на рис. 4 показывает выполнение операции "изменение кода блокировки экрана устройства" на предприятии с названием *vstu* по порту 1883. Запрос инициирует: *mqtt/R/vstu/1883/mqtt-server/POST/block*. Ответ определен следующим образом: *mqtt/S/vstu/1883/a590g62f8b5c/POST/block*.

3. Вычислительный эксперимент

Проведем сравнение изменения латентности (*latency*) и джиттера (*jitter*) при приеме-отправке сообщений для каждого из исследуемых протоколов для системы УКМ. Тестовый прототип исследования описан далее.

Испытания проводили с использованием сервера со следующей конфигурацией: процессор Intel Core i3 3220 3,30 ГГц, 2 Гбайт ОЗУ под операционной системой Linux Ubuntu 12.04.5 x64 с предустановленными программами WireShark, Tshark и TCPDump для анализа трафика, подключенного через проводную сеть к сети Ин-

тернет со скоростью загрузки/скачивания в ~ 10 Mbps (Мбит/с). Серверные части написаны на языке PHP и работают на веб-сервере стека решений XAMPP.

Мобильным устройством, рассмотренным в экспериментах, является LG Nexus 5 под операционной системой Android 5.0, 16 Гбайт внутренней память, 2 а/б/г/н Гбайт ОЗУ и поддержка Wi-Fi 802.11. Устройство подключено через Wi-Fi к сети со скоростью загрузки 2 Mbps и со скоростью скачивания 4 Mbps.

На мобильное устройство было установлено приложение, реализующее сетевое взаимодействие с использованием различных протоколов (MQTT, XMPP и CoAP). Приложение написано на языке Java для Android-устройств на основе следующих библиотек для разработки: Paho (для MQTT) [26], Smack (для XMPP) [27], nCoAP (для CoAP) [28].

Для каждого протокола были проведены восемь измерений по увеличению числа сообщений с использованием показателей латентности и джиттера для оценки эффективности использования протокола в системе УКМ. Сообщения генерируются размером 254 байт в формате JSON и отправляются последовательно каждую миллисекунду в течение нескольких часов. Результаты проведенных испытаний показаны на рис. 5 (см. четвертую сторону обложки).

По сравнению с остальными протоколами, латентность при приеме-отправке сообщений протокола MQTT меньше, чем протоколов CoAP/ XMPP в ~ 2,5 раза и джиттер: в ~ 2 раза с протоколом CoAP и в 5,7 раз с XMPP.

На графике показано, что при увеличении числа сообщений, латентность и джиттер возрастают.

Таким образом, проведенный вычислительный эксперимент показал хорошую эффективность использования протокола MQTT для связи между мобильными устройствами и сервером с небольшим объемом информации в сообщении.

Заключение

В целом, можно сделать заключение, что использование M2M протоколов, и прежде всего протокола MQTT, дает более эффективную коммуникацию в системе УКМ. Предложенный подход повышает эффективность за счет уменьшения расхода заряда батареи, меньших накладных расходов на доставку данных между мобильными устройствами и обеспечение безопасности обмена сообщениями на базе трехуровневой архитектуры. Кроме того, совместное использование протоколов M2M на основе модели взаимодействия публикация/подписка повышает гиб-

кость системы. В связи с этим в данный момент протокол MQTT для MEAP-решений позволяет устранить проблемы обеспечения работоспособности при увеличении числа устройств в системе и экономить ресурсы системы.

Исследование выполнено при финансовой поддержке РФФИ (проект № 15-07-06254).

Список литературы

1. **Kravets A. G., Bui N. D., Al-Ashval M. S.** Mobile Security Solution for Enterprise Network // Knowledge-Based Software Engineering. Proc. of 11th Joint Conf., JCKBSE 2014. Volgograd, Russia, September 17–20, 2014 / ed. by A. Kravets, M. Shcherbakov, M. Kultsova, Tadashi Iijima. Volgograd State Technical University [et al.]. Springer International Publishing, 2014. V. 466. P. 371–382. (Series: Communications in Computer and Information Science).
2. **Аль-Ашваль М. С., Кравец А. Г.** Анализ показателей качества мобильных корпоративных сетей // Современные проблемы науки и образования: электрон. науч. журнал / РАЕ. 2014. № 6. URL: <http://www.science-education.ru/120-16129>.
3. **Аль-Ашваль М. С., Кравец А. Г.** Система критериев и показателей качества мобильных корпоративных сетей // Инновации на основе информационных и коммуникационных технологий. Инфо 2014. Матер. XI междунар. науч.-практ. конф., г. Сочи, 1–10 окт. 2014 г. М.: Национальный исследовательский ун-т "Высшая школа экономики" [и др.], 2014. С. 501–504.
4. **Кравец А. Г., Аль-Ашваль М. С.** Mobile corporate networks security control // International Siberian Conference on Control and Communications (SIBCON–2016) (Russia, Moscow, May 12–14, 2016): Proceedings / Tomsk IEEE Chapter & Student Branch, National Research University "Higher School of Economics". Moscow, 2016. С. 1–6 (DOI: 10.1109/SIBCON.2016.7491811).
5. **Буй Нгок Зьонг, Кравец А. Г., Ле Тхань Тунг Нгуен.** Безопасная аутентификация в системе управления корпоративной мобильностью // Известия ВолгГТУ. Сер. Актуальные проблемы управления, вычислительной техники и информатики в технических системах. Волгоград, 2015. № 13 (177). С. 45–51.
6. **Буй Нгок Зьонг, Кравец А. Г., Ле Тхань Тунг Нгуен.** Информационная безопасность процесса регистрации Android-устройства в системе управления корпоративной мобильностью // Вестник компьютерных и информационных технологий. 2016. № 8. С. 44–51.
7. **Буй Нгок Зьонг, Кравец А. Г., Ле Тхань Тунг Нгуен.** Проблема проверки root-прав на Android-устройстве в системе управления мобильными приложениями // Информационные технологии в науке, образовании и управлении: матер. XLIV междунар. конф. и XIV междунар. конф. молодых ученых IT + S&E'15 (Гурзуф, 22 мая – 1 июня 2015 г.). / под ред. Е. Л. Глориозова; М.: ООО "Институт новых информационных технологий", 2015. С. 420–426.
8. **Буй Н. З.** Архитектура системы управления мобильностью в корпоративной сети // Современная наука: актуальные проблемы теории и практики. Сер. Естественные и технические науки. 2016. № 09-10. С. 30–35.
9. **Push Technology: A Key Ingredient of Application Interactivity.** URL: http://www.seven.com/downloads/pdf/SEVEN_Push_Whitepaper.pdf (дата обращения: 20.01.2017).
10. **Буй Н. З., Нгуен Л. Т. Т., Кравец А. Г.** Разработка службы Push-уведомлений в системе управления корпоративной мобильностью с использованием Google Cloud Messaging // Юность и Знания — Гарантия Успеха — 2015: сб. науч. тр. 2-й междунар. науч.-практ. конф. (1–2 окт. 2015 г.). В 2 т. Т. 2. Курск: Юго-Западный гос. ун-т, ЗАО "Университетская книга" [и др.]. 2015. С. 28–30.
11. **Jung M., Kim J. H., Wi H. W., Kim S.** Things-to-cloud communication: technology overview and design considerations // 2015 5th International Conference on the Internet of Things (IoT), October 26–28, 2015 in Seoul, S. Korea.
12. **HTTP.** URL: https://en.wikipedia.org/wiki/Quality_of_service (дата обращения: 26.1.17).
13. **Разъяснения HTTP2.** URL: <http://habrahabr.ru/post/221427> (дата обращения: 05.02.2017).
14. **IBM. MQTT V3.1.** Protocol Specification // International Business Machines Corporation Eurotech. 2015.
15. **Quality of service.** URL: https://en.wikipedia.org/wiki/Quality_of_service (дата обращения: 26.01.17).
16. **ITU-T. Extensible Messaging and Presence Protocol (XMPP): Core** // RFC-3920. 2004.
17. **Shelby Z., Hartke K., Bormann C.** Constrained Application Protocol. URL: <https://datatracker.ietf.org/doc/rfc7252> (дата обращения 18.01.2017).
18. **Stephen N.** Power Profiling: HTTPS Long Polling vs. MQTT with SSL, on Android. URL: <http://stephendnicholas.com/archives/1217> (дата обращения: 28.01.17).
19. **Rodríguez-Domínguez C., Benghazi K., Noguera M., Garrido J. L., Rodríguez M. L., Ruiz-López T.** A Communication Model to Integrate the Request-Response and the Publish-Subscribe Paradigms into Ubiquitous Systems // E. T. S. I. I. T. University of Granada, C/Periodista Daniel Saucedo Aranda S/N, 18071 Granada, Spain, 2012.
20. **Jansen G.** Thoughts on RESTful API design / Geert Jansen [Электронный ресурс]. URL: <https://media.readthedocs.org/pdf/restful-api-design/latest/restful-api-design.pdf> (дата обращения 18.01.2017).
21. **Abbott M. L., Fisher M. T.** The art of scalability: Scalable Web Architecture, Processes, and Organizations for the Modern Enterprise. 2 ed., Pearson Education, Inc., 2015.
22. **Платформа для корпоративных мобильных приложений.** URL: <https://ru.wikipedia.org/wiki/> (дата обращения: 20.1.2017).
23. **Bui N. D., Кравец А. Г., Nguyen T. A., Nguyen L. T. T.** Tracking events in mobile device management system // IISA 2015 — 6th International Conference on Information, Intelligence, Systems and Applications (Corfu, Greece, 6 July 2015 — 8 July 2015): Conference Proceeding / Ionian University, Institute of Electrical and Electronics Engineers (IEEE) [Piscataway, USA]. 2015. 6 p. DOI: 10.1109/IISA.2015.7388127.
24. **Pongthawornkamol T., Nahrstedt K., Wang G.** The analysis of publish/subscribe systems over mobile wireless ad hoc networks // Proc. of ACM MobiQuitous. 07 Aug 2007, pp. 1–8.
25. **Message Broker.** URL: https://en.wikipedia.org/wiki/Message_broker (дата обращения: 26.08.16).
26. **Eclipse Paho MQTT.** URL: <https://eclipse.org/paho/> (дата обращения 18.01.2017).
27. **Smack 4.1** Readme and Upgrade Guide. URL: <https://github.com/igniterealtime/Smack/wiki/Smack-4.1-Readme-and-Upgrade-Guide> (дата обращения 18.01.2017).
28. **CoAP Client for Android based on nCoAP.** URL: <https://github.com/okleine/spitfirefox> (дата обращения 18.01.2017).

Analysis of Interaction Method and Models Between Components of Enterprise Mobility Management System

M2M (Machine-2-Machine) communication is the new technology that allows seamless interaction of a wide various devices over different networks without the need of human intervention between the components in distributed computing systems, especially in the enterprise mobility management system (EMM). Nowadays, there are a large of specialized energy-efficient protocols (M2M protocols) in Application layer with Low power and Lossy Networks (LLNs). Most popular of them is HTTP, MQTT, XMPP, CoAP and others. This article analyzes these protocols that enable communication between the server and devices in EMM system and also new formal messaging model based on asynchronous transfer technology is present. Experiments aimed at testing the latency and jitter receiving-sending messages between mobile device and EMM server based on these protocols.

Keywords: enterprise mobility management, resource-oriented architecture, push notification service, REST architecture, publish/subscribe model, request/response model, messaging model, asynchronous transfer, MQTT, CoAP, XMPP, HTTP, message broker

References

1. Kravets A. G., Bui N. D., Al-Ashval M. S. Mobile Security Solution for Enterprise Network, *Knowledge-Based Software Engineering: Proc. of 11th Joint Conf., JCKBSE 2014*, Volgograd, Russia, September 17–20, 2014, ed. by A. Kravets, M. Shcherbakov, M. Kultsova, Tadashi Iijima; Volgograd State Technical University [etc.]. Springer International Publishing, 2014, vol. 466, pp. 371–382. (Series: Communications in Computer and Information Science).
2. Al'-Ashval' M. S., Kravets A. G. Analiz pokazatelej kachestva mobil'nyh korporativnyh setej, *Sovremennye problemy nauki i obrazovaniya*, RAE, 2014, no. 6, available at: <http://www.science-education.ru/120-16129> (in Russian).
3. Al'-Ashval' M. S., Kravets A. G. Sistema kriteriev i pokazatelej kachestva mobil'nyh korporativnyh setej, *Innovacii na osnove informacionnyh i kommunikacionnyh tehnologij*. Info 2014: mater. XI mezhdunar. nauch.-prakt. konf. (g. Sochi, 1–10 okt. 2014 g.), Moscow, Nacional'nyj issledovatel'skij un-t Vysshaja shkola jekonomiki, 2014, pp. 501–504 (in Russian).
4. Kravec A. G., Al'-Ashval' M. S. Mobile corporate networks security control, *International Siberian Conference on Control and Communications (SIBCON-2016)*, Russia, Moscow, May 12–14, 2016: Proceedings, Moscow, 2016, pp. 1–6. DOI: 10.1109/SIBCON.2016.7491811 (in Russian).
5. Buj Ngok Zyong, Kravets A. G., Le Than' Tung Nguen. Bezopasnaja autentifikacija v sisteme upravlenija korporativnoj mobil'nost'ju, *Izvestija VolgGTU. Ser. Aktual'nye problemy upravlenija, vychislitel'noj tehniki i informatiki v tehnikeskikh sistemah*, 2015, no. 13 (177), pp. 45–51 (in Russian).
6. Buj Ngok Zyong, Kravets A. G., Le Than' Tung Nguen. Informacionnaja bezopasnost' processa registracii Android-ustrojstva v sisteme upravlenija korporativnoj mobil'nost'ju, *Vestnik komp'juternyh i informacionnyh tehnologij*, 2016, no. 8, pp. 44–51.
7. Buj Ngok Zyong, Le Than' Tung Nguen, Kravets A. G. Problema proverki root-prav na Android-ustrojstve v sisteme upravlenija mobil'nymi prilozhenijami, *Informacionnye tehnologii v nauke, obrazovanii i upravlenii: mater. XLIV mezhdunar. konf. i XIV mezhdunar. konf. molodyh uchjonyh IT + S&E'15*, Gurfuz, 22 maja — 1 ijunja 2015 g., vesennaja sessija, ed. E. L. Gloriov; Institut novyh informacionnyh tehnologij, Moscow, 2015, pp. 420–426 (in Russian).
8. Bui N. D. Architecture of Mobility Management system in Corporation Nextwork, *Sovremennaja nauka: aktual'nye problemy teorii i praktiki, Ser. Estestvennye i tehnikeskie nauki*, 2016, no. 09–10, pp. 30–35 (in Russian).
9. Push Technology: A Key Ingredient of Application Interactivity. Available at: http://www.seven.com/downloads/pdf/SEVEN_Push_Whitepaper.pdf (accessed: 20.1.2017).
10. Bui N. D., Nguyen L. T. T., Kravets A. G. Development of Push notification service in Enterprise Mobility Management system using Google Cloud Messaging. *Junost' i Znanija — Garantija Uspeha — 2015: sb. nauch. tr. 2-j mezhdunar. nauch.-prakt. konf. 1–2 okt. 2015 g. V 2 t. T. 2. Jugo-Zapadnyj gos. un-t, ZAO "Universitetskaja kniga" [i dr.]*, Kursk, 2015, pp. 28–30 (in Russian).
11. Jung M., Kim J. H., Wi H. W., Kim S. Things-to-cloud communication: technology overview and design considerations, *2015 5th International Conference on the Internet of Things (IoT), October 26–28, 2015, Seoul, S. Korea*.
12. HTTP. Available at: https://en.wikipedia.org/wiki/Quality_of_service (accessed: 26.01.17).
13. HTTP2. Available at: <http://habrahabr.ru/post/221427> (accessed: 05.02.2017).
14. IBM. MQTT V3.1. *Protocol Specification*, International Business Machines Corporation Eurotech. 2015.
15. Quality of service. Available at: https://en.wikipedia.org/wiki/Quality_of_service, accessed: 26.01.17.
16. ITU-T. Extensible Messaging and Presence Protocol (XMPP): Core, RFC-3920. 2004.
17. Shelby Z., Hartke K., Bormann C. Constrained Application Protocol. Available at: <https://datatracker.ietf.org/doc/rfc7252> (accessed: 18.01.2017).
18. Stephen N. Power Profiling: HTTPS Long Polling vs. MQTT with SSL, on Android. Available at: <http://stephendnicholas.com/archives/1217> (accessed: 28.01.17).
19. Rodríguez-Domínguez C., Benghazi K., Noguera M. et al. *A Communication Model to Integrate the Request-Response and the*

Publish-Subscribe Paradigms into Ubiquitous Systems, E. T. S. I. I. T. University of Granada, C/Periodista Daniel Saucedo Aranda S/N, 18071 Granada, Spain, 2012.

20. **Jansen G.** Thoughts on RESTful API design. Available at: <https://media.readthedocs.org/pdf/restful-api-design/latest/restful-api-design.pdf> (accessed: 18.01.2017).

21. **Abbott M. L., Fisher M. T.** *The art of scalability: Scalable Web Architecture, Processes, and Organizations for the Modern Enterprise*. 2 ed., Pearson Education, Inc., 2015.

22. **Platforma** dlya korporativnykh mobil'nykh prilozhenij. Available at: https://ru.wikipedia.org/wiki/Platforma_dlya_korporativnykh_mobil'nykh_prilozhenij (accessed: 20.01.2017).

23. **Bui N. D., Kravets A. G., Nguyen T. A., Nguyen L. T. T.** Tracking events in mobile device management system, *IISA 2015 — 6th International Conference on Information, Intelligence, Systems and Applications, Corfu, Greece, 6 July 2015 — 8 July 2015*:

Conference Proceeding, Ionian University, Institute of Electrical and Electronics Engineers (IEEE), Piscataway, USA. 2015. 6 p. DOI: 10.1109/IISA.2015.7388127.

24. **Pongthawornkamol T., Nahrstedt K., Wang G.** The analysis of publish/subscribe systems over mobile wireless ad hoc networks, *Proc. of ACM MobiQuitous, 07 Aug 2007*, pp. 1—8.

25. **Message** Broker. Available at: https://en.wikipedia.org/wiki/Message_broker (accessed: 26.08.16).

26. **Eclipse** Paho MQTT. Available at: <https://eclipse.org/paho/> (accessed: 18.01.2017).

27. **Smack 4.1** Readme and Upgrade Guide. Available at: <https://github.com/igniterealtime/Smack/wiki/Smack-4.1-Readme-and-Upgrade-Guide> (accessed: 18.01.2017).

28. **CoAP** Client for Android based on nCoAP. Available at: <https://github.com/okleine/spitfirefox> (accessed: 18.01.2017).

Адрес редакции:

107076, Москва, Строминский пер., 4

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

E-mail: it@novtex.ru

Технический редактор *Е. В. Конова*.

Корректор *Е. В. Комиссарова*.

Сдано в набор 09.11.2017. Подписано в печать 25.12.2017. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT118. Цена договорная.

Журнал зарегистрирован в Министерстве Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-15565 от 02 июня 2003 г.

Оригинал-макет ООО "Авансд солюшнз". Отпечатано в ООО "Авансд солюшнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.
